

ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

ДО
МИНИСТЪРСТВО НА ТРАНСПОРТА, ИНФОРМАЦИОННИТЕ
ТЕХНОЛОГИИ И СЪБЩЕНИЯТА

От Деян Прокопов

(собствено, бащино и фамилно име)

в качеството си на Управител (длъжност) Гама Консулт - Калинкин, Прокопов и с-ие
СД (наименование на участника) със седалище и адрес на управление гр. София п.к.1574
бул. Шипченски проход 63, ет.2, вписано в Търговския регистър с ЕИК 121562456,

УВАЖАЕМИ ГОСПОДА,

С настоящето Ви представяме нашето техническо предложение за изпълнение на поръчката в съответствие с техническата спецификация и изискванията на възложителя в обявеното от Вас, публично състезание за възлагане на обществена поръчка с предмет: „Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в съответствие с инженеринговите услуги, поддържащи прилагането на EUCISE2020 мрежата и съпровождащи услуги по време на демонстрационната фаза на обмен на данни“.

Предлагаме да изпълним поръчката съгласно изискванията на документацията за обществената поръчка и Техническата спецификация, а именно:

- Приложение 1: Предложение за организация за изпълнение и управление на поръчката, включително подробен график;
- Приложение 2: Предложение за управление на риска при изпълнение на поръчката;
- Приложение 3: Предложение за технологично решение за изграждане на EUCISE2020 Адаптер;
- Приложение 4: Предложение за поддръжка работоспособността на Адаптера за периода на тестова обмяна на данни.

/ Участникът изготвя предложение за изпълнение на поръчката, което включва:

- Предложение за организация за изпълнение и управление на поръчката;
- Предложено технологично решение за изграждане на EUCISE2020 Адаптер;
- Предложение за управление на риска при изпълнение на поръчката;
- Предложение за поддръжка работоспособността на Адаптера за периода на тестова обмяна на данни.
- Подробен график, в който следва да се конкретизират сроковете за изпълнение на всяка дейност и поддейност от настоящата поръчка.



EUCISE2020 has received funding from the European Union's seventh framework programme under grant agreement no: 608385



ДЕКЛАРИРАМ, ЧЕ:

- срокът за изпълнение на поръчката е до 10 (десет) месеца от влизане в сила на договора, включително сроковете на Етапите са, съгласно описаните в Документацията.
- съм запознат и съм съгласен с клаузите на приложения проект на договор за възлагане обществената поръчка с предмет: „Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в съответствие с инженеринговите услуги, поддържащи прилагането на EUCISE2020 мрежата и съпровождащи услуги по време на демонстрационната фаза на обмен на данни“.
- Направените от нас предложения и поети ангажименти за възлагането на обществената поръчка с предмет: „Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в съответствие с инженеринговите услуги, поддържащи прилагането на EUCISE2020 мрежата и съпровождащи услуги по време на демонстрационната фаза на обмен на данни“ са валидни за срок от 120 (сто и двадесет) дни, считано от датата, определена за краен срок за получаване на офертите. Офертата ще остане обвързваща за нас и може да бъде приета по всяко време, преди изтичане на този срок.
- При изготвяне на офертата за участие в публично състезание за възлагане на обществена поръчка с предмет: „Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в съответствие с инженеринговите услуги, поддържащи прилагането на EUCISE2020 мрежата и съпровождащи услуги по време на демонстрационната фаза на обмен на данни“ са спазени изискванията за задълженията, свързани с данъци и осигуровки, закрила на заетостта и условията на труд

Известна ми е отговорността по чл.313 от Наказателния кодекс на Република България за неверни данни.

Като неразделна част от Техническото предложение, прилагаме:

1. Документ за упълномощаване, в оригинал или нотариално заверено копие, когато е приложимо;
2. Декларация по чл.102, ал.1 и 2 от ЗОП, ако е приложимо, по образец № 2.


Дата: 09/03/2018г

(подпис и печат)

С уважение

По чл. 102, ал. 1 от ЗОП

Делян Прокопов



(име и фамилия на законния представител на участника или надлежно упълномощено лице)

Указания към попълване на Образец № 1:

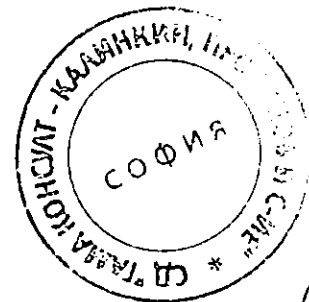
1. Техническото предложение по Образец № 1 е неразделна част от офертата на участника и се представя в запечатаната непрозрачна опаковка.

 EUCISE2020 has received funding from the European Union's seventh framework programme under grant agreement no: 608385





2. Образец № 1 се подписва от законния представител на участника или надлежно упълномощено лице.
3. Ако участникът е обединение, Образец № 1 се представя от името на обединението участник и се подписва от партньора, който представлява обединението за целите на обществената поръчка или надлежно упълномощено лице.
4. Документът за упълномощаване е неразделна част от Техническото предложение и се представя в оригинал или нотариално заверено копие.
5. Информацията се представя в съответствие с Техническата спецификация, както и Методиката за определяне на комплексната оценка и съдържа следните приложения:
 - 5.1. Организация за изпълнение и управление на поръчката.
 - 5.2. Предложение за управление на риска при изпълнение на поръчката.
 - 5.3. Предложено технологично решение за изграждане на EUCISE2020 Адаптер.
 - 5.4. Поддръжка работоспособността на Адаптера за периода на тестова обмяна на данни.
 - 5.5. Подробен график, в който следва да се конкретизират сроковете за изпълнение на всяка дейност и поддейност от настоящата поръчка.



EUCISE2020 has received funding from the European Union's seventh framework programme under grant agreement no: 608385

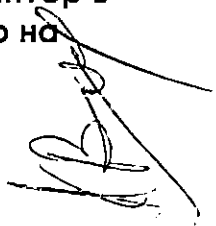


към

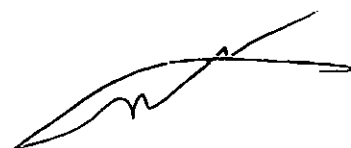
ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

за участие в открита процедура за възлагане
на обществена поръчка с предмет:

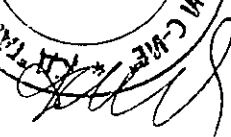
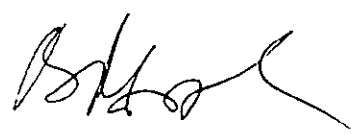
**„Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в
съответствие с инженеринговите услуги, поддържащи прилагането на
EUCISE2020 мрежата и съпровождащи услуги по време на
демонстрационната фаза на обмен на данни“**



Предложение за организация за
изпълнение и управление на поръчката,
вкл. подробен план-график



2018 г.

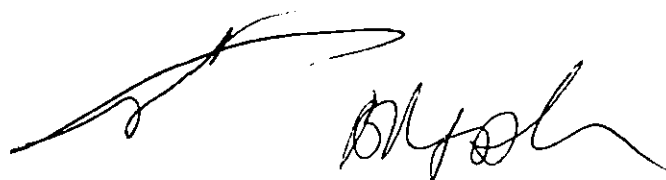


1
35

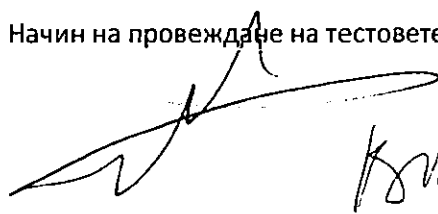

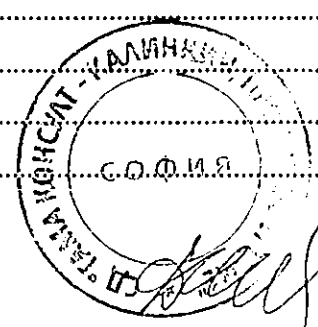


Съдържание

1. Въведение.....	5
2. Методология за управление на проекта	5
2.1. Общ подход за управление на проекта	5
2.2. Процеси за управление на проекта съгласно PMBOK	6
2.3. Процесни области при управление на проекта	6
2.3.1. Управление на интеграцията	6
2.3.2. Управление на обхвата	7
2.3.3. Управление на времето	8
2.3.4. Управление на разходите	8
2.3.5. Управление на качеството	9
2.3.6. Управление на човешките ресурси	9
2.3.7. Управление на комуникацията	10
2.3.8. Управление на риска	10
2.3.9. Управление на заинтересованите страни (stakeholders).....	11
2.3.10. Управление на доставките	11
2.4. Общи управленски дисциплини	12
2.5. Общи организационни принципи	12
2.6. Обосновка защо предложената методология ще допринесе за успешното изпълнение на поръчката	13
3. Предварителен план за управление на проекта.....	13
3.1. Организация на работа.....	13
3.1.1. Организация на взаимодействието между Изпълнителя и Възложителя.....	13
3.1.2. Органи за управление на проекта	14
3.1.3. Организационна диаграма.....	16
3.1.4. Роли и отговорности	16
3.1.5. Участие на експертите в дейностите по попоекта	21
3.1.6. Координация в рамките на итерациите.....	22
3.2. Управление на комуникацията.....	22
3.2.1. Проектна комуникация.....	22
3.2.2. Правила и канали за комуникация.....	22
3.2.3. Управление на документацията и инструменти за комуникация	23
3.2.4. Провеждане на срещи	24
3.2.5. Канали за комуникация	26
3.2.6. Докладване.....	26



3.3.	Управление на качеството.....	27
3.3.1.	Стратегия за качество в проекта	27
3.3.2.	Процедури за контрол на качеството.....	28
3.3.3.	Роли и отговорности в процесите по осигуряване и контрол на качеството	29
3.4.	Механизми за проследимост и отчетност на изпълнението	30
4.	Етапи, срок и график за изпълнение	33
4.1.	Срок за изпълнение на поръчката.....	33
4.2.	Етапи, срокове и резултати	34
4.3.	Подробен план-график за изпълнение	36
5.	Описание на подход и методи за ключови дейности от изпълнението на проекта.....	41
5.1.	Подход за бизнес анализ.....	41
5.1.1.	Основни области на знание (процеси).....	42
5.1.2.	Описание на процесите по бизнес анализ	45
5.1.3.	Адаптирана подхода за бизнес анализ.....	58
5.2.	Подход за проектиране на софтуер.....	58
5.2.1.	Процес за проектиране	59
5.2.2.	Софтуерна архитектура.....	62
5.2.3.	Обектно-ориентиран дизайн	65
5.2.4.	Моделиране с UML	66
5.2.5.	Софтуер за моделиране на UML – Sparx Systems Enterprise Architect.....	68
5.2.6.	Адаптиране на подхода за проектиране на софтуер.....	69
5.3.	Подход за софтуерна разработка	69
5.3.1.	Процес за софтуерната разработка	69
5.3.2.	Характеристики на процеса за софтуерна разработка	74
5.3.3.	Вътрешно-екипна координация в рамките на итерациите.....	75
5.3.4.	Обектно-ориентирано програмиране.....	77
5.3.5.	Дейности за интегриране на функционалности/компоненти в системата.....	79
5.3.6.	Стратегия за интегриране на функционалности/компоненти в системата	81
5.3.7.	Дневни билдове и тестове	82
5.3.8.	Управление на версиите.....	83
5.3.9.	Описание на инструментите, които ще се ползват от изпълнителя за извършване на разработка, тестване и внедряване	86
5.3.10.	Адаптиране на подхода за софтуерна разработка.....	87
5.4.	Подход за тестване	87
5.4.1.	Стратегия за тестване.....	88
5.4.2.	Начин на провеждане на тестовите.....	89

5.4.3.	Видове тестове	89
5.4.4.	Входни и изходни критерии.....	91
5.4.5.	Докладване на дефекти и несъответствия	93
5.4.6.	Инструменти за провеждане на тестовете	93
5.4.7.	Адаптиране на подхода за тестване.....	94
5.5.	Подход за внедряване.....	94
5.5.1.	Общи положения	94
5.5.2.	Стъпки за внедряване.....	94
5.6.	Механизъм и процедури за включване на граждани в процесите по разработка, тестване и идентифициране на пропуски на софтуера.....	95
6.	Обучение.....	96
6.1.	Учебни курсове.....	96
6.2.	Начин на провеждане на обучението	96
6.3.	Учебни материали.....	97
7.	Документация	97
7.1.	Изисквания към документацията	97
7.2.	Прозрачност и отчетност	98
7.3.	Спецификация на софтуерните изисквания	98
7.4.	Техническа документация.....	98
7.5.	Протоколи.....	99
7.6.	Комуникация и доклади.....	99

[Handwritten signature]

[Handwritten signature]



[Handwritten signature]

1. Въведение

Настоящото приложение към техническото предложение описва предложената организация за управление и изпълнение на поръчката, като включва:

- Методология за управление на проекта;
- Предварителен план за управление на проекта;
- Етапи, срок и график за изпълнение;
- Описание на подход и метод за ключови дейности от изпълнението на проекта;
- Обучение;
- Документация.

Представена е концепция, в която са описани всички етапи и действия по изпълнение на услугите и дейностите на поръчката, включително методите за тяхното осъществяване и резултатите, които ще бъдат постигнати от реализацията им.

Предложена е организация на работа, която показва разпределение на задачите и отговорностите между отделните експерти; показани са взаимовръзките между тях и способите за взаимодействие с възложителя ИАМА.

Предложената концепция съответства на предмета и обхвата на поръчката, разработена е съобразно структурната, смислова и логическа последователност на Техническата спецификация на възложителя и би осигурила необходимото качествено ниво на изпълнение.

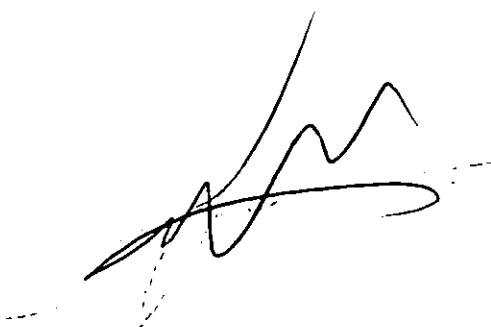
Представен е план-график за управление на изпълнението, който е съобразен с времетраенето и последователността на всеки етап.

Предложени са механизми за проследимост и отчетност на изпълнението, които да осигурят ефективно управление на проекта, така че да се гарантира степента на постигане на целите при съпоставяне на действителните и очакваните резултати от изпълнението.

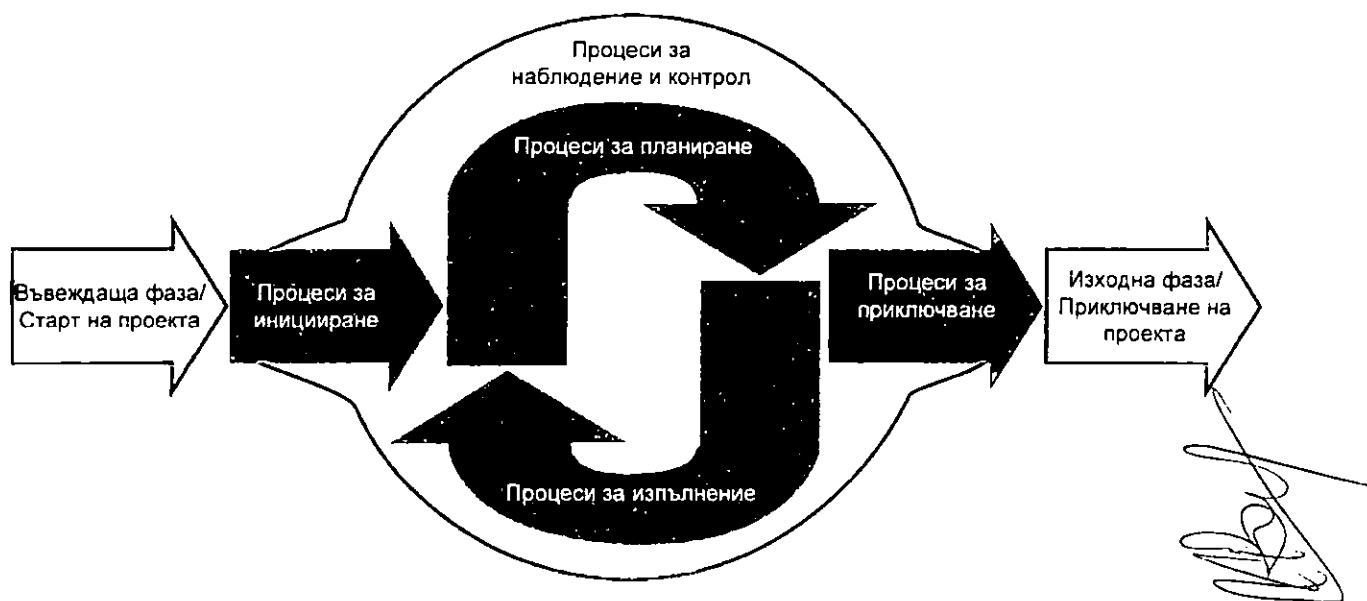
2. Методология за управление на проекта

2.1. Общ подход за управление на проекта

При управлението на проекта ще бъде използвана практикуваната при нас система от правила, принципи и стандарти на Института за Управление на Проекти (Project Management Institute, PMI), систематизирани в Ръководство за Система от Знания за Управление на Проекти (PMBOK Guide) – пето издание.



2.2. Процеси за управление на проекта съгласно PMBOK



Управлението на проекта представлява прилагане на знания, умения, средства и методи в дейностите, за да се изпълнят изискванията на проекта. Това изисква ефективно управление на съответните процеси. Процесите за управление на проекта са групирани в пет групи:

- Група процеси за инициране – тези процеси се провеждат с цел дефиниране на нов проект или фаза на съществуващ проект за получаване на разрешение за започване на проекта или фазата.
- Група процеси за планиране – тези процеси се изпълняват за установяване на обхвата на проекта, прецизиране на целите, дефиниране на курс на действие, необходим за постигане на тези цели.
- Група процеси за изпълнение – тези процеси се изпълняват за реализиране на определената работа в плана за управление на проекта, за да бъдат спазени спецификациите на проекта.
- Група процеси за наблюдение и контрол – тези процеси са необходими за проследяване, контрол и регулиране на напредъка и изпълнението на проекта, идентифициране на области, където са необходими промени в плана и инициране на съответните промени.
- Група процеси за приключване – процеси, които се изпълняват с цел финализиране на всички дейности от всички групи процеси за формалното приключване на проекта или дадена фаза от него.

2.3. Процесни области при управление на проекта

2.3.1. Управление на интеграцията

Управлението на интеграцията в проекта включва процеси и дейности, нужни за идентифициране, дефиниране, комбиниране, унифициране и координиране на разнообразни процеси и дейности за управление на проекта в рамките на групите процеси за управление на проекта. В рамките на контекста за управление на проекти, интеграцията включва характеристики като унификация, консолидация и интеграционни дейности, които са решаващи за завършването на проекта и успешното управление на очакванията на заинтересованите страни и покриване на изискванията. Управлението на интеграцията в проекта изисква вземането на решения относно разпределянето на ресурсите, балансиране между конкуриращи се цели и алтернативи и управлението на зависимости във всички области на управление на проекта.

Процеси при управление на интеграцията:

[Handwritten signature]



- Разработване на Проектна дефиниция (Харта на проекта) – процес на разработване на документ, който формално оторизира проекта или фаза от него и документира началните изисквания, които ще удовлетворят нуждите и очакванията на заинтересованите страни.
- Разработване на План за управление на проекта – процесът на документиране на действията, необходими за дефиниране, подготовка, интегриране и координиране на всички допълнителни планове.
- Направляване и управление изпълнението на проекта – процесът на изпълнение на работата, дефинирана в плана за управление на проекта за постигане на целите на проекта.
- Наблюдение и контрол на работата по проекта – процесът на проследяване, преглед и регулиране на напредъка с цел покриването на целите за производителност, дефинирани в плана за управление на проекта.
- Интегриран контрол на промените – процесът на преглед за всички искания за промяна, одобряване на промени и управление на промените към отчетните продукти, проектните документи, проектните активи на организацията и плана за управление на проекта.
- Затваряне на проекта или фаза от проекта – процесът на финализиране на дейностите в група процеси за управление на проекта с цел формалното приключване на проект или фаза на проект.

Очаквани резултати:

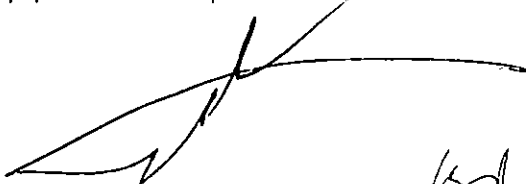
- Проектна дефиниция (Харта на проекта);
- План за управление на проекта;
- Изпълнени работни пакети (задачи) по проекта;
- Данни за напредъка и производителността;
- Заявки за промени и решения по тях;
- Научени уроци;
- Формално приключени фази и проект.

2.3.2. Управление на обхвата

Управлението на обхвата в проекта включва процесите, необходими за осигуряване на това, че проектът включва цялата изисквана работа и само работата, която се изисква за успешното му приключване. Управлението на обхвата най-вече се отнася до определяне и контролиране на това какво се включва и какво не в проекта.

Процеси при управление на обхвата:

- Събиране на изисквания – процесът на дефиниране и документиране на нуждите на заинтересованите страни за постигане на проектните цели.
- Дефиниране на обхвата – процесът на разработване на детайлно описание на проекта и продукта.
- Създаване на структура на пакетите от работа (WBS) – процесът на разделяне на проектните отчетни продукти и работа на по-малки и управляеми части.
- Проверка на обхвата – процесът на формалното приемане за завършените отчетни продукти в проекта.
- Контрол на обхвата – процесът на мониторинг на статуса на обхвата на проекта и продукта, и управление на промените.





7
40

Тези процеси си взаимодействат с всички останали процеси в другите области при управлението на проекта. Всеки от процесите може да изисква усилия от един или повече хора в зависимост от конкретните нужди в проекта. Всеки процес се появява поне веднъж в проекта и веднъж или повече в отделните фази на проекта, в случай че проектът е разделен на фази.

Очаквани резултати:

- Специфицирани изисквания и нужди на заинтересованите страни;
- Дефиниция на обхвата;
- Структура на пакетите от работа (WBS).

2.3.3. Управление на времето

Управлението на времето в проекта включва изпълнението на задачи, необходими за навременното приключване на проекта.

Процеси при управление на времето в проекта:

- Дефиниране на задачите – процесът по идентифициране на специфични задачи, които следва да бъдат изпълнение за създаване на отчетните продукти по проекта.
- Задаване на последователност на задачите – процесът на идентифициране и документиране на зависимостите между задачите в проекта.
- Оценка на ресурсите по задачи – процесът на оценка на вида и количеството на материали, хора, оборудване или доставки, необходими за изпълнението на всяка задача.
- Оценка на продължителността на задачите – процесът на определяне на ориентировъчна продължителност на работното време, необходимо за изпълнение на всяка от задачите с определените ресурси.
- Разработване на график – процесът по анализиране на последователността на задачите, тяхната продължителност, изисквания за ресурси и ограничения с цел създаване на график за изпълнение на проекта.
- Контрол на графика – процесът на мониторинг на статуса на проекта за актуализиране на напредъка и управление на промените спрямо приетия график.

Очаквани резултати:

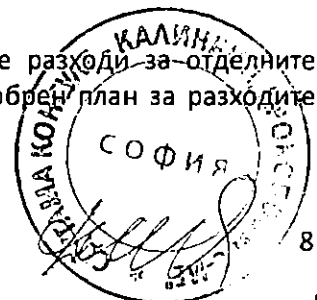
- Дефинирани на задачите;
- Последователност и зависимост на задачите;
- Оценка на ресурсите по задачи;
- Продължителност на задачите;
- График.

2.3.4. Управление на разходите

Управлението на разходите в проекта включва оценка, бюджетиране и контролиране на разходите така, че проектът да бъде завършен в рамките на одобрения бюджет.

Процеси:

- Оценка на разходите – процесът на разработване на ориентировъчна оценка на паричните средства за изпълнение на проектните задачи.
- Определяне на бюджет – процесът на обобщаване на оценените разходи за отделните задачи или пакети от работа за определяне и оторизиране на одобрен план за разходите (включващ разпределението им във времето).



- Контролиране на разходите – процесът на мониторинг на статуса на проекта за актуализиране на бюджета на проекта и управление на промените в приетия план за разходите.

Очаквани резултати:

- Оценка на разходите;
- Бюджет.

2.3.5. Управление на качеството

Управлението на качеството в проекта включва процесите и дейностите на изпълнителя, които определят политиките за качество, целите и отговорностите така, че проектът да удовлетвори нуждите, поради които е предприет. Това включва системи за управление на качеството, политики за качество и процедури за постоянно подобряване на процесите.

Процеси при управление на качеството:

- Планиране на качеството – процесът по идентифициране на изискванията за качество и/или стандартите, на които трябва да отговаря проекта и продукта, и документиране на това как тези стандарти и изисквания ще бъдат прилагани в проекта.
- Осигуряване на качеството – процесът на одититане на изискванията за качество и резултатите от измерванията за контрол на качеството, осигуряващи приетите стандарти. Осигуряването на качеството се прилага спрямо процесите в проекта.
- Контрол на качеството – процесът на мониторинг и регистриране на резултатите от изпълнение на дейностите по контрол на качеството за оценка на производителността и препоръчване на промени. Контролът на качеството се прилага спрямо продуктите.

Управлението на качеството се отнася към проекта и към продуктите от проекта. То се прилага във всички проекти, независимо от естеството на техния продукт.

Очаквани резултати:

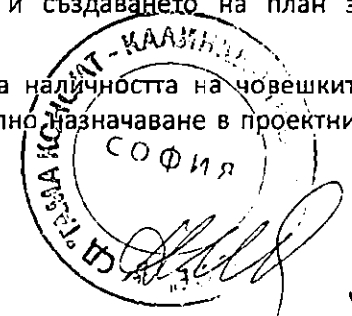
- План за управление на качеството;
- Осигурено ниво на качеството на продукта или услугата, разработвани в проекта;
- Осигурено спазване на процесите в проекта.

2.3.6. Управление на човешките ресурси

Управлението на човешките ресурси включва процесите на организиране, управление и лидерство на проектния екип. Проектният екип включва хора с определени роли и отговорности за изпълнение на проекта. Видът и броя на членовете на екипа може да се променя често в хода на изпълнение на проекта. Въпреки, че членовете на екипа имат възложени конкретни роли и отговорности, въвличането на всички членове при планирането и вземането на решения може да бъде полезно. Ранното включване и участие на членовете на екипа добавя техния опит в процеса на планиране и подсилва тяхната ангажираност към проекта.

Процеси при управление на човешките ресурси:

- Разработване на План за управление на човешките ресурси – процесът на идентифициране и документиране на ролите и отговорностите в проекта, необходимите умения и познания на членовете на екипа, йерархията и организацията в екипа и създаването на план за управление на персонала.
- Набиране на проектен екип – процесът на потвърждаване на наличността на човешките ресурси и присъединяването им към екипа на проекта. Формално Назначаване в проектния екип.



- Развиване на проектния екип – процесът по усъвършенстване на компетентностите, взаимодействието в екипа и цялостната среда в екипа с цел подобряване на производителността в проекта.
- Управление на проектния екип – процесът по проследяване на производителността на членовете на екипа, даване на обратна информация, разрешаване на казуси и управление на промени за оптимизиране на производителността в проекта.

Очаквани резултати:

- План за управление на човешките ресурси;
- Формиран екип за изпълнение на проекта.

2.3.7. Управление на комуникацията

Управлението на комуникацията в проекта включва процесите, необходими за осигуряване на навременно и точно генериране, събиране, разпространение, съхранение и структуриране на проектната информация. Проектните ръководители прекарват значителна част от времето си в общуване с членовете на екипа и заинтересованите лица – външни или вътрешни за организацията.

Процеси:

- Планиране на комуникацията – процесът на определяне на нуждите от информация на заинтересованите лица и определяне на подхода за комуникация.
- Разпространяване на информация – процесът по предоставяне на информацията на заинтересованите лица по проекта така, както е планирано.
- Управление на очакванията на заинтересованите лица – процесът на комуникация и работа със заинтересованите лица за посрещане на техните нужди и обсъждане на казуси при тяхната поява.
- Докладване на производителността – процесът на събиране и разпространяване на информация за производителността, включително доклади за състоянието и измерване на напредъка и прогнози.

Очаквани резултати:

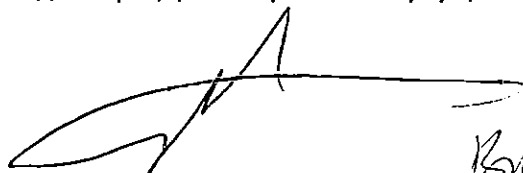
- План за комуникация;
- Артефакти (доклади, протоколи от работни срещи и др.) от комуникацията.

2.3.8. Управление на риска

Управлението на риска в проекта включва процесите по планиране управлението на риска, идентификация и анализ, планиране на ответни действия, мониторинг и контрол на рисковете в проекта. Целта на управлението на риска е повишаване на вероятността и влиянието от позитивните събития и понижаване на вероятността и влиянието на негативните събития върху проекта.

Процеси при управление на риска в проекта:

- Планиране на управлението на риска – процесът на дефиниране как ще бъдат управлявани дейностите по управление на риска в проекта.
- Идентифициране на рисковете – процесът по определяне на рискове, които могат да въздействат на проекта и документиране на техните характеристики.
- Качествен анализ на рисковете – процесът по приоритизиране и категоризиране на рисковете, както и оценяване на тяхната вероятност и влияние.
- Количествен анализ на рисковете – процесът по количествено анализиране на ефекта от идентифицираните рискове върху целите на проекта.





- Планиране на ответни действия – процесът на разработване на опции и действия за подсилване на положителните възможности и намаляване на заплахите спрямо целите на проекта.
- Наблюдение и контрол на рисковете – процесът по изпълнение на планираните ответни действия, регистриране и проследяване на идентифицираните рискове, мониторинг на остатъчните рискове, идентифициране на нови рискове и оценка на ефективността на процесите за управление на риска.

Очаквани резултати:

- План за управление на риска;
- Регистър на рисковете;
- Качествен и количествен анализ на рисковете;
- Планове за ответни действия при възникване на рискове.

2.3.9. Управление на заинтересованите страни (stakeholders)

Процеса по управление на заинтересованите страни започва с идентифицирането на тези хора и групи от организацията, които имат отношение към или в следствие на проектната реализация.

Идентифициране на заинтересованите лица – процесът на идентифициране на всички хора и организации, повлияни от проекта и документирането на информацията за техните интереси, отношение и влияние върху успеха на проекта.

Основната полза от управлението на този процес, е че позволява на Ръководителя на проекта да идентифицира най-подходящия фокус към всяка една от заинтересованите страни или група от хора имащи отношение към проекта.

Очаквани резултати:

- Регистър на заинтересованите лица.

2.3.10. Управление на доставките

Управлението на доставките в проекта включва процесите, необходими за доставка или придобиване на продукти и услуги извън проектния екип.

Процеси при управление на доставките:

- Планиране на доставките – процесът на документиране на решенията за доставки в проекта, специфициране на подхода и идентифициране на потенциалните доставчици (в конкретния случай доставчиците са определени).
- Провеждане на доставките – процесът по набиране на оферти от доставчици, избор на доставчик и сключване на договор (в конкретния случай набиране на оферти няма да е необходимо).
- Администриране на доставките – процесът по управление на взаимоотношенията при доставка, мониторинг на договора и правене на промени и корекции при необходимост.
- Приключване на доставките – процесът на завършване на всяка доставка в проекта. Включва и формалното приключване на договора.

Очаквани резултати:

- План за доставки;
- Документация за доставки;
- Извършени доставки.



44

2.4. Общи управленски дисциплини

Управление на конфигурацията – за продуктите, необходими за управление на конфигурацията, за които Възложителя не е посочил продукти или посочените от него продукти не са достатъчни, ще се използват продукти, интегрирани в единна система, посочени от Изпълнителя. Тези системи ще предоставят като минимум възможността за достъп до всички обекти в конфигурацията, маркиране на версиите и автоматизирано генериране на актуалната конфигурация и нейното съдържание. Достъп ще бъде осигурен на упълномощени от Възложителя лица в рамките на изпълнението на целия проект.

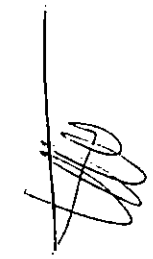
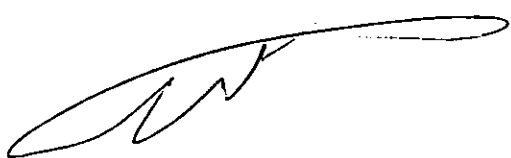
Управление на промените - спомага за оптимизиране на излагането на риск, неблагоприятното въздействие или прекъсването на нормалната експлоатация. Процесът включва подаване и регистриране на искания за промяна, оценка на въздействието, цената, ползите и рисковете, свързани с промяната, подготовка на бизнес обосновката и получаване на одобрение, управление и координация на изпълнението на промените, наблюдение и отчитане на дейностите по изпълнение на промените, затваряне и преглед на исканията за промяна. Исканията за промяна се подават само в писмен вид и могат да се инициират и от двете страни от упълномощени представители.

Изпълнителя ще използва специализирана система за целите на управление на промените. Ще бъде предоставено на Възложителя Детайлно описание на процедурата за управление на промените и достъп до системата. Записите за промените ще се преглеждат от ръководителите на Изпълнителя и Възложителя.

Управление на проблемите - включва грешки свързани с хардуер, софтуер и приложения. Идентифицирането на проблемите ще става с набор от автоматизирани и неавтоматизирани методи. Включва: контрол на проблема (идентифициране, регистриране, класифициране; проучване на причините; диагностика и идентифициране на мерки за решаването му; проследяване и наблюдение на проблемите), контрол на грешките (оценка и изпълнение на мерки за решаване на проблема; затваряне на проблема; проследяване и наблюдение на диагностицирани проблеми), предотвратяване на проблеми (анализ на тенденциите и извършване на насочени превантивни действия). Изпълнителят ще използва специализирани системи за целите на докладването на проблеми и управление на тяхното коригиране. Ще предостави детайлно описание на процедурата, по която ще управлява този процес, както и достъп до съответната система.

2.5. Общи организационни принципи

Задължително изискване е да се спазят утвърдените хоризонтални и вертикални принципи на организация на изпълнението на предмета на обществената поръчка за гарантирано постигане на желаните резултати от проекта, така че да се покрие пълният набор от компетенции и ноу-хау, необходими за изпълнение на предмета на поръчката, а също така да се гарантира и достатъчно ниво на ангажираност с изпълнението и проблемите на проекта:



- Хоризонталният принцип предполага ангажиране на специалисти от различни звена, така че да се покрие пълният набор от компетенции и ноу-хау по предмета на проекта и същевременно екипът да усвои новите разработки на достатъчно ранен етап, така че да е в състояние пълноценно да ги използва и развива и след приключване на проекта;
- Вертикалният принцип включва участие на експерти и представители на различните управленски нива, така че управленският екип да покрива както експертните области, необходими за правилното и качествено изпълнение на проекта, така и управленски и организационни умения и възможности за осъществяване на политиката във връзка с изпълнението на проекта. Чрез участие на ръководители на звената – ползватели на резултата от проекта, ще се гарантира достатъчно ниво на ангажираност на институцията с проблемите на проекта.

2.6. Обосновка защо предложената методология ще допринесе за успешното изпълнение на поръчката

Значителният обхват, кратките срокове на проекта и взаимодействието с множество заинтересовани страни изискват той да се управлява по стандартен унифициран начин. Затова сме избрали да приложим популярните добри практики и методология на световно утвърдена организация като Project Management Institute (PMI). Утвърдили сме добрите практики на PMI като основа за прилаганата от нас методология за управление на проекти. Екипът ни има необходимите познания и опит за прилагане на методологията, а проектният ръководител е професионалист с богат опит в областта с прилагането на този подход.

3. Предварителен план за управление на проекта

В настоящия предварителен план за управление на проекта са описани конкретни аспекти от проектното управление в съответствие с изискваната информация съгласно техническото задание. Представена е и информация, необходима за обективно оценяване на предложението. В случай, че бъдем избрани за изпълнител ще разработим подробен проектен план, обхващащ всички необходими области на проектното управление.

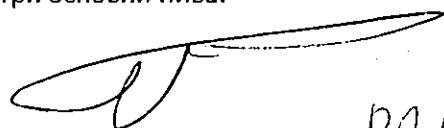
3.1. Организация на работа

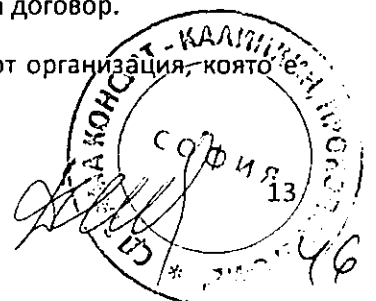
3.1.1. Организация на взаимодействието между Изпълнителя и Възложителя

Опитът показва, че успешното изпълнение на ИТ проекти зависи от ефективната и мотивирана съвместна работа на множество хора, с различни роли, знания и отговорности. Всички те трябва да работят в синхрон за постигане на резултатите от проекта в предвидените срокове и с необходимото качество. За тази цел е необходимо проектирането на ефикасна организация със съответните правила, роли и отговорности. В следващите параграфи ще опишем нашето предложение за съставянето на една такава организация. Методологията, в началната фаза на проекта в тази организация следва да бъде преразгледана, детайлизирана и съгласувана съвместно с Възложителя.

Проектната организация описва всички основни роли и функции, заедно с техните права и задължения, които най-общо могат да бъдат разделени на няколко нива. Дефинирането и ефективното създаване на проектната организация, както и нейното последващо функциониране и контролиране се извършва незабавно след стартиране на проекта и е един от най-важните аспекти на първата фаза на всеки един проект. Препоръчителните роли и процеси в проектната организация, ще бъдат съгласувани между Възложителя и Изпълнителя след подписване на договор.

Управлението на проекта, съгласно най-добрите практики, се осъществява от организация, която е разпределена в три основни нива:





Стратегическо ниво: Това е най-високото ниво, което обикновено функционира под формата на т.нар. *Комитет за управление на проекта* и в него страните обикновено се представят от своите Управители, Изпълнителни директори или други оправомощени представители, които имат правото и отговорността да дефинират, както и да променят, целите и обхвата на проекта.

Управленско ниво: Това е оперативното управляващо ниво, в което страните са представени от своите *Ръководители на проекта*, които имат правото и отговорността да взимат всички управленски решения, касаещи постигането на дефинираните от Стратегическото ниво цели и резултати, при управление на предварително договорените обхват, срокове и ресурси.

Техническо ниво: Това е оперативното експертно ниво, в което влизат отговорните технически експерти, в лицето на *Ръководителите на екипите по проектиране и разработка* (обикновено софтуерни и/или бизнес архитекти), които носят правото и отговорността за проектирането на предложеното решение, което следва да бъде реализирано.

В духа на добрата практика при изпълнение на проекти с висока сложност или важност, структурата на проектната организация трябва да е напълно симетрична, като за всяко от основните нива на управление има по един отговорен ръководител, съответно от страна на Възложителя и на Изпълнителя.

3.1.2. Органи за управление на проекта

При съставянето на проектната организация следва да бъдат спазени както хоризонталният, така и вертикалният принцип на управление. Накратко, тези два принципа предполагат, че от една страна ще бъдат дефинирани вертикални нива на отговорност, и от друга – че за всяко ниво ще бъде номинирано лице, носещо подходящите права и отговорности за вземане на решения.

Право и отговорност на Възложителя е да номинира и включи в проектната организация лица с необходимите компетенции, управленски и организационни умения и възможности, и най-вече – със съответния авторитет и пълномощия, така че да се гарантират адекватен процес на взимане на решения, устойчивост на резултатите, дългосрочно развитие и непрекъснато подобрене.

Хоризонталният и вертикалният принцип на управление трябва да бъдат съблюдавани от Собствениците на проекта и от двете страни, както при формиране на оперативните екипи, така и при изграждане на вертикалната организация за ескалация, контрол и отчетност.

Ние предлагаме управлението на проекта да се осъществява на трите основни нива, посочени в предходната точка.

Комитетът за управление на проекта, като най-високо ниво в проектната организация, отговаря за определянето, контрола и спазването на стратегическите цели на проекта. Този орган съблюдава спазването на тези цели, както и взема стратегически решения и разрешава несъответствията възникнали при промяна на общите цели на проекта, включително има решаваща роля при управлението на рискове и промени с голямо влияние и/или изискващи допълнителни ресурси. Комитетът разрешава спорове и разногласия, които не са могли да бъдат решени на управленското организационно ниво. Решенията на комитета са окончателни.

Срещите на Комитета за управление на проекта ще се провеждат по график, съгласуван при стартиране на проекта и според сроковете зададени в Техническото задание и Проектният график. Решенията на Комитета ще бъдат задължителни за двете страни по проекта. При необходимост, среща на Комитета за управление на проекта може да бъде свикана извънредно от Собственика на проекта от страна на Възложителя или на Изпълнителя. По своя преценка членовете на Комитета за управление могат да поканят участници от другите нива, например Ръководителите на проекта и

Техническите ръководители, както и други заинтересовани лица. С оглед на постигане на синхрон в изпълнението на дейностите в различните администрации и общите цели на проекта, участието на представители на тези администрации в структурата на проектното управление е от ключово значение и е препоръчително за безпроблемно изпълнение на проекта.

Ръководителите на проекта, като управленско ниво в проектната организация, ще отговарят за вземане на всички текущи решения, касаещи реализацията на проекта, както разрешаването на спорове и разногласия, които не са могли да бъдат решени от по-ниското, експертно, организационно ниво. В случай на невъзможност даден проблем да бъде разрешен на това ниво, както и в случаите на въпроси и рискове, водещи до съществени промени в обхвата, графиката или ресурсите на проекта, проблемът се ескалира към най-високото, стратегическо ниво.


Срещите на Ръководителите на проекта следва да се провеждат по график, съгласуван при стартиране на проекта, но не по-рядко от веднъж месечно. Решенията от срещите на Ръководителите на проекта ще бъдат задължителни за двете страни по проекта. При необходимост, среща на Ръководителите на проекта може да бъде свикана извънредно от Техническия ръководител от страна на Възложителя или на Изпълнителя. По своя преценка Ръководителите на проекта могат да поканят участници от другите нива, например Техническите ръководители и ръководителите на екипи, както и други заинтересовани лица, например Собствениците на проекта.

Техническите ръководители ще отговарят за вземане на всички оперативни решения, касаещи техническата реализация на внедряваните решения, изпълнението на задачите си по проекта, както и за разрешаване на спорове и разногласия, които не могат да бъдат решени вътрешно в дадения екип или между съответните два екипа, заети с конкретната технологична или проектантска задача. В случай на невъзможност даден проблем да бъде разрешен на това експертно ниво проблемът се ескалира към по-горното, управленско ниво.

Препоръчително е Техническите ръководители да имат пълната свобода и отговорност да свикват технически срещи с честота и програма по своя преценка, като особено във фазата на анализ и проектиране това следва да е не по-рядко от веднъж седмично.

Когато трябва да бъде взето решение от по-високо ниво, т.е. наложително е даден проблем да бъде ескалиран, той се описва по възможно най-добрия начин и се изпраща според правилата за проектна комуникация до съответното по-високо ниво в организацията, както е описано по-горе.

Контролните органи по проекта ще могат да участват с в комитета за управление. Докладите за изпълнението на проекта ще бъдат разпространявани и до контролните органи по проекта. Проектат ще бъде управляван прозрачно и всяка необходима информация ще бъде предоставяна на контролните органи.

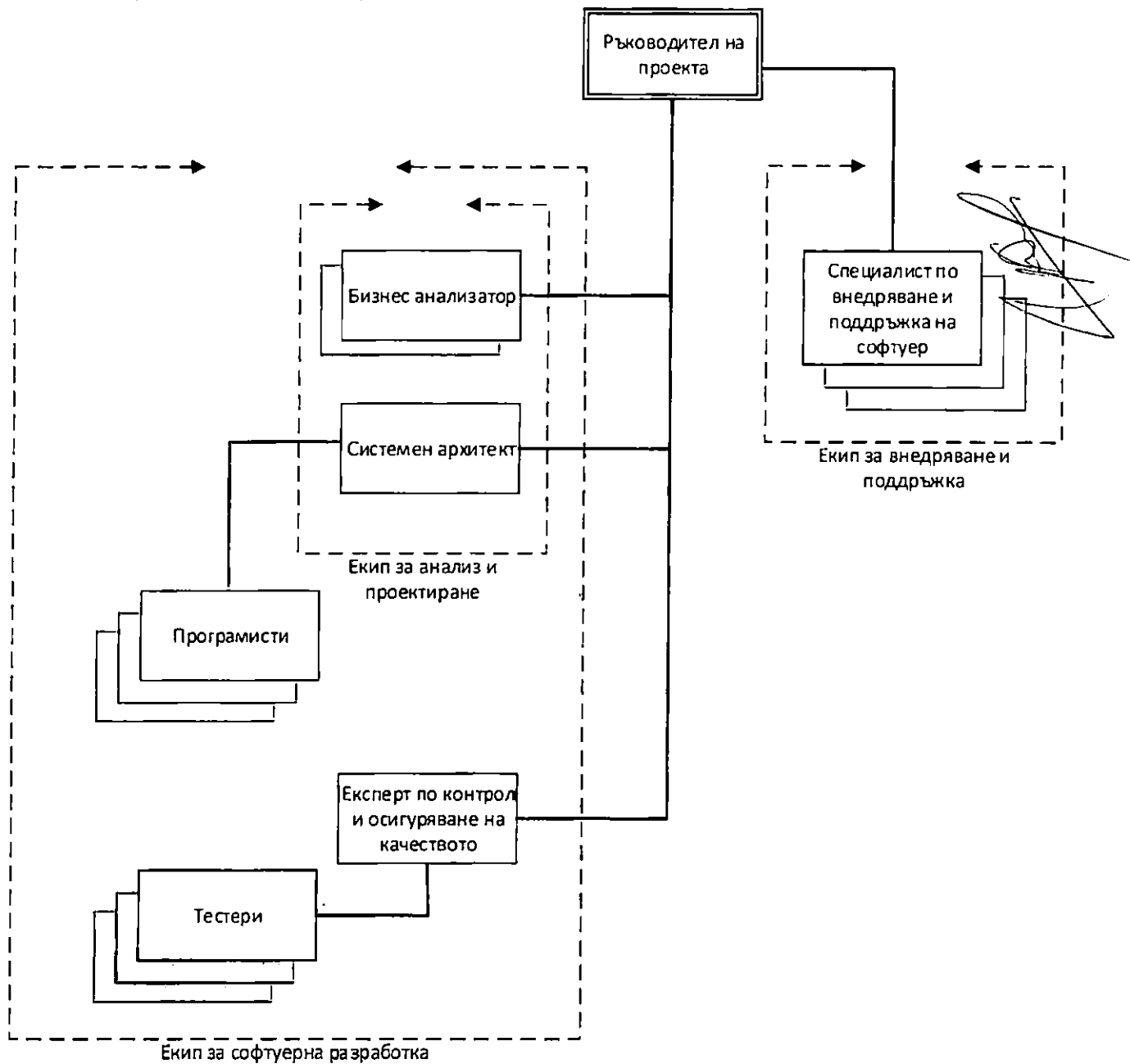


15

48



3.1.3. Организационна диаграма

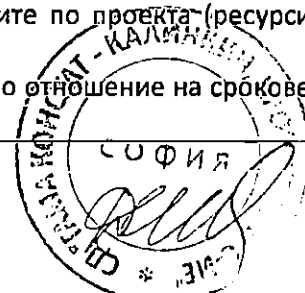


3.1.4. Роли и отговорности

Роля	Отговорности
Ръководител на проекта (Ключов експерт)	Отговаря за успешното изпълнение на проектите, които управлява; Отговаря за подготовката на изпълнението на проекта, включително: първоначална идентификация на рисковете; изготвяне на план за комуникация; идентификация и план за управление на заинтересовани лица; Отговаря за планиране и ре-планиране на дейностите по проекта (ресурси, план-график); Мониторинг и оценка на изпълнението на проекта (по отношение на срокове, бюджет и други дефинирани показатели);

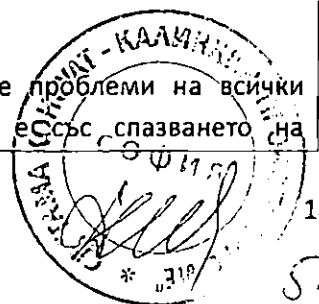
[Handwritten signature]

[Handwritten signature]

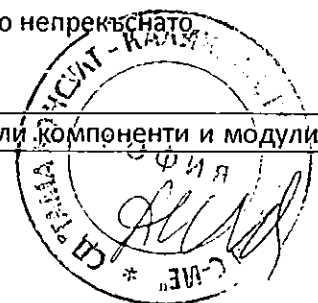


Роля	Отговорности
	<p>Контрол по изпълнението на проекта (иницииране или изпълнение на корективни действия);</p> <p>Отговорен е за управлението на рисковете свързани с работата по проекта (включително идентифициране, оценка, регистриране, предотвратяване или смекчаване на последствията при настъпване);</p> <p>Отговорен е за управлението на проблемите свързани с работата по проекта (включително идентифициране, оценка, регистриране и действия по разрешаване);</p> <p>Отговорен е за управление на промени и обхвата на проекта;</p> <p>Отговорен е за управление взаимоотношенията и комуникацията с всички заинтересовани лица (stakeholders), свързани с реализацията на проекта (напр. представители на клиента, партньори, под-изпълнители, ръководството на фирмата, екипа по изпълнение и т.н.);</p> <p>Отчитане на статуса (напредъка) по проекта - вътрешно (във фирмата) и външно (пред Възложителя), включително участие в съвет за управление на проекта;</p> <p>Участва в дейностите по одобрение на отчетената работа по проекта и месечното приключване;</p> <p>Отговаря по навременното изпълнение на всички административни дейности по обслужване на договора;</p> <p>Приключването на проекта – изготвянето на оценка от изпълнението на проекта, извличане и споделяне на научените уроци;</p> <p>Отговаря за дефинирането и изпълнението на индикатори за проектите, които управлява.</p>
<p>Бизнес анализатор (Ключов експерт)</p>	<p>Ролята на Бизнес системния анализатор се изразява най-общо в „Определяне, проучване, анализиране, систематизиране и документиране на дейностите, процесите и нуждите на клиента“. Той е посредникът между бизнес потребителите и техническия екип при разработката на софтуер.</p> <ul style="list-style-type: none"> • Работи съвместно с потребителите и екипа по разработка за определяне на оптималния за двете страни подход по отношение на изискванията, които са в рамките на проекта. Подпомага активно управлението на обхвата на проекта. • Документира резултатите от направените проучвания и анализи посредством изготвяне на функционални и други спецификации, както и изготвянето на прототипи на потребителския интерфейс. Актуализира изготвените спецификации през целия жизнен цикъл на проекта. • Представя резултатите от анализа пред потребителите и екипа по разработка, като подпомага двете страни в постигането на общо и пълно разбиране за бизнес изискванията. • Участва в процеса по промяна на изискванията, като обновява съответната документация и осигурява достигане на информацията за промяната до заинтересованите членове на екипа. • Подпомага екипа по поддръжка в изготвяне на потребителска документация и участва в провеждането на обучения за потребители. • Работи съвместно с екипа по осигуряване на качеството за осигуряване на съответствие на продукта с поставените изисквания, вкл. подпомага изготвянето на план за тестване и участва в тестове по приемане. • Контактна с екипа по осигуряване на качеството и екипа по разработка, като следи в хода на проекта за точното изпълнение на изискванията и проактивно взема мерки при установяване на несъответствия между

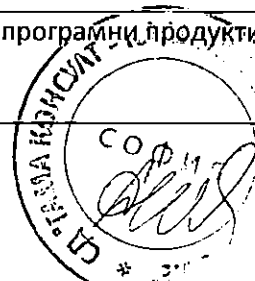
Роля	Отговорности
	<p>поставените изисквания и реализираните резултати.</p> <ul style="list-style-type: none"> Извършва оценка на резултатите от тестовете на базата на основните качествени показатели, нормативни изисквания, както и изискванията и очакванията на клиента.
<p>Системен архитект (Ключов експерт)</p>	<p><u>Проектни дейности (в ролята на ръководител на екип по проект):</u></p> <ul style="list-style-type: none"> Отговаря за навременното и коректно отчитане и одобрение на работата и дейностите, извършвани от хората в неговия екип; Отговаря за вътрешната организация и координация (работни процеси) на работата в екипа при изпълнението на проекти, както и за координирането на работата и резултатите на екипа с Ръководителя на проекта; Отговорен е за идентифицирането и предотвратяването на рисковете, свързани с работата на екипа. Ескалира рисковете към Ръководителя на проекта; Отговаря за дефинирането и изпълнението на цели за качество за екипа, който ръководи. <p><u>Технически дейности:</u></p> <ul style="list-style-type: none"> Верифициране и приемане на изискванията, предоставени от Бизнес анализатора; Проучва технологии и варианти за решения; Технически анализ на изискванията. Измисляне на технически решения на поставените за решаване проблеми и казуси в изискванията. Избор на решение и технически подход за реализация. Активно участие в процеса на оценка на работата по имплементиране и предоставяне на окончателната оценка на Ръководителя на проекта/екипа, преди да започне процеса по имплементация. Ако някои задачи не могат да бъдат оценени в разумно кратко време, дава оценка на времето, необходимо за изследване, анализ или прототипиране на решението, така че на следващ етап да може да бъде оценено. Отговаря впоследствие за изпълнението на задачите в срок. Участва активно във фазата на имплементация, като: <ul style="list-style-type: none"> Участва в разпределянето на задачите; Реализира част от задачите; Отговаря за решаването на техническите проблеми на всички програмисти в екипа. Пряко свързано със спазването на



Роля	Отговорности
	<p>сроковете за изпълнение на задачите на другите програмисти. Преценява до каква степен и как да включи други програмисти от екипа за решаването на техническите проблеми или да поиска външна помощ;</p> <ul style="list-style-type: none"> ○ Отговаря за координацията на работата м/у програмистите в екипа; ○ Отговаря за периодичното провеждане на прегледи (ревюта), включващи инспекция на кода и проверка на функционалността по отношение на това как са имплементирани изискванията, т.е. дали отговарят на базови технически критерии и функционални изисквания. Ревютата следва да се провеждат с целия екип, преди да се дадат задачите за тестване от тестера. В резултат от ревюто, задачите се приемат от Ръководителя за тестване или се връщат към програмистите за доработка; <ul style="list-style-type: none"> ● Подпомага Тестера да си изпълни ефективно своите задължения по тестване, като: <ul style="list-style-type: none"> ○ му помага да разбере как даденото изискване е реализирано или ○ като му предоставя скрипт или тестов код за генериране на данни или тестова функционалност, нужна за тестване на приложението; ● Отговаря за деплойването на приложението в тестова и продукционна среда; ● Отговаря за техническото обучение и навлизането на нови колеги в проекта.
<p>Експерт по Контрол и осигуряване на качеството (Ключов експерт)</p>	<ul style="list-style-type: none"> ● Отговаря за всички дейности по осигуряване и контрол на качеството, включително планиране на качеството, разработване на тестови сценарии и изпълнение на тестове; ● Разработва план за осигуряване на качеството в проекта, включващ мерки и процеси за осигуряване и контрол на качеството, които ще бъдат прилагани; ● Участва в извършването на контрол на качеството на разработваните програмни продукти чрез: <ul style="list-style-type: none"> ○ Разработва тестови сценарии; ○ Изпълнява тестови сценарии; ○ Разработва и прилага автоматизирани тестове; ○ Извършва вътрешни тестове на разработения софтуер, включително и автоматизирани тестове; ○ Координира работата на тестерите; ● Осигуряване на качеството на процесите и тяхното непрекъснато подобряване; ● Участва в изпълнението на приемни тестове.
<p>Програмист</p>	<p>Програмиране - създаване на софтуерни продукти или компоненти и модули</p>



Роля	Отговорности
	<p>от тях с приетите методи и средства; Спазване и прилагане на приетата технология за създаване на софтуер; Отговорност за работоспособността и ефективността на създадения софтуер; Първично тестване на модулите, върху които работи; Участва в код ревю.</p>
<p>Специалист по внедряване и поддръжка на софтуер</p>	<p>По отношение на внедряване:</p> <ul style="list-style-type: none"> • Разработване на план за внедряване; • Инсталиране и конфигуриране на базов софтуер; • Инсталиране и конфигуриране на специализиран разработен софтуер; • Параметризиране на софтуерни системи: настройки, работни процеси, шаблони за справки и др. • Инициализиране на софтуерни системи: въвеждане/зареждане на начални данни и номенклатури, настройване на права и роли и въвеждане на потребители и др. • Мигриране на данни от стари системи: анализ на съществуващите данни, планиране на миграцията, разработване на инструменти за мигриране на данните, зареждане и проверка на данните в новата система; • Провеждане на обучения на администратори, ключови потребители и потребители; • Подпомагане на потребителите при започване на работа със системата; • Следене и оптимизиране на системата при започване на експлоатация; • Планира провеждането на обучения за представители на Възложителя; • Подготвя провеждането на обучения – учебни материали, учебни казуси, тестове и т.н.; • Организира провеждането на обучения; • Участва в провеждането на обучения; • Администрира процеса по провеждане на обучения. <p>По отношение на поддръжката:</p> <ul style="list-style-type: none"> • Осъществява техническа поддръжка на приложен софтуер; • Осъществява техническа поддръжка на базов софтуер; • Подпомага работата на потребителите; • Регистрира работата по поддръжката в специализираните инструменти, включително прави описания на разрешените проблеми; • Следи за изпълнението на споразуменията за ниво на поддръжка за системите, за които отговаря; • Осъществява мониторинг на системите, които поддържа; • Създаване и изпълнение на процедури за резервиране на данни и системи; • Приложно и системно администриране.
<p>Тестер</p>	<p>Осъществява контрол на качеството на разработваните програмни продукти чрез:</p> <ul style="list-style-type: none"> • Разработва тестови сценарии;

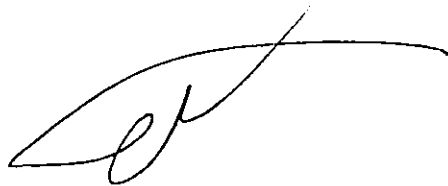


Роля	Отговорности
	<ul style="list-style-type: none"> Изпълнява тестови сценарии; Разработва и прилага автоматизирани тестове; Извършва вътрешни тестове на разработения софтуер, включително и автоматизирани тестове; Участва в приемните тестове; Документира резултатите от изпълнените тестове.

3.1.5. Участие на експертите в дейностите по проекта

№	Етап/Дейност	Подетап	Участващи експерти
1	Дейност 1: Анализ, проектиране и разработка на адаптера		
1.1	Етап 1: Бизнес анализ и Проектиране	Анализ на данните и изискванията	Бизнес анализатор Системен архитект
		Изготвяне на Спецификация на софтуерните изисквания	Бизнес анализатор Системен архитект Експерт по Контрол и осигуряване на качеството
1.2	Етап 2: Изграждане на Адаптора	Разработване на софтуерното решение	Бизнес анализатор Системен архитект Експерт по Контрол и осигуряване на качеството Програмист Специалист по внедряване и поддръжка на софтуер Тестер
		Тестване	Експерт по Контрол и осигуряване на качеството Програмист Тестер
2	Дейност 2: Тестване и съпровождане на адаптора		
2.1	Етап 3: Верификация на EUCISE2020 адаптор и съпровождане по време на тестовата обмяна на данни	Обучение	Бизнес анализатор Специалист по внедряване и поддръжка на софтуер
		Внедряване и съпровождане	Специалист по внедряване и поддръжка на софтуер Програмист Тестер

Ръководителят на проекта има участва във всички дейности, етапи и задачи в проекта



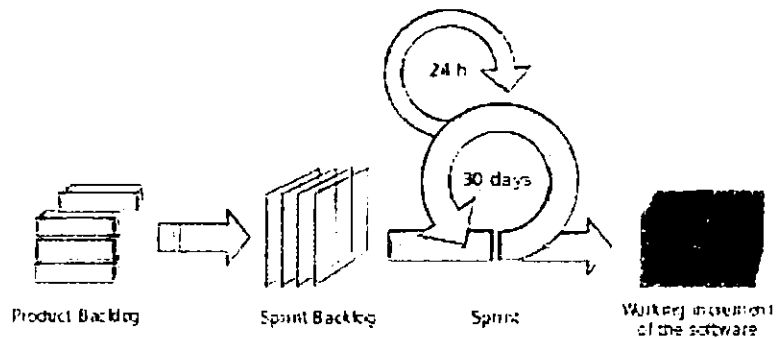


21

54

3.1.6. Координация в рамките на итерациите

Координацията на екипа ще бъде реализирана чрез прилагането на Scrum. Scrum е итеративна, инкрементална рамка за управление на проекти.



Представители на Възложителя могат да участват в планиращите и заключителните срещи на всеки спринт.

Подробно описание на подхода за екипна координация е представено по-долу в т. 5.3.3.

3.2. Управление на комуникацията

3.2.1. Проектна комуникация

Управлението на комуникацията в проекта включва процесите, необходими за осигуряване на навременно и точно генериране, събиране, разпространение, съхранение и структуриране на проектната информация. Проектните ръководители прекарват значителна част от времето си в общуване с членовете на екипа и заинтересованите лица – външни или вътрешни за организацията.

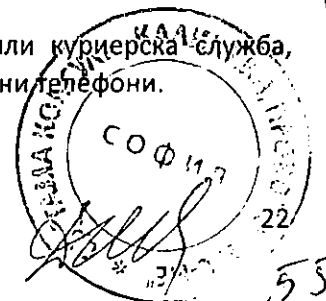
Ефективната съвместна работа на проектната организация налага въвеждането и съблюдаването на набор от формални правила за комуникация. Тези правила обикновено приемат формата на т.нар. *План за комуникация*, който зависи от големината, структурата, капацитета и зрелостта на проектната организация. Затова, правилата за комуникация в проекта следва да бъдат съгласувани между двете страни в стартовата фаза на проекта, едновременно със създаването на проектната организация. Нашият дългогодишен опит в реализацията на ИТ проекти с разнообразна сложност, големина и продължителност, показва, че като минимум, планът за комуникация трябва да включват всички аспекти на формалната комуникация: правила, канали и инструменти за обмен и контрол на информацията, подготовка и управление на документацията, организиране и провеждане на срещи. В духа на добрите практики при управлението на проекти, ние предлагаме спазването на посочените по-долу минимални препоръчителни правила.

3.2.2. Правила и канали за комуникация

За да бъде гарантирано успешното изпълнение на проекта при ефективно сътрудничество между всички членове на проектната организация, следните правила за формална комуникация е препоръчително да бъдат стриктно съблюдавани:

Определяне на език на комуникация: Езикът за комуникация в проекта, между екипите на Възложителя и Изпълнителя, е български.

Дефиниране на канали за комуникация: Например, традиционна поща и/или куриерска служба, избрана и одобрена от двете страни, електронна поща, мобилни и/или фиксирани телефони.



При комуникация чрез традиционна поща или предаване на документация на ръка в деловодство, всеки документ следва да носи уникален изходящ номер и дата на изпращача и уникален входящ номер и дата на получателя;

При комуникация чрез електронна поща всяко съобщение следва да съдържа: в полето „До: (To):“ електронния адрес на всеки получател, от когото се очаква отговор; в полето „Копие до: (cc):“ електронния адрес на всеки получател, за сведение на когото е съобщението; в полето „Относно (Subject):“: темата на съобщението, на български език; в полето „Съдържание (Body):“ самото съобщение, както и името и контактите на изпращачия го за улесняване на обратната връзка. Препоръчва се активирането на функцията „Връщане на потвърждение за прочитането (Read Receipt)“. Отговор трябва да бъде изпратен в рамките на 2 работни дни, а ако това не е възможно трябва да бъде отговорено със срока, в който се очаква да има отговор. При последваща комуникация по същата тема следва да се отговаря чрез даденото съобщение чрез функцията „Отговори на всички (Reply-to-all)“. В допълнение, страните могат да съгласуват и позволения формат и кодиране на електронните съобщения, например HTML или Plain Text и Cyrillic KOI8-R. В зависимост от зрелостта, капацитета и нуждите на проектната организация могат да бъдат одобрени и правила за комуникация чрез електронно подписани с удостоверения за универсален електронен подпис (УЕП) съобщения.

При комуникация чрез мобилни и фиксирани телефони е препоръчително първо да се използват мобилните номера (след съгласуване), като при пропуснато повикване търсеното лице има ангажимент да върне обаждането в рамките на работния ден. Добра практика е да се активира Call Waiting функцията чрез мобилните апарати.

Дефиниране на правила за работа с документи, например:

Работни документи се предават на адреса на електронната поща на съответните получатели, включително до Ръководителя(ите) на екипа(ите), на които получателите са членове.

Официални документи се предават в деловодството на Възложителя, съответно на рецепцията на адреса за кореспонденция на Изпълнителя, чрез традиционна поща и куриерска служба, избрана и одобрена от двете страни, или на ръка от представител на съответната страна-подател.

При поискване на коя да е от страните може да бъде осигурен работен документ в хартиен вид (не по ел. поща). Времето за отговор по работен документ е 2 работни дни. Времето за експертен отговор на официален документ, с изключение на работните документи, които се предават в писмен вид, е 5 работни дни. Изпратен документ след 17 часа се води за изпратен на следващия работен ден.

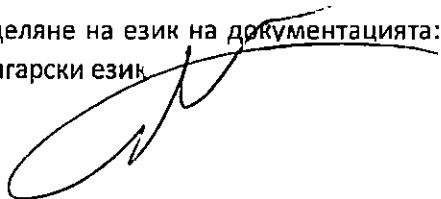
Всички официални отговори на официални документи, както и искания за промени в изискванията трябва да стават в писмен вид – хартиени документи изпратени чрез традиционна поща или електронни документи подписани с Универсален електронен подпис и изпратени чрез електронна поща.

Всички отговори на работни документи, както и искания за срещи трябва да стават чрез електронна поща.

3.2.3. Управление на документацията и инструменти за комуникация

В допълнение към предложеното, страните е препоръчително да съгласуват и правила за подготовка и управление на документацията, които в процеса на изпълнение на проекта да бъдат стриктно съблюдавани:

Определяне на език на документацията: Работните и официални документи се съставят и поддържат на български език



В отделни случаи, например при използването на специфична техническа литература, Ръководителите на проекта могат да се договорят конкретните документи да бъдат предоставени и само на английски език или друг работен език.

Поддържането на даден документ на два езика едновременно е по допълнителна предварителна договорка.

Определяне на лицата, които работят с даден документ, например:

За всеки документ се указва неговият собственик. Единствено собственикът на документа има право да съгласува и одобрява промени в документа.

Списъкът с получателите на даден документ се определя от собственика му, вписва се в документа и се довежда до знанието на Ръководителите на проекта.

За всеки документ се указват: името на проекта, името на документа, текущата версия и дата, история на промени, реквизити на съставилия и одобрилия го, съдържание, номерация на страниците и общ брой страници.

Добра практика е да бъде поддържан регистър на документите, в който се описват и актуализират всички документи със своите атрибути, като: наименование на документа, наименование и адрес на файла на документа в електронен вид, собственик на документа, версия и статус на документа.

Инструментите, използвани за целите на този проект, следва да бъдат съгласувани и одобрени в стартовата фаза на проекта. Тук сме предложили препоръчителен набор от популярни инструменти, които могат да бъдат заменени с други версии или с техни еквиваленти, според наличността на лицензи при Възложителя, удобството и компетентността на участниците в проекта:

Стандартен уеб-браузър, например Internet Explorer, версия 9.0 или по-висока;

Microsoft Office Outlook: ще бъде използван за обмяна на електронни съобщения, изпращане и получаване на покани за срещи, контакти и т.н.

Microsoft Office Project: ще бъде използван за проследяване на прогреса на проекта във времето чрез визуализиране на Проектния план.

Microsoft Office Word: ще бъде използван за обмяна на работните версии на всички документи по проекта (във формат .doc);

Microsoft Office Excel: ще бъде използван за обмяна и редакция на електронни таблици (във формат .xls);

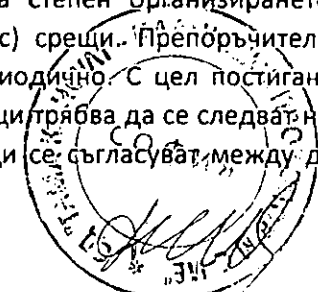
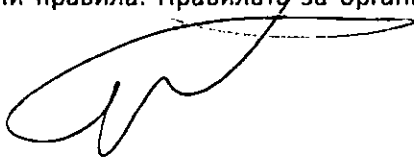
Microsoft Office PowerPoint: ще бъде използван за обмяна на презентации;

Microsoft Office Visio: ще бъде използван за обмяна и редакция на схеми, диаграми и т.н.

Microsoft SharePoint Portal: ще бъде използван по време на проекта за съхраняване на документи и колективна работа.

3.2.4. Провеждане на срещи

Постигането на целите и резултатите на проекта налага в различна степен организирането на технически (работни), управленски (проектни) и стратегически (бизнес) срещи. Препоръчително е срещи на участниците в проекта на всички нива да се провеждат периодично. С цел постигане на резултатност и ефективност при организирането и провеждането на срещи трябва да се следват някои формални правила. Правилата за организиране и провеждане на срещи се съгласуват между двете



страни в стартовата фаза на проекта. Препоръчително е спазването на следните минимални изисквания:

Определяне на необходимия срок на предизвестие при свикване на среща, например: Срещите за следващата седмица се планират не по-късно от 16 часа в четвъртък на настоящата седмица или срещите се организират с предизвестие не по-малко от 24 часа.

Предварително обявяване на програмата: Темите на всяка среща се дефинират и обявяват предварително от организатора на срещата, като се включва и преглед на темите и протокола от предишната среща.

Предварително осигуряване на документация: Всички документи, например протоколи, техническа или проектна документация, които са нужни за срещата, се предоставят не по-късно от обяд на деня предхождащ деня на срещата.

Подготовка на материали: Всички материали, необходими по време на срещата, например проектор и/или флипчарт, се осигуряват от домакина на срещата.

Определяне на език: Работният език на срещите е български.

Протоколиране на срещите: Организаторът на срещата има отговорността за нейното протоколиране (в случаите когато това не е описано в ТЗ), като той може да делегира отговорността за изготвяне на протокола чрез съгласуване с участниците и анонсиране на протоколчика в началото на всяка една среща.

Определяне на срок за изготвяне и разпращане на протокол от среща: Протоколи от срещите се подготвят и изпращат на всички участници не по-късно от два работни дни след приключване на срещата по предварително съгласуваните канали за комуникация (виж по-горе).

Съгласуване на протокол от среща: Ако до два работни дни след получаване на протокола от срещата участниците не уведомят организатора на срещата в писмен вид (в съответствие с одобрените канали за комуникация) за своите забележки към него, протоколът се счита за одобрен.

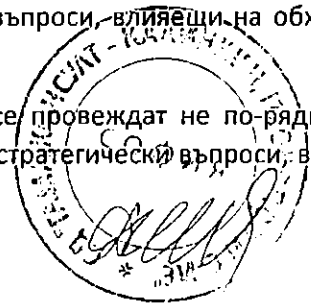
Аудио протоколи: Организаторът на срещата има правото да поиска използването на аудио протоколиране, съгласувано с участниците в срещата. В случай на съгласие, организаторът има отговорност да осигури необходимите технически средства, както и да разпространи протокола сред участниците в предварително уговорен формат (виж също инструменти за комуникация по-горе) и при спазването на всички останали правила за комуникация. Следвайки добрите практики, препоръчително е в случаите на аудио протоколиране да бъде изготвено резюме в писмен вид.

Свикване на срещи по спешност: Срещи извън посочения график могат да бъдат организирани при необходимост. Заявка за такава среща трябва да бъде изпратена от организатора ѝ, като се препоръчва толеранс от поне един ден за да се даде време на всички участници да се подготвят.

Препоръчително е срещи на Техническите ръководители или Ръководителите на екипи от двете страни да се провеждат ежеседмично. Основната цел на тези срещи е да се дискутират и съгласуват оперативни въпроси, които изискват координирани експертни действия на страните.

Препоръчително е срещи на Ръководителите на проекта от двете страни да се провеждат не по-рядко от веднъж месечно. Основната цел на тези срещи е да се дискутират въпроси, влияещи на обхвата, сроковете и ресурсите на проекта.

Препоръчително е срещи на Комитета за управление на проекта да се провеждат не по-рядко от веднъж на два месеца. Основната цел на тези срещи е да се дискутират стратегически въпроси, важни



за успешното приключване проекта, като съществени промени в целите и очакваните резултати от проекта, управление на рисковете и цялостен контрол на текущото състояние на изпълнение на проекта.

3.2.5. Канали за комуникация

Тип на комуникацията	Комуникационен канал	Страни в комуникацията
Официална (формална)	Писма, входящи в деловодството на организацията	Между проектните ръководители; Между ръководителите на организацията.
	Факс	
	Работни срещи, протоколирани с протокол	
Оперативна	Електронна поща	Между членовете от екипа, отговорни за темата на комуникацията.
	Телефон и др. средства за провеждане на разговори (Lync, Skype и т.н.)	
	Факс	
	Работни срещи	
	Портал за поддръжка	

3.2.6. Докладване

В хода на изпълнение на договора Изпълнителят ще изготви и предостави на Възложителя:

- Встъпителен доклад - Встъпителният доклад ще бъде предоставен в двуседмичен срок от подписването на договора и да съдържа описание минимум на:
 - Подробен работен план и актуализиран времеви график за периода на проекта;
 - Начини на комуникация;
 - Отговорни лица и екипи.

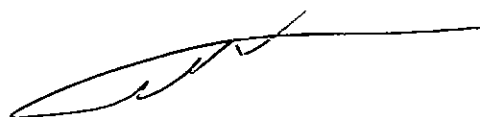
Встъпителният доклад следва да бъде одобрен от Възложителя.

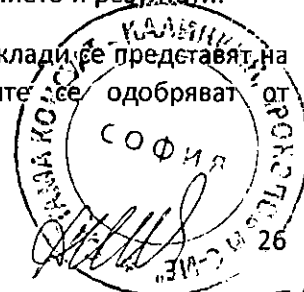
- Междинен доклад - Междинните доклади ще бъдат представяни и да се предават при приключване на всяка от дейностите и поддейностите и/или при настъпване на събитие. Междинните доклади ще съдържат информация относно изпълнението на дейностите и поддейностите по предварително изготвения проектен план. Като минимум междинните доклади ще съдържат следната информация:
 - Общ прогрес по дейностите през периода;
 - Постигнати проектни резултати за периода;
 - Срещнати проблеми, причини и мерки, предприети за преодоляването им;
 - Рискове за изпълнение на свързани дейности и на проекта като цяло и предприети мерки;
 - Актуализиран план за изпълнение, ако има такъв.

Всеки междинен доклад следва да бъде одобрен от Възложителя.

- Окончателен доклад - В края на периода за изпълнение ще бъде предоставен окончателен доклад. Окончателният доклад трябва да съдържа описание на изпълнението и резултати.

Докладите ще се изпращат до отговорния служител на Възложителя. Всички доклади се представят на български език в електронен формат и на хартиен носител. Докладите се одобряват от отговорния/отговорните служител/служители в срок до 5 работни дни.





Всички доклади ще бъдат представени на Възложителя на български език на хартиен и на електронен носител. Представянето на докладите трябва да се извършва чрез подписване на двустранни предавателно-приемателни протоколи, подписани от представители на Изпълнителя и на Възложителя.

Примерно съдържание на междинни доклади:

1. Въведение			
2. Общ преглед на състоянието на проекта			
Общ статус:		[Цвят]:	
[Резюме]			
3. Прогрес по конкретните дейности			
3.1. Дейност X			
3.1.1. Общ прогрес по дейността			
3.1.2. Постигнати проектни резултати			
3.1.4. Планирани дейности в следващия период			
4. Срещнати проблеми и отворени въпроси			
№	Описание	Предложение за решение	Решение/ коментар
5. Рискове			
Описват се във формата на регистър на рисковете, представен в плана за управление на риска.			
6. Препоръки за текущото състояние на проекта			
7. Актуализиран план за изпълнение на проекта			

3.3. Управление на качеството

3.3.1. Стратегия за качество в проекта

Стратегията за осигуряване на качеството определя дейностите, които ще осигурят, че контролът на качеството се прилага ефективно и механизмите за контрол на качеството са адекватно планирани, графици са спазвани, преглеждани и актуализирани.

Осигуряване на качеството (QA) се отнася до систематично измерване, сравнение със стандарти, мониторинг на процесите и свързаните с тях дейности за осъществяване на обратна връзка, която спомага за предотвратяването на грешки. Това контрастира с процеса по контрол на качеството, който се фокусира върху изхода на процесите.

Осигуряването на качеството се основава на два базови принципа:

- 1) продукта трябва да може да изпълнява предназначението си;
- 2) откритите грешки трябва да се отстраняват възможно най-бързо.

Осигуряването на качеството включва управление на качеството на всички базови продукти, компоненти на системата и процеси свързани с изграждането, включително тези на управление и мониторинг и контрол.



Критериите за определяне на качеството се определя от заинтересованите лица и основно потребителите на системата. Те са свързани и с разходите за прилагане на необходимите изисквания, които зависят както от наличните бюджети, потребителите, така и от самия продукт, тъй като определено ниво на качеството е свойствено за точно определени категории продукти.

Качеството на продукта може да бъде разглеждано от два аспекта:

- 1) Какви характеристики притежава;
- 2) До каква степен тези характеристики са реализирани.

Притежаваните характеристики се задават при изготвянето на функционалната спецификация на проекта. Притежаваните характеристики са ограничени от обхвата на договора, от разполагаемото време и от наличните ресурси. Предмет на тестването е да установи до каква степен тези характеристики са осъществени. Поради това то се изпълнява след завършване на разработката. При процес на бърза разработка, то се изпълнява паралелно с разработката, в края на всяка итерация, но е възможно да се окаже, че изисква значително повече ресурси спрямо посветеното тестване, в края на проекта. При този похват обаче се спестява време, тъй като голяма част от дейността се изпълнява паралелно на разработката. Частта от дейността, която не може да се изпълни паралелно е осъществяването на приемните тестове. Приемните тестове се осъществяват на базата на предварително изготвени и съгласувани с Възложителя приемни сценарии. Те имат за цел да покажат, че системата може да бъде въведена в експлоатация.

Като част от проектния план ще бъде разработен План за осигуряване на качеството. След като Планът за осигуряване на качеството бъде разработен и приет от Възложителя, той ще бъде водещият документ за организиране на процесите по управление на качеството в проекта.

Всеки етап в жизнения цикъл на проекта ще се отрази като артефакт, даващ основа за следващия етап. Чрез контрола на качеството тези артефакти ще бъдат проверявани преди да са използвани в следваща фаза, което ще гарантира верността ѝ. Те също ще бъдат верифицирани съобразно техническата спецификация.

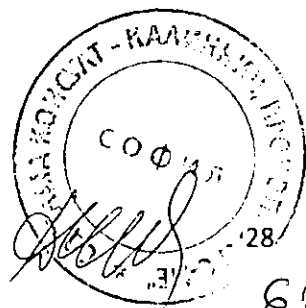
Всеки документ и материал, който ще бъде доставян на Възложителя или други външни лица подлежи на контрол на качеството.

За всеки един разработван компонент Изпълнителят ще покрие следните изисквания за гарантиране на качеството на извършваната разработка и на крайния продукт:

- Документиране на Адаптора в изходния код, минимум на ниво процедура/функция/клас;
- Покритие на минимум 50% от изходния код с функционални тестове [в случай на надграждане на съществуваща система – 50% от новата функционалност и 20% от съществуващата];
- Използване на continuous integration практики;
- Използване на dependency management.

3.3.2. Процедури за контрол на качеството

Съществуват няколко метода, които ще бъдат използвани при осигуряването на проекта и материалите по него в съответствие с подходящите стандарти за качество:



- Вътрешни прегледи на проекта - Това са работни сесии на екипа по проекта, в които екипът преглежда всички материали по конкретна фаза преди да определи методологията за официален преглед. Прегледът се извършва от ръководителя на проекта и ръководителя на разработката.
- Инспекции - Това е съвместен преглед на материали от представители на Възложителя и изпълняващия екип за целите на проверка и приемане.

3.3.3. Роли и отговорности в процесите по осигуряване и контрол на качеството
Ръководителят на екипа ще бъде отговорен за:

- Определяне на стратегия за осигуряване на качеството;
- Установяване на процедури за осигуряване на качеството;
- Определяне на задачи и отговорности на членовете на екипа свързани с изпълнението на процедурите по осигуряване на качеството;
- Наблюдение на процесите по осигуряване на качеството и извършване на коригиращи действия, при необходимост.

Експерт по контрол на качеството:

- Участва в определянето на стратегия и процедури за осигуряване на качеството;
- Организира изпълнението на процедурите и стратегията за осигуряване на качеството;
- Разработва подробен тестов план.

Всеки ръководител на технически екип:

- Ще бъде отговорен за спазването на установените в проекта стандарти и процедури за осигуряване на качеството в екипа, който ръководи;
- Ще сигнализира на Ръководителя на проекта за потенциални слабости в качеството на разработвания в екипа и другите екипи продукт и ще дава мотивирани предложения за подобряване на стандартите за качество.

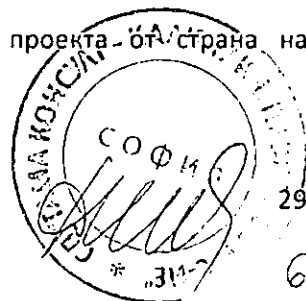
Всеки член на екипа:

- Ще бъде отговорен за прилагането на стандартите за качество на проекта при изпълнението на своята работа;
- Ще изпълнява възложените му задачи за осигуряване на качеството;
- Ще сигнализира за дефекти и грешки, намерени при изпълнение на собствените си задачи, включително когато не са пряко свързани с текущата задача.

Представители на клиента ще имат възможност да вземат участие в процесите за осигуряване на качеството.

Ръководителят на проекта ще бъде отговорен за общото качество на проекта - от страна на ИЗПЪЛНИТЕЛЯТ.

Ръководител на проекта ще бъде отговорен за:



- Определяне на стратегия за осигуряване на качеството;
- Установяване на процедури за осигуряване на качеството;
- Определяне на задачи и отговорности на членовете на екипа свързани с изпълнението на процедурите по осигуряване на качеството;
- Наблюдение на процедурите по осигуряване на качеството и извършване на коригиращи действия, при необходимост.

Експерт по контрол на качеството:

- Участва в определянето на стратегия и процедури за осигуряване на качеството;
- Организира изпълнението на процедурите и стратегията за осигуряване на качеството;
- Разработва подробен тестов план.

Всеки ръководител на технически екип:

- Ще бъде отговорен за спазването на установените в проекта стандарти и процедури за осигуряване на качеството в екипа, който ръководи;
- Ще сигнализира на Ръководителя на проекта за потенциални слабости в качеството на разработвания в екипа и другите екипи продукт и ще дава мотивирани предложения за подобряване на стандартите за качество.

Всеки член на екипа:

- Ще бъде отговорен за прилагането на стандартите за качество на проекта при изпълнението на своята работа;
- Ще изпълнява възложените му задачи за осигуряване на качеството;
- Ще сигнализира за дефекти и грешки, намерени при изпълнение на собствените си задачи, включително когато не са пряко свързани с текущата задача.

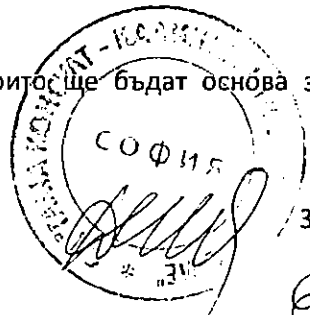
Представители на клиента ще имат възможност да вземат участие в процесите за осигуряване на качеството.

3.4. Механизми за проследимост и отчетност на изпълнението

Наблюдението и контролът на работата по проекта е процесът на проследяване, преглед и регулиране на напредъка с цел покриването на целите за производителност, дефинирани в плана за управление на проекта. Този процес е част от групата за управление на интеграцията в проекта.

Предлагаме механизми за проследимост и отчетност на изпълнението, които да осигурят ефективно управление на проекта. Ще бъде проследявана степента на постигане на целите при съпоставяне на действителните и очакваните резултати от изпълнението.

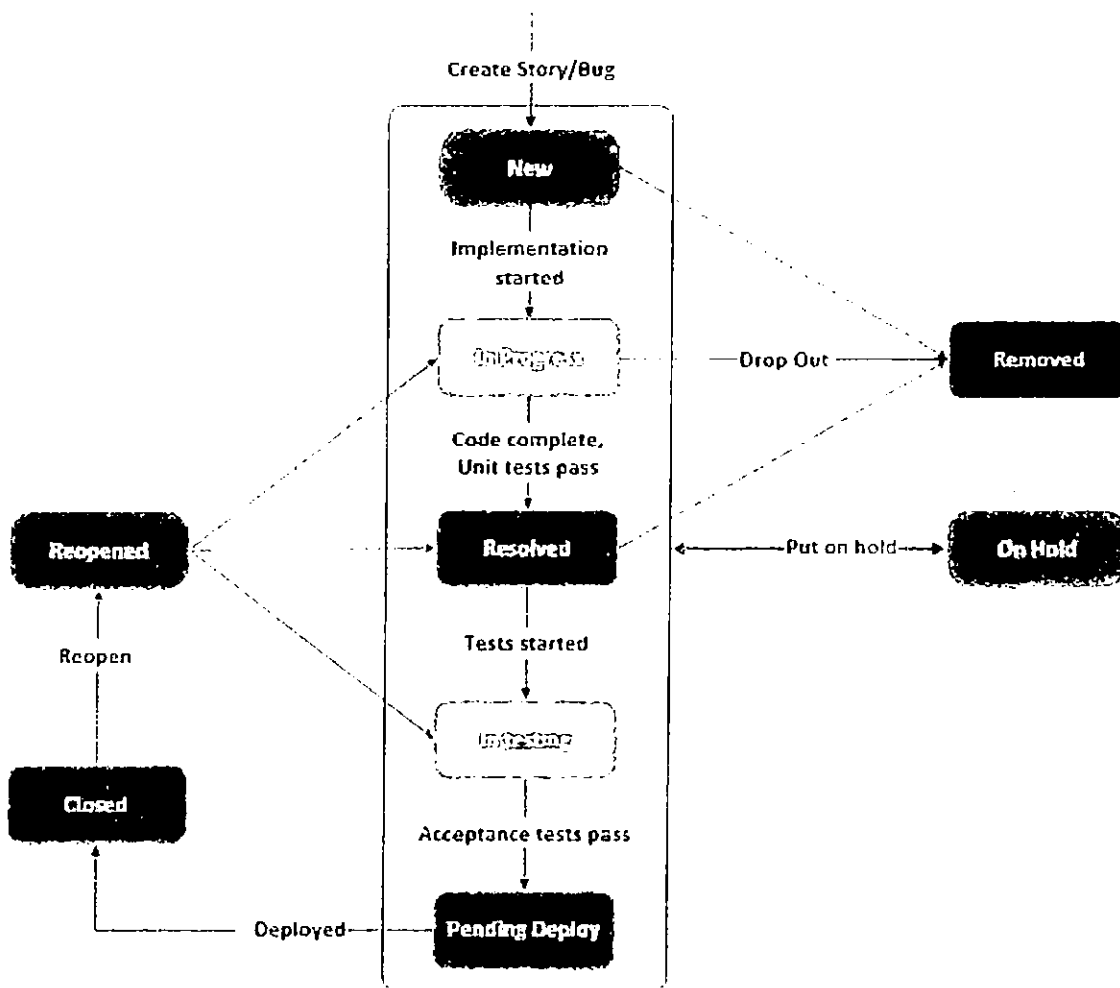
Във фазата на планиране ще бъдат разработени следните артефакти, които ще бъдат основа за бъдещия мониторинг:



- Детайлен план за изпълнение;
- Ключови индикатори на начини за измерването им;
- Бюджет на проекта.

Най-значителният ресурс, който ще бъде вложен в изпълнението на поръчката е времето на проектните екипи. Тяхната работа ще бъде планирана, разпределяна и регистрирана чрез Atlassian Jira – инструментална среда за управление на съвместната работа на софтуерни екипи.

За управлението на задачите (work items) ще се използват следния процес и статуси:



Handwritten signature

В инструменталната среда ще се въвежда и съхраняват данни за планирано и реално изработено време за всеки индивид в проектния екип.

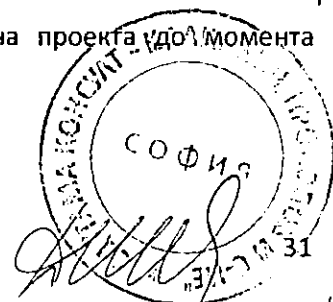
За анализ на разходваните средства и оценка на постигнатите резултати предлагаме да се използва методът на спечелената стойност (Earned value management / EVM). EVM е техника от дисциплината управление на проекта, която служи за измерване на напредъка на изпълнението и прогреса по обективен начин. Тази техника комбинира в едно измерването на напредъка относно обхвата, графика и разходите в една интегрирана система за оценка. Методът дава възможност за изготвяне на акуратна оценка на проблемите с производителността на проекта.

Спечелената стойност се дефинира като обема свършена от началото на проекта до момента запланувана работа или:

$$EV = \sum_{Start}^{Current} PV(Completed)$$

Handwritten signature

Handwritten signature



Цели: Анализът на спечелената стойност е най-често използваният начин за измерване на напредъка. Използва се за ревизия на планираните график и бюджет за оставащата част от проекта и/или за да се идентифицира нуждата от промяна на обхвата на проекта.

Начин на измерване: За да бъде осъществена EVM калкулацията е необходимо наличието на следните предусловия:

- Необходим е пълен списък със задачи;
- Необходим е списък на ресурсите с техните часови ставки;
- Необходим е списък на всички други разходи;
- Необходим е одобрен план-график;
- Необходим е текущ статус на проекта;
- Необходима е дата на изготвяне на доклада.

При измерването се използват следните индикатори:

- PV или често използвана като BCWS – Planged value/Планирана стойност, и е еквивалентна на Budgeted Cost of Work Scheduled/Бюджетна стойност на планираната работа;
- EV или често използвана като BCWP – Earn value/Спечелена стойност, и е еквивалента на Budgeted Cost of Work Planned/Бюджетна стойност на планираната работа;
- AC или често използвана като ACWP – Actual Cost/Реални разходи, и е еквивалентна на Actual Cost of Work Planned/Реални разходи за извършване на планираната работа;
- SV – Schedule variance/Разминаване в графика;
- CV – Cost variance/Разминаване в бюджета;
- EAC – Estimate at completion/Прогноза към датата на приключване;
- BAC – Baseline at completion/Планирана стойност към датата на приключване;
- ETC – Estimate to complete/Прогнозата за завършване на оставащата работа по проекта.

Използват се методи за оценка базирани на Изменението в графика и Изменението в разходите. Това става по следните формули:

- Изменение спрямо графика - Метода служи за управление на графика. Той не включва в себе си акумулиране на реалната стойност (AC), а се базира на оценка на прогреса спрямо времевия план. Метода се изпълнява самостоятелно в типичните малки и средни по големина проекти.

$$SV = EV - PV$$

Оценката става на следния принцип: SV по-големи от 0 показват, че проекта изпреварва графика. Разминаването в графика ще бъде 0 в края на проекта, тъй като всички планирани стойности са били „спечелени“.

- Използва се и така наречения Schedule Performance Index (Индекс за производителността спрямо графика). Той се изчислява съобразно формулата:

$$SPI = \frac{EV}{PV}$$

SPI > 1 се смята за много добър, тъй като тогава проекта изпреварва графика.

- Измерване спрямо разходите – Метода служи за управление на разходите. Измерва се общата планирана стойност (BCWS) на края на проекта. Ако в бюджета има резерви за управление те типично не се включват в оценката. Прилагат се следните формули:

$$CV = EV - AC$$

CV > 0 се считат за добри, тъй като проекта икономисва средства.

- Използва се и така наречения Cost Performance Index (Индекс за производителността спрямо бюджета). Той се изчислява съобразно формулата:



$$CPI = \frac{EV}{AC}$$

CPI > 1 са добри, тъй като проекта икономисва средства;

CPI = 1 означава, че проекта изразходва средства съобразно плана;

CPI < 1 означава, че проекта изразходва средства над планираните.

Твърде високи стойности на CPI (в някои случаи и 1.2 е високо) може да означава, че плана е твърде консервативен и производителността се измерва спрямо лош първоначален план.

Начин на отчитане и проследимост на извършената работа: Изготвянето на доклад за статуса на проекта посредством EVM е напълно автоматизирано с използването на предложения от нас софтуер за управление на проекта, а именно Microsoft Project 2010. Продукта поддържа и възможност за избор между това дали калкулацията на Индекса на бюджетните разходи на извършената работа (BCWP) да е базирана на % Complete или на Physical % Complete. Те кореспондират на напредъка спрямо графика и напредъка спрямо извършената работа. Всички от гореизброените стойности могат да бъдат автоматично изчислени на база на въведената информация. Възложителя ще има достъп до системата, а също така ще получава извадка за състоянието с регулярните отчети по договора.



Предлагаме да бъдат приложени следните видове мерки за контрол, относими към цялостния процес на изпълнение на проектните дейности, които ще прилага с цел качествено и навременно изпълнение на поръчката:

- Контрол на изпълнението на графика – чрез сравнение между базовия график и текущия напредък. Ще се използват Schedule Performance Index (SPI) и Schedule Variance (SV).
- Контрол на изразходване на бюджета – чрез сравнение между базовия бюджет и реално вложените разходи и време. Ще се използват Cost Performance Index (CPI) и Cost Variance (CV).
- Контрол на качеството, съгласно описаното в предложението.

4. Етапи, срок и график за изпълнение

4.1. Срок за изпълнение на поръчката

Периодът на изпълнение е 10 месеца считано от сключването на договора.

4.2. Етапи, срокове и резултати

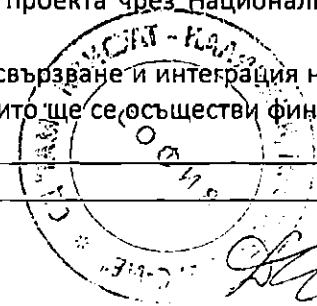
No	Етап/Дейност	Срок	Резултати
1	Дейност 1: Анализ, проектиране и разработка на адаптера		
1.1	Етап 1: Бизнес анализ и Проектиране	1 месец след подписване на договора	<ul style="list-style-type: none"> • Спецификации и доклади • Спецификация на софтуерните изисквания за национален неклассифициран адаптор за EUCISE2020 • Софтуер за национален адаптор за EUCISE2020 (архитектурен прототип) • Програмен код на софтуер за национален адаптор за EUCISE2020 (архитектурен прототип)
1.2	Етап 2: Изграждане на Адаптора	4 месеца след подписване на договора	<ul style="list-style-type: none"> • Препоръки за необходими промени и доработки в Националната система за управление на корабния трафик (VTMS) във връзка с адаптиране към изискванията на EUCISE2020 за обмен на неклассифицирани данни и използване на услуги • Софтуер за национален адаптор за EUCISE2020 • Програмен код на софтуер за национален адаптор за EUCISE2020 • План за тестване • Тестови сценарии • Спецификация на реализацията • Ръководство на потребителя • Ръководство на администратора • Ръководство за интеграция с националната система за управление на корабния трафик (VTMS) • Проект за конфигуриране и разполагане на компонентите на адаптора върху ИКТ инфраструктурата, която ще бъде осигурена • Проект за мрежова свързаност на ИКТ инфраструктурата на адаптора • ИКТ инфраструктура за периода на тестване и съпровождане на адаптора • Интернет-свързаност за работа на адаптора за периода на тестване и съпровождане • Протоколи за извършени инженерингови услуги за свързване и интеграция на националната система (VTMIS) през Демонстрационната фаза на проекта чрез Националния EUCISE2020 адаптор • Протоколи за извършени инженерингови услуги за свързване и интеграция на Националния EUCISE2020 адаптор с EUCISE2020 нодовете, чрез които ще се осъществи финалната връзка с EUCISE2020 мрежата
2	Дейност 2: Тестване и съпровождане на адаптера		

Handwritten signature

67

Handwritten signature

Handwritten signature



Handwritten signature

2.1	Етап 3: Верификация на EUCISE2020 адаптор и съпровождане по време на тестовата обмяна на данни	10 месеца след подписване на договора	<ul style="list-style-type: none"> • Протоколи за изпълнени тестове; • Протоколи за извършени услуги за техническа асистенция; • Актуализиран софтуер за национален адаптор за EUCISE2020 в следствие на коригирани на установени пре тестването недостатъци; • Програмен код на актуализирания софтуер за Национален адаптор за EUCISE2020; • Актуализирани документи – спецификации, ръководства и др.
-----	--	---------------------------------------	---

Handwritten signature and scribble.

68

Handwritten signature.

Handwritten signature.



4.3. Подробен план-график за изпълнение

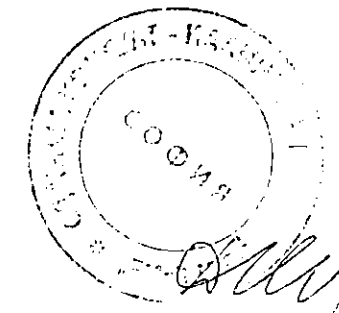
ID	Задача	Продълж.	Начало	Край	Qtr 2, 2018			Qtr 3, 2018			Qtr 4, 2018			Qtr 1, 2019					
					Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar		
1	Срокове	306 days	02.04.2018	02.02.2019															
2	Условно начало на проекта	0 days	02.04.2018	02.04.2018	◆ 02.04														
3	Срок за изпълнение на Етап 1 (1 мес.)	0 days	02.05.2018	02.05.2018	◆ 02.05														
4	Срок за изпълнение на Етап 2 (4 мес.)	0 days	02.08.2018	02.08.2018	◆ 02.08														
5	Срок за изпълнение (10 мес.)	0 days	02.02.2019	02.02.2019	◆ 02.02														
6	Дейност 1: Анализ, проектиране и разработка на адаптера	122 days	02.04.2018	01.08.2018	[Timeline bar from 02.04.2018 to 01.08.2018]														
7	Етап 1: Бизнес анализ и Проектиране	30 days	02.04.2018	01.05.2018	[Timeline bar from 02.04.2018 to 01.05.2018]														
8	Анализ на данните и изискванията	12 days	02.04.2018	13.04.2018	[Timeline bar from 02.04.2018 to 13.04.2018]														
9	Проучване на данни и изисквания	10 days	02.04.2018	11.04.2018	[Timeline bar from 02.04.2018 to 11.04.2018]														
10	Разработване на план за адресиране на идентифицираните нужди	5 days	09.04.2018	13.04.2018	[Timeline bar from 09.04.2018 to 13.04.2018]														
11	Изготвяне на Спецификация на софтуерните изисквания	23 days	09.04.2018	01.05.2018	[Timeline bar from 09.04.2018 to 01.05.2018]														
12	Определяне на концепция за адаптера	3 days	09.04.2018	11.04.2018	[Timeline bar from 09.04.2018 to 11.04.2018]														
13	Дефиниране на детайлни изисквания и бизнес процеси	12 days	12.04.2018	23.04.2018	[Timeline bar from 12.04.2018 to 23.04.2018]														
14	Дизайн на информационната система	10 days	19.04.2018	28.04.2018	[Timeline bar from 19.04.2018 to 28.04.2018]														
15	Дизайн на хардуерната и комуникационната инфраструктура	3 days	26.04.2018	28.04.2018	[Timeline bar from 26.04.2018 to 28.04.2018]														
16	Определяне на потребителски интерфейс	3 days	24.04.2018	26.04.2018	[Timeline bar from 24.04.2018 to 26.04.2018]														
17	Изготвяне на план за техническа реализация	3 days	29.04.2018	01.05.2018	[Timeline bar from 29.04.2018 to 01.05.2018]														
18	Спецификации и доклади	0 days	01.05.2018	01.05.2018	◆ 01.05														
19	Спецификация на софтуерните изисквания за национален неklasифициран адаптор за EUCISE2020	0 days	01.05.2018	01.05.2018	◆ 01.05														

69

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]



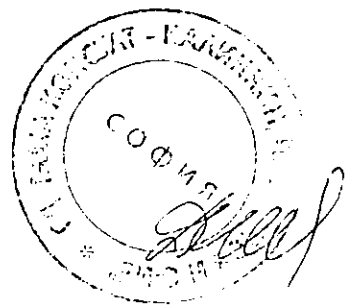
ID	Задача	Продълж.	Начало	Край	Mar	Qtr 2, 2018	May	Jun	Qtr 3, 2018	Aug	Sep	Qtr 4, 2018	Nov	Dec	Qtr 1, 2019	Jan	Feb	Mar	
						Apr			Jul			Oct							
20	Разработване на архитектурен прототип за доказване на работоспособност и изпълнимост	14 days	14.04.2018	27.04.2018															
21	Определяне на цели и изисквания за прототип	2 days	14.04.2018	15.04.2018															
22	Разработка на прототипа	10 days	16.04.2018	25.04.2018															
23	Оценка на проверяваните чрез прототипа варианти за решения	5 days	23.04.2018	27.04.2018															
24	Софтуер за национален адаптор за EUCISE2020 (архитектурен прототип)	0 days	27.04.2018	27.04.2018															
25	Програмен код на софтуер за национален адаптор за EUCISE2020 (архитектурен прототип)	0 days	27.04.2018	27.04.2018															
26	Етап 2: Изграждане на Адаптера	92 days	02.05.2018	01.08.2018															
27	Разработване на софтуерното решение	83 days	02.05.2018	23.07.2018															
28	Разработка на модулите на информационната система	83 days	02.05.2018	23.07.2018															
29	Итерация 1: Рамка на системата; Базова функционалност на CISE Adapter; Връзка с CISE Node;	40 days	02.05.2018	10.06.2018															
30	Итерация 2: Интерфейси за съществуващи системи; Администриране; Тестов клиент;	27 days	11.06.2018	07.07.2018															
31	Итерация 3: Оптимизиране и изчистване на дефекти	16 days	08.07.2018	23.07.2018															
32	Софтуер за национален адаптор за EUCISE2020	0 days	23.07.2018	23.07.2018															
33	Програмен код на софтуер за национален адаптор за EUCISE2020	0 days	23.07.2018	23.07.2018															

70

Handwritten signature

Handwritten signature

Handwritten signature



ID	Задача	Продълж.	Начало	Край	Timeline											
					Mar	Qtr 2, 2018		Qtr 3, 2018		Qtr 4, 2018			Qtr 1, 2019			
					Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
34	Вътрешно тестване и сценарии за приемни тестове	78 days	07.05.2018	23.07.2018	[Timeline bar from May to July]											
35	Провеждане на вътрешни тестове	78 days	07.05.2018	23.07.2018	[Timeline bar from May to July]											
36	Изготвяне на детайлни сценарии за провеждане на приемателните тестове	10 days	14.07.2018	23.07.2018	[Timeline bar in July]											
37	Тестови сценарии	0 days	23.07.2018	23.07.2018	◆ 23.07											
38	Документиране на разработката	30 days	14.06.2018	13.07.2018	[Timeline bar from June to July]											
39	Разработване на ръководства	15 days	14.06.2018	28.06.2018	[Timeline bar in June]											
40	Специфициране на реализацията	20 days	14.06.2018	03.07.2018	[Timeline bar from June to July]											
41	Разработване на ръководства и препоръки за интеграция	15 days	29.06.2018	13.07.2018	[Timeline bar from June to July]											
42	Спецификация на реализацията	0 days	13.07.2018	13.07.2018	◆ 13.07											
43	Ръководство на потребителя	0 days	13.07.2018	13.07.2018	◆ 13.07											
44	Ръководство на администратора	0 days	13.07.2018	13.07.2018	◆ 13.07											
45	Ръководство за интеграция с националната система за управление на корабния трафик (VTMS)	0 days	13.07.2018	13.07.2018	◆ 13.07											
46	Препоръки за необходими промени и доработки в Националната система за управление на корабния трафик (VTMS)	0 days	13.07.2018	13.07.2018	◆ 13.07											
47	Тестване	22 days	11.07.2018	01.08.2018	[Timeline bar from July to August]											
48	Подготовка за тестване	4 days	11.07.2018	14.07.2018	[Timeline bar in July]											
49	Разработване на методология за тестване	2 days	11.07.2018	12.07.2018	[Timeline bar in July]											
50	Разработване на план за тестване	4 days	11.07.2018	14.07.2018	[Timeline bar in July]											
51	План за тестване	0 days	11.07.2018	11.07.2018	◆ 11.07											

Handwritten signature and initials.

Handwritten signature.

Handwritten signature.



ID	Задача	Продълж.	Начало	Край	Qtr 2, 2018			Qtr 3, 2018			Qtr 4, 2018			Qtr 1, 2019		
					Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb
52	Осигуряване на среда и инсталиране и конфигуриране на адаптера	13 days	11.07.2018	23.07.2018												
53	Проектиране на ИКТ инфраструктурата	15 days	11.07.2018	15.07.2018												
54	Проект за конфигуриране и разполагане на компонентите на адаптора върху ИКТ инфраструктурата	10 days	15.07.2018	15.07.2018												
55	Проект за мрежова свързаност на ИКТ инфраструктурата на адаптора	10 days	15.07.2018	15.07.2018												
56	Съгласуване на проектите за конфигуриране и свързаност	3 days	16.07.2018	18.07.2018												
57	Осигуряване на необходимата ИКТ инфраструктура и свързаност	1 day	19.07.2018	19.07.2018												
58	ИКТ инфраструктура за периода на тестване и съпровождане на адаптора	0 days	19.07.2018	19.07.2018												
59	Интернет-свързаност за работа на адаптора за периода на тестване и съпровождане	0 days	19.07.2018	19.07.2018												
60	Конфигуриране на осигурената ИКТ инфраструктура и мрежова свързаност	2 days	20.07.2018	21.07.2018												
61	Инсталиране и конфигуриране на EUCISE2020 адаптора върху предоставената ИКТ инфраструктура	2 days	22.07.2018	23.07.2018												
62	Инженерингови услуги за свързване на EUCISE2020 адаптера	4 days	24.07.2018	27.07.2018												
63	Инженерингови услуги за свързване и интеграция на националната система (VTMIS)	4 days	24.07.2018	27.07.2018												
64	Инженерингови услуги за свързване и интеграция на Националния EUCISE2020 адаптор с EUCISE2020 нодовете	4 days	24.07.2018	27.07.2018												
65	Протоколи за извършени инженерингови услуги за свързване и интеграция на националната система (VTMIS)	0 days	27.07.2018	27.07.2018												
66	Протоколи за извършени инженерингови услуги за свързване и интеграция на Националния EUCISE2020 адаптор с EUCISE2020 нодовете	0 days	27.07.2018	27.07.2018												

22

Handwritten signature

Handwritten signature

Handwritten signature

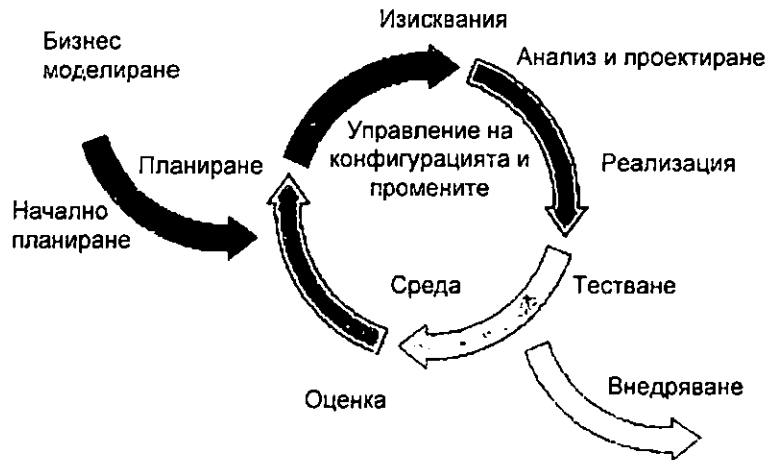


Предимство на този подход е предоставянето на използваем продукт в края на всяка итерация и навременното получаване на обратна връзка. Това дава възможност за гъвкавост при разработката при гарантиране на крайния резултат.

5.3.1. Процес за софтуерната разработка

Процесът на софтуерна разработка ще бъде цикличен и интерактивен посредством създаването на прототипи и одобрението им от потребителите. Това ще даде възможност за актуални прогнози и очаквания, като графика за предстоящата разработка ще бъде третиран като „жив“, променящ се документ, който отразява промените на изискванията и възникналите проблеми, съответно към намираните решения и успешното осигуряване на качеството.

Поради спецификите на проекта и кратките срокове за неговата реализация, проектът ще се изпълнява при ефективно съчетаване на класическия последователен подход (Waterfall) и итеративния (Agile) подход за реализация.

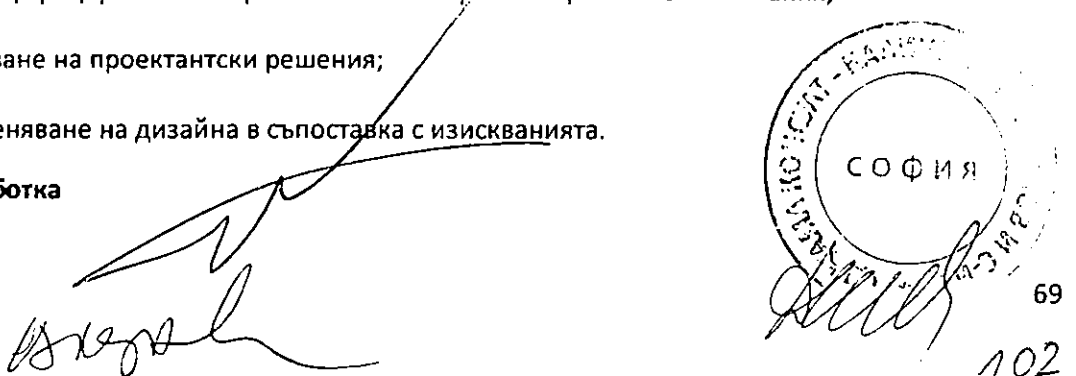


Когато програмната система се разработва от самото начало, процесът е същият като създаване на система по изисквания. Но след като веднъж системата придобие форма – или след като системата премине през първоначалния цикъл на разработка – всяка бъдеща доработка е процес на приспособяване на системата към новите или променени изисквания. Това се прилага през целия жизнен цикъл на системата.

Съществуват четири насочени към потребителя дейности по проектирането, за които е важно да започнат в най-ранната фаза на проекта:

- Разбиране и специфициране контекста на използване;
- Специфициране на потребителските и организационните изисквания;
- Даване на проектантски решения;
- Оценяване на дизайна в съпоставка с изискванията.

Итеративна разработка



UML не ограничава UML типовете елементи да принадлежат към конкретна диаграма. Като правило всеки UML тип на елемент може да се появи на почти всички типове диаграми. Тази гъвкавост е частично ограничена в UML 2.0. UML профилите могат да дефинират допълнителни типове диаграми или да разширяват съществуващи диаграми с допълнителни елементи. За съвместимост с традиционните инженерни чертежи, UML диаграмите позволяват поставянето на болезки или описание на предназначението, ограничения или намерения.

5.2.5. Софтуер за моделиране на UML – Sparx Systems Enterprise Architect
Използваният продукт за документиране и моделиране ще бъде Sparx Systems Enterprise Architect водещ продукт за моделиране с UML, поддържащ всички подходящи UML модели.

Със Sparx Enterprise Architect може да се създават и модели за стратегическо моделиране. Продуктът също поддържа моделиране на работните процеси с BPMN, както и организация и управления на изискванията, възможности за прототипиране на потребителски интерфейси и бяла дъска.

Продуктът може да бъде пригоден и за използване на модели, като ключовият за настоящия проект модел на бизнес дейностите.

Продуктът може да генерира документация на базата на поддържаните в него модели. Между голяма част от моделите може да се поддържат директни връзки.

5.2.6. Адаптиране на подхода за проектиране на софтуер
Дизайнът ще бъде съобразен с цялостната архитектура и специфики на CISE.

За целите на пилотното тестване ще се използва част от цялостния модел на данните.

5.3. Подход за софтуерна разработка

Методологии и добри практики:	Итеративна разработка Унифициран процес за разработка Обектно-ориентирано програмиране Непрекъснатата интеграция
-------------------------------	---

Средства и техники:	Scrum
Инструменти:	JetBrains IntelliJ Idea Sparx Systems Enterprise Architect Subversion Oracle Hudson Atlassian Jira

Източници:	www.ibm.com www.omg.org www.scrum.org
------------	--

Обосновка:	Поради спецификите на проекта и кратките срокове за неговата реализация, проектът ще се изпълнява при ефективно съчетаване на класическия последователен подход (Waterfall) и итеративния (Agile) подход за реализация. Етапите на разработка ще бъдат групирани в итерации по фази.
------------	--

UML моделите могат да бъдат автоматично трансформирани в други представяния (например Java) посредством подобни на QVT езици за трансформиране. UML е разширяем с двата си механизма за настройване: профили и стереотипи.

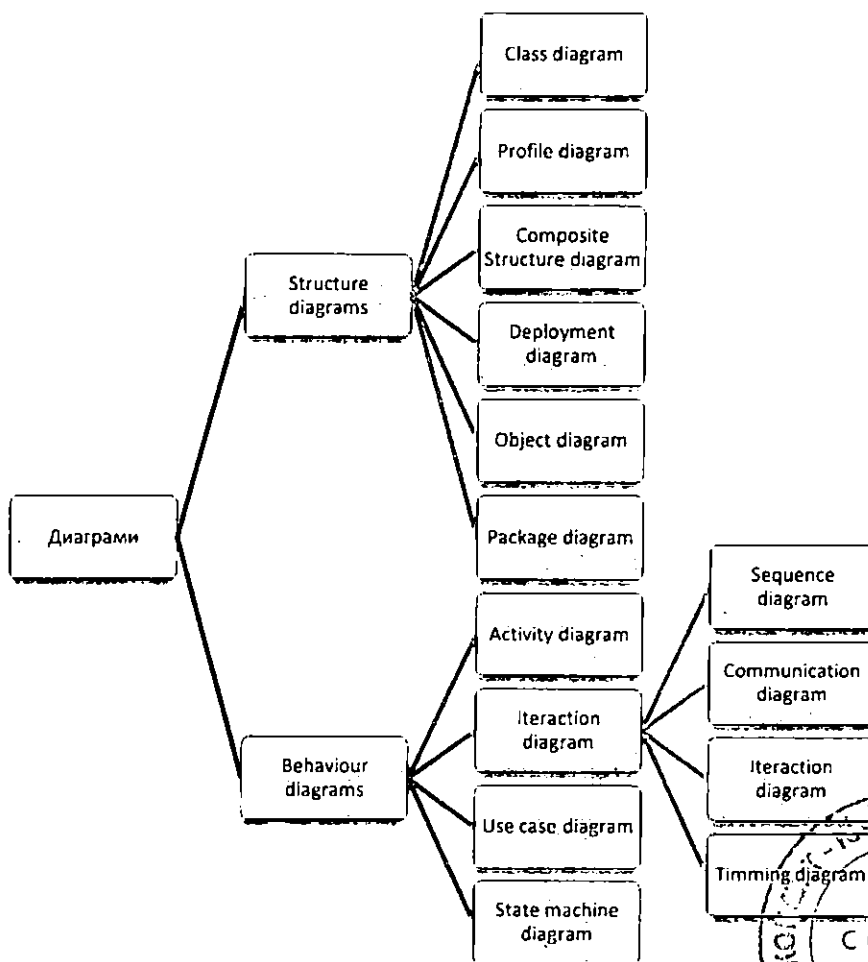
UML модели

Важно е да се направи разлика между UML модел и множество диаграми на дадена система. Диаграмата е частично графично представяне на модела на системата. Моделът съдържа информация, свързана с елементите и диаграмите му (като например текстови описания на изискванията на потребителите).

UML диаграмите представят два различни изгледа на модела на системата:

- 1) Статичен (или структурен) изглед – фокусът е върху статичната структура, като се използват обекти, атрибути, операции и взаимовръзки. Структурния изглед включва диаграма на класовете и диаграма на композитната структура.
- 2) Динамичен (или поведенчески) изглед – набляга на динамичното поведение на системата, показвайки взаимодействията между обектите и промените на вътрешните състояния на обектите. Този изглед включва диаграми като диаграма на последователностите, на действията и на състоянията.

UML 2.2 има 14 типа диаграми, разделени в двете категории. Седем от тях представят структурната информация, а останалите седем представят общите типове поведение, като четири от тях представят различни видове взаимодействие. Тези диаграми могат да бъдат категоризирани йерархично, по начина показан на следната клас диаграма:



СОФИЯ
67
100

- 3) Шаблоните за изпълнението адресират въпросите, свързани с изпълнението на приложението, включително стратегиите по изпълнение на система от задачи и работните блокове, осъществяващи синхронизацията между задачите.
- 4) Шаблонните стратегии за имплементирането адресират проблемите, свързани с имплементирането на програмния код и организацията на програмата, както и общите структури от данни специфични за паралелното програмиране.
- 5) Шаблоните за структурния дизайн адресират въпросите, свързани със структурите от високо ниво в разработваното приложение.

5.2.4. Моделиране с UML

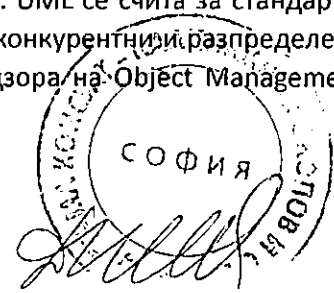
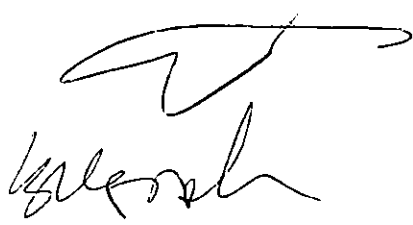
В проекта ще се използва методика за моделиране, основаваща се на унифицирания език за моделиране (Unified Modeling Language – UML).

Unified Modeling Language (UML) е приложим към всички фази на реализирането на продукта. В частност, в сферата на системната архитектура, той допълва модела на проектиране с възможности за унифицирано моделиране и техники за право и обратно генериране (forward/reverse engineering). UML е приложим към техниките за софтуерна разработка, както и към техниките за бизнес анализ споменати по-горе. Той е стандартизиран език за моделиране с общо предназначение, разработен специално за сферата на обектно ориентираното разработване. Той включва в себе си множество графични правила и техники за създаване на визуални модели на обектно ориентирани софтуерни системи.

UML се използва за специфициране, визуализиране, конструиране и документиране на такива системи. Той предлага стандартизиран подход за визуализиране на проектите на разработваната система, включвайки елементи като:

- действия;
- актьори;
- бизнес процеси;
- схема на данните;
- (логически) компоненти;
- изрази на програмен език;
- многократно използвани софтуерни компоненти.

UML комбинира техники от моделирането на данните (entity relationship diagrams), бизнес моделирането (work flows), обектното проектиране и компонентното моделиране. Той може да бъде използван заедно с всички процеси от жизнения цикъл на проект за софтуерна разработка и е независим от различните технологии за реализация. UML синтезира в себе си методите на Booch, обектно ориентираните техники (OMT) и обектно ориентираното проектиране и разработка на софтуер (OOSE), сливайки ги в един общ и широко прилаган език за моделиране. UML се счита за стандарт в езиците за моделиране, посредством който могат да бъдат моделирани конкурентни и разпределени системи. UML е де факто стандарт в индустрията и се развива под надзора на Object Management Group (OMG).



на способността за адаптиране към промените по време на целия жизнен цикъл на софтуера, за да се удължи продуктивният живот на софтуерната система.

5.2.3. Обектно-ориентиран дизайн

Ще бъдат прилагани принципите на обектно-ориентирания дизайн. Основните принципи, на които се обобщават в абривиатурата SOLID (Single responsibility, Open-closed, Liskov substitution, Interface segregation and Dependency inversion):

S	Принцип на конкретна отговорност (Single responsibility principle) Класовете трябва да имат конкретна, единична отговорност.
O	Принцип отворени/затворени (Open/closed principle) Софтуерните обекти трябва да бъдат отворени за разширяване и затворени за промяна.
L	Принцип на Лисков за заместването (Liskov substitution principle) Обектите в една програма трябва да бъдат заменими с инстанции от техни подтипове без да се променя коректността на програмата.
I	Принцип за разделяне на интерфейсите (Interface segregation principle) Много специфични за даден клиент интерфейси са по-добри от един интерфейс с общо предназначение.
D	Принцип за инверсия на зависимостите (Dependency inversion principle one) Нещата трябва да зависят от абстракциите, а не от конкретиката.

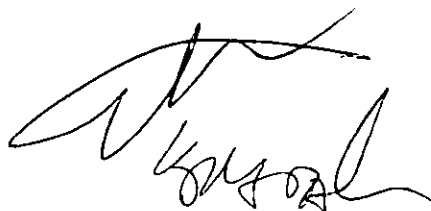
Тъй като дори и малките съвременни системи са изключително сложни за реализиране, има нужда от управление на обектно-ориентиран код, за да се избегне ненужната сложност. Тук се намесват шаблоните за дизайн, които са множество на добри практики за проектиране на кода.

В софтуерно инженерство, шаблонът за дизайн е решение за многократна употреба на проблеми, често срещани се в даден контекст, в софтуерния дизайн. Дизайн-модел не е завършен дизайн, който може да се трансформира директно в код. Това е описание или шаблон за това как да се реши даден проблем, който може да се използва в много различни ситуации. Обектно-ориентираните шаблони за дизайн показват взаимоотношенията и взаимодействията между класовете или обектите, без да се уточняват окончателните класове или обекти, които ще бъдат въввлечени. Много шаблони предполагат обектно-ориентирано или по-общо променящо се състояние, поради което не са подходящи при езиците за функционално програмиране, при които обикновено данните се третират като непроменящи се.

Шаблоните за дизайн са модел, работещ с модули и връзки между тях. На по-високо ниво те са архитектурни модели, които са по-големи по обхват, обикновено описващи цялостната картина, следвана от системата.

Има различни видове шаблони за дизайн, които могат да бъдат използвани при разработката:

- 1) Шаблоните за алгоритмични стратегии адресират въпросите, свързани със стратегиите от високо ниво, описващи как да се използват приложните характеристики на изчислителната платформа.
- 2) Шаблоните за изчислителен дизайн адресират въпросите, свързани с ключовите изчислителни проблеми.



Описание на решението трябва да бъде направено от различни гледни точки, които отразяват различни аспекти на системата като областта на приложение (домейна), функционалните изисквания, артефактите за разработката на системата, компоненти по време на изпълнение и тяхното поведение, съответствие с инфраструктурата, върху която ще се внедри решението. С цел да бъдат полезни документираните спецификации трябва да бъде едновременно достатъчно детайлни за анализ и с необходимото ниво на абстракция за разбиране.

Желателно е процеса на проектиране да приема добре дефиниран архитектурен стил с капсулиране на важните решения за архитектурни елементи и подчертаване важните ограничения на елементите и техните връзки.

Процесът обикновено е подкрепен с референтна архитектура, която включва набор от най-добри практики за домейна, на който трябва да функционира системата. В някои индустрии и технологични домейни, тази референтна архитектура е формализирана.

Анализ на архитектурни алтернативи

Целта на анализа на софтуерната архитектура е да се валидира, че изискванията за качество са били спазени в проектирането и са идентифицирани потенциалните рискове. Методите за оценка могат да се класифицират като базирани на софтуерни метрики, на сценарии или на модел на атрибутите. Методите базирани на софтуерни метрики използват идеите за ниска свързаност (зависимост) между модулите и кохезия, за да се идентифицират потенциалните проблеми с качеството. Методите базирани на сценарии се фокусират върху идентифициране на противоречи изисквания или непълна спецификация от гледна точка на заинтересованите страни. Методите за оценка, базирани на модел на атрибутите се фокусират върху оценка на качествените показатели. Техниките за оценка могат да бъдат беседа, контролни листове, формални методи, симулация или прототипи.

Проверка на съответствието на архитектурата

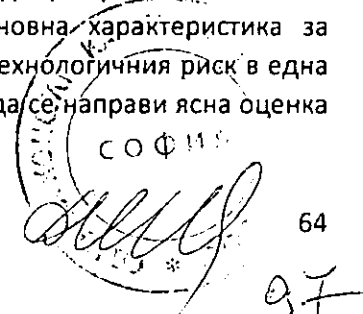
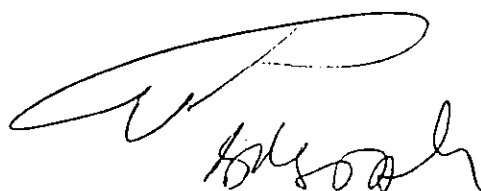
Често се налага на софтуерните архитекти да удостоверят, че системата е имплементирана в съответствие с планираната архитектура. По време на изпълнението и развитието на система, понякога се налагат отклонения от определената архитектура, дори и в първата версия, поради непознаване от разработчиците, противоречащи си изисквания, късни промени в изискванията, технически трудности, къс срок за изпълнение или малък бюджет. Тези отклонения обикновено се натрупват с течение на времето, което води до феномена, известен като архитектурен ерозия.

Съответствието на архитектурата е мярка в каква степен имплементираната архитектура в изходния код съответства на планираната архитектура. За оценка на съответствието, трябва да се създаде модел на действителната система, който да се сравни с планирания архитектурен модел, за да се установят нарушения, като например непозволенни зависимости.

Алтернативно, отклоненията от планираната архитектура могат да бъдат предотвратени чрез прилагане на определени шаблони за дизайн, които предлагат утвърдени решения за повтарящи се проблеми при проектирането, следващи архитектурните ограничения, използването на фреймуърци (рамки за разработка), чиято архитектура е в съответствие с търсената архитектура на решението или с помощта на инструменти, които генерират архитектурно-съвместим код.

Управление на промените в архитектурата

Способността на една софтуерна система лесно да се адаптира към бъдещи промени в бизнес изискванията, организационния контекст, както и технологии. Тя е основна характеристика за увеличаване на икономическата стойност на софтуера и минимизиране на технологичния риск в една организация. За софтуерните системи с дълъг жизнен цикъл, е необходимо да се направи ясна оценка



- Спецификацията на системата, която ще бъде имплементирана (блупринт);
- Средство за комуникация за постигане на общо разбиране (език);
- Обосновка за избора на решения, които ще бъдат имплементирани от системата;
- Документация за сегашните и бъдещите потребители и разработчици.

Множеството гледни точки водят до необходимостта от описание на софтуерната архитектура с различни методи и похвати. Методите за описание на софтуерната архитектура могат да бъдат класифицирани като:

- Артефакт ориентирани методи: фокусът е в артефактите, които трябва да бъдат произведени с акцент върху структурното описание на системата;
- Методи на потребителските случаи: използват описанието на потребителски случаи като основен артефакт;
- Методи ориентирани към качеството: използват качествените характеристики като основен двигател за избиране на едно архитектурно решение пред друго;
- Методи от областта на приложение: използват концепции от домейна (областта на приложение), при дефинирането на архитектурата.

Независимо от избраният метод (артефакти, случаи на употреба, ориентиран към качеството или домейна), процеса на архитектурното проектиране включва следните аспекти:

- Анализ на изискванията на архитектурата, идентифициране на рискове и процес за вземане на решения;
- Документиране на характеристиките и принципите на системата и нейното развитие, за да се улесни комуникацията им със заинтересованите страни на системата;
- Оценка и сравнение на архитектурни алтернативи;
- Проверка на съответствието на имплементацията на системата с архитектурното си описание;
- Управление на промените в архитектурата

Анализ на архитектурните изисквания

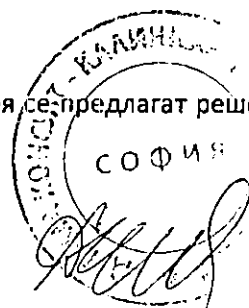
Анализът на архитектурните изисквания е процес, чрез който се идентифицират целите системата и очакваните качествени характеристики. Тези изисквания трябва да бъдат пряко свързани с бизнес целите и изрично да включват заинтересованите страни.

Една от най-важните стъпки в анализа е определянето на контекста: къде завършената и доставена система ще се впише в текущата оперативна среда. Идентифицирането на заинтересованите страни, системни цели, и на сценарии и случаи на употреба зависят от правилно дефинираният обхват.

Събирането на изисквания може да се направи с помощта на различни техники, като например използването на въпросници и анкети, интервюта и анализ на съществуващата документация. Груповото събиране на изисквания може да бъде използвано за ангажиране на заинтересованите страни и да предизвика по-пълно разбиране на нуждите и насърчаване на постигането на съгласие между различните заинтересовани страни. Тези техники включват брейнсторминг, фокус групи, и работни срещи за постигане на консенсус. Съществуват и по-официални методи, които се основават на подробен модел на събираната информация, като качествени характеристики и тяхното определение по отношение на качествените показатели.

Документиране на характеристиките

Тази дейност е в основата на процеса на разработване на архитектурата. В нея се предлагат решения от гледна точка на организация на системата и принципи за развитие.

- 10) Изисквания към поддръжката;
- 11) Предположения и ограничения;
- 12) Изисквания за тестване.

Техническата част на спецификацията трябва да включва различните аспекти на архитектурата на приложението, което ще бъде реализирано. Това означава, че трябва да бъдат анализирани и документиранни аспекти като:

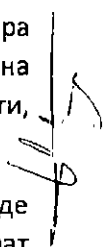
- 1) Сигурност;
- 2) Технология;
- 3) Интерфейси;
- 4) Миграция (Цифровизация);
- 5) Стандарти;
- 6) Производителност;
- 7) Скалируемост.



Верификация

Оценка на решенията и валидиране описва начина, по който бизнес анализаторите оценяват предложените решения. За да се определи кое решение най-добре отговаря на бизнес изискванията, трябва да се идентифицират пропуските и недостатъците в предложените решения и да се определят необходимите отклонения (заобикаляния) или промени на обхвата. Тази част от анализа също описва как бизнес анализаторите оценяват разгърнатите решения, за да се види как тези решения отговарят на първоначалните изисквания, така че спонсориращата организация да може да оцени изпълнението и ефективността на предложеното решение.

При проектирането на процесите на системата за управление е важно да използваме информацията за действащите и идеалните процеси и да съдействаме на ръководството на организацията да дефинира „желаните“ процеси. Решенията за реда за протичане на дейността са изцяло от компетенциите на мениджърите на организацията и не могат, и не бива да се прехвърлят върху екипа от специалисти, които са натоварени с разработването на системата.



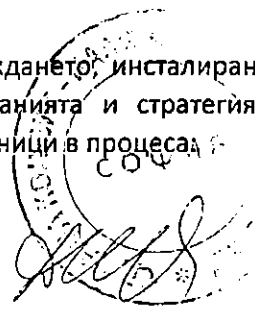
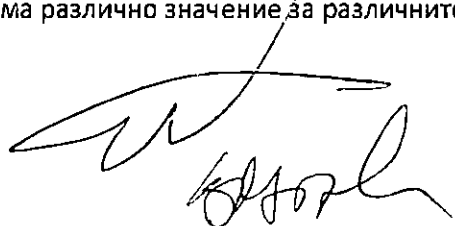
Процесът по верификация на изискванията следва този по документиране. Той може да бъде еднократен, т.е. да се верифицират изискванията наведнъж в една итерация или да се верифицират поетапно групи изисквания. Той има за цел проверка и формално одобрение на документираните изисквания.

Изискванията трябва да бъдат верифицирани. Документацията се представя на Възложителя и ако е необходимо се провеждат разяснителни сесии. Спецификацията задължително се документира.

5.2.2. Софтуерна архитектура

Методи за създаване на софтуерна архитектура

Софтуерната архитектура е от голямо значение при проектирането, изграждането, инсталирането и поддръжката на всяка софтуерна система. Тя определя целите, изискванията и стратегията за разработка. Архитектурата обаче има различно значение за различните участници в процеса:



- **Изучи (Study)** – използва се при PDSA цикъла - Изучаване и анализ на постигнатите резултати, с цел уточняване на по-нататъшните действия.

8) **Действие (Act)** - В зависимост от резултатите на проверката на този етап са възможни два вида действия:

- Въвеждане на постигнатото по-високо ниво като нов стандарт (формализиране) в организацията;
- Извършване на коригиращи и/или превантивни действия за подобряване на постигнатото и за постигане в следващия цикъл още по-високо ниво.

Документиране

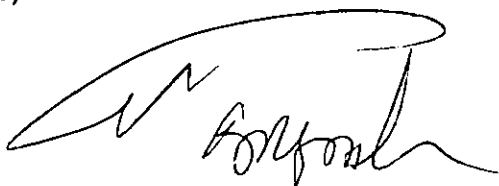
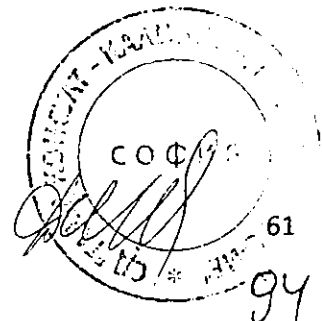
В рамките на под-процеса по Анализ ще се постави основата за реализацията и приемането на продукта. В под-процеса Документиране, се изготвя документ, който има за цел да формализира тази основа. Той трябва да бъде верифициран от Възложителя и в последствие ще бъде реализиран от екипа по разработка.

Документирането е процеса по записване на анализирания изисквания, било те функционални или технически. То се извършва паралелно на останалите процеси, но леко изостава във времето от тях. То служи като основа за следващия процес по верификация на изискванията, където възложителя трябва да потвърди коректността на извършения анализ.

При документирането първоначално се документират основните изисквания на заинтересованите лица, детайлизирани до нивото, до което те са известни. Във времето те се конкретизират и детайлизират. Поради това, процесът е итеративен и спираловиден. Преди да бъдат вписани изискванията, те трябва да бъдат SMART (виж по-горе).

Формата на документа може да бъде различен. Това може да е един документ, изброяващ всички изисквания, категоризирани по заинтересовани лица и приоритет, както и по-подробна форма, съдържаща описание, детайли и свързани документи. Този документ може да включва например следното съдържание:

- 1) Бизнес целите и целите на проекта за проследимост;
- 2) Функционалните изисквания, описващи бизнес процеса, информацията и взаимодействията с продукта или тези от тях, които са необходими;
- 3) Може да бъде текстов документ даващ списък на изискванията, моделите или и двете;
- 4) Нефункционални изисквания, като ниво на услугата, производителност, сигурност, съвместимост със стандарти и административни изисквания, възможности за поддръжка, и др.
- 5) Изисквания към качеството;
- 6) Бизнес правила;
- 7) Въздействие върху други области на организацията, като центъра за обслужване, технически групи и т.н.;
- 8) Въздействие върху други подразделения вътре и извън организацията;
- 9) Изисквания към обученията;

Цикъл на Шухарт

PDCA/PDSA

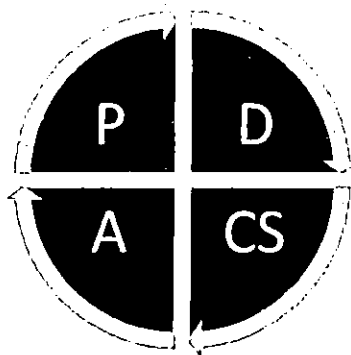
PDCA/SDCA

Цикълът на Деминг

Цикълът на Деминг показва пътя към подобрения. Той стои в основата на много от съвременните концепции за развитие на бизнеса. Известен е още като "Цикъл на Шухарт", "PDCA цикъл", "PDSA цикъл" или "SDCA цикъл". Съществуват и редица други модификации на цикъла, които доказват приложимостта му в много области и ситуации.

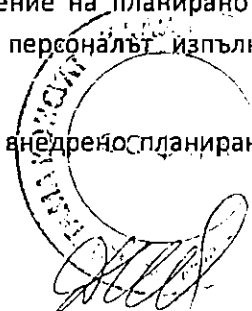
PDCA (PDSA) цикъл

Деминг развива идеите на Шухарт, като дава по-обща названия на всеки от етапите и добавя още един етап за анализ на получената на втория етап информация.



PDCA (PDSA) цикъл

- 5) **Планиране (Plan)** - Използва се подхода 5W 1H, при който трябва да се дадат отговори на следните шест въпроса:
 - какво? (what?) – дефиниране на целите и задачите
 - защо? (why?) – обосноваване на необходимостта;
 - кой? (who?) – определяне на отговорностите;
 - как? (how?) – определяне на метода (методите) на извършване;
 - къде? (where?) – ограничаване на полето на действие;
 - кога? (when?) – разработване на план (график).
- 6) **Изпълнение (Do)** - В началото на този етап се извършва необходимото обучение и квалификация на персонала, който ще извършва реалното изпълнение на планираното и неговото внедряване. След обучението (или паралелно с него) персоналят изпълнява планираното и го внедрява.
- 7) **Проверка (Check)** - На този етап се проверява как е изпълнено и внедрено планираното действие и се оценяват постиженията.



Средства и техники:	Microsoft Visio Sparx Systems Enterprise Architect – софтуер за моделиране на изисквания, бизнес процеси и архитектура. Продуктът поддържа всички описани по-долу видове диаграми.
Източници:	Object Management Group – сайт и документи: www.omg.org/ www.Uml.org http://www.sparxsystems.com.au/
Обосновка:	Използват се световно утвърдени процеси и принципи за проектиране, като цикъл на Деминг, SOLID, шаблони за дизайн, UML и др. UML и BPMN са стандарти в обектно ориентираното моделиране. Enterprise Architect е утвърден софтуерен продукт, който предлага необходимата функционалност и, с който нашият екип има опит. Продуктът е популярен и сред българските разработчици и възложители.

5.2.1. Процес за проектиране


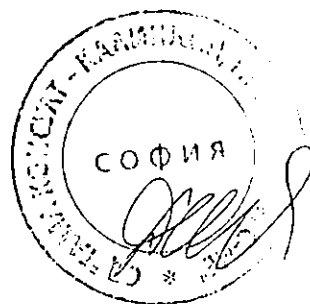
При проектирането на процесите и при разработването на ориентирана към процесите система, може да се използва и популярния цикъл на Деминг.

Важно условие за използването на цикъла на непрекъснато усъвършенстване е използването на подходящи методи за анализ на представянето на процесите.

Такива методи например могат да бъдат:

- 5) Съпоставка с „идеален“ процес. Цел: да се открият различията между действащата практика и определения като идеален вариант;
- 6) Проиграване на сценарии тип „какво ще се случи, ако...“. Цел: да се разработят няколко варианта за осъществяване на дадена дейност и да се избере най-добрият.
- 7) Съпоставка с предварително дефинирани ключови параметри на процеса. Цел: да се открият пропуски или възможности за подобрене при осъществяването на процеса.
- 8) Съпоставка между два и повече процеса в организацията. Цел: да се открият общи за организацията силни или слаби страни и да се търси начин на пренасяне на успешните практики от един процес към друг.

Използването на методи за анализ, помага на екипа наточен с разработка и внедряване на система за управление при изготвянето на описание на процесите и на тяхното взаимодействие. Методите за анализ следва да бъдат предвидени като част от системата за управление и да се превърнат в обичайна практика в организацията.

Най-често употребявани техники за валидиране на изискванията са:

- Непрекъснато потвърждаване (Continual Confirmation) – извършва се когато има готови изисквания. Не е еднократен процес и не се изчаква завършването на цялата документация. Потвърждение се появява всеки път, когато бизнес анализатор преглежда изискванията/ решението със заинтересованите страни.

Потвърждаването не е одобрение - само оторизирани заинтересовани страни могат да одобряват. Потвърждаването означава получаване на съгласие от заинтересованите страни, че решението, което ще бъде разработено, е добро и ще постигнете целите.

- Формален преглед на изискванията (Requirements Walkthrough) – техниката се използва, когато прави преглед на изискванията със заинтересованите страни с цел да се получи потвърждение, че посочените изисквания са валидни. Валидни изисквания са тези, които отразяват точно решението, което заинтересованите страни имат необходимост да бъде разработено и това решение е реализуемо. За провеждане на преглед на изискванията, бизнес анализаторът организира сесия като предоставя на заинтересованите страни (от страна на Възложителя и представители на различни роли в екипа на Изпълнителя) материалите и достатъчно време да се подготвят.

Тази сесия е една от възможностите за заинтересованите страни да повдигат въпроси, да търсят яснота и да изкажат опасенията си. Бизнес анализаторът може да използва тази сесия като възможност да се изчистят и затворят всички отворени въпроси, свързани с подготовката за приемане и окончателно одобрение на изискванията.

Одобрение на изискванията

Одобрението на изискванията е отделен процес от тяхното потвърждаване. След като изискванията/решението е потвърдено и валидирано, се пристъпва към формално одобрение на изготвената документация. Тъй като изискванията са предварително потвърдени, че са верни, точни, разбираеми, приложими и т.н., от всички отделни заинтересовани участници в проекта, получаването на одобрение обикновено е рутинна процедура.

5.1.3. Адаптиранена подхода за бизнес анализ

Бизнес анализът ще се фокусира върху спецификата на услугите и данните, предоставяни през CISE мрежата и съществуващите системи, които в бъдеще ще бъдат интегрирани.

5.2. Подход за проектиране на софтуер

Методологии и добри практики:	Unified Modelling Language - UML
	Business Process Modelling Notation – BPMN
	SOLID
	Цикъл на Деминг
	Шаблони за дизайн



Верифициране на изискванията

Процесът по верификация на изискванията следва този по документиране. Той може да бъде еднократен, т.е. да се верифицират изискванията наведнъж в една итерация, или да се верифицират поетапно групи изисквания. Той има за цел проверка на документираните изисквания от страна на екипа на Изпълнителя.

Техники за верифициране на изискванията:

- Преглеждане (Peer Review) - преглед от друг анализатор, който извършва аналогична дейност (могат да бъдат включени и тестер или специалист по качеството). Целта е в документацията да се открият дефекти и пропуски. Обичайно прегледът не е формализиран.
- Инспектиране (Inspection) - по-строга форма на преглеждането. Изискванията се проверяват за всички аспекти на „SMART“ принципа, както и за точност, пълнота, актуалност и т.н.

В рамките на този процес се осъществява взаимодействието с техническите екипи и екипа по осъществяването на качеството. Процесите по проектиране и осигуряване на качеството са интегрирани с този по анализ. Ролята, която отговаря за идентифициране и анализ на изискванията свързани с архитектурата, технологията и реализацията е системния архитект. В рамките на анализа се определя дали дадено изискване е в обхват или не. В някои проекти експерта по качеството има само консултираща роля, но в други участва и в процеса по определяне на изискванията свързани с качеството. Изискванията към качеството са ключов момент, който не трябва да бъде пропускан при анализа. Те са свързани със сценариите на ползване. Всички критерии за приемане трябва да бъдат докладвани като изисквания, тъй като в последствие трябва да бъдат реализирани и да бъде осъществена съответната подготовка за тестване и верифициране в края на проекта. Ако специфичните изискванията касаещи тестването и приемането не бъдат определени от Възложителя на този етап, по късно те няма да могат да бъдат проведени, поради отсъствието на съответната подготовка.

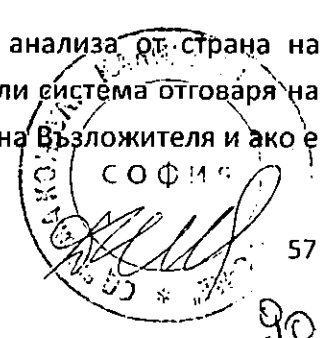
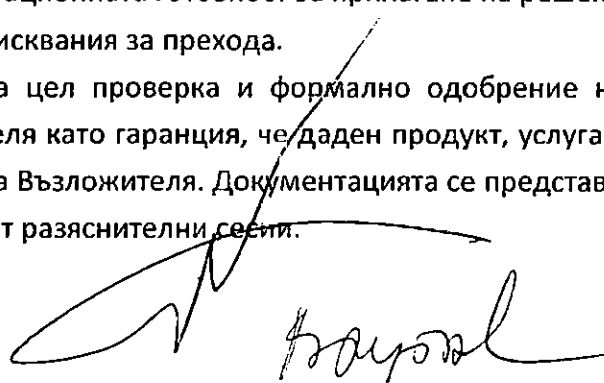
Валидиране на изискванията

Процесът по валидиране на анализа ще бъде базиран на зоната на знание от BABOK „Оценка и валидиране на решението“.

Зоната на знание от BABOK „Оценка и валидиране на решението“ включва:

- Оценка на предложеното решение за това до каква степен отговаря на нуждите;
- Алокиране на изисквания към решението;
- Оценка на организационната готовност за прилагане на решението;
- Определяне на изисквания за прехода.

Валидирането има за цел проверка и формално одобрение на анализа от страна на Възложителя и се определя като гаранция, че даден продукт, услуга или система отговаря на нуждите и очакванията на Възложителя. Документацията се представя на Възложителя и ако е необходимо се провеждат разяснителни сесии.



Със Sparx Enterprise Architect може да се създават и модели за стратегическо моделиране като Мисловни карти, Организационно дърво, Верига на стойността, Стратегическа карта, Balanced scorecard и др.

Продуктът поддържа и моделиране на работните процеси с BPMN, както и организация и управление на изискванията, възможности за прототипиране на потребителски интерфейси и бяла дъска.

Продуктът може да генерира документация на базата на поддържаните в него модели.

Управление на изискванията

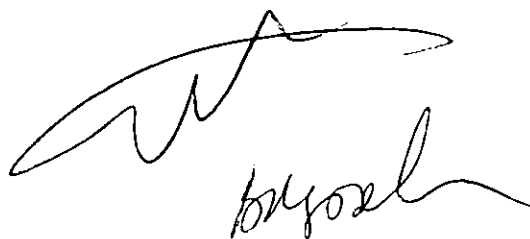
Това е процес по проследяване и мониторинг на изискванията и се състои от дейности, които гарантират, че изискванията са одобрени и се управляват през целия жизнен цикъл на проекта. По време на проследяване и мониторинг се изготвят матрица за проследимост на изискванията (traceability matrix) и свързаните с тях атрибути. Най-често използваните атрибути на изисквания са:

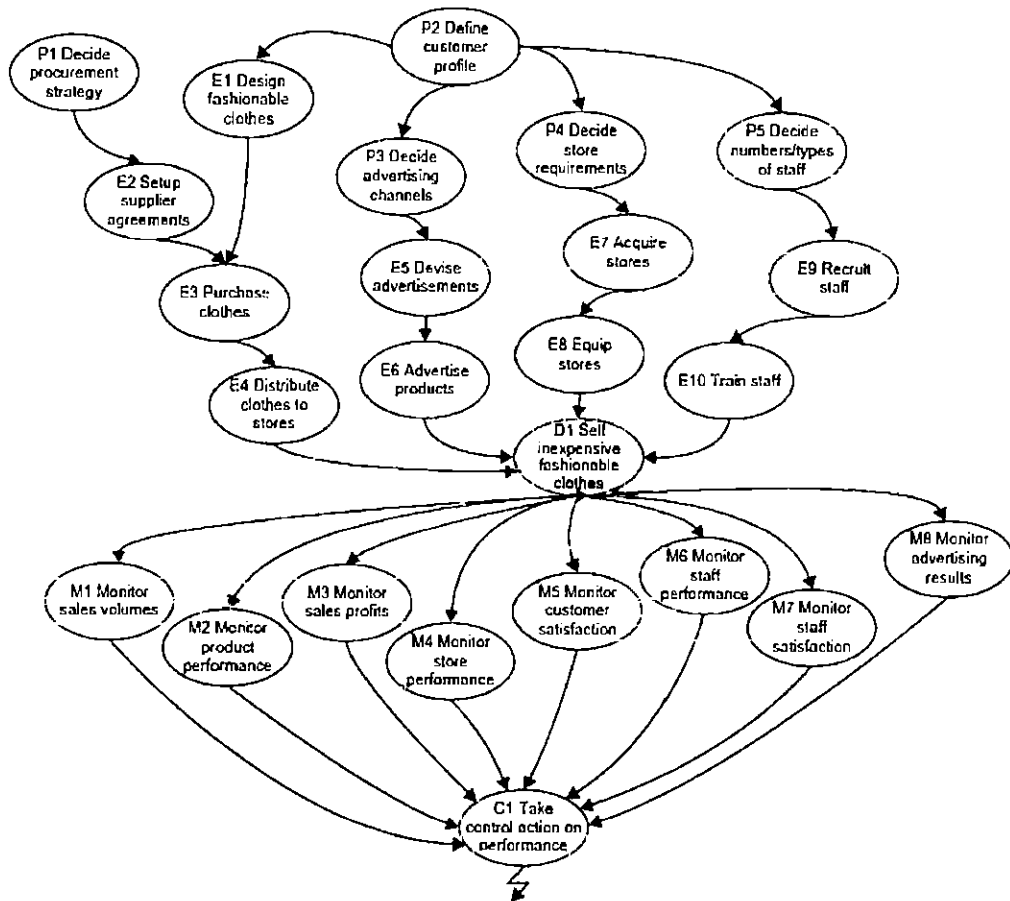
- Идентификатор на изискването (Requirement ID);
- Наименование на изискването;
- Кратко текстово описание (summary);
- Източник;
- Статус;
- Приоритет;
- Връзка с други изисквания;

Допълнително атрибутите могат да бъдат разширени с:

- Обосновка защо изискването е важно;
- Версия на изискването;
- Дата на завършване;
- Стабилност;
- Сложност;
- Критерии за приемане на функционалността (Acceptance criteria).

Матрицата за проследимост и поддържането на атрибути на изискванията подпомагат следенето и контрол на обхвата на проекта. Създава се и се поддържа основна версия на одобрените изисквания (baseline). При поява на нови изисквания или необходимост от промени по вече одобрени изисквания, те се документират, добавят към матрицата за проследяване, оценяват за въздействия върху проекта и продукта, и представят за одобрение на заинтересованите страни. Процесът включва проследяване и представяне на състоянието на всички изисквания, като се използват методи на комуникация, определени и одобрени в рамките на плана за бизнес анализ.

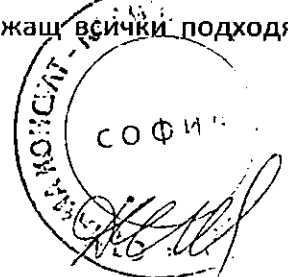




MoSCoW приоритизация - в рамките на анализа могат да се открият много изисквания, несъответствия и потенциални възможности, но може да се окаже, че не всички от тях ще могат да бъдат част от решението. В такива случаи ще се ползва подхода за приоритизация MoSCoW, базиран на следното:

- **Must have** (задължително) – изисквания, които са фундаментални и задължително следва да се реализират колкото се може по скоро;
- **Should have** (трябва да ги има) – изисквания, които са важни и трябва да се включат, но не задължително на първи етап;
- **Could have** (може да ги има) – изисквания, които могат да се оставят на заден план и ако има ресурси, да се изпълнят в последващ етап;
- **Won't have (but Would like in future)** (искат се) – неща, които са интересни, но не са във фокуса.

За подпомагане на моделирането, документирането и управлението на изискванията може да се използва специфичен софтуер. Пример за такъв инструмент е Sparx Systems Enterprise Architect - водещ продукт за моделиране с UML, поддържащ всички подходящи UML модели, ориентиран към Enterprise Architecture.



Handwritten signature

експертите и т.н. Обикновено като нотация за моделиране на работни процеси се използва BPMN (Business Process Model and Notation) или UML (Unified Modelling Language).

• UML моделиране - за описание на съществуващите системи може да се наложи тяхното моделиране с използване на UML за следните модели:

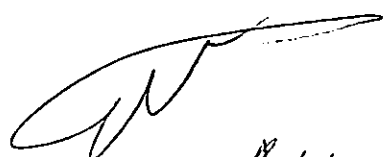
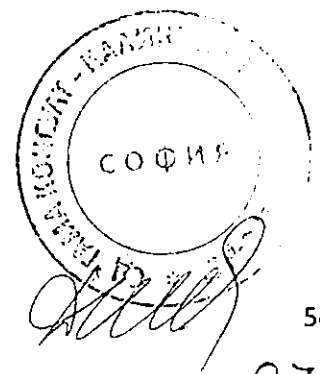

- Диаграми на разгръщане и диаграми на компоненти;
- Случаи на ползване - за критични случаи на ползване, съответно с диаграми на дейностите за сценариите на случаите на ползване;
- Домейн Модел – с използване на диаграми на класове;
- Логическо моделиране на критични данни – с използване на диаграми на класове;
- Диаграми на последователност – за описание на критични случаи на взаимодействие при интеграция между системни компоненти;
- Диаграми на състояния – за описание на важни аспекти на състоянията на една система или системен обект.

Тъй като UML е широко разпространен език за моделиране, няма да се спираме на описание на тези модели.

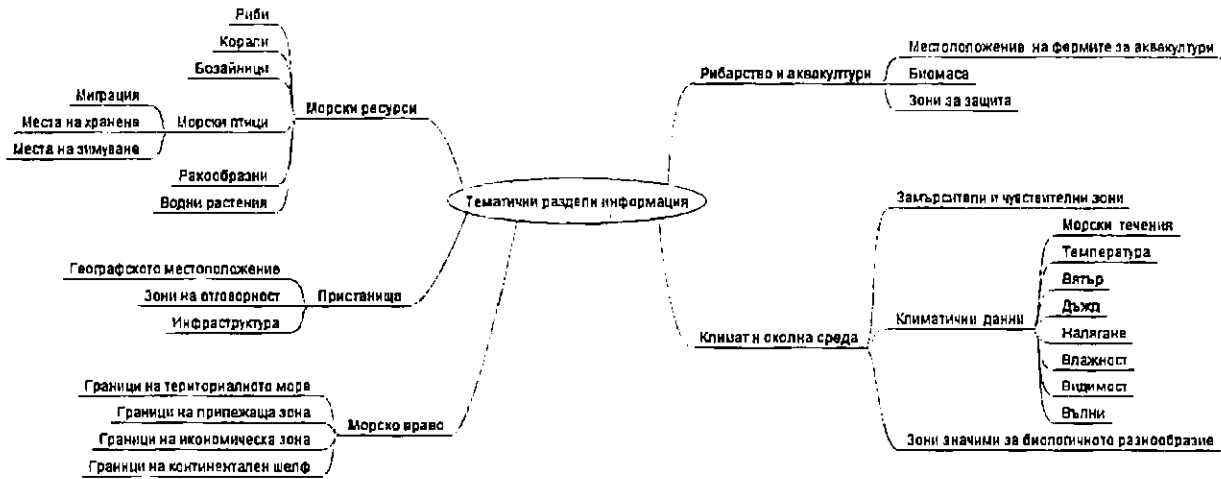
• Моделиране на бизнес дейности (Business activity modelling) - техника на моделиране в рамките на анализа, която представя на високо ниво бизнес дейности, които се изпълняват в една организация, и на логическите зависимости между тях. В този модел дейностите се класифицират по отделни видове - например планиращи, способстващи, извършващи, мониториращи и контролиращи.

Обикновено се ползва за да представи гледните точки на различни заинтересовани страни за развитието на бизнеса, за анализ на несъответствията (gap analysis) между тях и за откриване на потенциални възможности.

Пример за модел на бизнес дейности:

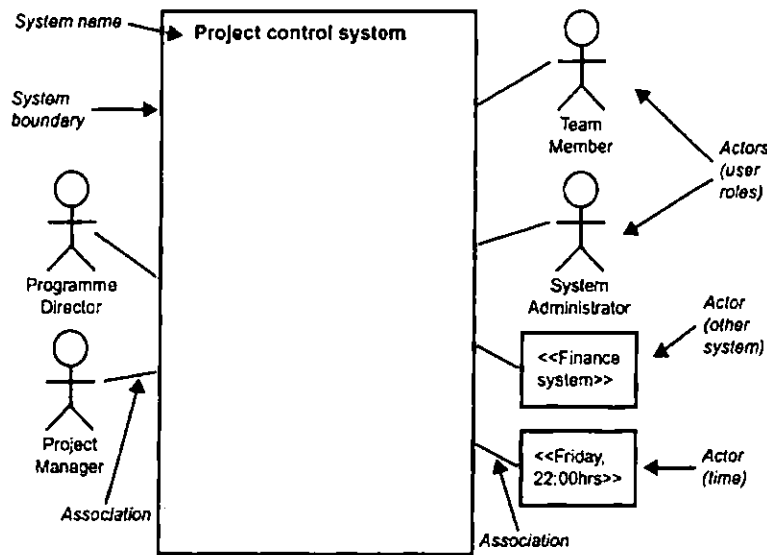



Пример за мисловна карта:



- Контекст диаграми - използват се за описание на отношенията на система с околния свят: кои са потребителите, които я ползват и кои системи обменят данни с нея. Тази техника се използва за документирание на изследваните системи на високо ниво. Подходяща е и при работни срещи.

Пример за контекст диаграма:



- Анализ на бизнес събития - занимава се с разглеждането на бизнес система или сфера на дейност, с цел да се идентифицират събитията, с които организацията трябва да се справя. Бизнес събитията помагат да се изяснят различните компоненти на работата, извършена в рамките на дадена област, и помагат да се идентифицират различните бизнес процеси, които трябва да се организират.

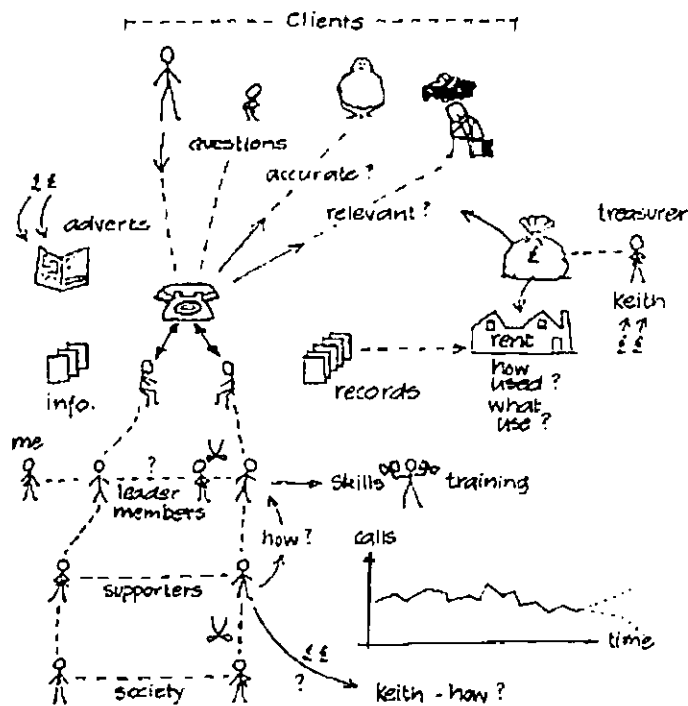
- Моделиране на бизнес процеси - широко разпространена техника за анализ, откриване на тесните места и подобряване на бизнес процесите. Текущите процеси в една организация могат да бъдат извлечени от всеки от елементите на анализ – например от документацията на текуща система, от нормативната уредба, от интервютата, проведени с

Handwritten signatures and a circular stamp are present at the bottom of the page. The stamp contains the text 'СЪСТАВИТЕЛ' and '2015'. The page number '53' is visible in the bottom right corner, along with the handwritten number '86'.

Инструменти за реализация на дейността

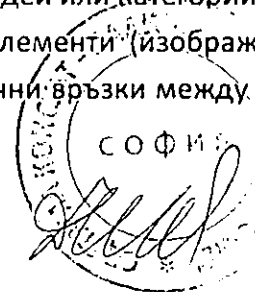
- **Богати картини (Rich pictures)** - Това е техника на документиране, популяризирана от методиката Soft System Methodology на проф. Питър Чекланд и неговите сътрудници. Идеята е в изобразителна форма да се уловят основните елементи на бизнес проблем или да се улесни цялостното представяне и анализ на определена ситуация. Няма правила за това какво може или не може да бъде обхванато в богатата картина, нито за това, какви символи да се ползват. Това е техника с много свободен формат. Дава възможност да се представят организацията и основните актьори в нея, виждания и идеи, процеси, основни проблеми, култура и климат и т.н.

Пример за богата картина:



Обикновено богати картини са основен елемент за визуално документиране на първоначалните елементи от анализа, като заедно с това да бъдат документираните основните вътрешни проблеми и да се подчертават потенциалните възможности за развитие (например от добри практики), както и първоначални бележки за несъответствията с други елементи на анализа.

- **Мисловни карти (Mind Maps)** - техника на документиране, която представлява диаграма, използвана за визуално представяне на информация (идеи, задачи, цели), свързана чрез ключова дума или идея. Могат да се използват за генериране, структуриране и нагледно представяне на идеи, както и като помощно средство за организиране на информация, разрешаване на конкретен проблем или вземане на решения. Основани са на използването на една централна идея, от която се разклоняват асоциираните с нея идеи или категории, като всяка отделна идея е представена чрез ключова дума и визуални елементи (изображения, цветове, схеми, форми). Отделните категории представляват семантични връзки между части от информацията.



- Основа (baseline) за валидиране на нуждите на заинтересованите страни;
- Основен документ, който се използва от проектантския екип, разработчиците, тестерите, и екипа по осигуряване на качеството;
- Основа за потребителски ръководства и друга документация;
- Отправна точка за оценка на решението;
- Възможност за употреба от други проектни екипи, които се нуждаят от разбиране на детайлите по проекта, докато той е в процес на изпълнение или след завършване на изпълнението му;

Въпреки важноста на документиране на изискванията, трябва да се имат предвид няколко фактора относно нея:

- Документацията е само една от няколко техники за осигуряване на консенсус между всички заинтересовани страни по отношение на поведението на системата/решението.
- Документацията не трябва да замества комуникацията и сътрудничеството.

В рамките на процеса по документиране на изискванията се включват следните дейности:

- Описание на бизнес целите и целите на проекта;
- Изготвяне на речник на използваните термини;
- Описание на функционалните изисквания, описващи бизнес процеса, информацията и взаимодействията с продукта, или тези от тях, които са необходими;
- Описание на нефункционалните изисквания като ниво на услугата, производителност, сигурност, съвместимост със стандарти и административни изисквания, възможности за поддръжка, и др.;
- Изисквания към качеството/ обучението/ поддръжката/ тестването;
- Дефиниране на бизнес правилата;
- Дефиниране на предположенията и ограниченията;
- Приоритизиране на изискванията.

Техническата част на спецификацията трябва да включва различните аспекти на архитектурата на приложението, което ще бъде реализирано. Това означава, че трябва да бъдат анализирани и документиращи аспекти като:

- Сигурност;
- Технология;
- Интерфейси;
- Миграция;
- Стандарти;
- Производителност
- Скалируемост.


Юлия



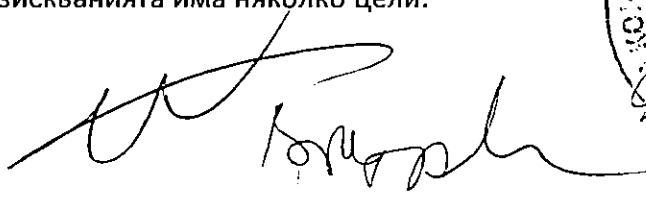
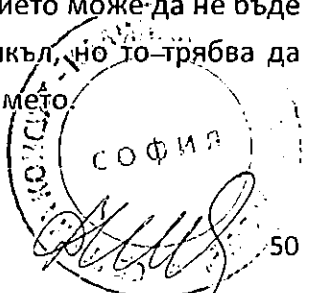
- 1) **Планиране (Plan)** - използва се подхода 5W 1H, при който трябва да дадат отговори на следните шест въпроса:
 - какво? (what?) – дефиниране на целите и задачите
 - защо? (why?) – обосноваване на необходимостта;
 - кой? (who?) – определяне на отговорностите;
 - как? (how?) – определяне на метода (методите) на извършване;
 - къде? (where?) – ограничаване на полето на действие;
 - кога? (when?) – разработване на план (график).
- 2) **Изпълнение (Do)** - в началото на този етап се извършва необходимото обучение и квалификация на персонала, който ще извършва реалното изпълнение на планираното. След обучението (или паралелно с него) персоналот изпълнява планираното.
- 3) **Проверка (Check)** - на този етап се проверява как е изпълнено планираното действие и се оценяват постиженията.
 - **Изучаване (Study)** - използва се при PDSA цикъла - изучаване и анализ на постигнатите резултати, с цел уточняване на по-нататъшните действия.
- 4) **Действие (Act)** - в зависимост от резултатите на проверката на този етап са възможни два вида действия:
 - Въвеждане на постигнатото по-високо ниво като нов стандарт (формализиране) в организацията;
 - Извършване на коригиращи и/или превантивни действия за подобряване на постигнатото и за постигане на още по-високо ниво в следващия цикъл.

Документиране на изискванията

Документиране на анализа се базира на няколко от зоните на знание от BABOK като по-долу са описани често използваните инструменти за реализация на дейността (извличане, моделиране и документиране на изискванията).

След анализа на информацията, която е била извлечена на предходния етап, и изготвянето на необходимите модели, бизнес анализаторът документира получените резултати във форма, подходяща за организацията, нуждите на проекта, и жизнения цикъл на проекта. Изготвя се документ (функционална и техническа спецификация), който описва изискванията и обхвата на решението на бизнес проблема или възможността. Бизнес анализаторът подготвя пакет с изискванията така, че Възложителят и екипът, който ще разработва решението, да уеднаквят разбирането си какво решение трябва да се разработи. Документацията може да бъде произведена с различно ниво на формалност и в различна форма, което често зависи от избрания жизнен цикъл на проекта. Решението може да не бъде цялостно решение, както е в случая на проект с адаптивен жизнен цикъл, но то трябва да представлява описание на наличната информация в този момент във времето.

Документирането на изискванията има няколко цели:



50
83

- Речник (Data Dictionary)
- Таблица/ диаграма на състоянията (State Table and State Diagram)
- Модели на потребителския интерфейс (Interface Models)
 - Скици (Wireframes)
 - Прототипи (Prototypes)

Проектиране като част от процеса по анализ и моделиране на изискванията

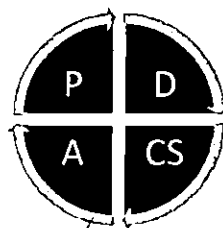
При проектирането на процесите и при разработването на ориентирана към процесите система може да се използва и популярния цикъл на Деминг (цикъл на непрекъснато усъвършенстване). Важно условие за използването на този цикъл е използването на подходящи методи за анализ на представянето на процесите. Такива методи например могат да бъдат:

- 1) Съпоставка с „идеален“ процес. Цел: да се открият различията между действащата практика и определения като идеален вариант.
- 2) Проиграване на сценарии тип „какво ще се случи, ако...“. Цел: да се разработят няколко варианта за осъществяване на дадена дейност и да се избере най-добрия.
- 3) Съпоставка с предварително дефинирани ключови параметри на процеса. Цел: да се открият пропуски или възможности за подобрене при осъществяването на процеса.
- 4) Съпоставка между два и повече процеса в организацията. Цел: да се открият общи за организацията силни или слаби страни и да се търси начин на пренасяне на успешните практики от един процес към друг.



Цикъл на Деминг

Цикълът на Деминг показва пътя към подобрения. Той стои в основата на много от съвременните концепции за развитие на бизнеса. Известен е още като "Цикъл на Шухарт", "PDCA цикъл", "PDSA цикъл" или "SDCA цикъл". Деминг развива идеите на Шухарт, като дава по-обща названия на всеки от етапите и добавя още един етап за анализ на получената на втория етап информация.



PDCA (PDSA) цикъл



Анализ, моделиране и изчистване на изискванията

Всички усилия по време на този процес са фокусирани върху, извлечената по време на предходната фаза информацията, която е свързана с изискванията. Анализът е процеса на изучаване, детайлизиране и синтезиране на информацията с цел по-нататъшното ѝ добро разбиране, допълване и подобрене.

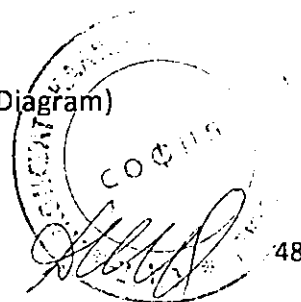
Процесът по анализ на изискванията е изцяло експертна работа. Анализът води до внимателно разглеждане на частите информация и как те се отнасят една към друга. Анализът включва прогресивна и итеративна работа чрез детайлизиране на информацията и често води до генериране на абстрактни модели, които да предоставят общ поглед върху организацията/процесите/системите/проекта.

Изискванията се дефинират така че да бъдат SMART (Specific/ Measurable/ Appropriate, Achievable/ Realistic, Relevant/ Timely). От етап анализ процесът може да се върне в извличане, за да бъде набавена дадена информация или в определяне на обхвата, за да бъде разширен или стеснен обхвата през формалния процес по промяна. Анализът върви леко назад във времето спрямо извличането и идентифицирането на изискванията.

В зависимост от нуждите и конкретния проект бизнес анализаторите изготвят модели. Моделът подпомага възприемането на информация чрез визуалното ѝ представяне, както абстрактно и конкретно. Моделите са диаграми, таблици, или структуриран текст. Моделите могат да бъдат създадени с различни инструменти, като се започне от инструментите формалното моделиране и се стигне до скициране на лист хартия или бяла дъска.

В зависимост от целта на модела могат да се използват различни техники за моделиране на изисквания, най-популярните от които са следните:

- **Модели на обхвата (Scope Models)**
 - Целеви модел и модел на бизнес целите (Goal Model and Business Objective Model)
 - Мисловна карта (Mind Map)
 - Контекстна диаграма (Context Diagram)
 - Диаграма на случаите на употреба (Use Case Diagram)
- **Процесни модели (Process Models)**
 - Диаграма на дейностите (Process Flow Diagram/Activity Diagram)
 - Случай на употреба (Use Case)
 - Потребителски случай (User Story)
- **Модели на правила (Rule Models)**
 - Каталог на бизнес правилата (Business Rules Catalog)
 - Дърво за вземане на решение (Decision Tree)
- **Модели на данни (Data Models)**
 - Диаграма на връзките между обектите (Entity Relationship Diagram)
 - Диаграма на протичане на данните (Data Flow Diagrams)

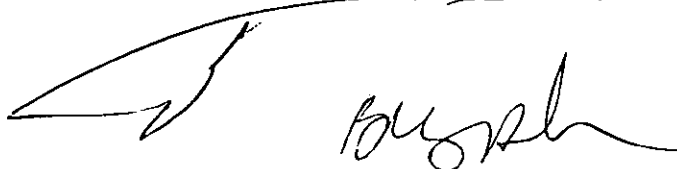


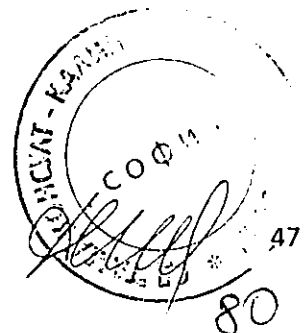
- Колективно обсъждане на проблем/ генериране и изчистване на идеи (brainstorming);
- Документален анализ - средство за извличане на информация чрез изучаване на налична документация. Документалният анализ може да включва анализ на бизнес планове, пазарни проучвания, договори, искания за представяне на предложения, становища, докладни записки, съществуващите насоки, процедури, наръчници за обучение, публикувани сравнителни коментари, доклади за проблеми и съществуващи системни спецификации. Документалният анализ включва и стратегически, концептуални и планиращи документи, както и запознаване с нормативната уредба.
- Работни срещи/ Фокус групи - по същество са събиране на група от заинтересовани страни в проект за целите на:
 - съгласуване посоката и обхвата на проекта;
 - идентифициране и съгласуване на бизнес и/или системни изисквания;
 - разглеждане на възможните решения на изискванията;
 - разглеждане и одобряване на продуктите за анализ, например
 - изисквания каталог и спецификация на изискванията.

В настоящия проект работни срещи ще се използват за:

- допълване на информацията, получена от документалния анализ;
- сравнение на архитектурните модели на всеки от елементите на анализ и обсъждане на предварително идентифицираните от Изпълнителя проблемни области, несъответствия, възможности;
- Интервюиране - една от основните техники за установяване на фактите, изследване или извличане, използвани от бизнес анализаторите. Обичайно се състои с срещи очи в очи между анализатора и представител на заинтересованите страни. В конкретния проект ще се използва за допълване на информацията, получена от документалния анализ.
- Наблюдение - форма на извличане, при която анализаторът наблюдава как се извършва дейността. Има няколко форми като структурирано наблюдение, следване като сянка, протокол анализ и др.
Целта е да се наблюдава последователността от действия, която бизнес потребител извършва (със система или в рамките на бизнес процес).
В настоящия проект може да се наложи да бъде използвана като съпътстваща техника при анализ на конкретен бизнес процес, за който не могат да бъдат открити документи или за който има противоречива информация.
- Прототипиране
- Въпросници и проучвания

- Документират се резултатите от извличането;
- Резултатите от извличането се проверяват и потвърждават.





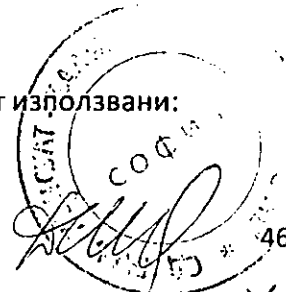
- определяне на заинтересованите страни (stakeholders), с които ще се работи по време на анализа;
- определяне на подходящото ниво на детайлност;
- определяне на необходимите ресурси от страна на Възложителя (технически, човешки и др.), необходими за реализацията;
- дефиниране на процес за приоритизиране на изискванията;
- дефиниране на процес за проследимост на изискванията;
- дефиниране на подход за комуникация;
- дефиниране на процес за вземане на решения;
- дефиниране на процес на верифициране и валидиране на изискванията;
- дефиниране на процес за управление на промените;
- планиране на работата по бизнес анализа:
 - идентифициране на отчетни резултати (deliverables);
 - определяне на задачи и дейности: планират се аналитичните дейности, използвани техники и средства, начин на моделиране и документиране, ;
 - определяне на времева рамка и последователност на задачите;
 - определяне на роли и отговорности;
 - идентифициране на източниците на информация;
 - определяне как ще се извършва управлението на екипа за анализ;
 - оценка на напредъка на работата.

Извличане на изискванията

Извличането на изискванията ще бъде базирано на зона на знание „Извличане на изискванията“. Интегриран с този процес е и процесът по определяне на обхвата, който започва от най-общите изискванията на заинтересованите лица, и завършва с изготвяне на детайлно функционално и техническо описание на проекта и продукта. Подготовката на документите, специфициращи обхвата, е критична дейност за успеха на проектите за разработка на софтуер. Определящи за обхвата са главните компоненти, които трябва да бъдат доставени, както и наложените ограничения и направените предположенията. Тези компоненти се определят при иницирането на проекта и могат да бъдат намерени в договора, техническото задание или отделен документ, предоставен от Възложителя. По време на анализа обхватът се специфицира и описва в по-голяма детайлност, като не може да излиза от границите определени в инициращите документи.

В рамките на процеса по извличане на изискванията се включват следните дейности:

- Определяне на източниците на информация;
- Определяне на заинтересованите страни;
- Техники за извличане на изискванията, които е възможно да бъдат използвани:



46
79

как тези решения отговарят на първоначалните изисквания, така че Възложителя да може да оцени изпълнението и ефективността на предложеното решение.

Ключови компетенции

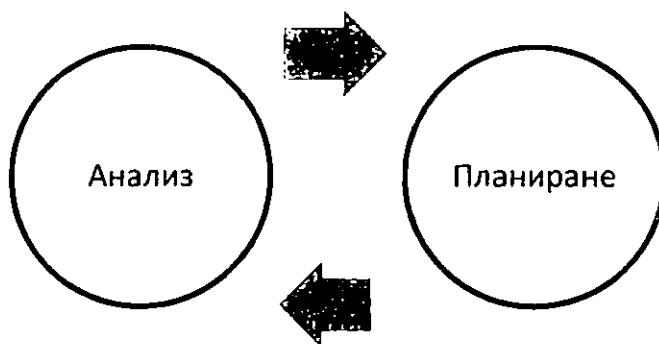
Ключови компетенции е част от бизнес анализа, която описва дейности, знания, както и други характеристики, които подпомагат ефективното изпълнение на бизнес анализа.

5.1.2. Описание на процесите по бизнес анализ

Планиране на анализа

Планирането на анализа ще бъде базирано на зона на знание „Планиране и наблюдение на бизнес анализа“.

Планирането и анализът са част от всяка итерация, но в началото на проекта трябва да им бъде отделено най-голямо внимание. Планирането и анализът като дейности са взаимосвързани, тъй като планирането изисква детайлен анализ, и същевременно може да породят допълнителни изисквания (обикновено това са ограничения), които да повлияят на спецификациите и прототипите.

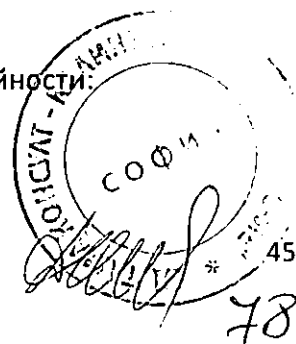


Обвързаност на анализа и планирането

В хода на изпълнение на проекта общия обем на извършвания анализ и планиране в рамките на итерациите намалява постепенно. Въпреки това необходимост от допълнителен анализ възниква, когато по време на разработката се откриват пропуски или ограничения налагащи промени в изискванията или Възложителя промени своите изисквания. Тези промени се управляват с формалния процес по управление на промените и налагат извършване на допълнителен анализ и допълнително планиране. Те могат да бъдат утежнени поради факта, че е възможно вече да има реализирана функционалност, която да трябва да бъде променена или такава, която е в процес на разработка в момента, чиято промяна да повиши възможността за възникване на програмни грешки. Това може да доведе до различни вариации, както от гледна точка на функциониране на системата, така и от гледна точка на управление и вложени ресурси.

В рамките на процеса по планиране на анализа се включват следните дейности:

- определяне и разбиране на контекста на проекта;
- уточняване и ясно дефиниране на обхвата на разработката;



След като извлечените изисквания се документират те трябва да се съгласуват с ръководството на организацията и с ключовите лица за отделните процеси. Практиката показва, че е по-удачно съгласуването да стане на самостоятелни срещи с всеки служител или мениджър, от чието потвърждение се нуждае екипът. Много е важно на този етап още веднъж да потвърдим, че описанието на процесите, което сме направили отразява реалните действия в организацията.

Анализ на организацията

Процесът по анализ на организацията (Enterprise Analysis) е свързан с това, как бизнес анализаторите идентифицират бизнес нуждите на организацията, усъвършенстват и изясняват определението на тази бизнес необходимост. Тази област от бизнес анализ описва дефинирането на проблеми и анализи, разработване на бизнес случаи, предварителни проектни проучвания, и определянето на обхвата.

За да функционира ефикасно, дадена организация трябва да идентифицира и управлява множество взаимосвързани дейности. Всяка дейност, която използва ресурси и се управлява, за да бъде възможно превръщането на входните елементи в изходни елементи, може да бъде разглеждана като процес. Често изходните елементи на един процес се явяват входни елементи за следващия процес. Прилагането на система от процеси в една организация, както и идентифицирането и взаимодействието на тези процеси и тяхното управление означава "подход базиран на процес".

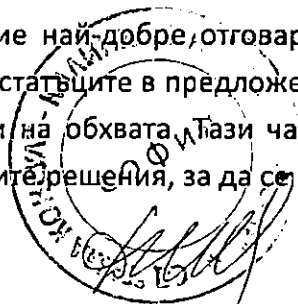
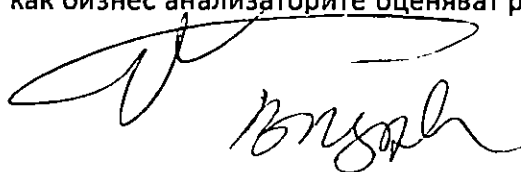
Едно от първите неща, с които започва екипът, ангажиран с анализ на организацията, е определяне на процесите, които протичат в организацията. На базата на информацията за тези процеси се извършва така наречения анализ на състоянието, чрез който се прави преглед на степента на съответствие на действащата система спрямо изискванията на нормативната база и изискванията на международния стандарт, които организацията е решила да внедрява. Анализът на състоянието е и в основата на планирането на необходимите промени, които следва да се въведат в организацията, за да се отговори напълно на изискванията.

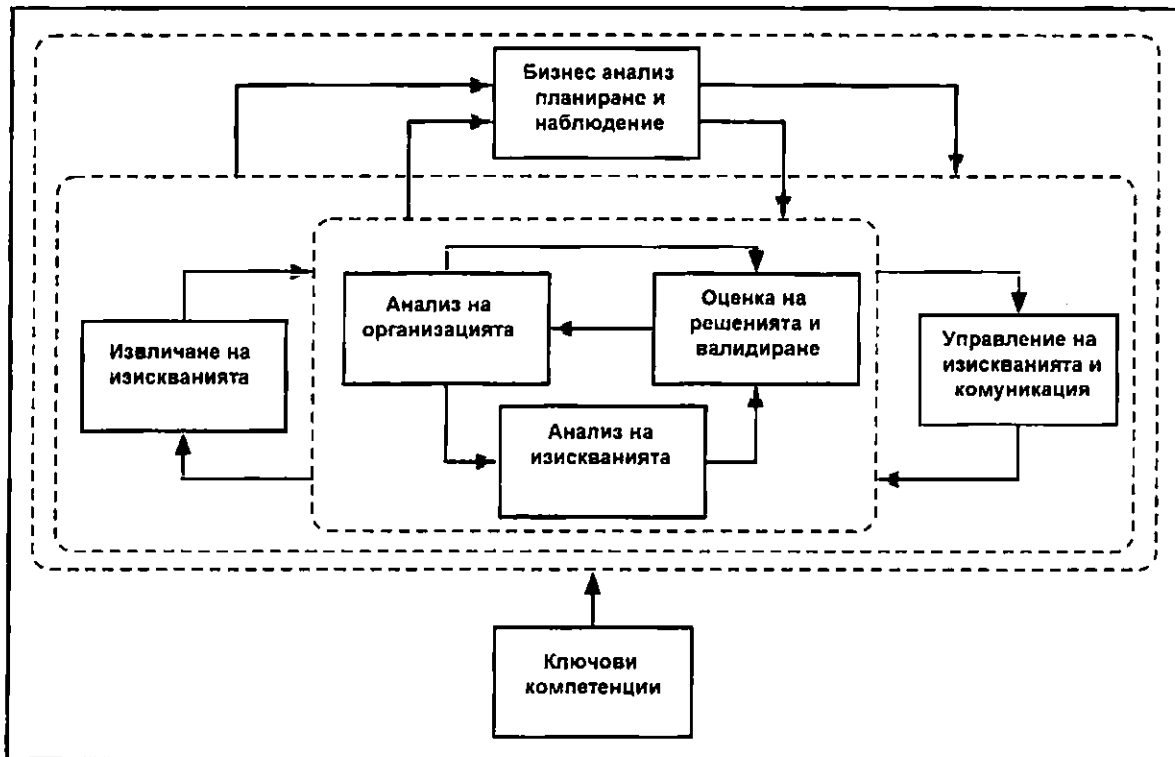
Анализ на изискванията

Процесът по Анализ на изискванията включва приоритизиране и постепенно разработване на изискванията с цел да се даде възможност на екипа на проекта да осмисли бизнес необходимостта, да идентифицира възможните решения и да избере за изпълнение решение, което ще отговори на нуждите на спонсориращата организация и заинтересованите страни. Това включва анализ на заинтересованите страни, които трябва да вземат решения, отговарящи на тези нужди, оценка на текущото състояние, идентифициране и препоръки за подобрения, както и проверка и валидиране на получените чрез тях изисквания.

Оценка на решенията и валидиране

Оценка на решенията и валидиране описва начина, по който бизнес анализаторите оценяват предложените решения. За да се определи кое решение най-добре отговаря на бизнес изискванията трябва да се идентифицират пропуските и недостатъците в предложените решения, и да определят необходимите отклонения или промени на обхвата. Тази част от анализа също описва как бизнес анализаторите оценяват разгърнатите решения, за да се види





Връзки между процесите в бизнес анализа

Планиране и наблюдение на бизнес анализа

Планиране и наблюдение е частта от бизнес анализа, която обхваща дейностите, свързани с определяне на това, кои дейности са необходими, за да се изпълни успешно бизнес анализа. Тя обхваща идентифициране на заинтересованите страни; избор на техники за бизнес анализ; процес, който ще се използва за управление на изискванията; как да се направи оценка на напредъка на работата. Извършените дейности в тази фаза на анализа управляват изпълнението на всички други задачи за бизнес анализ.

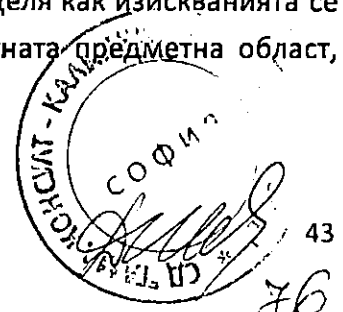
Извличане на изискванията

В рамките на този процес се определя и описва начина, по който бизнес анализаторите работят със заинтересованите страни, за да се идентифицират и да се разберат техните нужди и изисквания, и да се оцени бизнес средата, в която те работят. Целта на извличане на изисквания е да се гарантира, че действителните основните нужди на заинтересованите страни се разбират, а също така и да се определят основните и второстепенните изисквания.

Управление на изискванията и комуникация

Процесът по управление на изискванията и комуникацията описва как бизнес анализаторите управляват конфликти, проблеми и промени, за да гарантират, че заинтересованите страни и екипа на проекта са напълно наясно за обхвата на проекта и са съгласни с възникналите промени. Тази част от бизнес анализа определя как изискванията се представят на заинтересованите страни, и как знанията за конкретната предметна област, придобити от бизнес анализатора се запазват за бъдеща употреба.

Вършец



За основен приоритет на дейността си, членовете на организацията поставят стандартизирането и развитието на бизнес анализа като значима и престижна дейност в България, както и за подобряване на качеството на анализа.

Анализаторите трябва да познават добре отделните дисциплини (зони на знание) на областта на бизнес анализа, за да са в състояние да разберат задачите, бизнес процесите и да опишат изискванията. Анализаторите могат да извършват анализ използвайки препоръките на дисциплини последователно, итеративно, или едновременно. Отделните дисциплини могат да бъдат прилагани в произволен ред в зависимост от конкретните особености на проекта и бизнес процесите, които се анализират. Извършването на бизнес анализа може да започне с всеки един процес от коя да е дисциплина. Дисциплините не са на фази на изпълнение на проекта. Възможно и допустимо е анализа да започне с анализ на дейностите в организацията и да продължи с работа по анализ на изискванията, оценка на възможните решения и валидация на решението, което е избрано и предложено. Международния стандарт за бизнес анализ не поставя изисквания към самата последователност и това трябва да бъде решено индивидуално от бизнес анализатора.

При итеративната разработка процесът по изготвяне на функционална и техническа спецификация (основен резултат от бизнес анализа) се изпълняват като няколко под-процеса. Това са процесите Извличане, Анализ, Моделиране, Документиране, Верифициране и Валидиране. Тези процеси се изпълняват паралелно и итеративно, като от гледна точка на нарастването на обхвата им във времето те са и спираловидни.

Извличане

Анализ и моделиране

Документиране

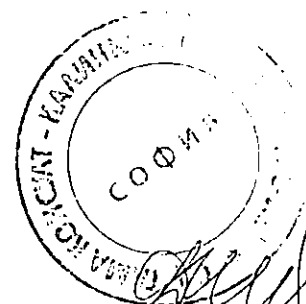
Верифициране

Валидиране

Процес по изготвяне на функционална и техническа спецификация

5.1.1. Основни области на знание (процеси)

Подходът за анализ на изискванията, използван при Кандидата се придържа към Business Analysis Body of Knowledge (BABOK). BABOK дефинира стандарти и добри практики в областта на бизнес анализа и представя основните процеси, наречени зони на знание, и съответните връзки между тях

5. Описание на подход и методи за ключови дейности от изпълнението на проекта

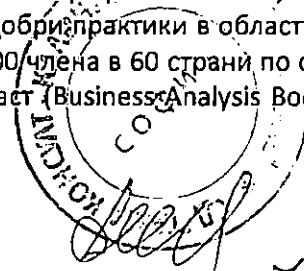
5.1. Подход за бизнес анализ

- Методологии и добри практики: IIBA, Business Analysis Body of Knowledge
International Requirements Engineering Board
- Основни средства и техники: Документен анализ;
Интервюиране;
Експертно мнение;
Работни срещи/Фокус групи;
Цикъл на Деминг;
UML моделиране;
Формален преглед на изискванията
- Софтуер за моделиране, документиране и управление на изискванията: Microsoft Office & SharePoint Portal и Sparx Systems Enterprise Architect
- Източници: Сайтове и документация:
- International Institute of Business Analysis;
 - International Requirements Engineering Board;
 - www.sparxsystems.com.au
- Обосновка: Ще бъдат приложени утвърдени практики на водещи световни професионални организации в областта на бизнес анализа.

Извършването на бизнес анализа ще премине през следните процеси:

- 1) Планиране на анализа.
- 2) Извличане на изискванията.
- 3) Анализ, моделиране и изчистване на изискванията.
- 4) Документиране на изискванията.
- 5) Управление на изискванията.
- 6) Верифициране на изискванията.
- 7) Валидиране на изискванията.
- 8) Одобрение на изискванията.

Кандидатът прилага препоръките на международния институт за бизнес анализ (IIBA), в чиито стандарт тези процеси са прецизирани. Международният институт по бизнес анализ е най-престижната професионална организация на бизнес анализатори в света. В нея членуват професионалисти, които работят за установяването и поддържането на общоприети стандарти и добри практики в областта на бизнес анализа. Създаден през 2003 г. в Канада, институтът има над 10 000 члена в 60 страни по света и създава най-пълната колекция от знания и добри практики в тази област (Business Analysis Body of Knowledge - BABOK).



ID	Задача	Продълж.	Начало	Край	Годишен график																
					Mar	Qtr 2, 2018 Apr	May	Jun	Qtr 3, 2018 Jul	Aug	Sep	Qtr 4, 2018 Oct	Nov	Dec	Qtr 1, 2019 Jan	Feb	Mar				
67	Провеждане на тестване	5 days	28.07.2018	01.08.2018																	
68	Провеждане на тестване на софтуерното решение	3 days	28.07.2018	30.07.2018																	
69	Отстраняване на евентуални дефекти, открити при тестването	5 days	28.07.2018	01.08.2018																	
70	Актуализиран софтуер за национален адаптор за EUCISE2020	0 days	01.08.2018	01.08.2018																	
71	Актуализиран програмен код на софтуер за национален адаптор за EUCISE2020	0 days	01.08.2018	01.08.2018																	
72	Актуализирана документация на системата	0 days	01.08.2018	01.08.2018																	
73	Дейност 2: Тестване и съпровождане на адаптера	185 days	02.08.2018	02.02.2019																	
74	Етап 3: Верификация на EUCISE2020 адаптор и съпровождане по време на тестовата обмяна на данни	185 days	02.08.2018	02.02.2019																	
75	Обучение	4 days	02.08.2018	05.08.2018																	
76	Подготовка за обучение	2 days	02.08.2018	03.08.2018																	
77	Провеждане на обучение за ключови потребители	1 day	04.08.2018	04.08.2018																	
78	Провеждане на обучение за администратори	1 day	05.08.2018	05.08.2018																	
79	Внедряване и съпровождане	185 days	02.08.2018	02.02.2019																	
80	Инсталиране и конфигуриране на продукционна инстанция на адаптера	3 days	02.08.2018	04.08.2018																	
81	Тестване	180 days	06.08.2018	01.02.2019																	
82	Провеждане на тестове за интеграция на националната система (VTMIS) чрез националния EUCISE2020 адаптор	180 days	06.08.2018	01.02.2019																	
83	Провеждане на тестове за интеграция на Националния EUCISE2020 адаптор с EUCISE2020 нодовете	180 days	06.08.2018	01.02.2019																	
84	Отстраняване на евентуални недостатъци на Националния EUCISE2020 адаптор	170 days	16.08.2018	01.02.2019																	
85	Отстраняване на евентуални недостатъци в спецификациите и др. Документи по проекта	170 days	16.08.2018	01.02.2019																	
86	Протоколи за изпълнени тестове	0 days	01.02.2019	01.02.2019																	
87	Актуализиран софтуер за национален адаптор за EUCISE2020	0 days	01.02.2019	01.02.2019																	
88	Програмен код на актуализирания софтуер за Национален адаптор за EUCISE2020	0 days	01.02.2019	01.02.2019																	
89	Актуализирани документи – спецификации, ръководства и др.	0 days	01.02.2019	01.02.2019																	
90	Услуги за техническа асистенция по време на Демонстрационната фаза на тестова обмяна и валидиране на информация	185 days	02.08.2018	02.02.2019																	
91	Протоколи за извършени услуги за техническа асистенция	0 days	02.02.2019	02.02.2019																	

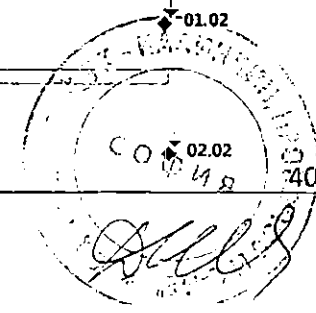
Handwritten signature, likely 'B. P. P.' written vertically.

Large handwritten scribble or signature.

Handwritten initials 'H S'.

Handwritten signature or initials at the bottom center.

Handwritten signature at the bottom right.



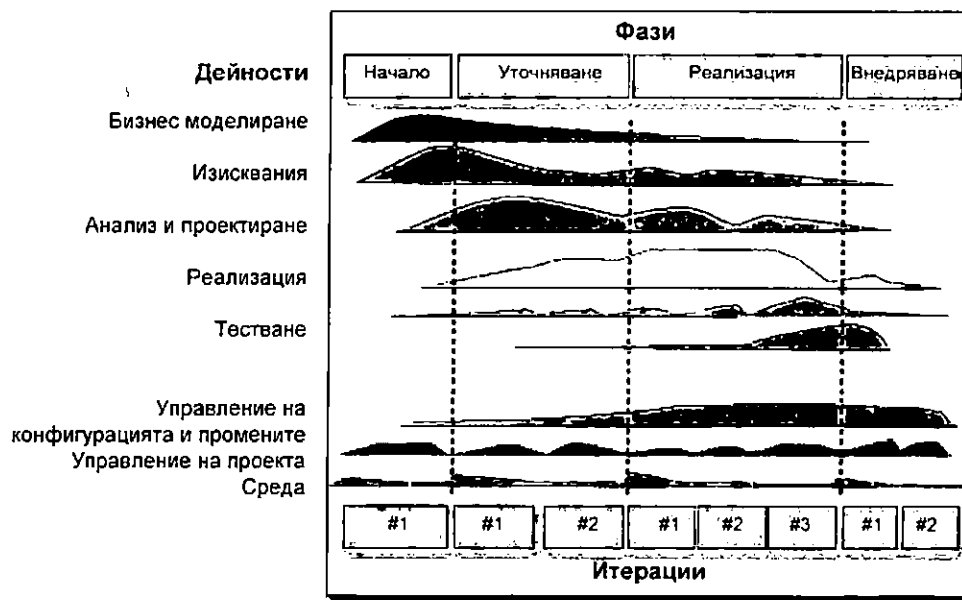
Проект, в който се използва итеративна разработка, притежава жизнен цикъл състоящ се от няколко итерации. Всяка итерация включва множество не-строго последователни дейности в областта на бизнес моделирането, определянето на изискванията, анализ и проектиране, реализация, тестване и внедряване. Те са в различно съотношение в зависимост от това къде в жизнения цикъл на разработката е разположена итерацията.

Итерациите в началната фаза и фазата на уточняване се фокусират върху управление на изискванията и дейности по проектирането. Итерациите във фазата на реализация се фокусират върху проектирането, разработката и тестването. Итерациите във фазата на внедряване се фокусират върху тестване и разпространение на системата. За всички итерации трябва да бъде прието, че са с определена времева рамка и графикът за изпълнението им да бъде фиксиран. Обхватът и съдържанието на итерациите трябва да се управлява активно така, че да се изпълни предвиденият график.

Итеративният подход превъзхожда линейния и каскадния (waterfall) поради няколко причини:

- Рисковете се намаляват рано т.к. елементите се интегрират прогресивно;
- Възприети са променящите се изисквания;
- Заложено е подобряването при изглаждането на продукта, което води до много по-качествен продукт;
- Организациите имат възможността да се учат от този подход и да подобряват техните процеси;
- Повишена е възможността за повторно използване на компоненти;

Унифициран процес на разработка



Препоръчва се да бъде приложен Унифициран процес за разработка. Той дава възможност за организиран подход на възлагане на задачи и отговорности в рамките на организацията на разработчика. Неговата цел е да осигури разработка на висококачествен софтуер, който удовлетворява нуждите на своите потребители и е с предвидим график и бюджет.

Основните фази са:

[Handwritten signatures]

[Circular stamp: КОМПАНИИ СОФИ]

Планиране (Подготовка) - Най-важната цел на началната фаза е да се постигне еднакво мислене на всички заинтересовани лица за целите през целия жизнен цикъл на проекта. Началната фаза е от решаващо значение най-вече за проекти, при които се започва разработка на нов продукт. При този вид проекти значителни рискове свързани с предметната област и изискванията могат да бъдат определени и предотвратени. За проекти, целящи разширяване на съществуващи системи, началната фаза е по-кратка, но също е фокусирана върху осигуряването на проекта, че ще се развива добре и че е възможен за изпълнение.

Проектиране - Целта на фазата е да начертае архитектурата на системата и да предостави основа за подробно проектиране и разработка във фазата на реализация. Системната архитектура се развива въз основа на най-значимите изисквания и на оценката на риска. Стабилността на архитектурата се оценява чрез един или няколко архитектурни прототипа.

Изграждане - Целта на фазата за реализация е да се изяснят неясните изисквания и да се завърши разработката на системата като се съблюдава начертаната архитектура. Фазата на реализация е един вид производствен процес, в който ударението е поставено върху управлението на ресурсите и контролирането на операциите така, че да се оптимизират разходите, графика и качеството.

Внедряване (Предаване) - Целта на фазата за Внедряване е да направи софтуера достъпен за неговите потребители. Фазата на внедряване може да премине през няколко итерации и включва тестване на продукта, подготовка за нови версии, корекции базирани на обратната връзка от потребителите. На този етап от жизнения цикъл, обратната връзка от потребителите трябва да е фокусирана основно върху прецизни настройки, конфигуриране, инсталиране и използваемост. Всички по-съществени структурни въпроси трябва да бъдат решени в по-ранните фази на проекта.

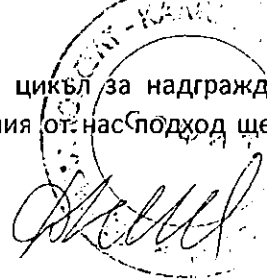
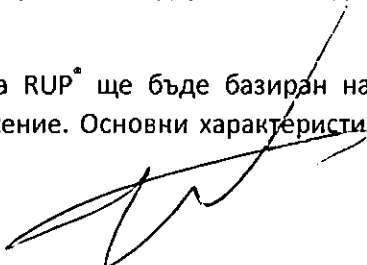
При изпълнението на проекта ще се използва методика за разработка на софтуер Rational Unified Process (RUP).

Управлението и изпълнението е организирано във фази, които могат да бъдат разделени на две или повече итерации. Основната цел на итерациите е да се постигне поетапно (итеративно) извършване на реализацията, тестването и внедряването. Итерациите по отделните фази обхващат една или повече дисциплини от цикъла за разработка на софтуерния продукт (бизнес моделиране, специфициране на изисквания, анализ и дизайн, реализация, управление на проекта, изграждане на среда за изпълнение).

Подход за приложение на RUP®

Методиката за разработка на софтуер, която ще приложим ще бъде IBM Rational Unified Process® (RUP®). Възникнали в хода на изпълнението въпроси, незасегнати в Техническото задание ще бъдат решавани съгласно изискванията на RUP®. За всички останали непокрити от Техническото задание или RUP® аспекти на проектното управление, като допълващи методи за управление, ще се придържаме към препоръките на ANSI/PMI 99-001-2008. Това са например Управление на доставките, управление на човешките ресурси и управление на финансовите разходи. В такива ситуации за управление на изискванията ще се придържаме към Business Analysis Body of Knowledge (BABOK®) Guide на IIBA®. По отношение на останалите дисциплини, аспектите непокрити от Техническото задание или RUP® ще бъдат изпълнявани под общоприети стандарти и методологии, като BPMN, UML, JEE, XML и други технологични стандарти.

Подходът ни за прилагане на RUP® ще бъде базиран на еволюционен цикъл за надграждане на съществуващо голямо приложение. Основни характеристики на прилагания от нас подход ще бъдат



Итеративната разработка, Управление на изискванията, Архитектура на отделните компоненти, Визуално моделиране, Непрекъсната проверка на качеството и Управление на промяната.

Начин на организация на итерациите

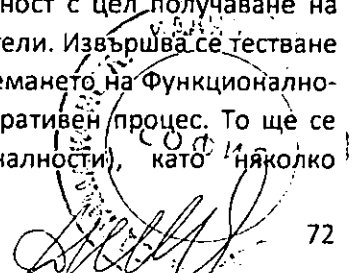
Съгласно RUP® жизнения цикъл на проекта ще бъде организиран във фази, които ще бъдат разделени на итерации. Дейностите ще бъдат разделени на фази по следния начин - Планиране (Inception), Проектиране (Elaboration), Изграждане (Construction) и Внедряване (Transition).

Целите, които си поставяме в края на Планирането е детайлизиран план за изпълнение на жизнения цикъл. В края на Проектирането, целта е да бъде изготвена архитектура и дизайн на системата. В края на Изграждането трябва да бъде постигнат базов оперативен капацитет на системите. В края на Внедряването трябва да бъде въведена в експлоатация окончателната версия на продукта.

В рамките на всяка една от фазите са обособени една или повече итерации. Изпълняваните дейности са свързани с определени отчетни продукти. Съгласуван списък с отчетни продукти и описание на съдържанието им в съответствие с RUP ще бъдат заложи в Плана за разработка на софтуерния продукт, който ще бъде утвърден от Възложителя в края на фаза Планиране. Допълнителна информация за конкретните фази на изпълнение:

- **Планиране/Inception** – В рамките на тази фаза ще си извършват предимно дейности по планиране от всички работни процеси. Паралелно с тях ще стартират дейности свързани с изпълнение на задачи по Проектиране, Изграждане и дори Внедряване с инициране на планиране на внедряването. Фазата на планиране ще завърши с предаването на изискваните съгласно Техническото задание отчетни продукти - План на проекта, План за управление на качеството, Списък на рисковете, Спецификация на функционалните изисквания, Спецификация на допълнителните изисквания и План за следващата итерация. Дейностите по планиране ще продължат, като преди началото на всяка итерация ще се извършва оценка на постигнатото в предходната итерация и планиране за следващата итерация. В началото на всяка фаза ще се извършва планиране на итерациите по фазите. Дейността по планиране ще бъде осъществена по отворен за проектния екип начин, така че да служат ефективно като средство за управление на риска и промяната.
- **Проектиране/Elaboration** – Ще бъдат разработени прототипи за доказване на изпълнимостта на продукта. Проектирането е предмет на планиране и в този смисъл то винаги следва дейностите по планиране. Същинското проектиране ще започне с приключването на фаза Планиране. Проектирането ще бъде извършено в началото на проекта в широчина на обхвата, като в рамките на всяка итерация ще бъде конкретизирано по компоненти. По време на проектирането ще се изпълняват и дейности от Изграждане и Внедряване. Проектирането изисква високо ниво на комуникация и съвместна работа с експерти на Възложителя. То ще завърши с предаване на отчетните материали за фазата, съгласно Техническото задание.
- **Изграждане/Construction** – Основните цели на тази фаза са разработката на софтуера и създаването на използваеми версии (алфа, бета и други тестови версии) в предвидените срокове и с необходимото качество. През тази фаза ще се приключи с дизайна, реализацията и алфа тестовете на цялата заявена функционалност с цел получаване на завършен продукт, годен за предаване на бъдещите му потребители. Извършва се тестване на база дефинираните критерии за оценка. То ще започне с приемането на Функционално-техническата спецификация. Изграждането е изключително итеративен процес. То ще се изпълнява по компоненти (капсуловани кратки функционалности), като няколко


72
105

компоненти ще се изпълняват паралелно. Точния график на итерациите ще бъде специфициран в Плана за разработка на софтуерния продукт. Всяка итерация ще бъде предхождана от дейности по Планиране и дейности по Проектиране. Итерациите ще завършват с дейности по оценка на изпълнението и с дейности по вътрешно тестване. Функционално завършени компоненти ще бъдат предавани на Възложителя за приемно тестване (част от фазата Внедряване). В този смисъл Изграждането ще завърши с получаване на одобрение от Възложителя за въвеждане на компонента в експлоатация, т.е. Внедряване.

- Внедряване/Transition – Основните цели на фаза Внедряване са: Провеждане на бета тестове за валидиране на системата от гледна точка на потребителските изисквания; Мигриране на стари данни, ако е необходимо; Специфични за етапа на внедряване дейности по инсталиране; Фини настройки като отстраняване на дефекти, подобряване на производителността и използваемостта на системата; Оценка на индикаторите за внедряване на база критериите за приемане на продукта; Получаване съгласие от Възложителя, че индикаторите за внедряване са изпълнени и съответстват на критериите за оценка. Внедряването на даден компонент ще започне след получаване на одобрение от Възложителя за въвеждане на компонента в експлоатация. Внедряването ще завърши с въвеждането в експлоатация на продуктите и получаването на съответното потвърждение от Възложителя.

Изборът ни на традиционен подход за разработка, базиран на Rational Unified Process се обосновава на следните факти:

- Поръчката се възлага чрез договор с фиксирани обхват, цена и срок. Предложеният подход съвпада с изискванията за етапи на изпълнение, съгласно Техническото задание.
- Срокът за изпълнение е изключително кратък, което предполага интензивен анализ на изискванията и проектиране в началото на изпълнението, както и придържане към твърд график по време на цялото изпълнение.
- Световно-утвърден и доказан подход за разработка.
- Позволява итеративно предоставяне на компоненти от разработката.

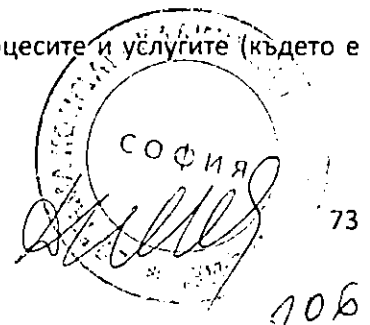
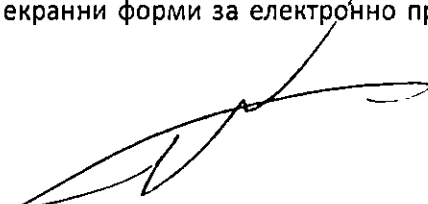

5.3.2. Характеристики на процеса за софтуерна разработка

На база на изготвени анализи, моделирани процеси и спецификации на програмни интерфейси за всеки елемент на системата, Изпълнителят ще започне неговата разработка, като изпълнява най-малко под-дейностите, които са приложими, както са дефинирани в следващите секции на ТС.

Разработка:

- Разработка на схеми за данни / структура на базата/ите данни;
- Разработка на изходен код за имплементиране на бизнес-логика
- Разработка / генериране на изходен код за автоматизирани процедурни тестове (Unit Tests)
- Разработка / генериране на изходен код за програмни интерфейси и клиентска интеграция;
- Идентифициране и специфициране на нужди от промени в програмни интерфейси (APIs) на други елементи или компоненти, които са необходими за реализация на разработваните функционалности и процеси, които зависят от тях;
- Разработка на екранни форми за електронно представяне на процесите и услугите (където е приложимо);

Тестване:



73

106

- Разработка на тестови сценарии за проверка на реализацията на системата;
- Тестване на разработената функционалност по съответния компонент;
- Документиране на резултатите от проведените тестове.

Представяне:

- Представяне на Възложителя на разработени функционалности чрез изпълнение на предварително дефинирани тестови сценарии;

Изпълнението на дейностите по разработка ще бъдат планирани, както по елементи, така и по дефинирани периодични итерации, включващи разработка и тестване на отделните елементи и компоненти. Трябва да бъде предвиден и координационен механизъм и процеси за обмен на спецификации за промени в програмни интерфейси между отделните екипи, изпълняващи дейности по разработка на елементи от системата, ако е приложимо.

Тестване и представяне на резултатите на Възложителя ще се извършва за всяка итерация, в оперативен порядък. В случай, че Възложителят има изисквания за корекции по реализираната функционалност, те трябва да бъдат отразени от Изпълнителя до представяне на следващата итерация по разработка на съответния компонент.

В случай, че Възложителят няма забележки, итерацията се приема и се интегрира в софтуерното решение.

Промяна:

- Стъпка „Промяна“ се извършва само в случаите, в които Възложителят е имал забележки и корекции по реализираната функционалност в една итерация, описана в предходната стъпка.

Интегриране:

- Стъпка „Интегриране“ обхваща дейностите, свързани с включване на разработената и одобрена от Възложителя функционалност в един компонент.

Документиране:

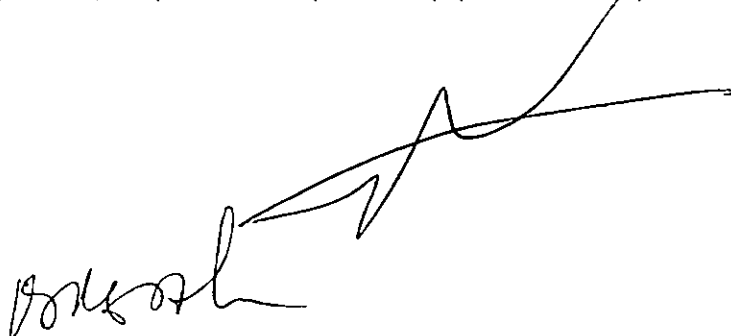
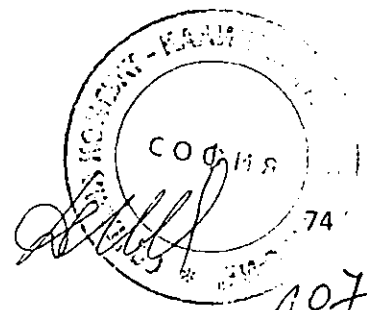
- Стъпка „Документиране“ включва генериране на документация за текущата (build) версия на програмните интерфейси за интеграция на отделните елементи, съгласно изискванията за документиране на програмни интерфейси, описани в т. 7.9.2 от ТС.

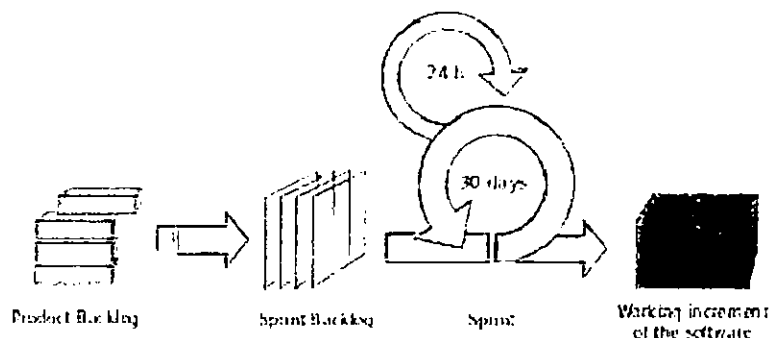
Като резултат от дейност „Разработка“ Възложителят ще получи:

- Работеща функционалност на системата (система, подсистема, модул, компонент);
- Изходен код с версия за публикуване (Release Version);
- Тестови сценарии и доклади от проведени тестове.

5.3.3. Вътрешно-екипна координация в рамките на итерациите

Вътрешно екипната координация ще бъде реализирана чрез прилагането на Scrum. Scrum е итеративна, инкрементална рамка за управление на проекти.



Макар че подходът на Scrum е първоначално предложен за управление на проектите за разработване на продукти, той се фокусира с времето върху управлението на софтуерни проекти и може да бъде използван, за да задвижва екипи за софтуерна поддръжка или като общ подход за проектoven програмен мениджмънт. Второто определение го прави доста ценно и нужно за България и българското образование, т. к. в страната ни има недостатъчно на брой специалисти от световно ниво по "product ownership" и "product development". Това е предпоставка за факта, че често разработваме софтуер, който някой друг продава, или за който някой друг вече е намерил клиентите.

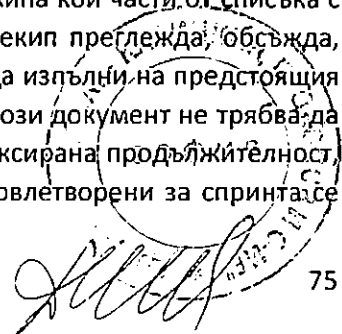
Scrum процесът се състои от отделни спринтове. Спринтовете могат да имат продължителност от една седмица до четири седмици. В края на всеки спринт, екипът разполага с работеща версия на продукта, която включва всички готови задачи от backlog-a.

SCRUM позволява организирането на самоорганизиращи се екипи като стимулира това всички членове на екипа да се намират на едно и също място и да си комуникират на живо. Ключов принцип на SCRUM е това, че се приема още в самото начало на проекта, че изискванията няма как да са пълни и напълно разбрани. Т.е. очакват се нововъведения от клиента – промяна на желанията му. SCRUM се фокусира върху способността на екипа да доставя бързо и да е готов да отговори бързо на неочакваните промени. Това е положителна черта за SCRUM, защото резките промени не могат да бъдат добре оборени с традиционните отгатващ или планиращ методи. Както в другите гъвкави методологии за управление на проекти, SCRUM може да се имплементира чрез различни видове инструменти. Най-много се използват спредшийтове. В тях се правят така наречените SCRUM „артефакти“, като sprint backlog-a. Други организации имплементират SCRUM с бели дъски, лепящи се листчета и хартиени документи.

Спринтът е най-малката единица време за разработване. Спринтовете са с постоянна дължина от 1 седмица до 1 месец. Всеки спринт е опит за подобрене, вкаран във фиксирани времеви рамки. Преди всеки спринт има среща за планиране на спринта. На нея се поставят измеримите цели за спринта и се идентифицират задачите, които ще бъдат свършени в неговите рамки. По време на всеки спринт екипът създава завършени парчета от даден продукт. Дейностите за всеки спринт се описват и взимат от „продуктовата опора“ или на английски „product backlog“. Често тези дейности са описани като характеристики, които продукта трябва да има и да бъдат постигнати за спринта. Т.е. product backlog е приоритизиран списък на изискванията. Какво от списъка да влезе в даден спринт се решава на планиращата среща преди спринта. Продуктовият собственик уведомява екипа кои части от списъка с изисквания иска да бъдат свършени на предстоящия спринт. Развойният екип предлага, обсъжда, решава и записва в Sprint Backlog кои от тези изисквания и цели ще успее да изпълни на предстоящия спринт. Sprint Backlog е собственост на развойния екип. Целите, вписани в този документ не трябва да бъдат променяни по време на спринта. За разработването се определя фиксирана продължителност, такава, че спринтът да свърши навреме. Изискванията, които не бъдат удовлетворени за спринта се

Handwritten signature

Handwritten signature



изключват и връщат към product backlog. След като спринтът е изпълнен, екипът демонстрира как се използва софтуерът.

По време на спринта всеки ден се провеждат т.нар. правостоящи срещи (stand-up meetings). Тези срещи продължават от 5 до 15 минути и се провеждат всеки ден в определен час. На срещата всеки от екипа абсолютно неформално разказва за три неща: какво е работил предишния ден, какво планира за предстоящия ден и какви проблеми е срещнал, които му пречат да работи.

Характерно за ежедневните срещи е следното:

- Срещата започва точно навреме.
- Мястото и часът са едни и същи всеки ден.
- Срещата трябва да се вмести точно в 15 минути.
- Може всички, но главно говорят основните роли в екипа.
- На срещата се обновява и backlog-ът като се отбелязва свършената работа. Ако се идентифицират някакви проблеми, те се решават колективно. Важно е да се отбележи, че това не са срещи за отчет пред ръководството, а за синхронизация (самоорганизация) на екипа и разкриване на потенциални пречки в работата.

Специфични практики:

- Ежедневни правостоящи срещи (Stand-up meetings)
- Backlog: списък със задачите за текущия спринт и тяхното състояние
- Самоорганизиращ се екип: екипът не следва предварително раздадени задачи, а всеки негов член се стреми да допринесе за постигане целите на спринта - всеки ден всеки си взема задачи, за които отговаря
- Работни срещи и обсъждания с клиента и с екипа след всеки спринт

Срещи след спринта:

- Среща за обзор на спринта: Преглежда се и работата, която е била свършена, и тази която не е. Представя се свършената работа на клиентите (показва се демо). - Несвършената работа няма как да бъде представена, но се описва. - Срещата трябва да се вмести в 4 часа.
- Спринт Ретроспекция - Всички членове на екипа си припомнят и обсъждат миналия спринт.

5.3.4. Обектно-ориентирано програмиране

Обектно-ориентираното програмиране (ООП) е парадигмата за програмиране, използваща "обекти" - структури от данни, състояща се от полета с данни и методи, в едно с техните взаимодействия. Използва се за проектиране на приложения и компютърни програми. Техниките за програмиране могат да включват функции като абстракция на данни, капсулиране, съобщения, модулност, полиморфизъм и наследяване. Много съвременни езици за програмиране, които сега поддържат ООП, поне като опция.

Дори простите ООП програми могат да представляват един "дълъг" списък от команди. По-сложните програми групират по-малки участъци от код във функции или така наречените подпрограми, всяка от които може да изпълнява определена задача. С дизайн от този вид става обичайна практика за някои от данни на програмата да са „глобални“, т.е. достъпни от всяка част на програмата. Когато програмите станат по-големи, и е позволено всяка функция да променя всяка част от данните, се оказва, че грешките могат да имат сериозни последици.

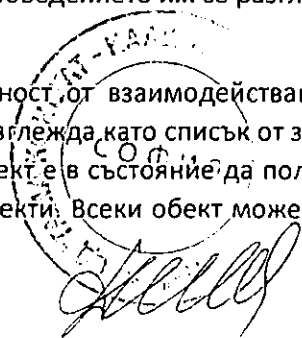
За разлика от този подход, обектно-ориентираното програмиране насърчава програмистите да поставят данните, където им е мястото, т.е. да не са директно достъпни от останалата част на програмата. Вместо това, данните са достъпни чрез използване на написани за целта функции, често наричани методи, които или вървят в пакет с данните, или се наследяват от „класа на обекта“. Те действат като посредници за извличане или промяна на данните, които контролират. Програмирането е конструкция, която комбинира данни и методи за достъп и управление на тези данни, нарича се обект. Практиката да се използват подпрограми за достъп и промяна на данни е използвана и в необектно ориентираното програмиране.

Обектно-ориентираните програми обикновено съдържат различни видове обекти, всеки вид, съответстващ на определен вид комплексни данни, които ще се управляват или отговарят на обект или понятие от реалния свят, като например банкова сметка, хокей играч или булдозер. Една програма може да съдържа няколко копия на всеки тип обект, по един за всеки обект от реалния свят. Например, може да има един обект на банкова сметка за всяка истинска сметка в определена банка. Всяко копие на обекта на банкова сметка, ще бъде с еднакви методи за манипулиране или четене на данни, но данните вътре във всеки обект ще се различават и ще отразяват различната историята на всяка сметка.

Обектът може да се разглежда като опаковка на данните, заедно с набор от функции, предназначени да гарантират, че данните се използват по подходящ начин и да съдейства в тази употреба. Методи на обекта, обикновено включват проверки и предпазни мерки, които са специфични за данните, които се съдържат в обекта. Един обект може да предлагат прости за използване, стандартизирани методи за извършване на определени операции върху данните си, като същевременно прикрива спецификата на това как се изпълняват тези задачи. По този начин могат да бъдат правени промени във вътрешната структура или методите на обект, без да се изисква модифициране на останалата част от програмата. Този подход може също така да бъде използван и за предлагане на стандартизирани методи в рамките на различните видове обекти. Като пример, няколко различни видове обекти може да предложи методи за печат. Всеки тип на обекта може да реализира този метод за печат по различен начин, отразяващ различни видове данни, но същевременно различните методи за печат могат да са кръстени по еднакъв и стандартизиран начин на други места в програмата. Тези характеристики стават особено полезни, когато повече от един програмист разработва код в проекта или когато целта е за повторно използване на код между проектите.

Ориентацията към обекти е свързана с поддържането на качеството на софтуера. Тя е насочена към общи проблеми, като набляга на създаването на дискретни, подходящи за многократна употреба модули от програмна логика. Технологиата се фокусира върху данните, вместо върху процесите, като програмата се състои от самодостатъчни модули („класове“), всяко копие, на които („обекти“) съдържа цялата информация, необходима, за да може да манипулира своята собствена структура от данни („членове“). Това е в контраст на съществуващите техники за модулно програмиране, които са били доминиращи в продължение на много години, които се фокусират върху функциите на един модул, а не върху конкретните данни, но по подобен начин предлагат възможност за повторно използване на код, самодостатъчни са и са подходящи за многократна употреба, дават възможност на сътрудничество чрез използването на свързани модули (подпрограми). Това е по-традиционния подход, който все още продължава, да се използва. При него данните и поведението им се разглеждат отделно едни от други.

Обектно-ориентираната програма може да се разглежда като съвкупност от взаимодействащи си обекти, за разлика от конвенционалния модел, в които програмата се разглежда като списък от задачи (подпрограми) да изпълнява. В обектно-ориентирания подход, всеки обект е в състояние да получава съобщения, да обработва данни и да изпраща съобщения към други обекти. Всеки обект може да се



разглежда като независима „машина“ с ясно изразена роля или отговорност. Действията (или „методите“) на тези обекти са тясно свързани с обекта. За пример, ООП структури от данни са склонни да „носят своите оператори заедно с тях“ (или най-малкото да ги „наследяват“ от подобни обекти или класове) - с изключение на случаите, когато те трябва да бъдат сериализирани.

За да бъде един програмен език обектно-ориентиран, той трябва не само да позволява работа с класове и обекти, но и трябва да дава възможност за имплементирането и използването на принципите и концепциите на ООП:

- Наследяване (Inheritance) - как йерархиите от класове подобряват четимостта на кода и позволяват преизползване на функционалност;
- Абстракция (Abstraction) – виждането на даден обект само от гледната точка, която ни интересува и да игнорираме всички останали детайли;
- Капсулация (Encapsulation) – скриване на ненужните детайли в нашите класове и предоставяне на прост и ясен интерфейс за работа с тях;

Полиморфизъм (Polymorphism) - работа по еднакъв начин с различни обекти, които дефинират специфична имплементация на някакво абстрактно поведение.

5.3.5. Дейности за интегриране на функционалности/компоненти в системата
В етапа на разработка, стъпка „Интегриране“ обхваща дейностите, свързани с включване на разработената и одобрена от Възложителя функционалност в един компонент.

Интеграцията е процес, при който се комбинират отделни софтуерни компоненти в единна система.

Интеграцията е и форма на тест – Тъй като тя се извършва след като разработката и тестването на отделните компоненти е приключило. Също така тя е свързана и с интеграционните тестове. Тя е достатъчно сложна, за да се разглежда като независима част от проекта – Правилната и внимателна интеграция води след себе си :

- По-лесна диагностика;
- По-малко дефекти;
- Съкратено време за завършването на продукта;
- Подобрено качество на кода;
- Повишени шансове проекта да приключи успешно;
- По-малко документация;

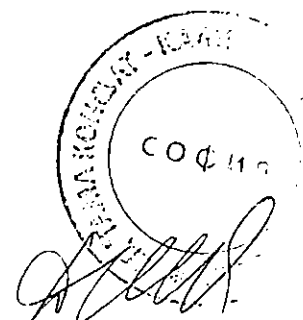
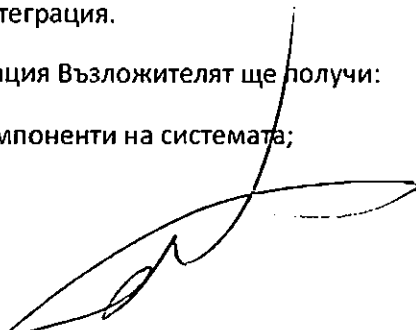
и много други.

В тази дейност се очаква да бъдат:

- Изградени съответните интеграционни интерфейси;
- Интегрирани съответните компоненти;
- Извършени тестове за интеграция.

Като резултат от дейност Интеграция Възложителят ще получи:

- Успешно интегрирани компоненти на системата;



- Доклади от проведени тестови изпитания и резултатите от тях.

Целият хардуер, компоненти, модули, части и софтуерни лицензи на специфицираното и предложено от Изпълнителя оборудване, ще бъдат сглобени, тествани, инсталирани и ще се валидира тяхната функционалност и работоспособност в работно помещение на Възложителя.

Съществуват два основни подхода за интеграция на компонентите:

- Единна интеграция;
- Нарастваща (инкрементална) интеграция.

Единна интеграция

Единната интеграция включва следните стъпки:

- 1) Дизайн, конструиране на кода и тестове
- 2) Всичко направено се събира заедно
- 3) Извършване на тестове върху така получената пълна система

Този вид интеграция дава предимство, че предварително могат да се планират и обхванат всички интеграционни задачи, но има и редица недостатъци, които го правят приложим само за по-малки системи.

- Не е ясно от къде идват проблемите;
- Този тип интеграция не може да започне в ранна фаза на проекта. Трябва да се изчака почти до самия му край.

Тъй като съществува голям брой класове, които до сега не са работили заедно, когато възникне грешка няма как да я идентифицираме лесно. Не е ясно, дали даден клас не е тестван добре, дали е грешка заради интерфейса, по който комуникират два компонента или пък е грешка от взаимодействието на два класа. Този недостатък до някъде се компенсира от факта, че всички проблеми се появяват едновременно – по този начин програмистите трябва да се справят не само с проблеми от взаимодействието на компонентите, но и с проблеми, които се появяват от поради наличието на няколко други проблема.

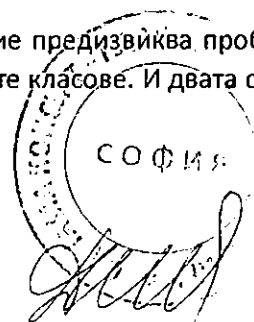
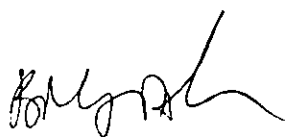
Нарастваща интеграция

При нея процесът протича по следния начин:

- 1) Разработва се малка част от системата – В общия случай за всеки проект се подхожда различно. Началния код, който се разработва може да е най-лесната или най-трудната част, както и някаква комбинация между тях.
- 2) Разработва се нов клас, тества се и т.н.
- 3) Новият клас се добавя към вече съществуващата част и така получената комбинация се тества отново – Ако всичко е наред с тестовете и е останало още за доразработване, се преминава обратно в стъпка 2.

Нарастващата интеграция дава следните предимства:

- Не е нужно добавянето да става клас по клас;
- Грешките се откриват по-лесно - Или интерфейсът за взаимодействие предизвиква проблема или е взаимодействието на новия клас, с някой от вече интегрираните класове. И двата случая е ясно къде да трябва да се погледне, за да се намери проблема.



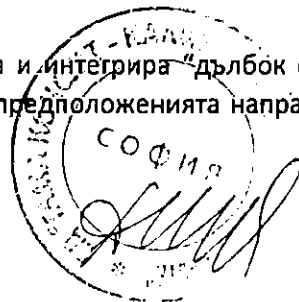
- Системата е видима в ранна фаза – Това подобрява морала на програмистите и атмосферата на работа, понеже виждат как системата се изгражда и работи. С единната интеграция винаги остава съмнението, че когато нещата се съберат заедно, нищо няма да сработи.
- Подобен контрол над прогреса – Много по-лесно е да определим как се движи проекта и дали е по график, когато виждаме 50% от него вече работещ.
- Клиентът е по-щастлив – Клиентите обичат да виждат признаци на прогрес. Нарастващата интеграция непрестанно им предоставя факти по напредъка.
- Модулите на системата се тестват по-пълноценно – Първо, тестовите почват в много по-ранна фаза. Второ, всеки клас се тества много по-често (с всяко добавяне на следващ модул).
- Системата се изгражда за по-кратко време – Ако интеграцията е планирана правилно, работата може да започне по някаква част от проекта, докато друга част все още е на етап дизайн. Това няма да намали общото време за реализация, но позволява част от работата да се свърши в паралел.

Съществуват различни видове стратегии за прилагане на нарастваща интеграция:

- **Top-Down интеграция** - При Top-Down интеграцията, класът стоящ най-високо в йерархията се пише и интегрира пръв. Започва се с базовия клас на системата (съдържащ main(), WinMain() и т.н.) Останалите класове често се пишат само като шаблони (съдържащи само декларациите на методите) и в последствие се дописват и добавят. Тъй като чистия Top-Down подход рядко се използва, негов по-практичен вариант е вертикалната Top-Down интеграция. Тя елиминира повечето проблеми на предишния подход и е често използвана в практиката.
- **Bottom-Up интеграция** - При Bottom-Up интеграцията, добавянето на класове върви в обратен ред (от долу нагоре). Основният проблем при Bottom-Up интеграцията е, че системните интерфейси в най-високото ниво се интегрират последни. Ако в системата има концептуална грешка или проблем свързан с дизайна, той остава скрит почти до края на интеграцията. Освен това, ако в такъв момент дизайнът се смени, може да се наложи част от класовете в ниските нива на йерархията да се пренапишат. Аналогично, за избягване на недостатъците на Bottom-Up, се използва вертикална Bottom-Up интеграция, точно както при Top-Down.
- **Sandwich интеграция** - Заради проблемите, които могат да възникнат при Top-Down и Bottom-Up интеграцията, някои експерти препоръчват интеграционния процес да протече така: първо се интегрират класовете от най-високо ниво, после тези от най-ниско ниво, а свързващото ги звено остава за последната фаза. Това е реалистичен и много практичен метод за интеграция.
- **Рисково-ориентирана интеграция** - Основният принцип при рисково ориентираната интеграция е "най-трудната част се интегрира първа". В общия случай, при този тип интеграция се интегрират класовете стоящи най-високо и най-ниско в йерархията. Този подход много прилича на Sandwich интеграцията, тъй като изследванията показват, че най-рисковите нива са в най-горната и най-долна част на йерархията.
- **Функционална интеграция** - При функционалната интеграция основната идея е, че основният интеграционен елемент е функцията (в смисъл на възможност или характеристика на приложението). Например, ако пишем текстообработващо приложение една функция може да бъде Fonts диалога, класът за изобразяване на ефекти върху шрифтовете, менютата и т.н. Ако някоя от функциите, които ще интегрираме се състои от множество класове, към тях отново може да се приложи стратегия за нарастваща интеграция. Т.е. стратегиите за нарастваща интеграция могат да се използват рекурсивно.
- **T-образна интеграция** - При T-образната интеграция се изгражда и интегрира "дълбок срез" вертикално през целия проект. По този начин се проверява дали предположенията направени



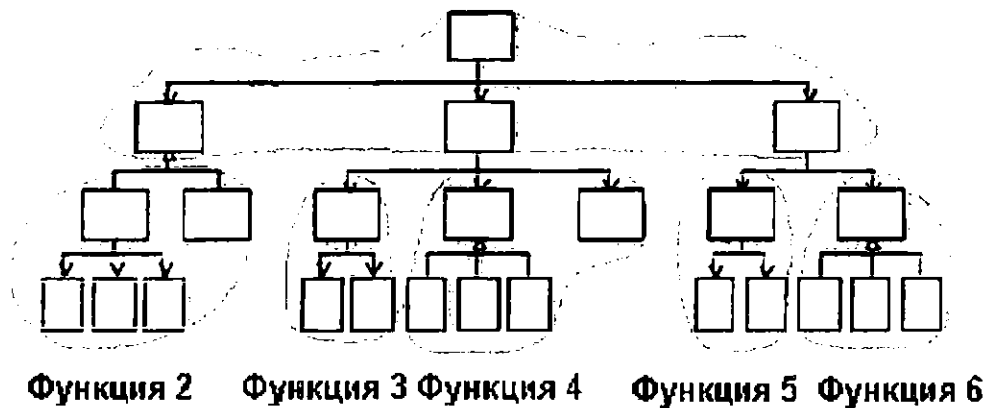




по време на проектирането на архитектурата са коректни. След като тази част приключи, всички останали модули се интегрират хоризонтално.

Няма една "най-добра" методология за интеграция. Никоя от тях не е универсална и не трябва да се спазва стриктно. Винаги се търси метод, по-скоро базиран на някоя от стратегиите, така че той да пасне най-добре на самия проект.

5.3.6. Стратегия за интегриране на функционалности/компоненти в системата
В етапа на анализ и проектиране ще бъде разработена детайлна системна и софтуерна архитектура. Като част от архитектурата ще бъдат поставени стандарти за интеграция на компонентите, както и ще бъдат дефинирани основните интеграционни интерфейси.



При разработката препоръчваме прилагането на стратегия за нарастваща функционална интеграция, която дава следните предимства:

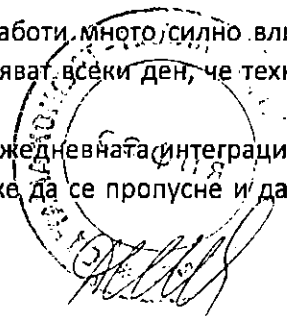
- Редукция на поддържащия код – Добавянето на новата функция към проекта не е свързано с модификация или създаване на поддържащ код. Подлежащата на интеграция функцията е независима и съдържа всичко нужно за работата си.
- Новоинтегрирания код води до нова функционалност – Това показва, че проекта върви напред. От една страна повдига духа на разработчиците, а освен това подобрява и отношенията с клиента.
- Добра съвместимост с ООП – Тъй като най-често обектите са свързани с някаква възможност или функционалност, този подход действа много естествено в обектно-ориентираните системи.

5.3.7. Дневни билдове и тестове

Добра практика при прилагането на методологията за интеграция е извършването на дневни билдове в комбинация с тестване, които водят до следните ползи:

- Намален риск на качеството – Отстранява се възможността някоя грешна интеграция да предизвика сериозни проблеми в системата. Проекта веднъж докаран до добро, работещо състояние се държи на това ниво и всеки ден се проверява, че всичко е наред.
- По-лесна диагностика на дефектите – Ако проекта се проверява всеки ден и един от дните спре, веднага е ясно, че проблемът е в код добавен в последния ден.
- Подобен морал на разработчиците – Самият факт, че продуктът работи много силно влияе върху морала на разработчиците. Чрез дневните билдове те се уверяват всеки ден, че техният продукт работи.
- Изважда на преден план "невидими" задачи – Страничен ефект от ежедневната интеграция е факта, че тя изважда на преден план доста работа, която иначе може да се пропусне и да се

[Handwritten signature]



появи чак в края на проекта. Екипи, които не са ползвали дневни билдове, първия път винаги се оплакват, че дневните билдове ги бавят. Това не е така, просто им се налага да отделят ежедневно ресурси за интеграция на кода, които иначе биха им се струпали накрая.

При изпълнението надневни билдове и тестове ще се прилагат следните принципи:

- Разработчиците трябва сами да тестват кода си, преди да го добавят към проекта – Един добър метод е всеки разработчик да си прави личен билд, да го тества и чак тогава да добави промените си към целия проект.
- Централно хранилище за код – Основна част от успехите на дневните билдове идват от факта, че знаем кои билдове са добри и кои са пропаднали. При малки и средни проекти за да се пази история се използва система за контрол на версиите. При големите проекти обикновено само билд групата има права да добавя код в главната системата за контрол на версиите.
- „Наказания“ за „счупване“ на билда – На разработчиците трябва да е ясно, че дневния билд винаги трябва да работи. Пропадането му е голямо изключение (никога практика) и всеки трябва да счита с най-висок приоритет задача по възстановяване на пропаднал билд. В Microsoft, разработчиците от Windows 2000 и Office групата носят денонощно пейджъри и биват викани и в 3:00 през нощта, ако се окаже че техен код проваля билда.
- Билдовете приключват сутрин – Добра идея е билдовете да минават нощем, тестовете да се правят рано сутринта и процеса да приключва още преди обяд. Така тестерите могат всяка сутрин да тестват с нов билд. Сутрин колегите се откриват най-лесно (в случаи, когато с билда има проблем). Викането на хора посред нощ поставя излишно напрежение в екипа.
- Дневни билдове и тестове трябва да се извършват без значение натовареността – Когато времето напредва и напрежението нараства, разработчиците губят дисциплина. Никога не трябва да се допуска това да влияе върху дневните билдове и тестове.

5.3.8. Управление на версиите

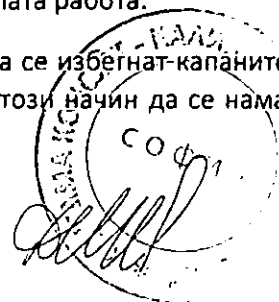
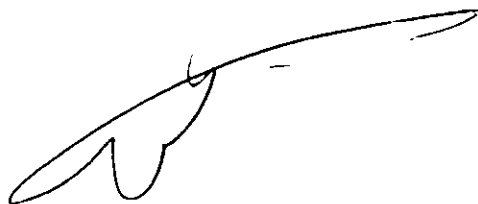
Непрекъснатата интеграция

В софтуерно инженерство, непрекъснатата интеграция (CI) осъществява непрекъснат процес на прилагане на контрол на качеството - малки усилия, които се прилагат често. Непрекъснатата интеграция има за цел да подобри качеството на софтуера и да намали времето, необходимо той да бъде доставен, чрез заместване на традиционната практика на прилагане на контрол на качеството, след завършване разработката.

При навлизането на промяна, разработчиците вземат копие на текущата кодова база, върху която да работят. Когато други разработчици направят промени в изходния код, те го изпращат в хранилището на версиите. Когато разработчиците представят код на хранилището, те трябва първо да актуализират своя код, за да отразяват промените в хранилището, тъй като там е най-актуалното копие. Колкото повече промени съдържа хранилището, толкова повече работа ще свършат разработчиците, преди да предадат своите собствени промени.

В крайна сметка, хранилището може да стане толкова различно от основната линия на разработчиците, че те да попаднат в така наречения "интеграционен ад", където времето, необходимо да се интегрират надхвърля времето, необходимо, за да направят промените. В най-лошия сценарий, разработчиците могат да изхвърли своите промени и да извършат наново цялата работа.

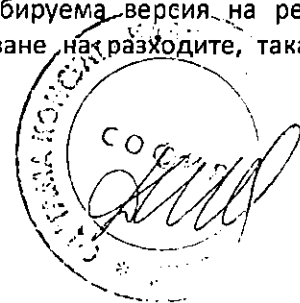
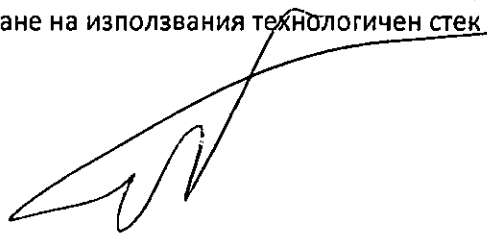
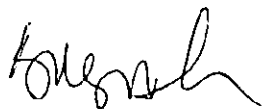
Непрекъснатата интеграция предполага интегриране рано и често, така че да се избегнат капаните на "интеграционния ад". Практиката има за цел да намали преработките и по този начин да се намалят разходите и времето.



Непрекъснатата интеграция - практика на често интегриране на нов или променен код в хранилище на съществуващия код - ще се случва достатъчно често, така че да не остава прозорец между качване на код и компилиране, и поради това да не може да възникне грешка без програмиста да я забележи и коригира незабавно. Нормалната практика е да се задействат при всяко качване на код, а не периодично. Тези практики в среда от много програмисти с бързи комити (включване на програмен код в интеграционната среда) се преобразува в такава, при която се задава кратък интервал след всяко качване на код и чак след това се пуска компилирането.

За да постигне своите цели, непрекъснатата интеграция разчита на следните принципи:

- 1) Поддържане на хранилище на код - Тази практика се застъпва за използването на система за контрол на ревизии на изходния код на проекта. Всички артефакти, които са необходими за изграждане на проекта ще бъдат поставени в хранилището. Важна практика е да бъде възможно системата да се компилира от пряко извлечен код от системата без да е необходимо удовлетворяването на допълнителни зависимости. Използването на клонове ще бъде сведено до минимум. Промените ще бъдат бързо интегрирани вместо да се създават множество разклонения. Основната линия (trunc) ще бъде място за работеща версия на софтуера.
- 2) Автоматизиране на компилацията – ще може да се извърши с една команда. Автоматизация на компилацията ще включва автоматизиране на интеграция, която често включва внедряване в производствената среда. В много случаи, скрипта за компилация не само компилира изпълнимите файлове, но също така генерира документация, страниците на уебсайта, статистика и окончателната дистрибуция (например Windows MSI файлове, RPM или DEB файлове).
- 3) Компилацията ще може да се тества самостоятелно - След като кодът е компилиран, всички тестове би трябвало да се стартират, за да потвърдят, че той се държи така, както програмистите очакват да се държи.
- 4) Ежедневно качване в базовата линия - чрез редовно качване в базовата линия, всеки програмист може да се намали броят на конфликтните промени. Седмично качване води до риск от увеличаване на конфликтните промени, които може да са трудни за разрешаване. Ранните, малки конфликти в дадена областта карат програмистите да общуват относно промените, които правят.
- 5) Всяко качване (в базовата линия) ще се компилира - Системата трябва компилира текущата версия, за да определи дали направените промени са правилно интегрирани. Честа практика е да се използват автоматизирани средства за непрекъснатата интеграция, въпреки че това може да се направи и ръчно. За много хора, непрекъснатата интеграция е синоним на използването на сървър за автоматизирана непрекъснатата интеграция, който автоматично стартира процеса по компилиране.
- 6) Компилацията ще бъде поддържана бърза – ще се изпълнява бързо за да може, ако възникне проблем с интегрирането да бъде идентифициран бързо.
- 7) Тестване в копие на производствената среда – използването на тестова среда може да доведе до провал на внедряването в реалната производствена среда, тъй като е възможно те да се различават по някакъв начин. Изграждането обаче, на копие на производствената среда може да е скъпоструващо. Вместо това пред-производствената среда ще е изградена като мащабируема версия на реалната производствена среда, което води както до намаляване на разходите, така и до запазване на използвания технологичен стек и нюанси.



- 8) Да могат лесно да се вземат най-актуалните артефакти – ако резултатите са налични винаги за заинтересованите лица и тестерите, това може да намали много количеството преработки, необходими за компилиране на новите версии. В допълнение по-ранното тестване намалява шансовете на дефектите да оцелеят до внедряването. Ранното откриване на грешките в някои случаи намалява количеството работа, необходима за отстраняването им.
- 9) Всеки да може да види резултатите от последната компилация – ще е лесно да бъдат намерени без значение дали компилацията е успешна или не, така че да може да бъдат направени необходимите промени.
- 10) Автоматизирано внедряване - повечето CI системи позволяват стартирането на скриптове, след като завършва компилацията. В повечето случаи е възможно да се напише скрипт за внедряване на прилагането на тестовия сървър, така че всеки да може да го погледнете. По-нататъшно подобрене в начина на внедряване може да се реализира като се направи така, че софтуера да бъде внедрен автоматично на сървъра, често съпроводено от допълнително автоматизиране, за да се предотвратят дефектите и загубата на функционалности.

Автоматизирано управление на кода

Автоматизираното управление на кода, известно също като контрол на версиите и контрол на кода (и е един от аспектите на управлението на софтуерната конфигурацията или SCM), е управлението на промените в документи, програми и друга информация, която се съхранява като компютърни файлове. Най-често се използва при разработката на софтуер, където екип от хора могат да се променят едни и същи файлове. Към всяка версия е асоцииран кодов номер, наречен "номер на ревизията". Всяка ревизия е свързана с момент от времето и лицето, което е направило промяната. Ревизиите могат да бъдат сравнявани, възстановени, а някои типове файлове дори и обединявани.

Системите за контрол на версиите (VCS) най-често работят като самостоятелни приложения, но също така са вградени в някои видове софтуер. Софтуерните инструменти за контрол на версиите са от съществено значение за организацията на проектите с много разработчици.

В софтуерната разработка, контрола на версиите е практика, която следи и осигурява контрол над промени в изходния код. Разработчиците на софтуер понякога използват софтуера за контрол на версиите, за да съхраняват документация и конфигурационни файлове, както и изходен код.

Докато екипа проектира, разработва и внедрява софтуера, се инсталират различни версии на един и същи софтуер на различни локации, като и програмистите работят едновременно по обновяванията. Грешки или промени на софтуера често са налични само в определени версии (поради коригирането на грешки и появата на нови с еволюционното развитие на софтуера). Ето защо, за целите на локализирането и отстраняването на грешки е жизнено важно да може да се изтеглят и инсталират различни версии на софтуера, за да се определи в коя версия възниква проблемът. Също така може и да е необходимо да се разработват паралелно две версии на софтуера (например, ако една от версиите има коригирани грешки, но не и нови характеристики (клон), а другата версия е мястото, където се разработват новите характеристики).

В най-елементарно ниво, програмистите биха могли просто да запазят различни копия на програмата и да ги обозначават по подходящ начин. Този прост подход е бил използван при много големи софтуерни проекти. Макар че този метод може да работи, той е неефективен, когато ще се поддържат много, почти идентични копия. Това изисква много самодисциплина от страна на разработчиците и често води до грешки. Като резултат са били разработени системите за автоматизиране на някои или всички от процеса по управление на версиите.

Освен това, в разработката на софтуер, правната и бизнес практики и други среди, става все по-често за един-единствен документ или фрагмент от код, да бъдат редактирани от екип, членовете, на който могат да бъдат географски разпръснати и могат да преследват различни и дори противоречащи интереси. Сложните системи за контрол на версиите, който следи и съхранява информацията за източника на промени в документи и кода могат да бъдат изключително полезни или дори необходими в такива ситуации.

Управление на дистрибуциите

Процесът по управление на дистрибуциите е сравнително нова, но бързо развиваща се дисциплина в рамките на софтуерното инженерство.

Тъй като софтуерни системи, разработката и софтуерните процеси и ресурси стават все по-разпределени, те неизменно стават по-специализирани и по-сложни. Освен това, софтуерните продукти (особено уеб приложения) обикновено са в непрекъснат цикъл на разработка, тестване и внедряване. Добавете към това и еволюцията и нарастващата сложност на платформите, на които тези системи работят, и става ясно, че има много съставни части, които ще работят без проблеми заедно, за да се гарантира успеха и дълготрайната устойчивост на продукта или проекта.

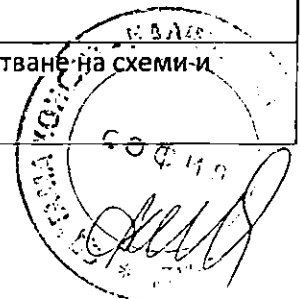
Следователно, съществува нужда от посвещаването на ресурси, които да наблюдават тази интеграция и поток на разработка, тестване, внедряване и поддръжка на тези системи. Въпреки че мениджърите на проекти са правили това в миналото, те обикновено са по-загрижени за аспектите на високо ниво в даден проект, и така често не разполагат с време, за да наблюдават някои от по-техническите или ежедневни аспекти. Управлението на дистрибуциите отговаря на тази необходимост. Те ще имат общи познания на всеки аспект от процеса на разработка на софтуер, различни приложими операционни системи и приложен софтуер или платформи, както и различни бизнес функции и перспективи.

Някои от техниките за управление на дистрибуциите включват управление на:

- Софтуерните дефекти;
- Проблемите;
- Риска;
- Исканията за промяна;
- Исканията за разработка на нови функционалности (допълнителни характеристики);
- Внедряване и пакетиране;
- Нови задачи за разработка.

5.3.9. Описание на инструментите, които ще се ползват от изпълнителя за извършване на разработка, тестване и внедряване

Инструмент	Описание
JetBrains IntelliJ Idea	Среда за разработка
Microsoft Visual Studio	Инструмент за моделиране и изработване на схеми и диаграми

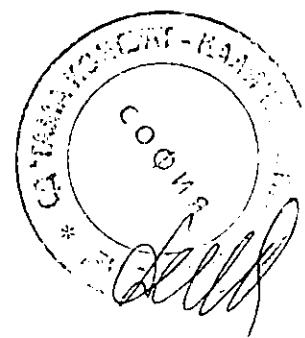


Sparx Systems Enterprise Architect	Специализиран софтуер за моделиране и управление на изисквания: <ul style="list-style-type: none"> - Business Process Modeler; - UML Modeler; - Управление на изискванията.
Microsoft Office Word	Редактор за документи
Microsoft Office Visio	Редактор на векторна графика – създаване на диаграми и схеми за визуализация
Microsoft Office Excel	Електронни таблици
Microsoft Office SharePoint Server	Портал за екипна работа, съхранение и управление на версии на документи
Subversion	Система за контрол на версиите
Oracle Hudson	Специализиран софтуер за създаване на версия на системата (Builds Generator)
Selenium	Автоматизирано тестване на потребителския интерфейс
TestLink	Софтуерна платформа за управление на тестването на софтуерни продукти. Системата предоставя възможност за управление на тестови случаи (test cases), тестови пакети (test suites), тестови планове (test plans), тестови проекти (test projects), а също и разнообразни отчети и статистики.
Atlassian Jira	Възлагане и отчитане на задачи по разработка, тестване и внедряване. Управление на дефекти и несъответствия.

5.3.10. Адаптиране на подхода за софтуерна разработка
Предвиждаме да бъде разработен CISE Adapter и специализиран тестови клиент.

Разработката ще протече в следните итерации:

Handwritten signatures



- 1) Итерация 1:
 - a. Рамка на системата;
 - b. Базова функционалност на CISE Adapter;
 - c. Връзка с CISE Node;
- 2) Итерация 2:
 - a. Интерфейси за съществуващи системи;
 - b. Администриране;
 - c. Тестов клиент;
- 3) Итерация 3:
 - a. Оптимизиране и изчистване на дефекти.

5.4. Подход за тестване

Методологии и добри практики: International Software Testing and Qualifications Board

Разработка на тестови сценарии за проверка на реализацията на системата;

Тестване на разработената функционалност по съответния компонент;

Документиране на резултатите от проведените тестове.

Основни средства и техники: Разнообразни видове тестове

Софтуер: Selenium

Jira

Източници: <https://www.istqb.org>

Обосновка: Ще бъдат приложени утвърдени практики на водещи световни професионални организации в областта на софтуерното тестване. Тестването ще бъде планирано и ще бъдат разработени тестови сценарии, което гарантира покритието на продукта с тестове. Ще бъдат приложени различни видове тестове, чрез които ще се тестват различни аспекти на разработката.

5.4.1. Стратегия за тестване

Тестването е етап на изпълнение, по време на който разработения продукт се проверява за съответствие с техническата и функционална спецификация на проекта. Тестването се съотнася с процеса по контрол на качеството. Тестването се изпълнява паралелно или след процеса по разработка. Задължителен вход за процеса по тестване са резултатите от процеса по разработка. Тестването е тясно свързано с процеса по осигуряване на качеството и се изпълнява едновременно/интегрирано с процесите по анализ, планиране, проектиране, разработване и внедряване. Това осигурява превантивното откриване и ранно отстраняване на грешки.

Тестването ще бъде проведено в три фази:

- Фаза1: Първичен тест – включва проверка на разработените модули и функционалности, изпълнени върху тестова подсистема (среда);



- Фаза 2: Вторичен тест – включва проверка на разработените модули и функционалности, изпълнени върху продукционна и резервна подсистема, след успешно приключване на първичния тест;
- Фаза 3: Тестове за натоварване и надеждност – включва тестове на Системата, които гарантират кохерентната работа на съвкупността от компоненти като тестове за натоварване, тестове за непрекъснатост на работата, тестове за възстановяване на предишна стабилна работеща версия на Системата;

И ще се изпълняват за всеки от етапите на изпълнение на поръчката.

Стратегията за тестване се състои от серия различни тестове, които ще проверят работоспособността на системата, включвайки всички нейни компоненти. Основната цел на тези тестове е да разкрие ограниченията на системата и да измери възможностите и.

Тестването започва след етапа на разработка на тестваема част от завършени функционални изисквания за съответния и компонент и следва да се планира като дейност, която е част от общото изпълнение на проекта и преди приемането на Системата. Процесът по тестване ще бъде итеративен и интегриран с цикъла за софтуерна разработка. Тестовите ще бъдат планирани на базата на одобрените изисквания.

Ще бъде прилагана методологията, която разчита на повтарящи се много къси цикли за разработка (итерации). В итеративния подход се приоритизира реализацията на базата на риска, нуждите на крайните потребители и важността за архитектурата, след което се преминава към по-малките компоненти на системата. Този подход ще се използва през целия последващ цикъл на тестване и внедряване. По този начин тестовия екип получава модули за тестване постепенно и има достатъчно време за тестване преди приключването на проекта. Всяка версия дава възможност за оценка на качеството на крайния продукт. Това също така дава възможност на екипа по разработка да извършва малки промени, когато разходите за прилагането им са ниски. Това намалява риска и увеличава качеството.

Планирането и подготовката на тестовите се извършва преди разработката, веднага след поставянето на изискванията. Още във фазата на проектиране за всички функционални и някои нефункционални изисквания се определят тестовите сценарии, с които те трябва да бъдат покрити. Изготвя се дизайн на това как ще бъде тестван продукта и неговите компоненти. Тестовите се разработват преди да бъдат разработени самите компоненти. При планирането на тестовите се прилага принципа на Парето известен още като „правило 80/20“. Тестовите трябва да се приоритизират и фокуса да бъде върху най-приоритетните причини, които пораждат множеството проблеми. Тестовите на функционалните и нефункционалните изисквания се групират в тестови сценарии. Изпълнението на тестовите сценарии може да бъде автоматизирано или неавтоматизирано. Решение какъв тест да бъде извършен се взема в етапа на планиране на тестването, като решението е базирано на технологичните особености на конкретната функционалност. В хода на самата разработка процента на реализираните и преминалите тестове постепенно нараства. В края на итерацията тестовите за итерацията трябва да са проведени и да са преминали успешно. В края на разработката абсолютно всички тестове трябва да са проведени и да са преминали успешно.

Възприети са различни видове тестове на софтуерните продукти. В зависимост от конкретния случай се използват различни техники за тестване. Техниките на тестване са в съответствие с избраните технологии, използвани за разработка. Изборът на това кой конкретен вид тест да бъде използван зависи от конкретния казус (модул, функционалност и др.). Резултатите от тестовите и докладите за дефекти се регистрират, за да може да се извършва контролна функция.

Приемането от Възложителя ще се извършва, при приключване на разработката на пълната функционалност на всички компоненти на Системата, описани в спецификацията на софтуерните изисквания от изпълнявания етап на поръчката. Резултатите от изпълнението ще бъдат нанесени в съответния протокол.

Списък с планираните видове тестове и кратко обяснение за тях следва по-долу.

5.4.2. Начин на провеждане на тестовете

При разработката се извършват следните дейности, свързани с тестването:

- Разработка на тестови сценарии за проверка на реализацията на системата;
- Тестване на разработената функционалност по съответния компонент;
- Документиране на резултатите от проведените тестове.

Изпълнението на дейностите по разработка трябва да бъдат планирани, както по елементи, така и по дефинирани периодични итерации, включващи разработка и тестване на отделните елементи и компоненти. Тестване и представяне на резултатите на Възложителя ще се извършва за всяка итерация, в оперативен порядък. В случай, че Възложителят има изисквания за корекции по реализираната функционалност, те ще бъдат отразени от Изпълнителя до представяне на следващата итерация по разработка на съответния компонент. На база извършената разработка ще бъде извършена интеграция и тестване в цялостното технологично решение на системата.

5.4.3. Видове тестове

В хода на тестовия процес ще бъдат използвани следните видове тестове, за функционални, нефункционални и приемателни тестове :

Тестове за проверка на изискванията

Това е преглед на спецификацията на изискванията. Целта е да се установи дали те отговарят на изискванията на Възложителя, описани в техническата спецификация и дефинирани в обхвата на проекта. Прегледа на изискванията се извършва от екипа по осигуряване на качеството при изготвяне на спецификацията на изискванията и се одобрява от Възложителя при предаването и.

Тестове за проверка на архитектурата

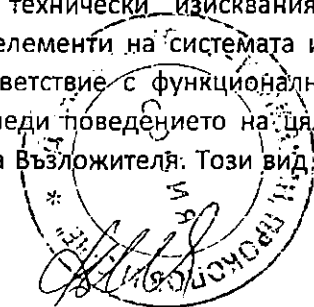
Архитектурата е техническата концепция за реализацията на изискванията. Може да се опише и като проект на системата на по-високо ниво. Архитектурата ще бъде тествана за съответствие със заданието и цялостните изисквания за архитектурата на Системата.

Функционално тестване

С провеждането на функционалните тестове се цели да се съпоставят реалните действия и състояние на системата с тези описани във функционалната спецификация. Целите на тези тестове са да се провери правилното приемане, обработка и извличане на данни, и правилното осъществяване на бизнес процесите. При наличие на разминавания с функционалната спецификация следва да се направят предложения и препоръки за тяхното отстраняване. Този вид тестване ще се провежда за всяка фаза, за всеки етап и компонент от разработката на системата.

Системно тестване

Системното тестване започва след като всички функционални и технически изисквания са имплементирани. Целта е да бъдат проверени всички функционални елементи на системата и да гарантира, че системата изпълнява описаните бизнес процеси в съответствие с функционални и технически изисквания. Този начин на тестване позволява да се проследи поведението на цялата системата и всичките и компоненти и дали тя отговаря на очакванията на Възложителя. Този вид тест ще се проведе за Фаза 1: Първичен тест и Фаза 2: Вторичен тест.



Интеграционно тестване

Интеграционните тестове ще бъдат изпълнявани, за да се гарантира, че компонентите в изпълнението на системите работят коректно след като са комбинирани, за да изпълняват определена функционалност, както и възможността за обмен на информация с други информационни системи, които са в обхвата на проекта. Целта е компонентите или множествата от компоненти да бъдат проверени за незавършеност или грешки при реализацията. Най-често това са грешки в интерфейсите между компонентите. Този вид тест ще се проведе за Фаза1: Първичен тест.

Тестове за натоварване и производителност

Тестовите за натоварване и производителност целят да докажат приемлива производителност (съгласно заложените параметри за производителност в спецификацията) в смисъла на изпълнение и време за отговор на компонента, като се има предвид очаквания брой потребители на всеки един от компонентите на системата. Целта на теста е да се проверят определени функционални изисквания или бизнес процеси при предварително определени условия, като се проследят данните за производителността на компонента, чрез симулация на очаквания брой потребители на системата, които да я натоварят. По време на тестовите ще бъде измерено времето за отговор на компонента. Този вид тест ще се проведе за Фаза 3: Тестове за натоварване и надеждност.

Тестове за сигурност

Тестовите за сигурност ще определят колко е сигурна новата система. Тестовите трябва да докажат, че е възпрепятстван неоторизиран достъп до поверителна информация, промяна или унищожаване на данните.

Бета тестове

Целта на бета тестовите е да се провери дали софтуерът е готов и може да бъде използван от крайните потребители за изпълнение на техните функции и задачи, за което продуктът е създаден. Бета тестовите ще започнат на етап, при които софтуера е функционално напълно завършен, но все още има вероятност да има неточности. Целта е крайните потребители да имат възможност да проверят дали той работи коректно спрямо функционалните спецификации на мястото, на което работят и в средата, където се очаква да работи системата. Основна роля при този вид тестване имат потребителите на системата. Бета тестовите за Фаза1: Първичен тест ще се проведат в предварително подготвена тестова подсистема, а бета тестовите за Фаза 2: Вторичен тест ще се проведат на продукционната и резервната подсистеми.

Тестове за използваемост

Тестовите за използваемост ще определят колко лесно и интуитивно е използването на разработената система от гледна точка на потребителски интерфейс и с каква лекота биха могли крайните потребители да постигнат специфичните си цели. Тестовите за използваемост включват и оценка за ефективността и бързината, с която потребителите биха могли да завършат задачите си в приложението, след като са запознати с начина му на ползване. Основна роля при този вид тестване имат потребителите на системата. Този вид тест ще се проведе за Фаза 1: Първичен тест.

Тестове за възстановяване

Тестовите за възстановяване целят да проверят колко бързо и ефективно може да се възстанови работа при:

- авария на продукционната подсистема, в които се налага прехвърляне на обслужването на потребителски заявки към резервната подсистема и обратно;
- загуба на комуникация между продукционната и резервната подсистема;

- възстановяване от архив;
- възстановяване на предишна стабилна работеща версия на Системата.

Този вид тест ще се проведе за Фаза 3: Тестове за натоварване и надеждност.

Приемни тестове

Приемните тестове ще бъдат последното тестово действие преди внедряване на системата. Целта е да се изгради доверие в софтуера, да се оцени дали системата е готова за внедряване и да се удостовери, че за всички бизнес правила има реализирана функционалност и тя отговаря на функционалната спецификация. Приемните тестове ще се проведат след успешното протичане на трите фази от тестването Фаза 1: Първичен тест, Фаза 2: Вторичен тест и Фаза 3: Тестове за натоварване и надеждност за всеки един от етапите на изпълнението на поръчката.

За провеждането им ще бъдат подготвени предварително тестови сценарии, ще се провеждат с участието на Възложител и Изпълнител и ще завършват с подписване на протокол.

5.4.4. Входни и изходни критерии

Критерии за започване на тестовия процес

Критерий за започване на тестовия процес е всички необходими документи по проекта да са изготвени от Изпълнителя и одобрени от Възложителя.

Критерии за преустановяване на тестването

Ако се открият дефекти, които сериозно се отразяват на напредъка на тест, Ръководител тестване може да избере да спре тестването. Критерии, които ще оправдаят преустановяване на тестването са:

- Хардуера/софтуера, необходими за тестването не са налични;
- Разработената функционалност има един или повече критични дефекти (бъгове), които сериозно възпрепятства или ограничава тест прогреса;
- Ресурсите определени за тестване не са на разположение, когато са необходими на testerите;
- Постъпило е искане за промяна, което засяга функционалностите, които са застъпени в настоящата итерация.

Критерии за подновяване на тестването

Ако тестването е преустановено, подновяването му ще се случи, след като причините предизвикали спирането са отстранени. Когато критичен дефект е причината за спирането, отстраняването му трябва да бъде проверено.

Критерии за спиране на тестовия процес

При покриването на следните критерии се счита, че тестовия процес е преминал успешно:

- ✓ 100% от планираните тестови сценарии трябва да са изпълнени;
- ✓ 100% от всички документирани критични програмни грешки и дефекти трябва да бъдат затворени;
- ✓ 100% от всички документирани програмни грешки и дефекти с високо въздействие трябва да бъдат затворени;
- ✓ Фаза 1 : Първичен тест е изпълнен успешно;
- ✓ Фаза 2 : Вторичен тест е изпълнен успешно;

Koloznak

[Handwritten signature]



927

- ✓ Фаза 3: Тестове за натоварване и надеждност са изпълнени успешно;

Критерии за приемане

Системата и нейните компоненти ще се приемат в експлоатация когато:

- ✓ Отговарят на изискванията на настоящата техническа спецификация и на утвърдената от Възложителя методология за изпълнение на дейностите;
- ✓ Успешно са преминали всички дефинирани тестове;
- ✓ Не произвежда неправилни резултати (и не изпада в състояние да не произведе резултат) при правилни входни данни;
- ✓ Не нарушава целостта си и целостта на съхраняваната информация в следствие на некоректни входни данни, програмни или други грешки;
- ✓ Не изпада в недетерминирани състояния ("блокира", "заспива") в следствие на некоректни входни данни, програмни и други грешки или продължителна работа;
- ✓ Няма логически грешки;
- ✓ Няма грешки или съществен спад на производителността, проявяващи се по време на претоварване, увеличаване на капацитета на базата от данни или автоматични действия по архивиране, индексирание и т.н.;
- ✓ Няма грешки, зависещи от настъпването и взаимодействието на асинхронно възникващи събития, както и от забавянето на отговора/реакцията на други приложения;
- ✓ Няма грешки, проявяващи се след системен срив или системно аварийно възстановяване след изключителни събития (напр. отпадане на захранването или апаратна повреда);
- ✓ Съществуват показания, че информационната система може да обработи големи обеми данни без значителен спад на производителността;
- ✓ Произвежда резултат в очакваното време за отговор;
- ✓ Успешно са изпълнени Фаза 1 : Първичен тест, Фаза 2 : Вторичен тест и Фаза 3: Тестове за натоварване и надеждност.

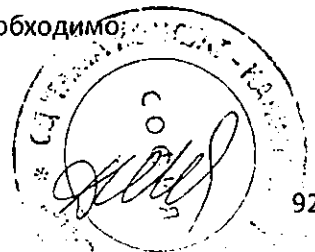
5.4.5. Докладване на дефекти и несъответствия

Когато са открити дефекти, тестерите ги отразяват в система за проследяване на дефекти като използват следните атрибути:

- Заглавие – кратко и ясно описание на проблема;
- Стъпки за възпроизвеждане – описание на пътя за проявяване на проблема;
- Резултат – действителния резултат, получен при изпълнението на тестовия сценарий;
- Очакван резултат – резултатът, който е съобразен със спецификацията на продукта;
- Въздействие – въздействието на проблема върху работоспособността на системата;
- Среда – средата (подсистемата), на която е открит проблема, ако е необходимо;
- Прикачени файлове (снимка, видео, файл) – ако е необходимо.

Handwritten signature

Handwritten signature



Ръководителят на екипа следи и анализира вписаните дефекти и по негова преценка ги обработва по един от следните начини:

- Коригирането му се възлага като задача за изпълнение;
- Връща го за допълнително изследване към екипа по осигуряване на качеството като посочва причината за връщането. В този случай статусът се променя на "Изчакващ";
- Приема, че не е дефект (ако като дефект е вписано нормално поведение на системата).

Коригирането на дефекта се възлага като обособена задача и управлението и се осъществява на база правилата за управление на задачи и статусът на дефекта се променя на "В процес на поправка". След завършване на задачата статусът на дефекта се променя на "Разрешен". Тестер задължително го верифицира и ако установи, че дефектът е отстранен го маркира като „Затворен“ (т.е. отпада от списъка на текущите дефекти).

Система за проследяване на дефекти трябва да е достъпна до тестери, разработчици и всички членове на екипа по проекта.

Дефектите се регистрират в специализирана система за управление на дефекти и несъответствия, базирана на Atlassian Jira.

5.4.6. Инструменти за провеждане на тестовете

Selenium	Автоматизирано тестване на потребителския интерфейс
TestLink	Софтуерна платформа за управление на тестването на софтуерни продукти. Системата предоставя възможност за управление на тестови случаи (test cases), тестови пакети (test suites), тестови планове (test plans), тестови проекти (test projects), а също и разнообразни отчети и статистики.
Atlassian Jira / Team Foundation Server	Възлагане и отчитане на задачи по разработка, тестване и внедряване; Управление на дефекти и несъответствия.

5.4.7. Адаптиране на подхода за тестване

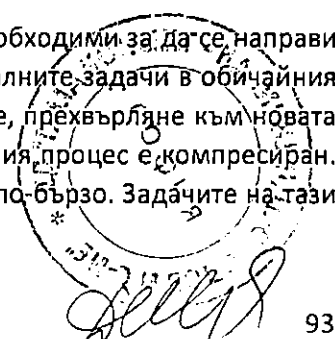
За изпълнение на тестването ще бъде разработен тестови клиент. Ще се използва и олекотен клиент и симулатор за кораби, предоставени от Възложителя. Ще бъде поставен фокус върху тестването на интеграция с други системи.

Поради минималния потребителски интерфейс, тестове за използваемост няма да се използват.

5.5. Подход за внедряване

5.5.1. Общи положения

Внедряването на софтуер е процес, обхващащ всички дейности, които са необходими за да се направи софтуерния продукт достъпен за използване. Внедряването обобщава финалните задачи в обичайния процес по софтуерна разработка, включително миграция на данни, тестване, прехвърляне към новата система и потребителски обучения. Сравнено с традиционните методи, целият процес е компресиран. Като резултат, новата система е изградена, доставена и пусната в действие по-бързо. Задачите на тази



фаза са миграция на данни, тестване на цялата система, преход към новата система и потребителски обучения.

Целта на фазата за Внедряване е да направи софтуера достъпен за неговите потребители. Фазата на внедряване може да премине през няколко итерации и включва тестване на продукта, подготовка за нови версии, корекции базирани на обратната връзка от потребителите. На този етап от жизнения цикъл обратната връзка от потребителите трябва да е фокусирана основно върху прецизни настройки, конфигуриране, инсталиране и използваемост. Всички по-съществени структурни въпроси трябва да бъдат решени в по-ранните фази на проекта.

Като цяло процесът за внедряване се състои от няколко взаимосвързани дейности, както и възможните преходи между тях. Тези дейности могат да се извършват в офисите на възложителя, на изпълнителя или и на двете места.

Тъй като всяка софтуерна система е уникална не могат да се дефинират точни процеси или процедури за внедряване. Поради това внедряването трябва да се тълкува като общ процес, който трябва да бъде персонализиран според специфичните изисквания или особености.

5.5.2. Стъпки за внедряване

Ще бъде приложен процеса по внедряване, заложен в Rational Unified Process, според който се извършват следните стъпки:

1) Подготовка:

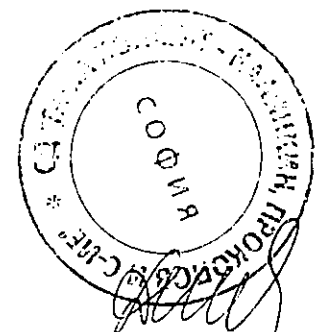
- 1.1) Предварителен анализ на готовността за внедряване
- 1.2) Предварително събиране на необходимата за внедряването изходна информация от Клиента.
- 1.3) Планиране на внедряването
- 1.4) Съгласуване на плана за внедряване
- 1.5) Предварително запознаване на Възложителя и страните с ангажиментите им по внедряването
- 1.6) Разработване на подробно постъпково ръководство за внедряване
- 1.7) Обучение на екипа от внедрители
- 1.8) Изготвяне на материали за обучения
- 1.9) Изготвяне на документация
- 1.10) Разработване на тестови сценарии за интеграционни и приемни тестове
- 1.11) Разработване и съгласуване на процедури за поддръжка
- 1.12) Организиране на help-desk

2) Изпълнение на внедряването:

- 2.1) Инсталиране и конфигуриране на техническата инфраструктура
- 2.2) Инсталиране и конфигуриране на базовия софтуер
- 2.3) Инсталиране и конфигуриране на подсистемите
- 2.4) Въвеждане на начални данни
- 2.5) Конфигуриране на интеграционните интерфейси
- 2.6) Провеждане на интеграционни тестове
- 2.7) Оптимизиране на системата
- 2.8) Провеждане на приемни тестове
- 2.9) Обучение на потребителите и администраторите
- 2.10) Стартиране на работа

3) Последващи действия:

- 3.1) Подпомагане на работата със системата



3.2) Поддържане на help-desk за внедрителските екипи

3.3) Управление на версиите на подсистемите

При стартирането на работа със системата, екипи на Изпълнителя ще подпомагат работата със системата:

- На място;
- Чрез портал за подпомагане на потребителите;
- Чрез гореща телефонна линия за подпомагане на потребителите.

Инсталациите и конфигурациите ще бъдат извършени с участие на администратори от страна на Възложителя така, че да бъде извършено предаване на знания.

Допълнително ще бъдат обучени ключовите потребители на системата. Те ще имат осигурен достъп до тестова среда така, че да придобият увереност за работа със системата преди въвеждането и в експлоатация.

5.6. Механизъм и процедури за включване на граждани в процесите по разработка, тестване и идентифициране на пропуски на софтуера

Изпълнителят ще извърши анализ на възможностите за включване на граждани в процесите по разработка, тестване и идентифициране на пропуски на софтуера. Този анализ ще бъде изпълнен по начин, даващ възможност за активно участие от страна на гражданите и събиране на информация с цел обратна връзка за функционалната реализация на разработеното приложение. Като предварителна визия предлагаме да бъдат използвани следните механизъм и процедури за реализирането на такива процеси:

- Да бъдат публикувани за достъп от граждани спецификациите и ръководствата, резултат от изпълнението на поръчката (по преценка на Възложителя).
- Да бъдат публикувани спецификациите за интеграция и протоколите за обмен на данни, така че да бъде възможно свързването към CISE Adapter (по преценка на Възложителя).
- Да бъде публикувана в Интернет точка за достъп до тестова инстанция на интеграционния интерфейс на адаптера (по преценка на Възложителя).
- Изпълнителят ще предаде пълният изходен код на разработката. По преценка на Възложителя, той може да бъде публикуван в хранилището, поддържано от Държавна агенция „Електронно управление“.
- Ще бъде осигурен канал за получаване на обратна връзка от граждани чрез специален адрес на електронна поща.

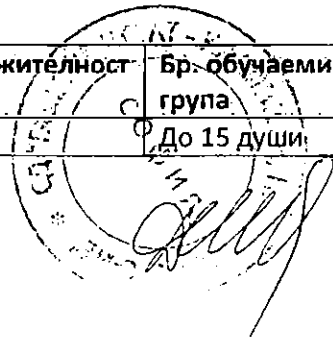
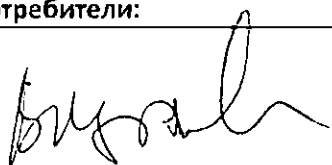
6. Обучение

6.1. Учебни курсове

Изпълнителят ще организира и ще проведе обучения за ключови потребители и администратори на Националния EUCISE2020 Адаптор.

Ще бъдат проведени следните учебни курсове:

Учебен курс	Продължителност	Бр. обучаеми в група
Обучение за ключови потребители:	1 ден	До 15 души



<ul style="list-style-type: none"> • Работа с Адаптора; • Интеграция на Адаптора с VTMISS; • Интеграция на Адаптора с EUCISE2020 Node; • Настройки и конфигурация; • Процедура за поддръжка. 		
Обучение за администратори: <ul style="list-style-type: none"> • Инсталиране; • Базов софтуер и ИТ инфраструктура; • Настройки и конфигурация; • Възможни проблеми и техните решения; • Процедура за поддръжка. 	1 ден	До 10 души

За провеждането на обученията Изпълнителят ще осигури за своя сметка:

- Необходимия хардуер;
- Необходимия софтуер;
- Зала/Зали за провеждане на обученията;
- Учебни материали;
- Лектори.

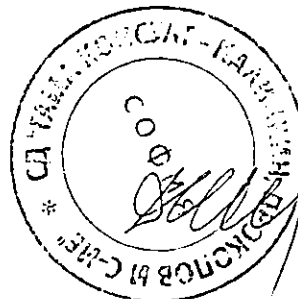
6.2. Начин на провеждане на обучението

При провеждане на обученията ще използваме семинарна форма на обучение, която ще бъде комбинирана с практически упражнения. Обучението ще се проведе със системата, която ще бъде въведена в експлоатация.

Обучението ще бъде проведено в две части - групово (посредством мултимедийна презентация) и индивидуално - чрез реална работа със системата. Учебният материал при груповото обучение ще бъде представен като поредица от модули. Всеки модул ще съдържа лекции, демонстрации с внедрените технологии и разработената система, споделяне на добри практики при работа със системата. Втората част от модула ще се състои от лабораторни занятия, които всеки обучаем трябва да изпълни индивидуално. В края на част от модулите ще бъде обсъдена примерна ситуация за конкретен потенциален случай при работа с внедрените технологии и разработената система. Ситуациите следва да бъдат разработени в зависимост от конкретните категории потребители, работещи със системата.

Примерна дневна програма за обучение:

Сесия	Време
Първа сесия	9:00 – 10:30
Кафе пауза	10:30 – 10:45
Втора сесия	10:45 – 12:00
Обяд	12:00 – 13:00
Трета сесия	13:00 - 15:15
Кафе Пауза	15:45 – 16:00
Обобщаване на материала и решаване на казуси	16:00 – 16:30
Въпроси	16:30 – 17:00



Обучението ще се проведе на територията на гр. София в учебен център, осигурен от Изпълнителя.

6.3. Учебни материали

Преди провеждането на обучението, ще предоставим всички необходими материали, които обучаемите ще ползват по време на обучението. Тези материали ще бъдат предоставени на хартиен или електронен носител в зависимост от необходимостта и обема по време на обучението.

В рамките на изпълнение на поръчката, ще подготви учебните материали в съответствие с ролята на служителите (оператори или администратори), в електронен и хартиен вид на български език:

- Презентации и ръководства за потребителите;
- Презентации и ръководства за инсталиране, конфигуриране, поддръжка и администриране на системата.

По време на обучението, на обучаемите ще предоставим всички необходими материали за провеждането му.

Материалите ще бъдат изготвени по начин, който да позволи те да се използват и за последващо опресняващо знанията и допълващо самостоятелно обучение на служители на Възложителя.

7. Документация

7.1. Изисквания към документацията

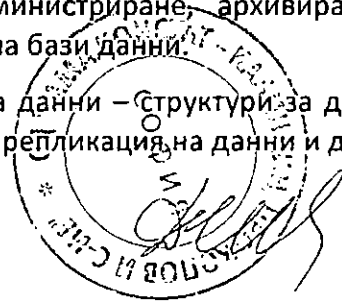
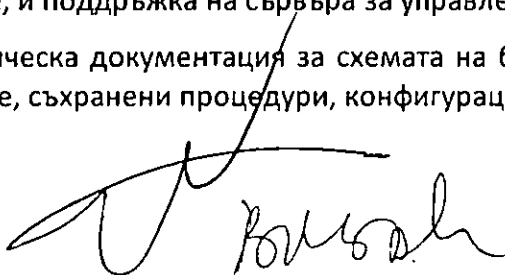
Цялата документация и всички технически описания, ръководства за работа, администриране и поддръжка на Адаптора, включително и на нейните съставни части, ще бъдат налични и на български език.

Всички документи ще бъдат предоставени от Изпълнителя в електронен формат (ODF/ /Office Open XML/MS Word DOC/RTF/PDF/HTML или др.), позволяващ пълнотекстово търсене/търсене по ключови думи и копиране на части от съдържанието от оригиналните документи във външни документи, за вътрешна употреба на възложителя;

Навсякъде, където в документацията има включени диаграми или графики, те ще бъдат вградени в документите в оригиналния си векторен формат;

Изпълнителят ще изготви и предостави на Възложителя:

- Детайлна техническа документация на програмния приложен интерфейс (API), включително за поддържаните услуги, команди, структури от данни и др.
- Детайлна техническа документация за схемата на базата данни – структури за данни, индекси, дялове, съхранени процедури, конфигурации за репликация на данни и др.
- Ръководства на потребителя и администратора за работа и администриране на Адаптора
- Обща информация, инструкции и процедури за администриране и поддръжка на приложните сървъри, сървърите за бази данни и др.
- Обща информация, инструкции и процедури за администриране, архивиране и възстановяване, и поддръжка на сървъра за управление на бази данни.
- Детайлна техническа документация за схемата на базата данни – структури за данни, индекси, дялове, съхранени процедури, конфигурации за репликация на данни и др.



Цялата документация по изпълнението ще отговаря на определени стандарти и утвърдени добри практики по отношение на оформление, структура, контрол на версиите, критерии за проследимост и контрол на качеството на документите. Изпълнителят ще предложи стандарти за подготовка на документация, съобразени с предлаганата от него проектна методология.

7.2. Прозрачност и отчетност

Всички документи ще се изготвят на български език и се предоставят в един хартиен екземпляр и на електронен носител.

7.3. Спецификация на софтуерните изисквания

Изпълнителят на настоящата поръчка ще дефинира в детайли конкретния обхват на реализация на софтуерната разработка и ще документира изискванията към софтуера в детайлна техническа спецификация (спецификация на софтуерните изисквания), която ще послужи за пряка изходна база за разработка.

При документирането на изискванията, с цел постигане на яснота и стандартизация на документите, ще се използва утвърдена нотация за описание на бизнес модели. Изготвената детайлна техническа спецификация (спецификация на софтуерните изисквания) ще бъде представена за одобрение на Възложителя. В случай на забележки, корекции или допълнения от страна на Възложителя Изпълнителят е ще ги отрази в детайлната техническа спецификация (системен проект).

7.4. Техническа документация

При изпълнение на дейностите по настоящата поръчка, изпълнителят ще изготви следната техническа документация, съдържаща:

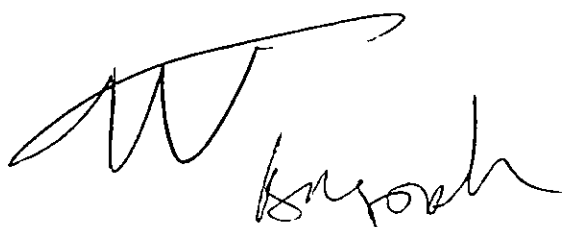
- Ръководство за инсталиране, конфигуриране и администриране;
- Ръководство за потребителите;
- Детайлно техническо описание на софтуерните модули и базата данни;
- Описание на изходния програмен код

7.5. Протоколи

Изпълнителят ще изготвя протоколи от изпълнението на различните етапи на проекта заедно със съпътстващите ги документи – резултати от изпълнението на етапите.

7.6. Комуникация и доклади

Виж. по-горе т. 3.2 – Управление на комуникацията.



към

ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

за участие в открита процедура за възлагане
на обществена поръчка с предмет:

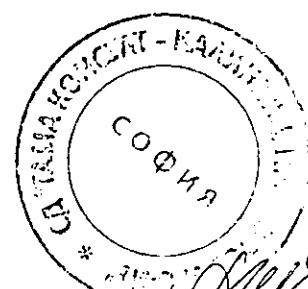
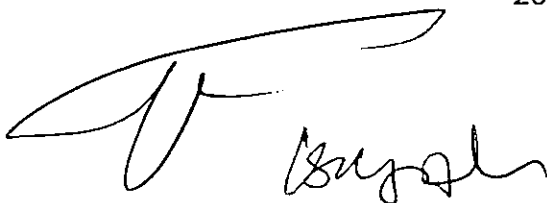
**„Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в
съответствие с инженеринговите услуги, поддържащи прилагането на
EUCISE2020 мрежата и съпровождащи услуги по време на
демонстрационната фаза на обмен на данни“**



Предложение за управление на риска при
изпълнение на поръчката



2018 г.



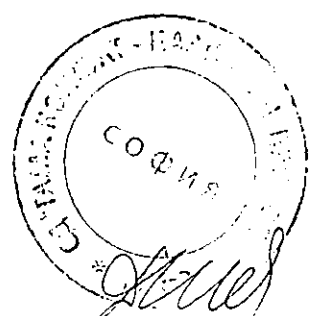
Съдържание

1. Въведение.....	3
2. Подход за управление на риска	3
2.1. Планиране на управлението на риска	4
2.2. Идентифициране на рисковете	4
2.3. Качествен анализ на риска.....	4
2.4. Количествен анализ на риска.....	4
2.5. Планиране на ответни действия	8
2.6. Наблюдение и контрол на рисковете	9
3. План за управление на риска.....	10
3.1. Процес.....	10
3.2. Роли и отговорности	10
3.3. Таксономия и категоризиране на рисковете.....	11
3.4. Таксономия на рисковете при софтуерна разработка.....	11
3.5. Регистър на рисковете	13
3.6. Инструменти за управление на риска	15
4. Конкретни рискове и предпоставки за успешно изпълнение на поръчката	16
4.1. Анализ на конкретни рискове, идентифицирани от Възложителя.....	16
4.2. Анализ на допълнителни, относими към поръчката рискове	25
5. Предпоставки за успешно изпълнение на поръчката	34

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]



1. Въведение

В настоящото приложение 2 към Техническото си предложение сме представили:

- Подход за управление на риска, който ще прилагаме при изпълнението на поръчката;
- Анализ на идентифицираните в Техническата спецификация на Възложителя възможни рискове;
- Анализ на идентифицирани от нас допълнителни, относими към поръчката рискове;
- Предпоставки за успешно изпълнение на поръчката.

Рисковете са анализирани по отношение на вероятността от тяхното настъпване, въздействието им върху изпълнението на дейностите и сферата им на влияние. Предложени са мерки и обосновка за предотвратяване или минимизиране на влиянието за всеки един от рисковете. Предложените мерките са аргументирани, приложими и изпълними за премахване/минимизиране на всеки един от посочените рискове.

2. Подход за управление на риска

През времето за изпълнение на проекта Изпълнителят ще следи рисковете, ще оценява тяхното влияние, ще анализира ситуацията и ще идентифицира (евентуално) нови рискове. В хода на изпълнение на поръчката Изпълнителят ще докладва състоянието на рисковете на месечна база.

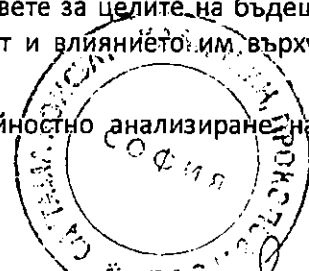
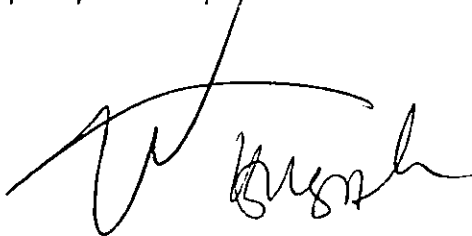
При управлението на проекта ще бъде използвана практикуваната при нас система от правила, принципи и стандарти на Института за Управление на Проекти (Project Management Institute, PMI), систематизирани в Ръководство за Система от Знания за Управление на Проекти (PMBOK Guide) – пето издание. Това включва и Система за управление на риска, която е представена по-долу.

Рискът се отнася до отклонението от един или повече резултати на едно или повече бъдещи събития от тяхната очаквана стойност. Технически, стойността на тези резултати може да е позитивна или негативна. Положителният риск се разглежда като възможност, а при общата употреба на думата риск се фокусира само върху потенциалната вреда (загуба на позитивни резултати), която може да възникне от бъдещо събитие, което да произтече или от влизане в разноси ("риск от снижение" - на английски: downside risk), или от неспособност да се придобие някаква печалба ("риск на изкачването" - на английски: upside risk). Рискът по дефиниция е събитие, което е извън нашия пряк контрол и може да повлияе отрицателни или положително върху постигането на целите на проекта.

Управлението на риска в проекта включва процесите по планиране, идентификация, анализ, планиране на ответните действия и наблюдение и контрол на рисковете. Целта на управлението на риска в проекта е да се повиши вероятността и влиянието на позитивните събития и да се намали вероятността и влиянието от негативните събития в проекта.

Процесите по управление на риска в проекта включват:

- **Планиране на управлението на риска** – Процесът на определяне как ще се изпълняват дейностите по управление на риска в проекта.
- **Идентифициране на рисковете** – Процесът на определяне кои рискове могат да повлияят на проекта и документиране на техните характеристики.
- **Качествен анализ на риска** – Процесът на приоритизиране на рисковете за целите на бъдещ анализ, както и действията по оценка на вероятността да се случат и влиянието им върху проекта.
- **Количествен анализ на риска** – Процесът на количествено и стойностно анализиране на ефекта върху проекта при случване на риска.



3
134

- **Планиране на ответни действия** – Процесът на разработване на опции и действия за повишаване на благоприятните възможности и намаляване на заплахите пред проектните цели.
- **Наблюдение и контрол на рисковете** – Процесът на изпълнение на планираните ответни действия, проследяване на идентифицираните рискове, наблюдение на остатъчните рискове, идентифициране на нови рискове и оценка на процесите по управление на риска в проекта.

Описаните процеси си взаимодействат с останалите процеси по управление на проекта. По всеки от процесите работят един или повече хора в зависимост от спецификата на проекта. Всеки от процесите се изпълнява поне веднъж във всеки проект и се появява в една или няколко от фазите на проекта.

Въпреки, че процесите са описани като дискретни елементи с ясно обособен вход и изход, на практика те се допълват и си взаимодействат.

Проектният риск винаги е в бъдещето. Рискът е несигурно събитие или условие, което ако се случи, ще повлияе върху поне една от целите на проекта. Целите включват обхват, график, качество и цена. Рискът може да се случи поради една или множество причини и ако се случи може да повлияе на проекта по един или няколко начина. Причините могат да се състоят в появата на изискване, допускане, ограничение или условие, което създава вероятност за позитивен или негативен резултат.

Ако някое неочаквано събитие се случи, но може да повлияе върху разходите, графика или изпълнението. Потенциални рискове следват и от приетите практики – недостатъчно зрял процес за управление на проекти, липса на интегрирана система за управление, множество конкурентни проекти, зависимости от външни участници, които не могат да бъдат контролирани.

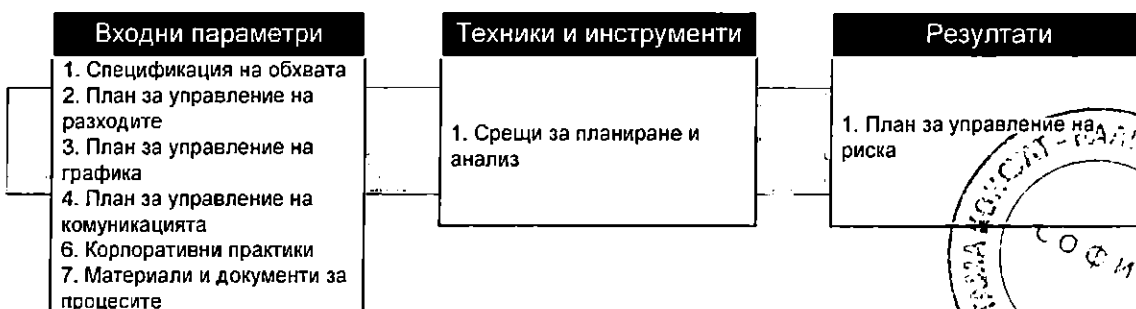
Проектният риск произтича от несигурността, съществуваща във всеки един проект. Известни рискове са тези, които са били идентифицирани и анализирани, което прави възможно да се планират ответни действия.

Организациите възприемат риска като ефект на несигурност на техните проекти или цели. Организациите и мениджмънта са склонни да приемат различна степен на риск. Това се нарича толеранс към риска. Рисковете, които са заплаха към проекта могат да бъдат приети, ако са в този толеранс, и ако и приемането на риска ще донесе достатъчно полза. Например, приемането на график с множество дейности, изпълнявани паралелно е риск, който се приема заради ползата от по-ранна дата на завършване.

За да бъдат успешни, организациите трябва да приемат, че ще управляват риска проактивно и непрекъснато през целия проект. Рискът съществува в момента, в който възниква проекта. Управлението на риска е динамичен процес, който се изпълнява в хода на целия проект. Проектните рискове могат да се променят в хода на проекта, като съществуващи рискове да отпаднат, да бъдат идентифицирани нови рискове, както и да се променя състоянието на съществуващи такива. Списъкът с актуални рискове следва да се актуализира посредством анализ на периодичните отчети по проектите. Напредването на проекта без фокус върху проактивното управление на риска води до повишаване на влиянието, което рискът може да окаже при възникването си и потенциално може да доведе до провал на проекта.

Несъобразяване с кой да е **КРИТИЧЕН ФАКТОР ЗА УСПЕХ** генерира допълнителен риск.

2.1. Планиране на управлението на риска



[Handwritten signature]

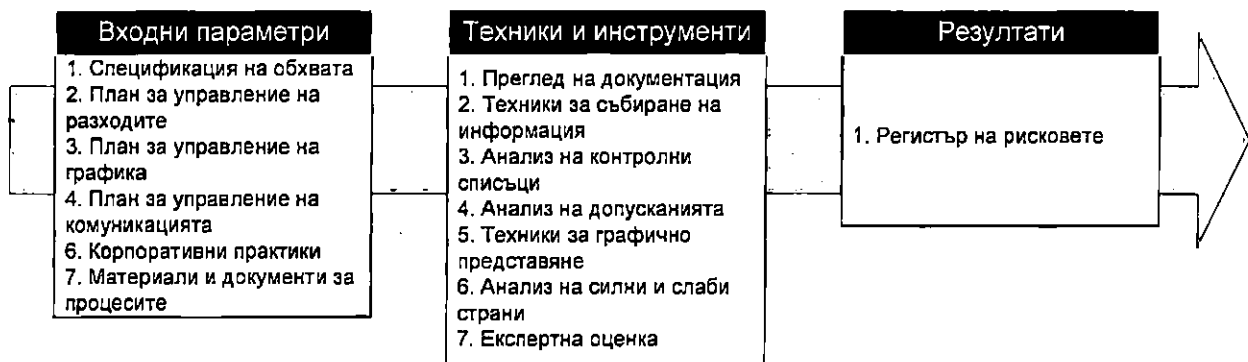


Планиране на управлението на риска е процесът на определяне как ще се изпълняват дейностите по управление на риска в проекта. Внимателното и посветено планиране повишава вероятността за успех на останалите процеси за управление на риска. Планирането на процесите за управление на риска е важно, за да се осигури, че степента, типа и подхода за управление на риска са адекватни на потенциалните рискове и важността на проекта за съответната организация.

Планирането също е важно за осигуряването на достатъчно ресурси и време за дейностите по управление на риска. Процесът по планиране управлението на риска трябва да стартира с възникването на проекта и трябва да се изпълнява заедно с останалото планиране по проекта.

- **Входни параметри**
 - Спецификация на обхвата
 - План за управление на разходите
 - План за управление на графика
 - План за управление на комуникацията
 - Практики
 - Материали и документи за процесите
- **Техники и инструменти**
 - Среци за планиране и анализ
- **Резултати**
 - План за управление на риска

2.2. Идентифициране на рисковете



Идентифициране на рисковете е процесът на определяне кои рискове могат да повлияят на проекта и документиране на техните характеристики. Участници в процеса на идентификация могат да бъдат: проектния ръководител, членовете на проектния екип, екипът за управление на риска, клиенти, експерти в предметната област, експерти извън проектния екип, други проектни ръководители, заинтересовани лица и експерти по управление на риска.

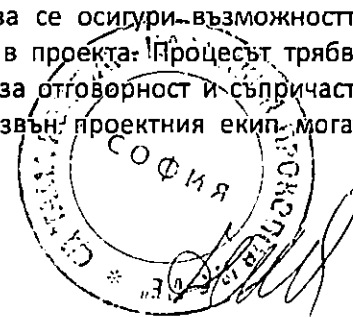
Въпреки, че въпросните хора са ключови участници в идентификацията на риска, всички членове на екипа се окуражават да участват в идентифицирането на рисковете.

Идентифицирането на рисковете е итеративен процес т.к. през целия цикъл на проекта е възможно да възникват нови рискове или да стават известни съществуващи рискове. Честотата на итерациите и участниците в процеса зависят от конкретната ситуация.

Форматът на описване на рисковете трябва да бъде еднотипен, за се осигури възможността за сравнение на ефекта от риска спрямо този на останалите рискове в проекта. Процесът трябва да включва участие на проектния екип, така че той да развие чувство за отговорност и съпричастност спрямо рисковете и ответните действия. Заинтересованите лица извън проектния екип могат да предоставят допълнителна обективна информация.

- **Входни параметри**

Handwritten signature



- Спецификация на обхвата
- План за управление на разходите
- План за управление на графика
- План за управление на комуникацията
- Корпоративни практики
- Материали и документи за процесите

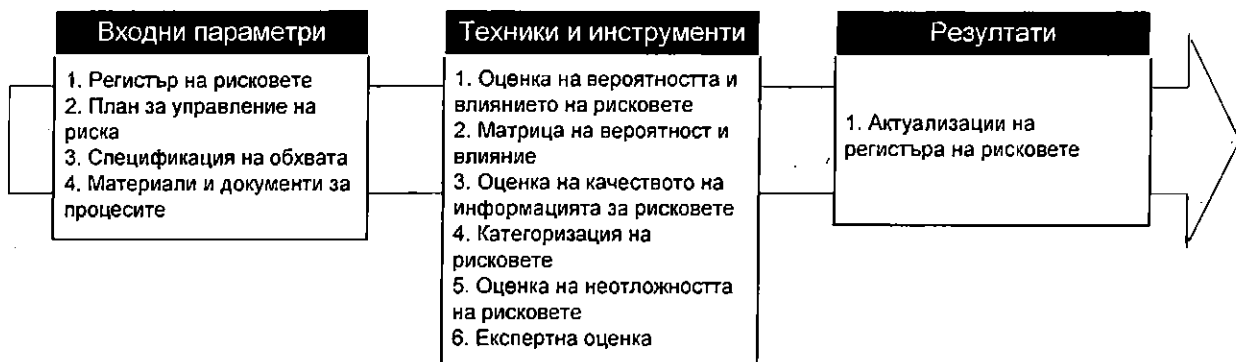
- **Техники и инструменти**

- Преглед на документация
- Техники за събиране на информация
- Анализ на контролни списъци
- Анализ на допусканията
- Техники за графично представяне
- Анализ на силни и слаби страни
- Експертна оценка

- **Резултати**

- Регистър на рисковете

2.3. Качествен анализ на риска



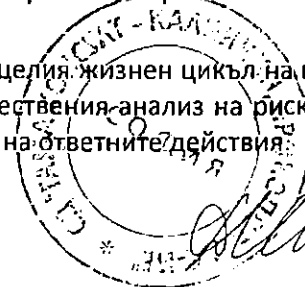
Качествен анализ на риска е процесът на приоритизиране на рисковете за целите на бъдещия анализ, както и действията по оценка на вероятността да се случат и влиянието им върху проекта. Организациите могат да подобрят изпълнението на проекта чрез фокусиране върху рисковете с висок приоритет. При качествения анализ на риска се оценява приоритета на идентифицираните рискове, като се вземе предвид вероятността за случване на риска, степента на влияние върху проектните цели, а също и други фактори като време за реакция, приет от организацията толеранс към риск и приетите ограничения на проекта – цена, график, обхват и качество.

Тези оценки се отразяват върху отношението на проектния екип и другите заинтересовани лица към риска. И т.к. отношението към риска води до отклонение при анализа на идентифицираните рискове, трябва да се обърне специално внимание върху оценката на това отклонение и коригирането му.

Установяването на определени нива на вероятност и влияние може да намали влиянието на отклоненията. Оценката на качеството на наличната информация за проектните рискове също спомага за по-прецизно определяне на важността на риска за проекта.

Качественият анализ на риска трябва да се изпълнява по време на целия жизнен цикъл на проекта за да бъде актуален спрямо промените в проектните рискове. От качествения анализ на риска може да се продължи към количествен анализ или директно към планиране на ответните действия.

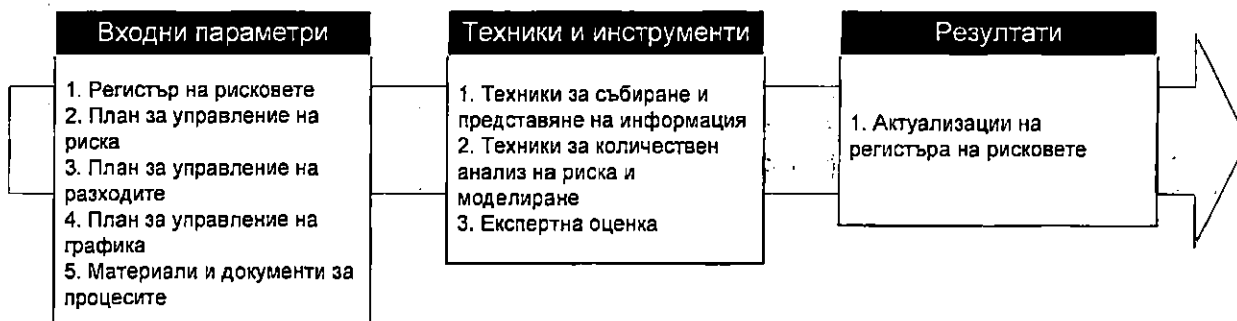
[Handwritten signature]



[Handwritten signature]
6
137

- **Входни параметри**
 - Регистър на рисковете
 - План за управление на риска
 - Спецификация на обхвата
 - Материали и документи за процесите
- **Техники и инструменти**
 - Оценка на вероятността и влиянието на рисковете
 - Матрица на вероятност и влияние
 - Оценка на качеството на информацията за рисковете
 - Категоризация на рисковете
 - Оценка на неотложността на рисковете
 - Експертна оценка
- **Резултати**
 - Актуализации на регистъра на рисковете

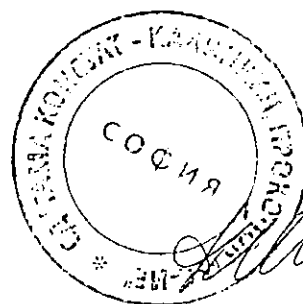
2.4. Количествен анализ на риска



Количествен анализ на риска е процесът на количествено и стойностно анализиране на ефекта върху проекта при случване на риска. Количественият анализ се изпълнява върху рисковете, които са приоритизирани при качествения анализ като значими за успешното изпълнение на проекта. Чрез този процес се дават числови изражения на ефекта от възникване на рисковете – за конкретен риск или общо. Количествената информация дава изходна информация при взимането на решения относно управлението на рисковете.

Изпълнението на количествен анализ обикновено следва процеса на качествения анализ. В някои случаи не се налага количествен анализ за планирането на ответни действия. Наличието на време и бюджет, както и конкретната нужда за качествен и количествен анализ определя подхода във всеки конкретен проект. Количественият анализ трябва да се повтори след планиране на ответните действия, а също така и по време на наблюдението и контрол на рисковете. Тенденциите в проекта показват нуждата от повече или по-малко действия за управление на риска.

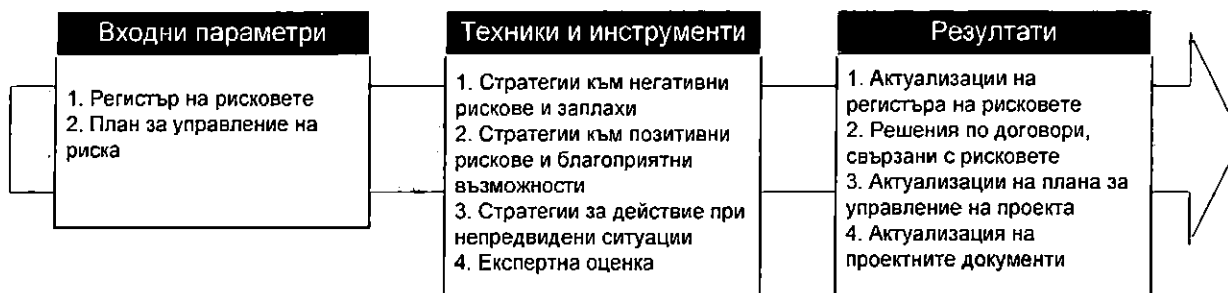
- **Входни параметри**
 - Регистър на рисковете
 - План за управление на риска
 - План за управление на разходите
 - План за управление на графика
 - Материали и документи за процесите



[Handwritten signature]

- **Техники и инструменти**
 - Техники за събиране и представяне на информация
 - Техники за количествен анализ на риска и моделиране
 - Експертна оценка
- **Резултати**
 - Актуализации на регистъра на рисковете

2.5. Планиране на ответни действия

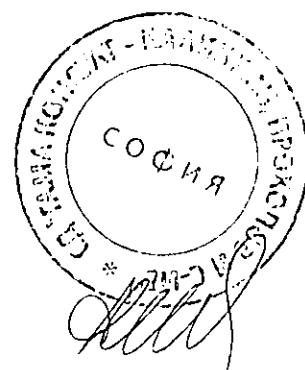


Планиране на ответни действия е процесът на разработване на опции и действия за повишаване на благоприятните възможности и намаляване на заплахите пред проектните цели. Той следва изпълнението на количествен и качествен анализ. Включва определяне на конкретен човек - отговорник за изпълняване на ответните действия при проява на риска. Планирането на ответни действия се отнася към рисковете в зависимост от техния приоритет, при това се включват съответни дейности в графика, бюджета на проекта и плана за управление.

Планираните ответни действия трябва да бъдат подходящи в зависимост от значимостта на рисковете, ценово ефективни, реалистични в контекста на проекта, съгласувани между заинтересованите страни и с назначен конкретен отговорник. Ответните действия трябва да бъдат навременни.

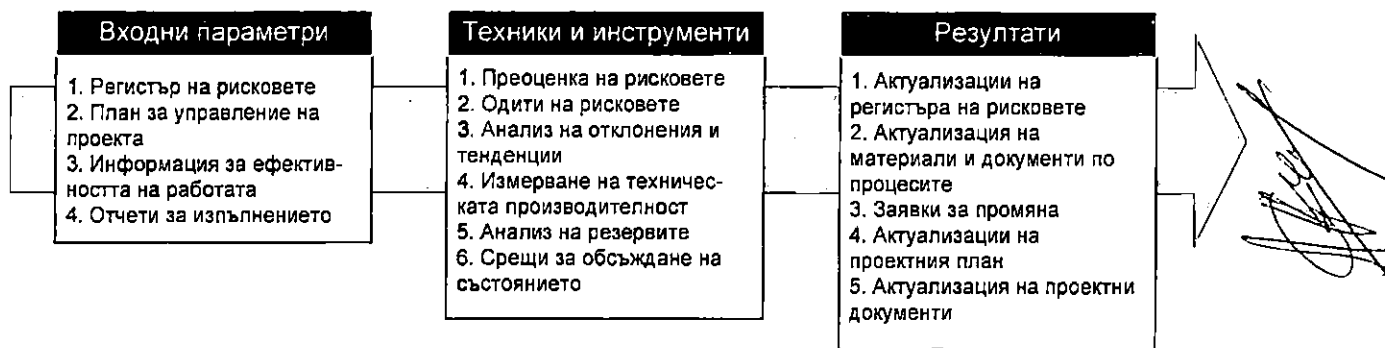
Използват се следните подходи за ответни действия при негативни рискове: избягване, прехвърляне, смекчаване и приемане. При позитивните рискове подходите са възползване и споделяне.

- **Входни параметри**
 - Регистър на рисковете
 - План за управление на риска
- **Техники и инструменти**
 - Стратегии към негативни рискове и заплахи
 - Стратегии към позитивни рискове и благоприятни възможности
 - Стратегии за действие при непредвидени ситуации
 - Експертна оценка
- **Резултати**
 - Актуализации на регистъра на рисковете
 - Решения по договори, свързани с рисковете
 - Актуализации на плана за управление на проекта
 - Актуализация на проектните документи



Handwritten signatures at the bottom of the page.

2.6. Наблюдение и контрол на рисковете



Наблюдението и контрола на рисковете е процесът на изпълнение на планираните ответни действия, проследяване на идентифицираните рискове, наблюдение на остатъчните рискове, идентифициране на нови рискове и оценка на процесите по управление на риска в проекта.

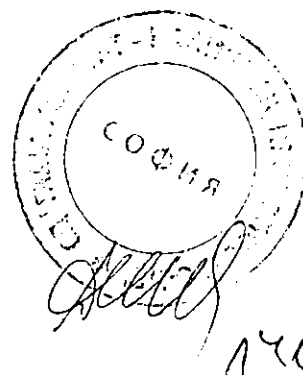
Планираните ответни действия, включени в проектния план се изпълняват по време на целия жизнен цикъл на проекта. Проектът трябва да бъде постоянно наблюдаван за нови, променени или отпаднали рискове.

При наблюдението и контрола на рисковете се прилагат техники като анализ на отклонения и тенденции, които изискват използване на информация за изпълнението. Друга цел на процеса е да се определи дали:

- проектните допускания са валидни;
- оценените рискове са се променили или отпаднали;
- се спазват политиките и процедурите за управление на риска;
- финансовите и времеви резерви трябва да се променят в зависимост от текущата оценка на рисковете.

Наблюдението и контролът на рисковете може да включва избор на алтернативни стратегии, изпълнение на резервен план, прилагане на коригиращи действия и промяна на проектния план. Отговорникът за риска периодично докладва на проектния мениджър ефективността на плана, неочаквани ефекти и необходимостта от корекции, необходими за адекватно управление на риска. Процесът също включва актуализацията на материали и документи по процесите, базата данни с „научени уроци“ и шаблони, които да бъдат използвани при бъдещи проекти.

- **Входни параметри**
 - Регистър на рисковете
 - План за управление на проекта
 - Информация за ефективността на работата
 - Отчети за изпълнението
- **Техники и инструменти**
 - Преоценка на рисковете
 - Одити на рисковете
 - Анализ на отклонения и тенденции
 - Измерване на техническата производителност
 - Анализ на резервите
 - Среци за обсъждане на състоянието
- **Резултати**



- експерти в предметната област;
- представител на спонсора на проекта или партньорска организация.

Всеки член на екипа участва в процеса на идентифициране на рисковете. Качествения анализ на риска се извършва от борда за управление на риска. Количественият анализ на риска и планирането на ответни действия се извършва от проектния ръководител, подпомогнат от борда за управление на риска. За изпълнението на количествен анализ и планиране на ответни действия проектният ръководител може да използва външни експерти в специфичната проектна област.

Наблюдението и контролът на рисковете се извършва от борда за управление на риска.

Минимум веднъж на няколко дена, а при необходимост и по-често Проектният ръководител докладва текущото състояние на проектните рискове. Регистърът на рисковете е достъпен за всички заинтересовани лица.

3.3. Таксономия и категоризиране на рисковете

Използват се следните категории при качествения анализ на рисковете:

- Категории за „Вероятност от настъпване“: малка, средна, голяма;
- Категория „Степен на въздействие“: малко, средно, голямо;
- В зависимост от вероятността и въздействието следва приоритетът на риска:

Вероятност Въздействие	Ниска	Несъществена	Умерена	Съществена	Висока
Много голяма (81 - 100%)	Среден приоритет	Среден приоритет	Висок приоритет	Висок приоритет	Висок приоритет
Голяма (61 - 80%)	Среден приоритет	Среден приоритет	Висок приоритет	Среден приоритет	Висок приоритет
Средна (41 - 60%)	Нисък приоритет	Среден приоритет	Среден приоритет	Среден приоритет	Висок приоритет
Ниска (21 - 40%)	Нисък приоритет	Среден приоритет	Среден приоритет	Среден приоритет	Висок приоритет
Минимална (1 - 20%)	Нисък приоритет	Нисък приоритет	Среден приоритет	Среден приоритет	Среден приоритет

При количественият анализ вероятността се оценява в проценти, а влиянието в пари – разход при възникване на риска.

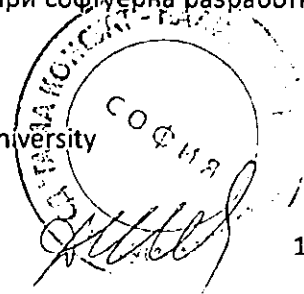
При управлението на риска се използват следните състояния за проследяване на рисковете: постоянен, бъдещ, текущ, затихнал, отминал.

3.4. Таксономия на рисковете при софтуерна разработка

Т. нар. таксономия на рисковете е стандартен начин за класифициране и организиране на рисковете. При изпълнението на проекта ще бъде прилагана таксономия на рисковете при софтуерна разработка на Института по софтуерно инженерство към Университет „Карнеги Мелън“:



Software Engineering Institute, Carnegie Mellon University

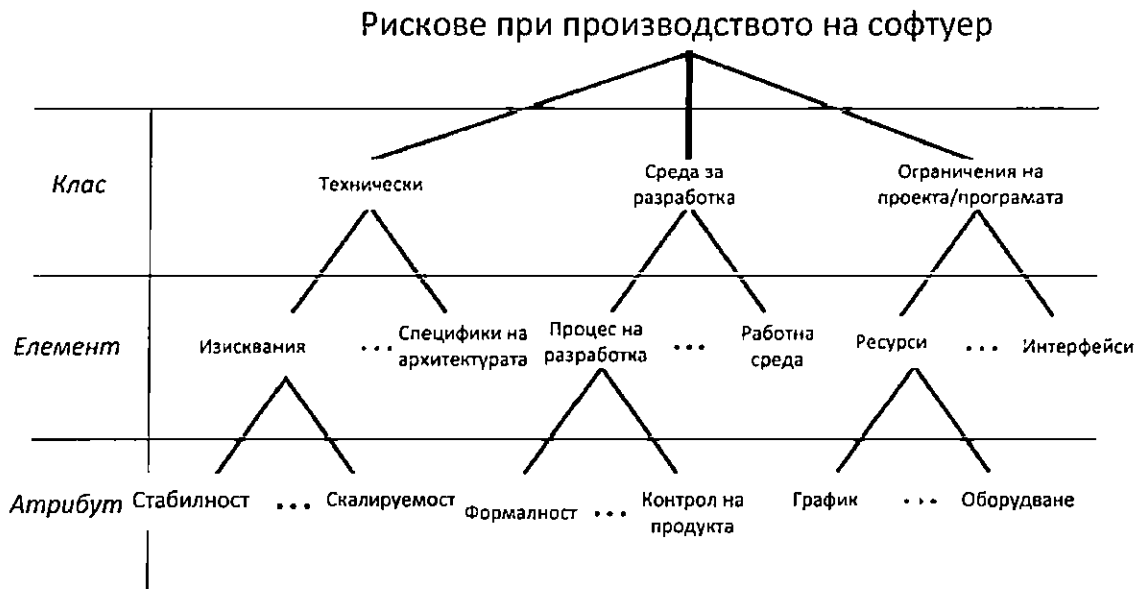


Предложената класификация следва процесите на софтуерното производство. Използването на стандартна класификация, извлечена от опита в различни проекти спомага при идентификацията на рисковете да се минимизират възможностите за пропуск.

Методът за идентификация на рисковете на Института по софтуерно инженерство се базира на следните приемания:

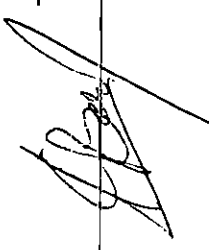

- Рисковете при софтуерното производство обикновено са известни на проектните екипи, но често са лошо комуникирани.
- За последователно управление на рисковете е нужен структуриран и повторяем метод за идентифициране.
- Ефективното идентифициране на рисковете трябва да покрива всички ключови области от разработката и поддръжката на проекта.
- Процесът за идентифициране на рисковете трябва да създаде и поддържа среда за събиране на информация, която да избягва използването на субективна, спорна и неубедителна информация.
- Не трябва да се прави оценка за успеха на проекта на базата на броя и естеството на откритите рискове.

Таксономията на рисковете е организирана на три нива: класове, елементи и атрибути:



Таксономия на рисковете при софтуерна разработка:

<p>A. Технически</p> <p>1. Изисквания</p> <p>a. Стабилност</p> <p>b. Пълнота</p> <p>c. Яснота</p> <p>d. Валидност</p> <p>e. Осъществимост</p> <p>f. Прецедент</p> <p>g. Скалируемост</p> <p>2. Дизайн</p> <p>a. Функционалност</p> <p>b. Леснота</p> <p>c. Интерфейси</p> <p>d. Производителност</p>	<p>B. Среда за разработка</p> <p>1. Процес за разработка</p> <p>a. Формалност</p> <p>b. Приложимост</p> <p>c. Контрол на процеса</p> <p>d. Познаване на процеса</p> <p>e. Контрол на продукта</p> <p>2. Инструменти за разработка</p> <p>a. Капацитет</p> <p>b. Приложимост</p> <p>c. Използваемост</p> <p>d. Познаване</p> <p>e. Надеждност</p>	<p>C. Ограничения на проекта и програмата</p> <p>1. Ресурси</p> <p>a. График</p> <p>b. Персонал</p> <p>c. Бюджет</p> <p>d. Оборудване</p> <p>2. Договор</p> <p>a. Тип на договора</p> <p>b. Ограничения</p> <p>c. Зависимости</p> <p>3. Интерфейси</p> <p>a. Клиент</p> <p>b. Партньори</p>
---	---	--

<ul style="list-style-type: none"> e. Тестваемост f. Хардуерни ограничения g. Невъзможност за разработка 	<ul style="list-style-type: none"> f. Поддръжка g. Възможност за доставяне 	<ul style="list-style-type: none"> c. Подизпълнители d. Основен контрактор e. Управление на компанията f. Доставчици g. Политика
<ul style="list-style-type: none"> 3. Код и тестване на модулите (unit test7) <ul style="list-style-type: none"> a. Изпълнимост b. Тестване c. Кодиране/Реализация 4. Интеграция и тестване <ul style="list-style-type: none"> a. Среда b. Продукт c. Система 5. Специфики на архитектурата <ul style="list-style-type: none"> a. Възможност за поддръжка b. Надеждност c. Безопасност d. Сигурност e. Човешки фактори f. Спецификации 	<ul style="list-style-type: none"> 3. Процес за управление <ul style="list-style-type: none"> a. Планиране b. Организация на проекта c. Управленски опит d. Интерфейси на програмата 4. Методи за управление <ul style="list-style-type: none"> a. Мониторинг b. Управление на персонала c. Осигуряване на качеството d. Управление на конфигурацията 5. Работна среда <ul style="list-style-type: none"> a. Отношение към качеството b. Кооперативност c. Комуникация d. Морал 	 

Посочената класификация на рисковете ще бъде разширена с релевантни категории рискове предвид широкия обхват на проекта.

3.5. Регистър на рисковете

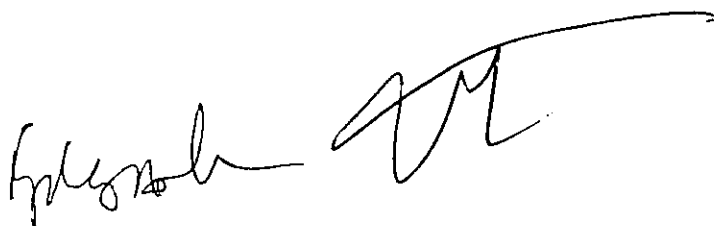
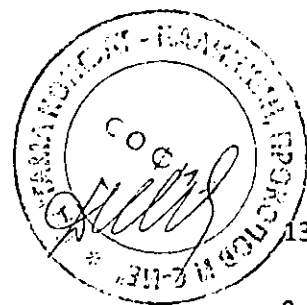
Рисковете за проекта ще се следят през целия период на изпълнение. За тази цел ще се поддържа списък на рисковете. Той се подготвя в началото на проекта и се поддържа в актуално състояние по време на целия проект. Преглежда се на всяка среща.

При оценката на всеки един от рисковете е оценена следната задължителна информация по рискове:

1. Обхват и степен на въздействие (последствия) на риска върху изпълнението на обществената поръчка;
2. Мерки за недопускане/предотвратяване на риска, в приложимите случаи, съответно обосновка за невъзможността да се предприемат подобни мерки в конкретния случай;
3. Мерки за преодоляване на последиците при настъпване на риска.

С помощта на този регистър ще може да се проследяват всички потенциални и възникнали рискове по време на изпълнение на проекта, което от своя страна ще позволи навременно извършването на превантивни действия с цел минимизиране на риска или реагиране при възникването на някой от тях.

Списъкът на рисковете представлява структурирано описание на известните и реално стоящи рискове пред проекта, подредени в низходящ ред по значение. Към всеки риск се привързват мерки за ограничаване на последствията или действия при настъпване на риска. Списъкът на рисковете трябва да отразява критичните и сериозни рискове. Изготвя се в следния примерен вид:

744

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка

Рискът се идентифицира с пореден номер, който се записва в първата колона. Втората колона съдържа описание на риска, а третата – резюме на въздействие върху изпълнението на проекта. В колона "Отговорник" се посочва(т) лицето или организацията (лицата или организациите), което отговаря (които отговарят) за преодоляването на съответния риск. Следваща колона е „Вид“ и съответно се определя дали е технически, външни, организационни, управленски. Степен на въздействие – отклонението от планираните графици, усилия и разходи, ако рискът действително се реализира.

Степен на въздействие	Вероятност от настъпване
Висока	Много висока (81-100%)

Степен на въздействие

Изберете стойността, която най-добре описва какво ще е отклонението от планираните графици, усилия и разходи, ако рискът действително се реализира.

Вероятност от настъпване - Вероятността рискът действително да се реализира (обикновено се изразява като процент).

Вероятност от настъпване	Индикатор
Много висока (81-100%)	Вероятност от настъпване Изберете стойността, която най-добре описва вероятността рискът действително да се реализира.

Индикатор - За всеки риск от списъка се идентифицира подлежащо на измерване състояние, настъпването на което означава, че рискът действително се е реализирал. Тези състояния представляват индикатори (лакмуси) за появата на риска (с чиято помощ разбираме, че рискът е вече реалност).

Стратегия за ограничаване - Разработването на планове за ограничаване на риска означава да се намалят последствията от настъпването на риска. За някои рискове се изисква да се опишат и действията, чието изпълнение зависи от настъпването на риска. Тук се разглеждат всички мерки за недопускане/предотвратяване на риска, а в някои случаи, съответна обосновка за невъзможността да се предприемат подобни мерки в конкретния случай.

[Handwritten signatures]

[Circular stamp and handwritten signature]

3.6. Инструменти за управление на риска

Предложената методика за управление на риска ще използва голям набор от инструменти и практики за постигане на целите на проекта. Всички инструменти са детайлно описани по-горе в частта методологии за управление на риска. Предлагаме съкратен списък на инструментите за управление на риска:

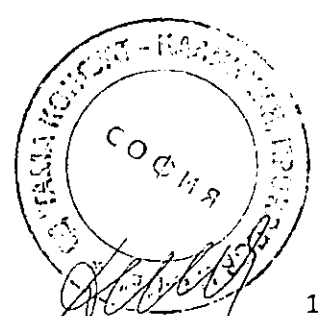
- Срещи за планиране и анализ;
- Преглед на документация;
- Анализ на контролни списъци;
- Анализ на допусканията;
- Анализ на силни и слаби страни;
- Експертна оценка;
- Оценка на вероятността и влиянието на рисковете;
- Матрица на вероятност и влияние;
- Оценка на качеството на информацията за рисковете;
- Категоризация на рисковете;
- Оценка на неотложността на рисковете;
- Стратегии към негативни рискове и заплахи;
- Стратегии към позитивни рискове и благоприятни възможности;
- Стратегии за действие при непредвидени ситуации;
- Преоценка на рисковете;
- Одити на рисковете;
- Анализ на отклонения и тенденции;
- Анализ на резервите;
- Срещи за обсъждане на състоянието.

За подпомагане поддържането на регистъра на рисковете се използва специализиран портал за управление на проекта, разработен от нас на базата на Microsoft Office SharePoint:

Списък на рисковете

Регистър на риска

№	Описание на риска	Вероятност	Влияние	Категория	Степен	Състояние	Действие	Срок	Забелешки
1	Риск от промяна в изискванията за функционалност	Висока	Високо	Критичен	Висок	Активен	Изяснение на изискванията	15.12.2019	20
2	Риск от изоставяне на работата на разработчиците	Средна	Средно	Среден	Среден	Активен	Мониторинг на работата	15.12.2019	20
3	Риск от промяна в приоритетите на разработчиците	Средна	Средно	Среден	Среден	Активен	Мониторинг на приоритетите	15.12.2019	20
4	Риск от изоставяне на работата на тестерите	Средна	Средно	Среден	Среден	Активен	Мониторинг на работата	15.12.2019	20
5	Риск от промяна в изискванията за качество	Средна	Средно	Среден	Среден	Активен	Изяснение на изискванията	15.12.2019	20
6	Риск от промяна в изискванията за време за пускане	Средна	Средно	Среден	Среден	Активен	Мониторинг на времето	15.12.2019	20
7	Риск от промяна в изискванията за цена	Средна	Средно	Среден	Среден	Активен	Мониторинг на цената	15.12.2019	20
8	Риск от промяна в изискванията за документация	Средна	Средно	Среден	Среден	Активен	Мониторинг на документацията	15.12.2019	20
9	Риск от промяна в изискванията за обучение	Средна	Средно	Среден	Среден	Активен	Мониторинг на обучението	15.12.2019	20
10	Риск от промяна в изискванията за поддръжка	Средна	Средно	Среден	Среден	Активен	Мониторинг на поддръжката	15.12.2019	20
11	Риск от промяна в изискванията за тестове	Средна	Средно	Среден	Среден	Активен	Мониторинг на тестовете	15.12.2019	20
12	Риск от промяна в изискванията за отчети	Средна	Средно	Среден	Среден	Активен	Мониторинг на отчетите	15.12.2019	20
13	Риск от промяна в изискванията за срещи	Средна	Средно	Среден	Среден	Активен	Мониторинг на срещите	15.12.2019	20
14	Риск от промяна в изискванията за комуникация	Средна	Средно	Среден	Среден	Активен	Мониторинг на комуникацията	15.12.2019	20
15	Риск от промяна в изискванията за документация	Средна	Средно	Среден	Среден	Активен	Мониторинг на документацията	15.12.2019	20
16	Риск от промяна в изискванията за обучение	Средна	Средно	Среден	Среден	Активен	Мониторинг на обучението	15.12.2019	20
17	Риск от промяна в изискванията за поддръжка	Средна	Средно	Среден	Среден	Активен	Мониторинг на поддръжката	15.12.2019	20
18	Риск от промяна в изискванията за тестове	Средна	Средно	Среден	Среден	Активен	Мониторинг на тестовете	15.12.2019	20



4. Конкретни рискове и предпоставки за успешно изпълнение на поръчката

4.1. Анализ на конкретни рискове, идентифицирани от Възложителя

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
1	Недобра комуникация между екипите на Възложителя и Изпълнителя по време на аналитичните етапи на проекта	<p>В резултат на което може да се получи некачествено и забавено изпълнение на етапа за анализ и проектиране.</p> <p>Забавяне изпълнението на задачите по проекта.</p> <p>Проблеми с дефинирането на правилните изисквания при реализацията.</p> <p>Бавно вземане на решения.</p> <p>Некачествените и непълни изисквания могат до доведат до влагане на време и ресурс за разработка в погрешна посока.</p> <p>Сферата на влияние е в рамките на всички етапи на проекта, като</p>	Изпълнителя Възложителя	10	Висока	Ниска (21 - 40%)	<p>Забавяне на отговори, срещи и дейности критични за изпълнението на проекта.</p> <p>Непълни, неясни или често променящи се изисквания.</p> <p>Преразглеждане на взети решения.</p>	<p>Стриктно прилагане на предложената методология за изпълнение, която включва план за комуникация. Допълнително всяка седмица ще се провежда среща (разговор) между ключови представители на екипа. Регулярно предаване на доклади за напредъка по проекта. Определяне на хора за ключови роли в проектните екипи от двете страни. Прилагане на процедура за преглед (ревию) на резултатите от аналитичните етапи на проекта с участие на екипа на изпълнителя. Това ще спомогне еднозначното и пълно разбиране на изискванията и системния проект.</p> <p>По този начин възможността за пропуски в изпълнението се свежда до минимум и всяко</p>

147

Handwritten signature

Handwritten signature and stamp

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		може да се разпростре е върху бъдещи проекти за надграждане на системата и интегриране на други системи с нея.						отклонение ще бъде идентифицирано преди да е довело до реален проблем.
2	Ненавременно изпълнение на всяко от задълженията от страна на Изпълнителя	<p>Забавяне на изпълнението на поръчката;</p> <p>Необходимост за влагане на допълнителни ресурси за паралелна работа;</p> <p>Влошаване на качеството;</p> <p>Санкции.</p> <p>В зависимост от вида на забавянето, то може да повлияе конкретни задачи, етапи от изпълнението на поръчката или цялата поръчка. Значително забавяне на поръчката може да</p>	Изпълнителя	4	Съществена	Минимална (1 - 20%)	<p>Липса на съгласуван детайлен план-график;</p> <p>Липса на ангажимент от страна на Възложителя за зависимостите в графика;</p> <p>Забавяне на междинни срокове.</p>	<p>За минимизиране:</p> <p>Разработване и съгласуване на детайлен план-график за изпълнение;</p> <p>Ясно комуникиране на зависимостите с Възложителя;</p> <p>Определяне и проследяване на критичен път на задачите в проекта;</p> <p>Регулярен мониторинг и контрол на изпълнението;</p> <p>За минимизиране на последствията:</p> <p>Бързо предприемане на коригиращи действия при необходимост;</p> <p>Паралелна работа по различни задачи при необходимост да се компенсира забавяне;</p> <p>Влагане на допълнителни</p>

Handwritten signature and scribbles on the left margin.

Handwritten number '2' at the bottom left.

Handwritten signature at the bottom center.

Handwritten signature at the bottom right.

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка	
		доведе до забавяне на други зависещи от нея поръчки и проекти.						ресурси в някои от задачите при необходимост да се компенсира забавяне. Предложените мерки съответстват на най-добрите и утвърдени световни практики за управление на проекти, като тези на PMI.	
3	Неправилно и неефективно разпределяне ресурсите отговорностите изпълнението договора	и на и при на	Забавяне на изпълнението. Изпълнение с недостатъчно ниво на качество. Оскъпяване на изпълнението. Сферата на влияние може да бъде ограничена в рамките на конкретни задачи т.к. се предполага, че ще бъдат приложени мерки за предотвратяване и евентуално за минимизиране на последствията.	Изпълнителя	4	Съществена	Минимална (1 - 20%)	Неясно дефинирани отговорности. Липса на план за наличност и капацитет на ресурсите.	Ясно дефиниране на организационната структура, ролите и отговорностите – както за екипа на Изпълнителя, така и за Възложителя. Детайлно планиране на ресурсите по задачи в началото на проекта. Изпълнителят да даде препоръка за необходимите от страна на възложителя ресурси. Основният ресурс за изпълнение на поръчката са експертните екипи от страна на двете страни. Задачите по проекта трябва да бъдат обезпечени с необходимия ресурс съобразно графика за

1906


[Handwritten signatures and initials]

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
								изпълнение. Освен като наличност за конкретни задачи, експертите и екипите трябва да притежават необходимите знания, умения и подготовка за изпълнение на конкретните задачи.
4	Забавяне при изпълнение на проектните дейности, опасност от неспазване на срока за изпълнение на настоящата поръчка	Забавяне на следващи процедури, зависещи от настоящата; Необходимост за влагане на допълнителни ресурси за паралелна работа; Влошаване на качеството; Санкции. В зависимост от вида на забавянето, то може да повлияе конкретни задачи, етапи от изпълнението на поръчката или цялата поръчка. Значително забавяне на	Изпълнителя	4	Съществена	Минимална (1 - 20%)	Липса на съгласуван детайлен план-график; Липса на ангажимент от страна на Възложителя за зависимостите в графика; Забавяне на междинни срокове.	Мерки за предотвратяване: Разработване и съгласуване на детайлен план-график за изпълнение; Ясно комуникиране на зависимостите с Възложителя; Определяне и проследяване на критичен път на задачите в проекта; Регулярен мониторинг и контрол на изпълнението; Мерки за минимизиране на последствията: Бързо предприемане на коригиращи действия при необходимост; Паралелна работа по различни задачи при необходимост да се компенсират забавяне; Влагане на допълнителни

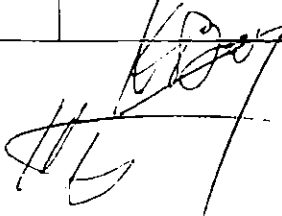
Handwritten signature/initials in the left margin.

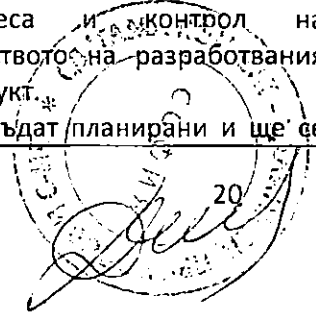
Handwritten signature and a circular stamp in the bottom right corner.

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		поръчката може да доведе до забавяне на други зависещи от нея поръчки и проекти.						ресурси в някои от задачите при необходимост да се компенсира забавяне. Предложените мерки съответстват на най-добрите и утвърдени световни практики за управление на проекти, като тези на PMI.
5	Грешки при разработване на функционалностите на адаптера	Невъзможност на системата да функционира правилно; Невъзможност на системата да отговаря на изискванията за бързодействие; Затруднена интеграция с други системи. Благодарение на заложените процеси за осигуряване и контрол на качеството евентуални грешки би трябвало да бъдат установявани основно в етапа на разработка. Предвижда се установяване и	Изпълнител	4	Съществена	Минимална (1 - 20%)	Регистриране на инциденти в системата	Изпълнителят ще следва стриктно процесите и добрите практики за разработка на софтуер. Прилагането на добри процеси и практики води до получаване на качествен продукт в предвидим срок и бюджет. При проектиране на системата се използват SOLID принципите за софтуерна архитектура – качеството ще бъде заложено в дизайна на системата. Процесите за софтуерна разработка включват мерки за осигуряване на качеството на процеса и контрол на качеството на разработвания продукт. Ще бъдат планирани и ще се

Ваня


151



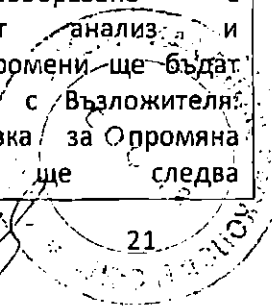


№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		коригиране на грешки и в етапа на тестване на системата. Сферата на влияние следва да се ограничи в рамките на поръчката.						<p>провеждат всички необходими тестове. За целта ще бъдат разработени детайли тестови сценарии.</p> <p>Предвидени са време и ресурс за отстраняване на грешки. Предвижда се прилагането на процедура и система за регистриране и отстраняване на грешки. Грешките ще бъдат отстраняване в предвидените срокове, съобразно техния приоритет.</p>
6	Недостатъчна яснота по правната рамка и/или променяща се правна рамка по време на изпълнение на проекта	<p>Може да доведе до концептуални непълноти и разминавания между цели и резултати.</p> <p>Промяна на изискванията, което ще доведе до увеличаване на сроковете или бюджета за реализиране на проекта. Промяна на вече изградени функционалности.</p>	Възложителя	5	Висока	Минимална (1 - 20%)	<p>Възложителя променя изисквания по техническата спецификация след започване на проекта. Промени спрямо утвърдените в Етап 1 спецификации. Разширяване на обхвата. Отмяна на взети решения.</p>	<p>По време на фазата на анализ ще се анализират действащите норми и проектите за техните промени по време на реализацията. Ще се проведат проучвания за намеренията за промени свързани с проекта. Техническото изпълнение ще бъде съобразено с направеният анализ и всякакви промени ще бъдат съгласувани с Възложителя. Всяка заявка за промяна стриктно ще следва</p>

Handwritten signature

Handwritten signature

Handwritten signature



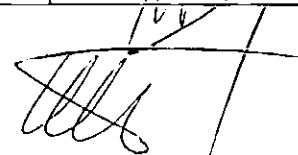
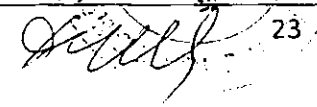
152

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		Сферата на влияние е в рамките на поръчката.						процедурата за управление на промените. Системата ще бъде гъвкава и в нея лесно биха могли да се реализират някои видове промени.
7	Липса на задълбоченост при изследването и описанието на бизнес процесите и данните	Недостатъчно добри изисквания към системата; Пропуснати изисквания; Неадекватна архитектура и оразмеряване. В случай на неадекватна архитектура, сферата на влияние може да се разпростре и върху други системи, чрез поставяне на неподходящи изисквания за интеграция.	Изпълнителя Възложителя	10	Висока	Ниска (21 - 40%)	Непълна и некачествена спецификация на изисквания; Изисквания, които не са SMART (Specific, Measurable, Attainable, Realisable, Traceable); Участие в изпълнението на ключови експерти, които не са добре запознати с предметната област.	Участието в проекта на правилните експерти от двете страни е решаващо за постигане на задълбоченост при изследването и описанието на бизнес процесите и данните. Особено важно е експертите на Възложителя да имат предвидено време за задачите по проекта и тези задачи да са с висок приоритет. Прилагане на процедура за преглед (ревию) на резултатите от аналитичните етапи на проекта с участие на екипа на изпълнителя. Това ще спомогне еднозначното и пълно разбиране на изискванията. За верифициране на изискванията, бизнес

153

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
								процедите и данните може да си използва съпоставяне със сходни проекти в други държави. Освен, че ще спомогне за гарантиране на качество, използването на добрия опит също може да улесни и ускори изпълнението.
8	Неинформиране на Възложителя за всички потенциални проблеми, които биха могли да възникнат в хода на изпълнение на дейностите	Невъзможност за планиране и предприемане на адекватни и навременни коригиращи действия. Сферата на влияние е в рамките на конкретната поръчка.	Изпълнителя. Възложителя	4	Съществена	Минимална (1 - 20%)	Липса на план за комуникация или план за комуникация, в който не е указано как протича комуникацията за проблемите (issues); Липса на план за управление на проблемите; Неспазване на плановете за комуникация и управление на проблемите; Установяване на наличен проблем, който не е	Разработване и съгласуване на план за управление на проблемите; Поддържане и регулярно комуникиране на регистър на проблемите; Разработване и съгласуване на план за комуникация, който включва комуникация на проблемите; Включване на темата за проблемите в регулярните срещи и доклади по проекта; Публикуване на регистъра на проблемите в общодостъпна среда. Предложените мерки са съобразени със световно утвърдени добри практики за

151

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
							комуникиран.	управление на проекти – конкретно PMBOK (Project Management Body of Knowledge) на PMI.
9	Риск за администриране на адаптера след изтичане на периода на гаранционна поддръжка	Невъзможно или затруднено администриране на системата след приключване на поддръжката по Дейност 2. Сферата на влияние се отнася до конкретната система, резултат от поръчката в следващ период.	Възложителя	4	Съществена	Минимална (1 - 20%)	Липса на обучени администратори от страна на Възложителя; Липса на ръководство за администриране на системата.	За предотвратяване на риска е необходимо да бъдат определени конкретни експерти от страна на Възложителя, на които да бъде вменена отговорност за бъдещото администриране на системата. Те трябва да бъдат обучени да извършват дейностите по администриране. За осигуряване на второ ниво на поддръжка следва да бъде подписан договор за извънгаранционна поддръжка. За минимизиране на последствията при възникване на риска, може да се сключи договор за пълно администриране на системата с външен изпълнител.

Handwritten signature and scribbles on the left side of the page.

155

Handwritten signatures and a circular stamp on the bottom right side of the page.

4.2. Анализ на допълнителни, относими към поръчката рискове

№	Описание	Последствия и сфера на влияние	Отговорник	Общ експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
1	Неочаквани промени в изискванията поради външни причини – закони, стандарти, оперативни решения на ръководни органи на страната, международни задължения и др.	Промяна на изискванията, което ще доведе до увеличаване на сроковете или бюджета за реализиране на проекта. Промяна на вече изградени функционалности. Сферата на влияние е в рамките на поръчката.	Възложителя	5	Висока	Минимална (1 - 20%)	Възложителя променя изисквания по техническата спецификация след започване на проекта. Промени спрямо утвърдените в Етап 1 спецификации. Разширяване на обхвата. Отмяна на взети решения.	По време на етапа на анализ ще се направи анализ на действащите норми и проектите за техните промени по време на реализацията. Ще се проведат проучвания за намеренията за промени свързани с проекта. Техническото изпълнение ще бъде съобразено с направеният анализ и всякакви промени ще бъдат съгласувани с Възложителя. Всяка заявка за промяна стриктно ще следва процедурата за управление на промените. Системата е гъвкава и в нея лесно биха могли да се реализират някои видове промени.
2	Слаба ангажираност на представители на крайните потребители на системата.	Получаване на непълни и неясни изисквания. Забавяне при предоставянето на информация. Трудно внедряване на системата в реалната работа на организацията. Сферата на влияние е в	Възложителя	6	Умерена	Ниска (21 - 40%)	Потребителите подават неясна, непълна или противоречива информация. Забавят се отговори и решения. Крайните потребители не	Определяне на конкретни ключови потребители за съответните области още в началото на проекта. По време на обученията ще се проведе максимално запознаване на потребителите с функционалностите на системата и ползата за тях от нея. В обученията трябва да бъдат

Handwritten signature and number 156.

Handwritten signature and number 25.

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		рамките на поръчката, но може да се разпростре и при интегрирането с други системи, в случай, че останат незабелязани дефекти.					използват новоизградената система.	включени точните потребители/служители, които ще продължат да имат отношение към експлоатацията на системата.
3	Невъзможност за провеждане или сериозни затруднения за провеждане на ефективни тестове.	Липса на сигурност в безпроблемното функциониране на системата. Сферата на влияние се ограничава в Дейност 2 по настоящата поръчка.	Изпълнителя Възложителя	3	Умерена	Минимална (1 - 20%)	Документирана невъзможност за провеждане на тестове. Голям брой грешки във внедрената система по-висок от допустимия праг.	По време на цялата реализация Изпълнителят ще провежда вътрешни хардуерни и софтуерни тестове на всички елементи преди тяхното предаване за тестване от Възложителя. Всички съвместни тестове ще бъдат предварително проверени и системите ще са претърпели предварителни тестове преди заявяване на готовност. С прилагането на най-добрите практики за системна интеграция възможността за невъзможност за провеждане на ефективни тестове е сведена до минимум. Изпълнителят ще осигури достъп на Възложителя до неговата тестова среда. Осигуряване на необходимото време за провеждане на тестове.

157

[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

[Circular stamp]

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
4	Затруднения при осигуряване в периода на обучение на подходящите за обучение лица.	Част от служителите не са преминали обучение и им липсват специфични умения за работа със системата. Да бъдат обучени служители, които няма да имат пряко отношение към експлоатацията на системата за сметка на такива, които имат пряко отношение. Сферата на влияние се свежда до настоящата поръчка. Рискът би се отразил на Дейност в и последващата експлоатация на системата.	Възложителя	1	Ниска	Минимална (1 - 20%)	По-малък брой присъстващи на обученията от предвиденото. Включените в обучение служители нямат определена роля по отношение на експлоатацията на системата.	Преди определяне на обучаемите, Възложителят трябва да определи служителите, които ще имат отношение към експлоатацията и поддръжката на системата и техните роли и отговорности. Изпълнителят ще предостави препоръки към обучаемите за да може Възложителят да подбере най-добрия възможен екип, които да бъде обучен. Графикът за обучения ще бъде направен няколко седмици преди провеждането на обученията. По време на изготвяне на графика всеки служител ще потвърди възможността си за участие. Обучението може да бъде заснето и публикувано в портала.
5	Поява на неизяснени изисквания в късен етап от проекта и промяна на вече утвърдени изисквания.	Влагане на допълнителни ресурси. Забавяне. Предизвикване на промени в други зависими системи. Сферата на влияние се разпростира върху етапите на разработка и тестване от	Изпълнителя Възложителя	12	Съществена	Средна (41 - 60%)	Промяна на утвърдени изисквания. Възникване на нови изисквания след фазата на дизайн.	Добро документиране на бизнес анализа и дизайна на системите. Прототипиране. Доброто документиране е предпоставка за еднозначно и пълно разбиране на изискванията. Комуникация при всяка една итерация за разработка. Добрата комуникация предполага, че неяснотите ще бъдат изяснявани

158

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		настоящата поръчка						на _____ момента. Допускане на промени само за най-важните изисквания – за да не се налагат значителни преработки и забавяния в следствие на тях.
6	Да се изгради система, която не отговаря на бизнес процесите и изискванията на възложителя.	Невъзможност за реална експлоатация на системата и постигане на целите на проекта. Ще окаже влияние върху целия проект.	Изпълнителя Възложителя	5	Висока	Минимална (1 - 20%)	Нееднозначно разбиране на изисквания. Поява на липсващи изисквания.	Задълбочаване на разработката едва след съгласуване на изискванията с Възложителя. Това ще предотврати влагането на усилия в разработка по неправилни изисквания. Прилагане на итеративен подход за разработка с участие на представители на Възложителя в церемониите по приключване на спринтовете (демо). Регулярното участие на Възложителя предполага навременна обратна връзка.
7	Възникване на проблеми с натоварването на системата - недостатъчен хардуер и съответно лицензи за да се покрие натоварването на системата.	Прекъсване в работата или неприемливо бавна работа на системата. Влияние върху изпълнението на Дейност 2 и бъдещата експлоатация на системата.	Възложителя	4	Съществена	Минимална (1 - 20%)	Голямо натоварване, водещо до прекъсване в работата или неприемливо бавна работа на системите.	Разработване и готовност за прилагане на план за скалиране на хардуерната и софтуерна инфраструктура. Базиране на оценките на инфраструктурния модел върху прогнози за натоварване.

Handwritten signature and initials on the left side of the page.

Handwritten signature and initials at the bottom right of the page.

Official circular stamp of a company with the word "СОФИЯ" (SOFIA) visible in the center.

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
8	Информацията в системата да не бъде достатъчно защитена.	Изтичане на информация. Спиране на работа на системата. Влиянието се разпростира върху системата, предмет на поръчката.	Изпълнителя Възложителя	4	Съществена	Минимална (1 - 20%)	Липса на стратегия и процедури за защита на информацията. Неприлагане на процедурите за защита на информацията.	Разработване на стратегия за защита на информацията. Прилагане на стратегията за защита при инсталацията на експлоатационната среда. Редовно изпълнение на процедурите за защита.
9	Забавяне при вземането на решения или промяна на взети решения.	Закъснения. Необходимост от преправяне на вече извършена работа. Сферата на влияние е в рамките на поръчката.	Възложителя	3	Умерена	Минимална (1 - 20%)	Забавяне на отговори. Забавяне на решения. Промяна на взети решения.	Изготвяне и спазване на план за комуникация. Изготвяне и спазване на процедура за управление на промените. (И двете са част от плана за управление на проекта.)
10	Невъзможност за администриране и конфигуриране на техническа и софтуерна инфраструктура.	Невъзможност за извършване на инсталации и промяна на конфигурации. Необходимост от извършване на преинсталации. Загуба на информация. Влияние в рамките на поръчката.	Възложителя	3	Умерена	Минимална (1 - 20%)	Невъзможен администраторски достъп до инфраструктурни системи	Разработване и прилагане на процедури за администриране и конфигуриране. Осигуряване на отдалечен достъп до ИТ инфраструктурата.
11	Поставяне на неизпълними, абстрактни или неясно дефинирани	Невъзможност за изпълнение или погрешно изпълнение. Забавяне на	Възложителя	4	Съществена	Минимална (1 - 20%)	Нееднозначно разбиране на изисквания.	Въвличане на служители от съответната бизнес дирекция при детайлизиране на системата. Разработване на ясни,

160

29

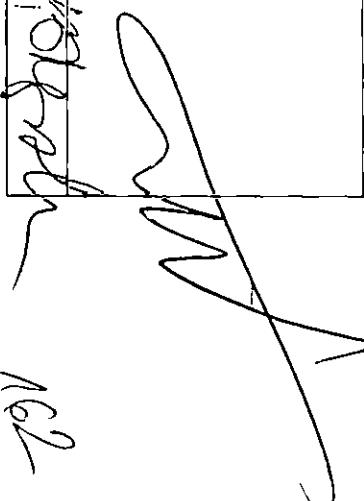
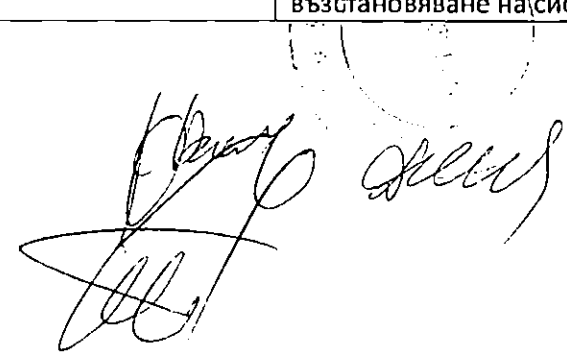
№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
	изисквания.	детайлизирането и изграждане на лош дизайн. Влияние в рамките на поръчката. Допускането на лош дизайн може да се отрази върху бъдещите интеграции със системата и възможността за нейното надграждане.						еднозначни и достатъчно подробни спецификации. Реализиране на системата по гъвкав начин, чрез който лесно да могат да се реализират евентуални промени. Стартиране на разработка след наличие на достатъчно детайлни изисквания.
12	Неустановена практика по процедурата за промени	Забавяне на искани промени поради бавна или неработеща процедура, а и от там на сроковете по проекта. Влияние в рамките на поръчката.	Изпълнителя Възложителя	12	Умерена	Голяма (61 - 80%)	Липса на двустранно утвърдена процедура за управление на промени.	Въвеждане на ясна процедура за управление на промените. Приоритизиране на исканията за промяна и реализиране само на най-приоритетните при спазване на обхвата на проекта.
13	Екипът на Възложителя е недостатъчен, не е на разположение за проекта или задачите по проекта са с нисък приоритет спрямо останалата работа.	Забавяне вземането на решения и като следствие забавяне на работата по проекта. Вземане на грешни решения и като резултат необходимост от повторно извършване на части от работата по проекта.	Възложителя	4	Съществена	Минимална (1 - 20%)	Забавяне на отговори. Забавяне на вземането на решения. Промяна на вече взети решения.	Регламентиране по вътрешен ред на приоритетите при изпълнение на работата на екипа. Предоставяне на допълнителни членове към екипа с необходимите компетенции. Осигуряване на допълнителна ангажираност на експертния персонал на Възложителя. Привличане на външни

169

30

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
		Влияние в рамките на поръчката.						консултанти.
14	Трудности при осигуряване на условия за извършване на приемни тестове	Забавяне на приемните тестове. Влияние в рамките на поръчката, конкретно – Дейност 2.	Възложителя	3	Умерена	Минимална (1 - 20%)	Липса на потвърждение за готовност за приемни тестове.	Предварително изготвени изисквания за провеждане на приемни тестове и осигуряване на условия. Осигуряване на резерв от време за извършването на приемните тестове.
15	Системата да не бъде подготвена за посрещане на заплахи	Срив на системата в случай на проявяване на заплаха. Влияние върху CISE Adapter и обмяна на информация с останалите системи.	Изпълнителя Възложителя	3	Умерена	Минимална (1 - 20%)	Липса на стратегия за резервираност. Липса на инфраструктура за резервираност.	Разработване на стратегия за резервираност, която включва: анализ на рисковете за изгражданото решение, възможните източници на заплаха, възможните слаби места в структурата на системата и т.н. Осигуряване на необходимата инфраструктура за резервираност. Наличие на процедури за възстановяване на системата.

10/12/2012

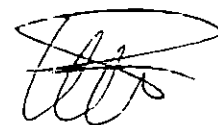
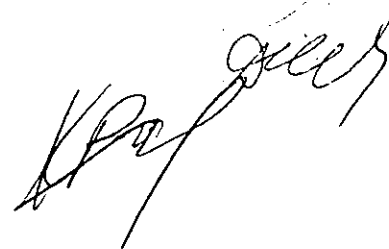
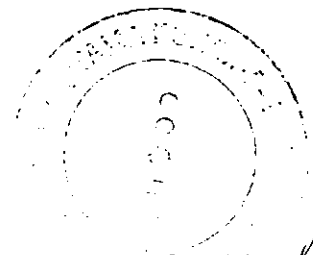



№	Описание	Последствия и сфера на влияние	Отговорник	Усещане на експозицията	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
16	Опасност от надхвърляне обхвата на изискванията за софтуера	Забавяне на Надвишаване на бюджета Въздействие на качеството на изпълнение на проекта. Влияние върху конкретната поръчка.	Изпълнител	8	Съществена	Ниска (21 - 40%)	Доработване на нови функционалности, които са извън утвърдената спецификация; Голям брой заявки за промяна	Разбрани, съгласувани и утвърдени изисквания в Етап 1 на проекта; Внимателно комуникиране на изискванията; Представяне на прототип; Въвличане на ключови потребители по итерации и разработка
17	Невъзможност за пълно тестване на системата	Наличие на дефекти в системата. Влияние върху конкретната поръчка.	Изпълнител	10	Висока	Ниска (21 - 40%)	Регистриране на инциденти в системата; Висок брой на дефектите	Прецизно планиране на тестването; Разработване на пълен набор от тестори сценарии; Използване на инструменти за автоматизирано тестване; Използване на симулатори за системите, които ще бъдат интегрирани към IDS; Осигуряване на пълна тестова среда.
18	Невъзможност за изпълнение на параметрите на SLA.	Превишаване на времето за престой на системата. Сферата на влияние е както за конкретната система, така за системите, които обменят информация чрез CISE Adapter.	Изпълнител, Възложител	10	Висока	Ниска (21 - 40%)	Неспазване на сроковете за реакция и остраняване на проблеми; Надвишаване на времената за престой и броя на планираните спириания	Сключване и спазване на SLA за работата на техническата инфраструктура т.к. без работеща инфраструктура няма как да работи самата система; Прилагането на проактивно наблюдение на системата чрез автоматизирани инструменти (от типа на Zabbix) значително ще ускори времето за реакция.

163

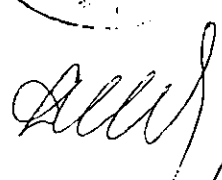
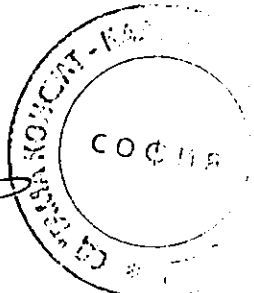

№	Описание	Последствия и сфера на влияние	Отговорник	Обща експозиция	Степен на значимост	Вероятност от настъпване	Индикатор	Мерки за минимизиране или предотвратяване на негативното въздействие и тяхната обосновка
							системата.	Еднозначно регламентиране на ангажиментите по SLA за Изпълнителя и Възложителя; Стриктно спазване на процедурите поддръжка; Прилагане на мероприятия за проактивна поддръжка.
19	Затруднения при надграждане и интеграция на системите, участващи в обмена	Забавяне или невъзможност за интеграция с конкретна система. Непълна функционалност. Сферата на влияние се разпростира върху системите, които участват в обмена на данни.	Възложител, Изпълнител	12	Съществена	Средна (41 - 60%)	Липса препоръки надграждане интеграция системите. Липса съдействие доставчика системата.	на за и на на от на Уточняване на препоръки за надграждане и изграждане на интерфейси към съществуващите системи във възможно най-ранен етап от изпълнението; Осигуряване на съдействие от доставчиците на системите, с които се извършва интеграция. За да бъдат препоръките към съществуващите системи оптимални и изпълними е желателно активното участие в процеса на екип от страна на организацията, които използват системите или техни подизпълнители.

164

5. Предпоставки за успешно изпълнение на поръчката

- Ангажираност на Възложителя в проекта – да определи проектен екип и ръководител, за които задачите по проекта ще бъдат с висок приоритет и ръководителят ще има необходимата оторизация.
- В проектния екип на Възложителя да бъдат подбрани експерти с познания в предметната област, които да могат да изпълняват ролята на ключови потребители.
- Да бъде осигурена информация и документация за съществуващите интернет страници на органите на изпълнителната власт в Сектор "Правосъдие". Желателно е тази информация да бъде подсигурана предварително.
- Да бъде осигурен достъп до непубличната част на интернет-страниците, които ще бъдат анализирани.
- Предварително да бъдат подбрани аналогични портали за достъп на граждани и бизнес до информация е-услуги и е-правосъдие.
- При изготвянето на архитектурата на системата да се вземат предвид добри и утвърдени световни практики.
- При изготвяне на спецификациите и тръжната документция да бъдат взети предвид, освен нормативните изисквания, и добри практики в областта.
- Съгласуване и приемане на съвместен план-график за изпълнение.
- Съгласуване и прилагане на план за управление на проекта.
- Поддържане на матрица за проследяване на изискванията (requirements traceability matrix).
- Установяване на процедура за управление на промените в началото на изпълнението.
- Поставяне на специфични, измерими, постижими, и изпълними изисквания (SMART - Specific, Measurable, Attainable, Realisable, Traceable).
- Следване на утвърдени методи и нотации за специфициране на изискванията и архитектурата, като Унифициран език за моделиране (UML – Unified Modelling Language) и др.



34
165

към

ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

за участие в открита процедура за възлагане
на обществена поръчка с предмет:

**„Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в
съответствие с инженеринговите услуги, поддържащи прилагането на
EUCISE2020 мрежата и съпровождащи услуги по време на
демонстрационната фаза на обмен на данни“**

Предложение за технологично решение за
изграждане на EUCISE2020 Адаптер






Съдържание

1. Въведение.....	3
2. Архитектура на EUCISE2020 Адаптер.....	3
2.1. Принцип на работа и общи характеристики на предлаганата архитектура	3
2.2. Диаграма и описание на софтуерната архитектура	7
2.3. Описание на компонентите	9
2.4. Обмен на информация с компонентите на мрежата EUCISE2020	11
2.5. Технологии и средства за разработка	12
2.5.1. Технологии.....	12
2.5.2. Средства за разработка	13
2.5.3. Базов софтуер	13
2.6. Модел на данните.....	13
2.7. Приложимост към съществуващата информационна среда	14
2.8. Възможности за бъдещо развитие.....	15
3. Функционални характеристики на EUCISE2020 Адаптер	16
3.1. Диаграма на функционалностите	17
3.2. Описание на ключови функционалности.....	17
3.2.1. Протоколи за комуникация.....	20
3.2.2. Обмен на съобщения.....	20
3.3. Възможности за оптимизация на вътрешносистемните процеси и информационни потоци ..	26
3.4. Ограничения и предположения	26
4. Нефункционални характеристики на EUCISE2020 Адаптер.....	27
5. Мотивация за предложеното технологично решение	33




2
167

1. Въведение

В настоящия документ сме представили технологично решение за разработване и внедряване на EUCISE2020 Адаптер и начина на реализация, надвишаващо изискванията на възложителя, съгласно техническата спецификация. Във функционално отношение е разписан принципа на работа и обмен на информация между отделните компоненти на системата, като е предоставена информация за ключови функционалности и експлоатационни качества. Предлагащото технологично решение е конкретно, обвързано с предназначението на системата, показва приложимостта към съществуващата информационна среда, разработено е изцяло в контекста на поставената цел, осигурява необходимия качествен резултат от нейното внедряване, и заедно с това съдържа възможности за оптимизация на вътрешносистемните процеси и информационни потоци и осигурява условия за бъдещо развитие и усъвършенстване.

Технологичното решение отговаря на изискванията на Възложителя, посочени в указанията и техническата спецификация, действащото законодателство, съществуващите изисквания и стандарти и съдържа и следните надграждащи елементи:

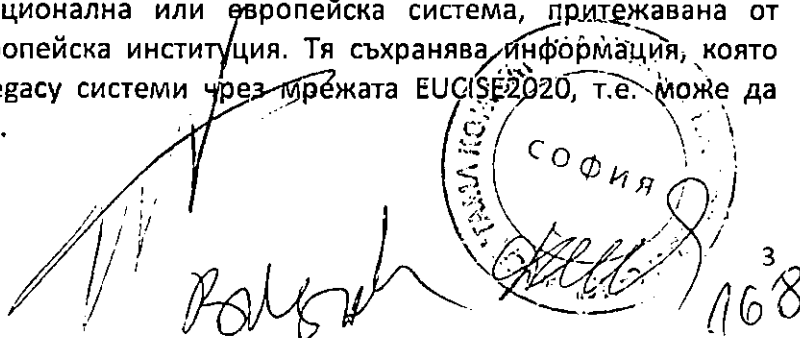
- а) съдържа освен описание на начина на изпълнение на предмета на поръчката, но и анализ на предложените дейности посредством които ще постигне очакваните резултати с цел максимално задоволяване на потребностите и нуждите на Възложителя и заинтересованите страни – информацията е представена детайлно в Приложение 1 към Техническото предложение – „Предложение за организация за изпълнение и управление на поръчката“, т. 4, т. 5 и т. 6;
- б) представената организационна структура, разпределението на ключовите експерти по функции, мобилизацията, разпределението на задачите и отговорностите между ключовите експерти, включва и отношенията, връзките на контрол и взаимодействие, освен в екипа, така и с Възложителя и останалите участници в изпълнението на дейността, включително контролните органи - информацията е представена детайлно в Приложение 1 към Техническото предложение – „Предложение за организация за изпълнение и управление на поръчката“, т. 3;
- в) в допълнение към представения от участника обхват, вероятност за настъпване и степен на въздействие върху всеки един от идентифицираните от Възложителя рискове са предложени мерки за недопускане/преодоляване на всеки един от тях, посредством които мерки може да се предотврати възникването или да ограничи негативното влияние на риска – информацията е представена детайлно в Приложение 2 към Техническото предложение – „Предложение за управление на риска при изпълнение на поръчката“.

2. Архитектура на EUCISE2020 Адаптер

2.1. Принцип на работа и общи характеристики на предлаганата архитектура

EUCISE2020 мрежата е среда за обмен на информация между Legacy системи посредством общ комуникационен протокол.

Legacy система е съществуваща национална или европейска система, притежавана от национален публичен орган или европейска институция. Тя съхранява информация, която може да бъде споделена с други Legacy системи чрез мрежата EUCISE2020, т.е. може да предоставя или получава информация.

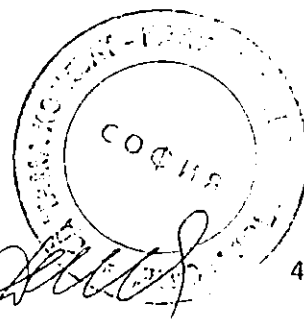
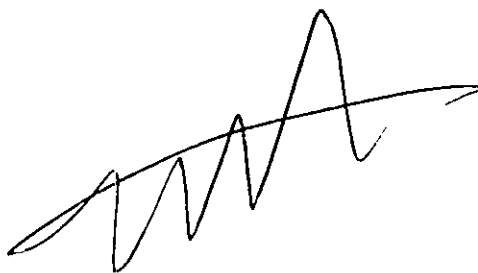


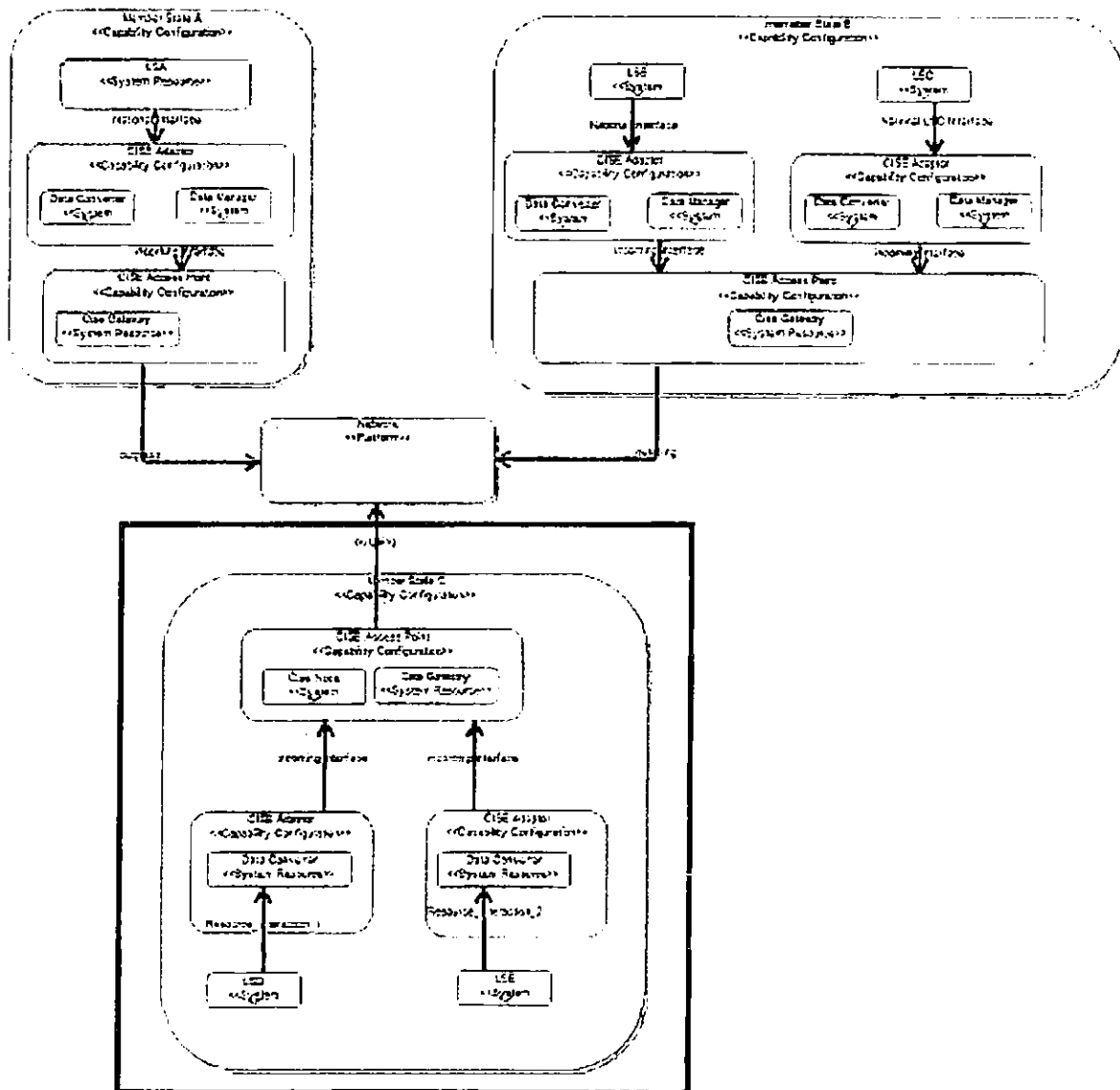
Всяка Legacy система обменя информация чрез свой EUCISE2020 адаптер. Адаптерът преобразува комуникационния протокол и моделът на данните на Legacy системата в EUCISE2020. По този начин адаптерът ще позволи на Legacy системата да предоставя и консумира информационни услуги през EUCISE2020 мрежата с другите свързани Legacy системи като запази своята автономност по отношение на реализираните бизнес процеси и модели на данни.

Конфигурацията на EUCISE2020 мрежата включва следните компоненти:

- Съществуващи (Legacy) системи, заинтересовани от обмена на информация;
- EUCISE2020 адаптери, които позволяват на Legacy системите да се свързват с EUCISE2020 мрежата. Те преобразуват данните от Legacy системите в общия модел на данни EUCISE2020 и гарантират, че съответната Legacy система е в състояние да предоставя и използва услугите на EUCISE2020;
- Gateway EUCISE2020 (GW) трансферира CISE съобщения и реализира мрежови протоколи за обмен на данни с CISE адаптера и с другите CISE Gateways / Nodes.
- EUCISE2020 Node (NODE) е подобрен шлюз, способен да предоставя услуги с добавена стойност като синтез на данни и съхранение на информация.

Видовете конфигурации и начините на присъединяване на Legacy системи към EUCISE2020 мрежата е в съответствие със следната схема:





[Handwritten signature]

[Handwritten signature]

Всеки участник в EUCISE2020 мрежата (държави-членки и публичните органи) може да избира между три различни модела на свързване - А, В и С съгласно схемата. Всеки модел е в съответствие с CISE Hybrid Vision архитектурата и определя как държавните власти на държавата-членка ще свържат своите системи в мрежата.

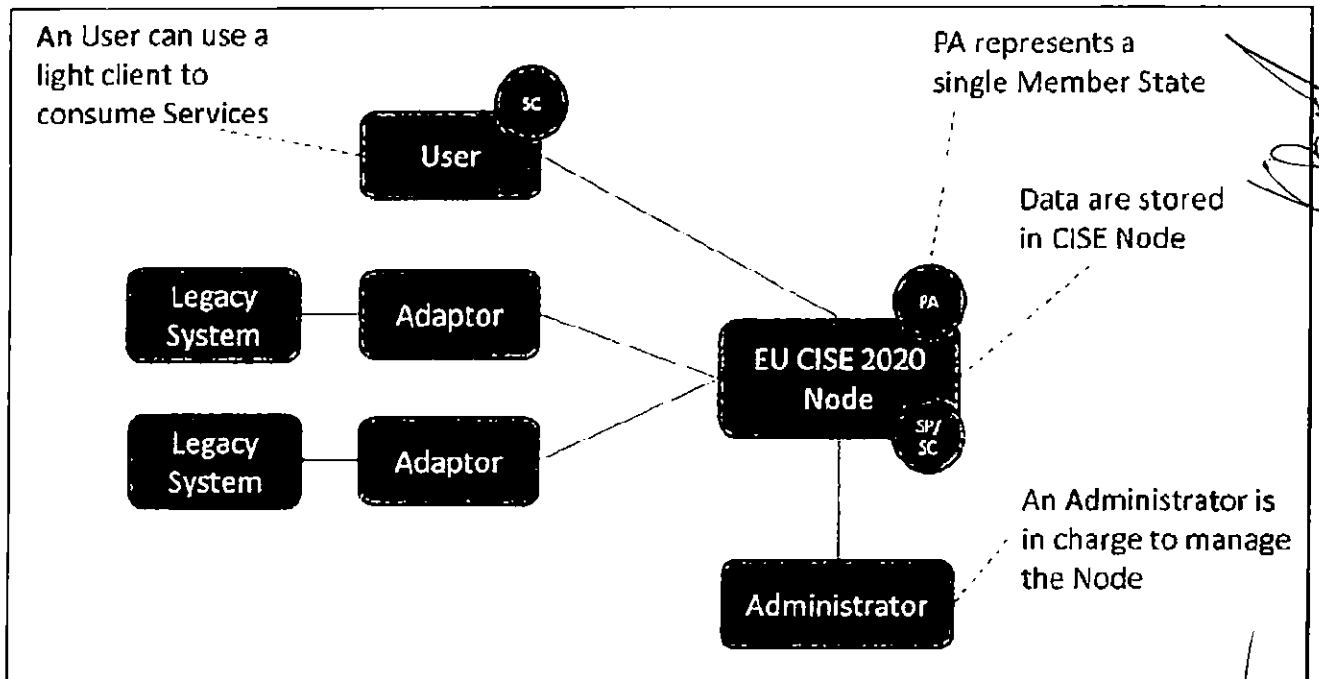
ИАМА ще обменя данни през EUCISE2020 мрежата на основата на конфигурация С.

В конфигурация С няколко държавни органа от съответната държава-членка свързват свои собствени Legacy системи към EUCISE2020 мрежата с единствен EUCISE2020 възел тип С.

Всяка Legacy система осигурява и консумира EUCISE2020 услугите чрез отделен EUCISE2020 адаптер.

[Handwritten signature]
[Handwritten signature]
 СОФИЯ
 5
 120

В тази конфигурация съществува връзка "1-1" между адаптера и Legacy системата и "1-N" между EUCISE2020 възел тип C и N на брой адаптери съгласно схемата:



EUCISE2020 Configuration C

EUCISE2020 адаптера, предмет на настоящата разработка, ще бъде проектиран и изпълнен с помощта на многослойна Java EE базирана модулна архитектура.

Модулният принцип на адаптера осигурява по-лесното му надграждане, разширяване и обслужване. Общите за всички следващи внедрявания на EUCISE2020 адаптера функционалности ще бъдат обособени в отделни модули с цел преизползване без необходимост от промени при запазен обектен и функционален модел на EUCISE2020 мрежата.

Всеки от слоевете на EUCISE2020 адаптера ще бъде базиран на актуални и съвременни набори от библиотеки, в максимална степен покриващи изискванията за производителност и сигурност. В архитектурата се предвижда залагането на системни компоненти и библиотеки, осигуряващи възможност за скалируемост и резервиране.

Заложената архитектура и всички системни и приложни софтуерни компоненти ще бъдат подбрани и/или разработени така, че да осигуряват работоспособност и отказоустойчивост на адаптера, както и недискриминационно инсталиране, т.е. без различни условия за инсталиране върху физическа и виртуална среда.

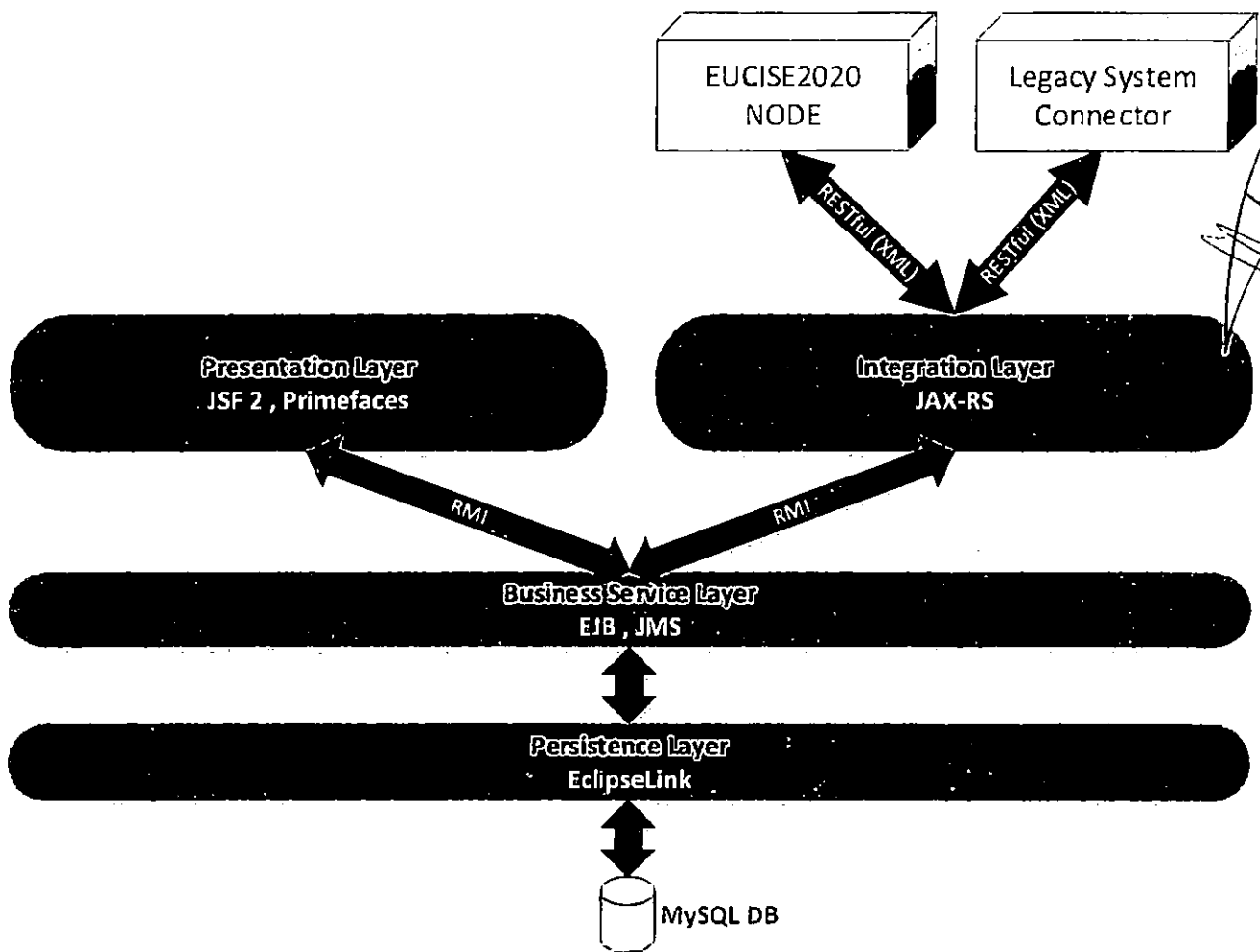
Комуникацията с EUCISE2020 NODE-а и Legacy системата ще се осъществява изцяло чрез RESTful уеб услуги върху HTTPS протокол. От страна на EUCISE2020 NODE-а ползването на предоставените услуги ще се осъществява през тяхната RESTful реализация. За комуникация с

Handwritten signatures and a circular stamp are present at the bottom of the page. The stamp contains the text 'ДИРЕКЦИЯ ЗА ПРОЕКТИРОВАНИЕ И ИЗПЪЛНЕНИЕ НА ПРОЕКТИ' and 'ПАИС'. A handwritten number '6' is visible on the right side.

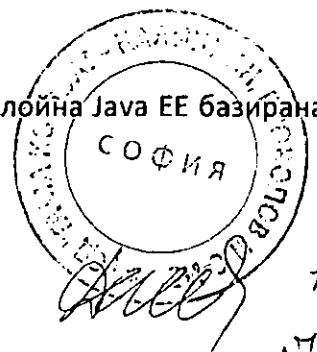
Legacy системите ще бъде проектиран интерфейс също базиран изцяло на RESTful услуги тъй като са популярен и широко приложим метод за комуникация между системи с максимална гъвкавост и независимост по отношение на инфраструктурата.

Предлаганата архитектурата на адаптера се базира изцяло на технологии, системни компоненти и библиотеки с отворен код, които едновременно с това са платформено независими. Това прави архитектурата и разработените базови компоненти на адаптера (комуникационен протокол и интеграция с мрежата) приложими и за следващи интеграции на национални системи с EUCISE2020.

2.2. Диаграма и описание на софтуерната архитектура



EUCISE2020 адаптера ще бъде изграден на модулен принцип с многослойна Java EE базирана архитектура, представена на горната схема.



7
172

Database Layer

Database слой ще бъде реализиран с релационна база данни MySQL Community Edition. Избраната база данни е с високата производителност, надеждност и сигурни механизми за защита на данните. В същото време не се изисква платен лиценз за ползването ѝ.

При проектирането и работата с базата данни ще бъдат следвани добрите практики за дизайн и разработка, осигуряващи оптимален, лесен за поддържане и разширения дизайн, бързодействие и консистентност на данните.

Persistence Layer

Persistence слой е междинен слой между базата данни и Business Service слой и служи за изолиране на горните слоеве от конкретно избрана база данни и физическата структура на данните в нея. По този начин има възможност за представяне на структурата на данните, независимо от реалната база данни и физическо представяне на данните. Физическата база и/или физическия модел на данните по този начин стават независими от другите слоеве и могат да бъдат подменени без това да налага промени в тях. Независимата от програмния код подмяна на физическата база създава възможност за поддържане на тестови инсталации и инсталации за обучение без необходимост от ползване на високо производителни бази, изискващи лицензи, хардуерен ресурс и специална инсталация и поддръжка. Друго предимство на Persistence слой е по-доброто капсулиране на логиката на базата данни в един слой.

За Persistence слой ще бъде използвана EclipseLink имплементацията за JPA (Java Persistence API). Основните предимства на избраната имплементация са следните:

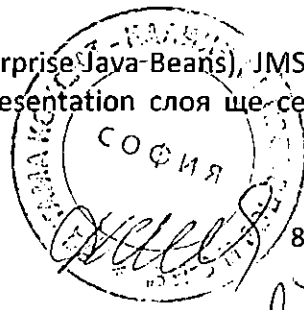
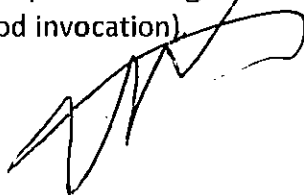
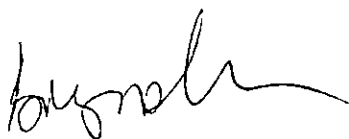
- Посредством заложените механизми за кеширане в EclipseLink се постига по-добра производителност при работа с базата данни;
- EclipseLink притежава вграден механизъм за поддържане на версии на данните и одит информация;
- EclipseLink е независим от конкретната база данни, което дава възможност при необходимост да бъдат създавани тестови инсталации и инсталации за обучение с ползване включително на бази със свободен лиценз.

Business Service Layer

В Business Service слой ще бъде имплементирана:

- бизнес логиката на EUCISE2020 адаптера;
- комуникационния протокол на адаптера с Legacy системата и EUCISE2020 NODE-a
- трансформацията между моделите на Legacy системата и EUCISE2020 NODE-a.

За целта ще се използват стандартни Java EE библиотеки като EJB (Enterprise Java Beans), JMS (Java Message Service) и други. Комуникацията с Integration слой и Presentation слой ще се осъществява чрез Java RMI (Remote Method invocation).



DB Data Model модулът ще реализира модела на данните в базата данни. Модулът може да остане непроменен и при бъдещи внедрявания на адаптера за връзка на други Legacy системи с EUCISE2020, ако не е необходимо разширение на функционалността, изискващо промяна на модела.

CISE Data Model модулът ще реализира общия EUCISE2020 модел на данните. Модулът може да остане непроменен и при бъдещи внедрявания на адаптера за връзка на други Legacy системи с EUCISE2020, ако бъде запазен текущия EUCISE2020 модел.

LS Data Model модулът ще реализира модела на данните от Legacy системата, предмет на интеграцията.

Transformation модулът ще реализира трансформацията на данните от общия EUCISE2020 модел към модела на Legacy системата и обратно.

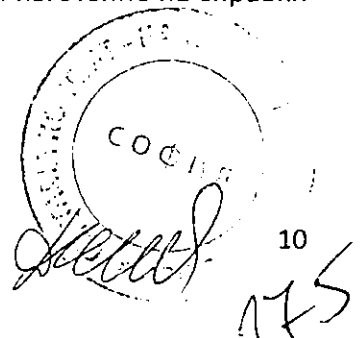
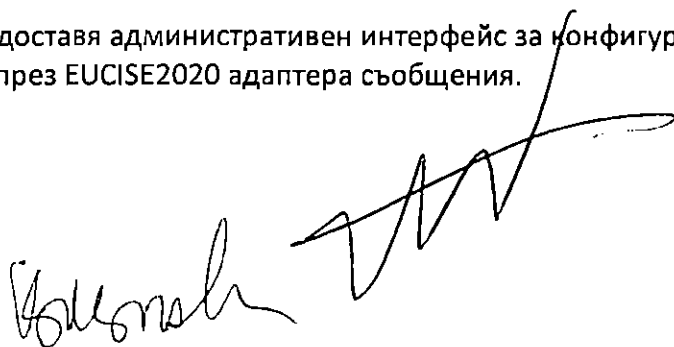
Digital Signature модулът ще реализира поточно електронно подписване (с цел да се избегне зареждането на цялото съдържание в оперативната памет) с всички видове електронни подписи издадени от Доставчици на доверителни услуги в ЕС, които отговарят на изискванията за унифициран профил на електронните подписи, съгласно подзаконовите правила към Регламент ЕС 910/2014, които влизат в сила и са задължителни от 1 януари 2017 г. Той ще подписва сигурен хеш-ключ, генериран на базата на образа/съдържанието на документа. Минимално сигурният алгоритъм за хеширане, който ще се използва е SHA-256.

Ще има възможност всяко съобщение, рутинано в EUCISE2020 мрежата да бъде подписано преди изпращането му както и ще бъде верифициран подписа на всяко съобщение, получено от Националния EUCISE2020 адаптер.

Core модулът е основният модул на системата. Чрез него ще се осъществява запис и четенето от базата данни. В него ще бъде имплементирана бизнес логиката по общия EUCISE2020 комуникационен протокол на адаптера с Legacy системата и EUCISE2020 NODE-а. При ползване на услуга през адаптера ще бъде идентифициран заявителя и ще бъде правена проверка за правата му за достъп до заявената услуга, по предвидения от протокола и правилата за достъп ред.. Модулът ще се грижи и за правилното изпълнение на услугите съгласно заложените по протокол правила така, че всяка от комуникиращите страни да има навременна и коректна информация за статуса на изпълнение. Модулът може да остане непроменен и при бъдещи внедрявания на адаптера за връзка на други Legacy системи с EUCISE2020, ако бъде запазен текущия EUCISE2020 комуникационен протокол.

Integration модулът ще осъществява комуникацията с Legacy системата и EUCISE2020 NODE-а, чрез RESTful уеб услуги имплементирани с помощта на стандартната Java EE библиотека JAX-RS (Java API for RESTful Web Services).

UI модулът предоставя административен интерфейс за конфигуриране и изготвяне на справки за обменяните през EUCISE2020 адаптера съобщения.



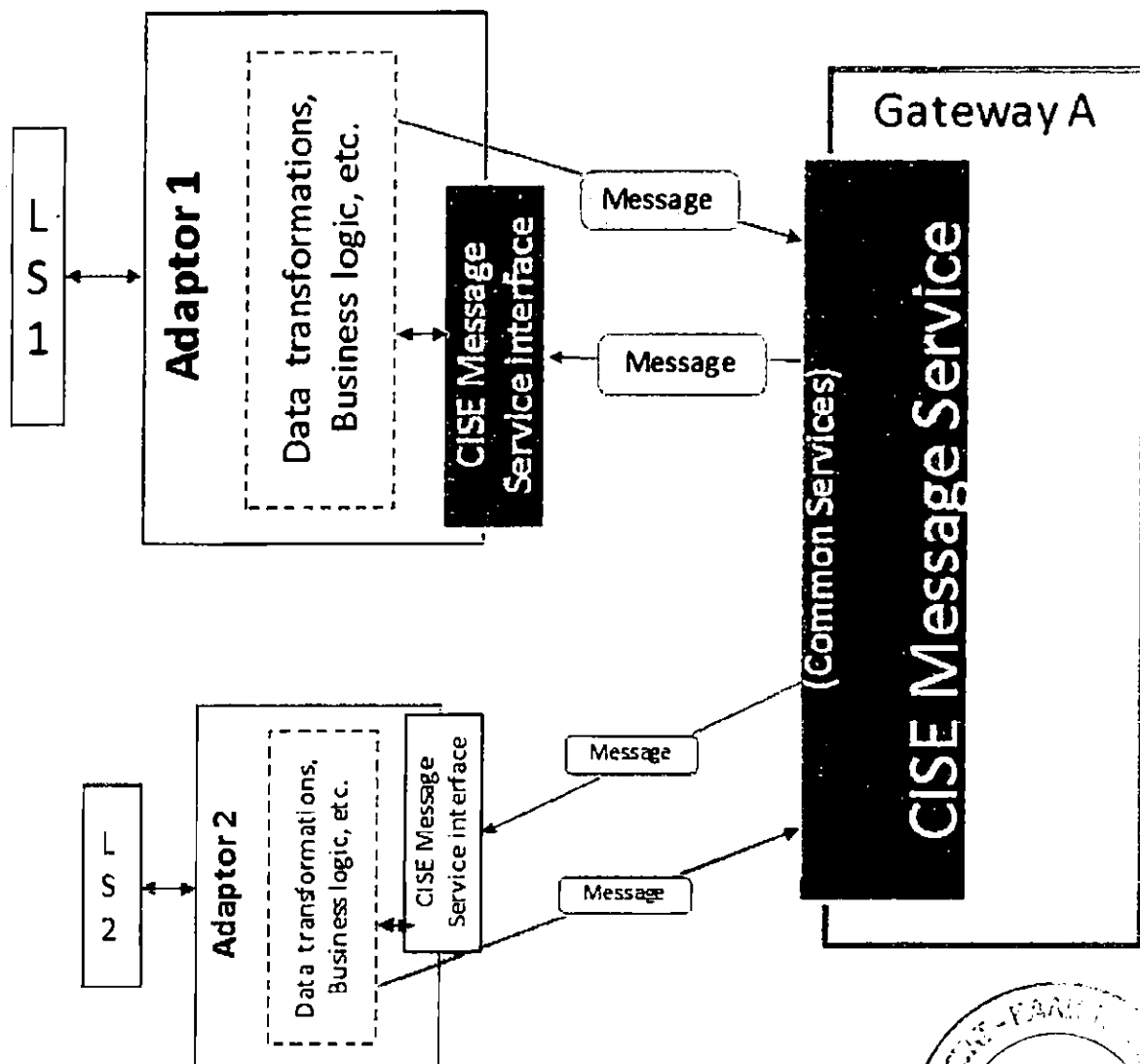
През този модул администраторът ще има възможност да конфигурира и лимитите за достъп до отделни веб-услуги, включително на параметрите на поддържани на приложно ниво "Rate Limiting" и/или "Throttling" на заявки от един и същ клиентски адрес.

Модулът ще предоставя възможност за достъп до справки за бизнес информацията, обменена през EUCISE2020 адаптера както и информация за дата и час на обмена, компонент (модул или система), от който е осъществен обмена, IP адрес на заявителя и др.

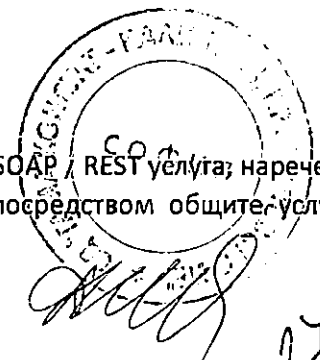
2.4. Обмен на информация с компонентите на мрежата EUCISE2020

EUCISE2020 адаптерът е връзката между Legacy системата и Common Services, достъпни в Gateway.

Следната диаграма изобразява моделът за обмен на информация между Legacy система и EUCISE2020 мрежата през адаптера:



Интерфейсът за услуги на EUCISE2020 мрежата е реализиран под формата SOAP / REST услуга; наречена "CISE Message Service". Всеки реален обмен на данни през мрежата посредством общите услуги



представлява поредица синхронни и/или асинхронни съобщения, базирани на дефинирани общи правила и модел на данните. Интеграцията с Legacy системата обаче е специфична и напълно персонализирана и зависи от нейните технически и функционалните възможности и нужди.

Адаптерът има две основни функции, действа като потребител и доставчик, по време на процеса на обмен на информация между Legacy системата и EUCISE2020 мрежата.

В ролята си на потребител, адаптерът предоставя част от общият интерфейс, посредством който Common Services могат да се свържат с него и да предадат информацията, изпратена от друг участник. Чрез извикване на аналогичен, предоставен от Common Services интерфейс адаптерът изпраща информация до EUCISE2020 мрежата.

2.5. Технологии и средства за разработка

2.5.1. Технологии

Java: обектно-ориентиран език за програмиране. Кодът, написан на Java, не се компилира до машинен код за определен процесор, а до специфичен за езика код, наречен байт код. Поради това за изпълнението на програма, написана на Java, е необходима т. нар. виртуална машина. Подобната реализация има своите предимства, като лесната преносимост между различните платформи, висока степен на сигурност и други.

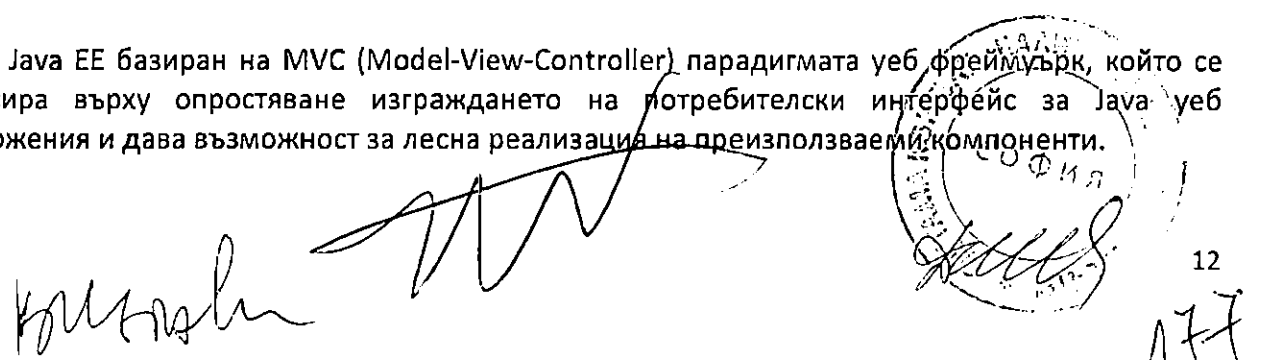
EclipseLink: имплементация на JPA (Java Persistence API). EclipseLink е обектно-реляционна картографска библиотека за Java, която осигурява рамка за картографиране на обектно-ориентиран домейн в традиционна реляционна база данни със заложен механизъм за кеширане, поддръжане на версии на данните и одит информация.

EJB (Enterprise Java Beans): Java API за модулно изграждане на високо мащабируем и надежден корпоративен софтуер. EJB е софтуерен компонент на сървъра, който реализира бизнес логиката на дадено приложение. EJB уеб контейнерът осигурява среда за изпълнение за уеб компоненти, включително компютърна сигурност, управление на жизнения цикъл на Java servlet, обработка на транзакции и други уеб услуги.

JMS (Java Message Service): част от Java EE, дефинирана от спецификацията JSR 914. Това е стандарт за съобщения, който позволява компонентите на приложението, базирани на Java EE да създават, изпращат, получават и четат съобщения. Тя позволява комуникацията между различните компоненти на разпределено приложение да бъде свободно изградена, надеждна и асинхронна.

JAX-RS (Java API for RESTful Web Services): Java EE библиотека, която осигурява поддръжка при създаването на уеб услуги в съответствие с архитектурния модел на Representational State Transfer (REST).

JSF 2: Java EE базиран на MVC (Model-View-Controller) парадигмата уеб фреймуърк, който се фокусира върху опростяване изграждането на потребителски интерфейс за Java уеб приложения и дава възможност за лесна реализация на преизползваеми компоненти.



Primefaces: Фреймуърк с повече от 100 визуални компоненти за JSF 2 с отворен код.

2.5.2. Средства за разработка

Системата ще бъде разработвана на IntelliJ IDEA, интегрирана Java среда за разработка (IDE) на компютърен софтуер.

За компилиране на проекта ще бъде използван Maven. Maven е инструмент с отворен код, традиционно използван в Java и Java EE проекти за компилиране на изходни файлове, изпълнение на единични тестове и съставяне на артефакти за разпространение.

2.5.3. Базов софтуер

За разработка на системата ще бъде използван JDK SE Development Kit 8.

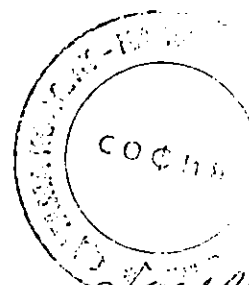
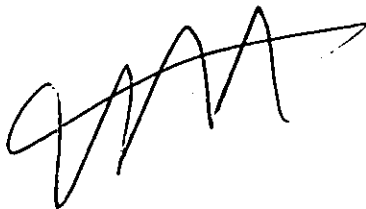
За приложен сървър ще бъде използва Oracle Glassfish Full Platform. Той е референтна реализация на Java EE и като такава поддържа Enterprise Java Beans, JPA, Java Server Faces, JMS, RMI, Java Server Pages, Java Servlet и др. Това позволява разработването на приложения, които са преносими, мащабируеми и интегрирани с традиционните технологии. Административният му интерфейс позволява лесна конфигурация и поддръжка.

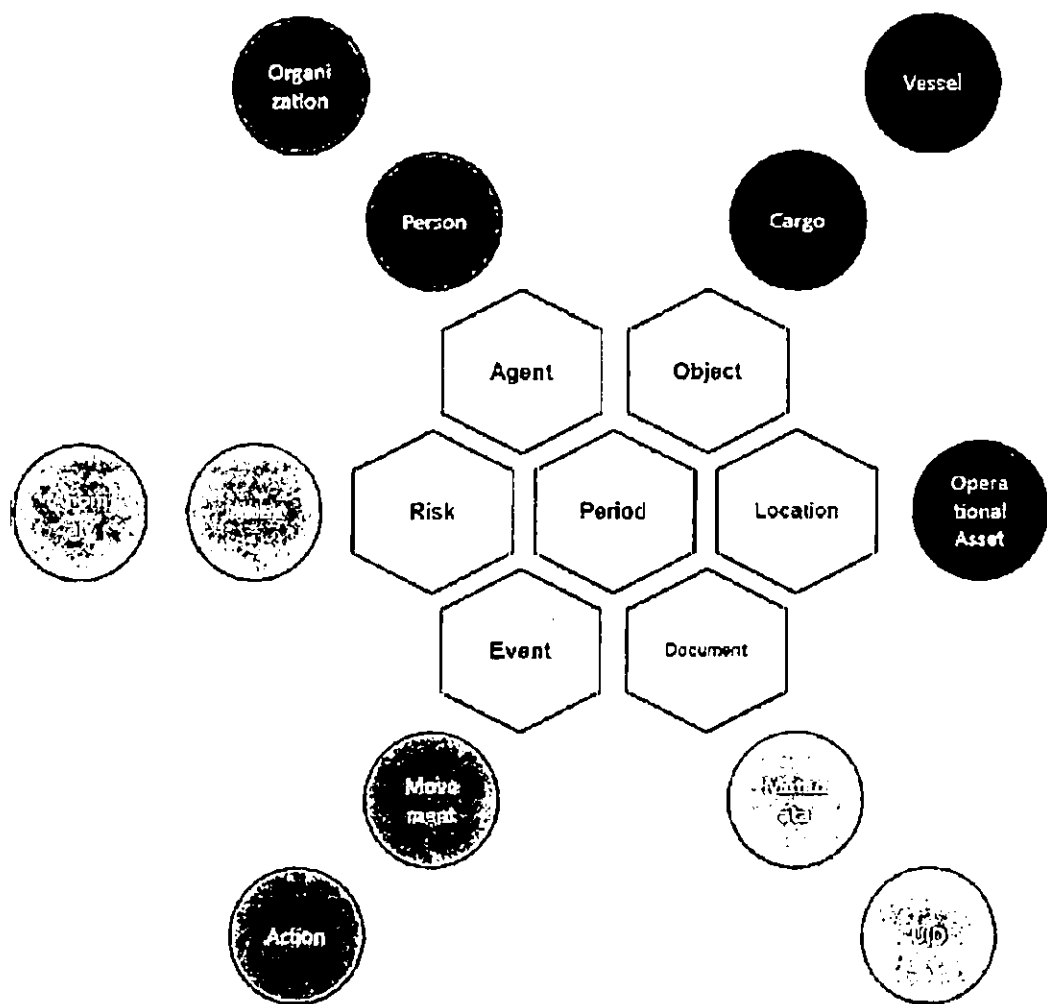
За база данни ще бъде използван MySQL Community Edition, поради високата си производителност, надеждност, сигурни механизми за защита на данните, възстановяване след срив без загуба на данни.

Предвиденият за разработката базов софтуер не изисква закупуване на лицензи като в същото време не се прави компромис с никой от задължителните за EUCISE2020 адаптера параметри като производителност, сигурност, скалируемост, отказоустойчивост и възможност за резервиране.

2.6. Модел на данните

В EUCISE2020 адаптера ще бъде реализиран цялостния модел на данните на EUCISE2020:





[Handwritten signature]

[Handwritten signature]

От страна на Legacy системата ще бъде реализиран модел, съответстващ на съществуващите бизнес обекти, относими към общия обектен модел на мрежата.

Една от основните роли на адаптера е преобразуването на данните от Legacy системата към общия модел на мрежата при предоставяне на информация и обратно при получаване на информация. В частност конкретната реализация ще покрива преобразуване на моделите на данните при обмен на съобщения между системата VTMISS (Национална система за събиране на данни за корабоплаването) и EUCISE2020 мрежата.

2.7. Приложимост към съществуващата информационна среда

EUCISE2020 мрежата е предназначена за обмен на информация между Legacy системите на държави-членки и публичните органи. Обменът на информация в самата мрежа се осъществява посредством единен модел на данните и предварително описани и реализирани правила (протокол). Всяка Legacy система се свързва с EUCISE2020 мрежата посредством адаптер. Адаптерът имплементира общия протокол и модел на данните на мрежата. Тази негова функционалност би трябвало да бъде обща и независима от конкретната Legacy

[Handwritten signature]

[Handwritten signature]

[Official stamp and handwritten signature]

[Handwritten mark]

система, към която осъществява връзка. От друга страна адаптерът поддържа модел на данни, базирани на обектите в Legacy системата и се грижи за трансформацията между двата обектни модела. Адаптерът реализира и специфична имплементация на правилата за обмен от страната на Legacy системата, съобразени с нейните технологични и бизнес особености.

По този начин без необходимост от съществени промени и надграждане, всяка Legacy система посредством адаптер може да обменя информация с други такива системи като се базира на собствения си модел на данни и бизнес процеси. Така съществуващата информационна среда остава непроменена от гледна точка на бизнес правила и данни като в същото време се дава възможност на всяка система да обменя данни с други системи, независимо от техните вътрешни правила и структура на данните.

Адаптерът, предмет на настоящата разработка, ще бъде проектиран и изпълнен така, че частта, реализираща общия EUCISE2020 модел на данните и протокола на EUCISE2020 мрежата, да бъдат обособена в отделен набор базови преизползваеми модули, позволяващи самостоятелно надграждане и прилагане при следващи внедрявания.

Специфичните за конкретната Legacy система обектен модел и правила за обмен ще касаят единствено конкретното внедряване и ще бъдат обособени във втори специфичен за него набор модули.

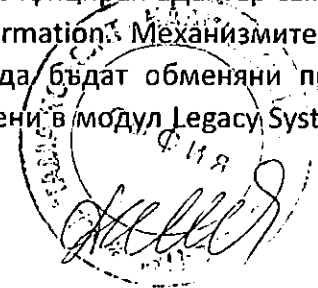
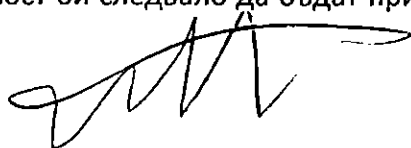
2.8. Възможности за бъдещо развитие

При необходимост от разширяване на обхвата на услугите, ползвани от Legacy системата или на данните, предоставяни от нея, ще бъде необходимо единствено разширяването на обектния модел (модул LS Data Model) и надграждане на модула за трансформация на данните (Transformation).

Ако е необходим трансфер на данни през адаптера от дадена Legacy система с голям обем и интензитет, архитектурата позволява прилагане на клъстеризация и балансиране на натоварването с минимална промяна на програмния код.

Всяко едно бъдещо внедряване на адаптера за връзка на друга Legacy система към EUCISE2020 мрежата ще изисква единствено разработка на специфичен модел на данните за самата Legacy система (LS Data Model), алгоритъм за трансформацията им от и към общия обектен модел на мрежата (Transformation) и при необходимост допълване и/или промяна на протокола за обмен с Legacy системата, ако текущо реализирания не е приложим поради технологични или бизнес особености на самата система. Всички останали модули на адаптера ще могат да бъдат използвани без необходимост от промяна при непроменен обектен и функционален модел на EUCISE2020 мрежата.

Разработеният адаптер може да бъде база за реализиране и на класифициран-адаптер само с подмяна на двата специфични модула LS Data Model и Transformation. Механизмите за преобразуване и филтриране на данни, които не би трябвало да бъдат обменяни през EUCISE2020 мрежата с цел сигурност би следвало да бъдат приложени в модул Legacy System



Connector на самата Legacy система. По този начин от гледна точка на адаптера цялата достигнала до него информация ще подлежи единствено на трансформация между моделите на данни, но не и на анализ по отношение на права за достъп и трансфер през EUCISE2020 мрежата.

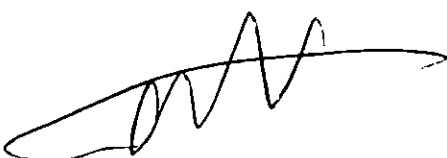

3. Функционални характеристики на EUCISE2020 Адаптер

Националният EUCISE2020 Адаптер ще бъде връзката между Legacy Системата VTМIS и Общите услуги (Common Services), достъпни в Gateway/Node в мрежата EUCISE2020. Той ще има връзка с набор от услуги и операции, които позволяват обмена на обектите на модела EUCISE2020. Адаптерът ще бъде реализиран като независим софтуерен компонент и ще осигурява презентирането на данни от VTМIS към мрежата EUCISE2020 и обратно. Ще изпълнява две интеграционни функции:

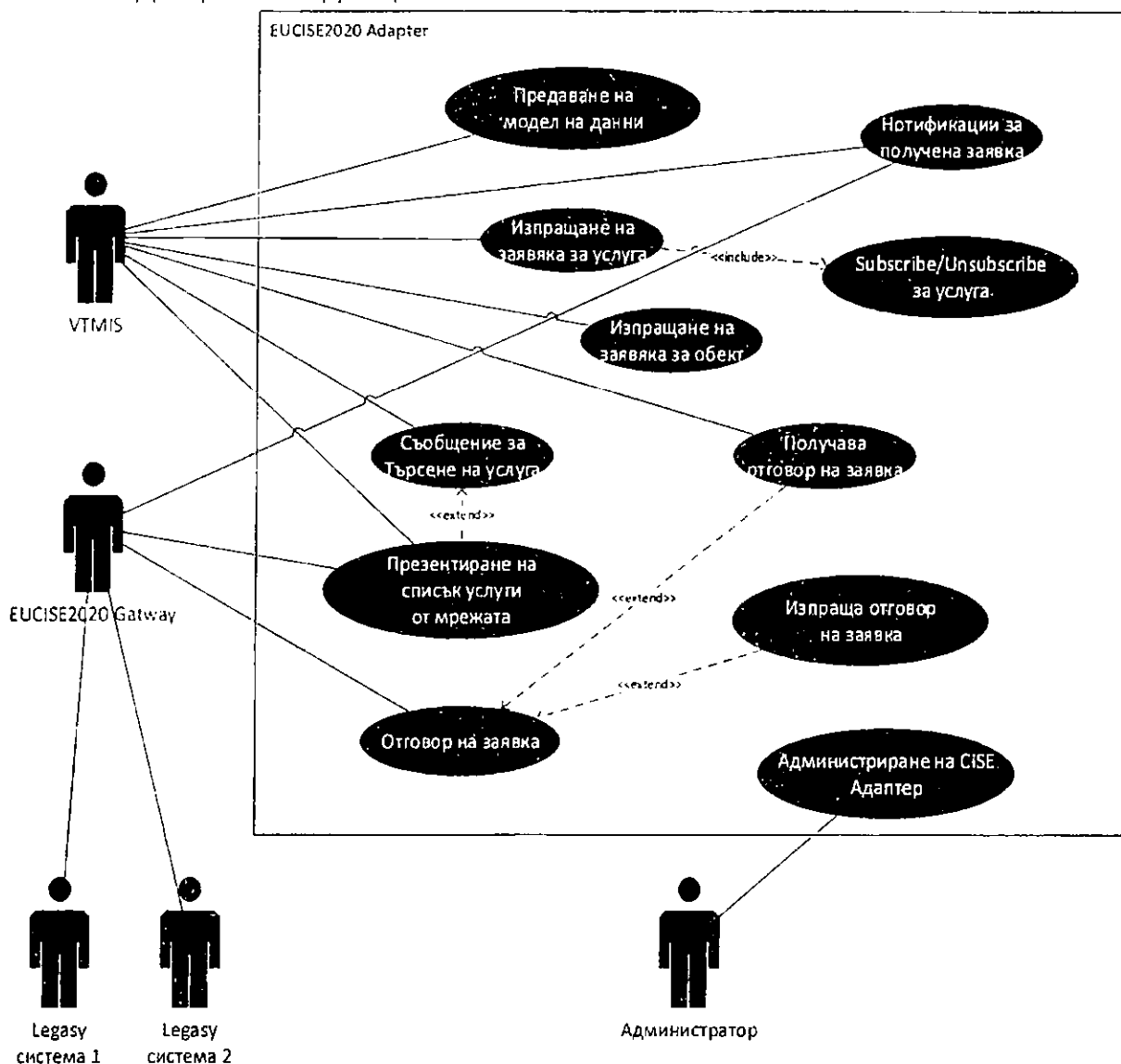
- изходяща – за свързване към Gateway/Node, за да изпраща съобщения и на други Legacy Системи;
- входяща – с цел да се позволи приемането на съобщения.

Докато интеграцията със Legacy системата VTМIS е напълно персонализирана и зависи от техническите и функционалните възможности и нужди, интеграцията с Общите услуги се основава на добре познати интерфейси и правила.

По отношение на интеграцията с Общите услуги, адаптерът ще има две основни функции, ще действа като потребител и доставчик, по време на процеса на обмен на информация между системата Legacy и мрежата EUCISE2020.

3.1. Диаграма на функционалностите



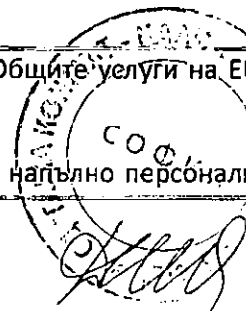
3.2. Описание на ключови функционалности

В таблицата по-долу са представени ключови функционални изисквания към Националния EUCISE2020 Адаптер:

№ по ред	Изискване
FREQ.1.	Националният адаптер ще бъде реализиран като независим софтуерен компонент. Ще осигурява последваща независимост при развитие на Legacy системата VTMISS.
FREQ.2.	Адаптерът ще има две основни функции, ще действа като потребител и доставчик, по време на процеса на обмен на информация между Legacy системата VTMISS и мрежата EUCISE2020.
FREQ.3.	Адаптерът ще бъде интегриран с Legacy системата VTMISS и Общите услуги на EUCISE2020 Gateway/Node. Докато интеграцията на Адаптера с Legacy системата VTMISS е напълно персонализирана и

[Handwritten signature]

[Handwritten signature]



№ по ред	Изискване
	зависи от техническите и функционалните възможности и нужди, то интеграцията му с Общите услуги ще се основава на добре познати интерфейси и правила.
FREQ.4.	<p>Компонентите на VTМIS, които ще участват в разработването на адаптера и влизани в обхвата на проекта са:</p> <ul style="list-style-type: none"> • <u>MSW.VTМIS.BG</u>; • <u>AIS.VTМIS.BG</u>. <p>За двете системи са налични модел на данните за обмен и разработени интерфейси, които подават данни към външни системи.</p>
FREQ.5.	Националният адаптер ще има възможност да осигури ползването на EUCISE2020 услуги от Националната система VTМIS, както и ще предоставя такива на други системи, включени в EUCISE2020 мрежата.
FREQ.6.	<p>Националният адаптер ще:</p> <ul style="list-style-type: none"> • превежда данните между вътрешния модел на данни на Националната система VTМIS и модела на данни на EUCISE2020; • имплементира EUCISE2020 модела на услуги.
FREQ.7.	Националният EUCISE2020 адаптер ще осигурява възможност на потребителите на Националната система VTМIS да достъпват EUCISE2020 мрежата, посредством получаване на данни от нея чрез разработения интерфейс Човек-Машина на самата система.
FREQ.8.	Адаптерът ще предоставя възможност за предоставяне и получаване на данни базирана на обмен на съобщения от вида PULL и PUSH.
FREQ.9.	Националният EUCISE2020 адаптер ще може да изложи на свързаната Национална система VTМIS списък на достъпните EUCISE2020 услуги.
FREQ.10.	Националният EUCISE2020 адаптер ще може да заявява EUCISE2020 услуга от EUCISE2020 Gateway/Node съгласно Модела на услугите на EUCISE2020.
FREQ.11.	Националният EUCISE2020 адаптер ще осигурява предаване на потребителските данни за достъп на потребителските профили от VTМIS към EUCISE2020 Gateway/Node. Валидирането и верифицирането ще се извършва в EUCISE2020 Gateway/Node съгласно предварително установените права за достъп.
FREQ.12.	Националният EUCISE2020 адаптер ще получава отговор на заявка за EUCISE2020 услуга от EUCISE2020 Gateway/Node. Отговорен за рутирането на заявката е EUCISE2020 Gateway/Node.
FREQ.13.	Националният EUCISE2020 адаптер ще получава отговор на заявка за EUCISE2020 обект от EUCISE2020 Gateway/Node. Отговорен за рутирането на заявката е EUCISE2020 Gateway/Node
FREQ.14.	Ще бъде реализирана интеграция на Адаптера с Общите услуги (Common Services), предлагани от EUCISE2020 Node. Адаптерът ще има две основни функции, ще действа като потребител и доставчик, по време на процеса на обмен на информация между Legacy

Handwritten signature

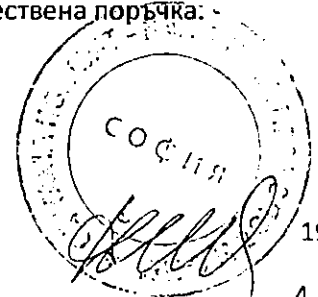
Handwritten signature

Official stamp and handwritten signature

№ по ред	Изискване
	системата и мрежата EUCISE2020.
FREQ.15.	<p>Реализирането на заявките към EUCISE2020 Gateway/Node и VTMISS, и отговорите съдържащи заявените метаданни ще бъде осъществено посредством услуга, наречена "CISE Message Service".</p> <p>Тази услуга предлага операция "изпращане", която получава типа съобщение. Този тип съобщение е родителска единица на няколко други, които представляват шаблоните за обмен на съобщения. Адаптерът ще бъде мост между Legacy Системата VTMISS и Общите услуги (Common Services) и като такъв ще има две интеграционни точки.</p>
FREQ.16.	Адаптерът трябва да реализира интерфейса CISEMessageService, с който ще комуникира Gateway Common Services.
FREQ.17.	Националният EUCISE2020 адаптер ще обработва заявката към кореспондиращата Национална система съгласно Модела на данни на EUCISE2020 използвайки стандартен протокол (като NMEA 0187, Asterix).
FREQ.18.	Националният EUCISE2020 адаптер ще превежда данните от Националната система на езика на Модела на данните на EUCISE2020.
FREQ.19.	Националният EUCISE2020 адаптер ще може да преведе данните на EUCISE2020 обекта на езика на Модела на данни на Националната система за да е сигурно, че EUCISE2020 обекта е показан коректно на интерфейса Човек-Машина на Националната система.
FREQ.20.	<p>Националният EUCISE2020 адаптер ще може да покаже на Legacy системата VTMISS достъпните EUCISE2020 услуги за сътрудничество, за да подsigури възможността за потребителите за взаимодействие през техния собствен интерфейс Човек-Машина.</p> <p>Най-малко следните услуги за сътрудничество ще бъдат достъпни (във фонов режим или на интерфейса Човек-Машина на Националната система):</p> <ul style="list-style-type: none"> • Услуга за електронна поща Email (изпращане/получаване/получатели); • Услуга за трансфер на файлове - Услуга за управление на трансфера на файлове сред EUCISE2020 участниците (качване/сваляне/изтриване); • Услуга за споделени документи (списък/създаване/изтриване/преименуване); • Услуга за споделен календар – споделяне на календари и събития (списък/създаване/споделяне на събития); • Услуга за видео и аудио конференция (изпращане/получаване); • Услуга за мигновени съобщения (Instant Message) (изпращане/получаване/получатели); • Услуга за Бяла дъска (White Board).

Следва описание на ключови функционалности, които Кандидатът ще реализира при разработването на EUCISE2020 Адаптера, ако бъде избран за Изпълнител на настоящата обществена поръчка:

[Handwritten signature]



184

3.2.1. Протоколи за комуникация

Оперативната съвместимост между Адаптера и Общите услуги (Common Services) ще бъде осъществяван на базата на технически интерфейс, който ще се използва от адаптера, за да комуникира с мрежата EUCISE2020 и обратно. Този интерфейс ще бъде реализиран на базата на SOAP/REST протоколи. Услугата, която ще се предоставя е наречена "CISE Message Service". Тази услуга ще предлага операция "изпращане", при която ще се получават различни по тип съобщения. Типът на съобщението е родителска единица на няколко други характеристики, които формират шаблоните за обмен на съобщения.

За изграждане на функционалностите на Националния EUCISE2020 адптер, Изпълнителят предвижда да използва REST протокол. Изпълнителят ще осигури необходимата информация така че да бъдат извършени необходимите конфигурации по време на регистрацията на услугата в EUCISE2020 Gateway/Node.

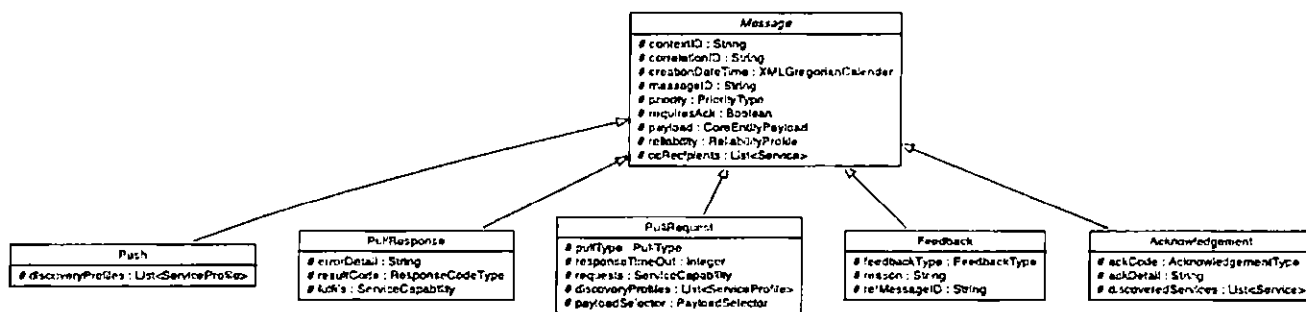
3.2.2. Обмен на съобщения

Интерфейсът на Общи услуги (Common services) за изпращане на съобщения – CISEMessageService, определя интерфейса, използван от адаптерите, за да комуникира с мрежата EUCISE2020 и обратно.

Адаптерът ще предоставя необходимия на Общите услуги интерфейс, който ще бъде разработен от Изпълнителя и е необходим за свързването и предаването на изпратената от друг участник информация.

Съображения:

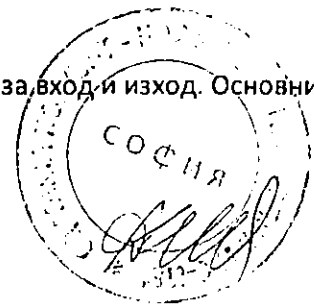
- Адаптерът ще използва интерфейса CISEMessageService, за комуникация с Общите услуги (Common Services) на Gateway/Node.
- Изпълнителят ще разработи собствен CISEMessageService и ще бъде отговорен за внедряването



на бизнес логиката вътре в него (CISEMessageService).

Поддържаните операции в интерфейса CISEMessageService на Общите услуги (Common Services) на Gateway/Node са представени на фигурата по-долу:

Всички налични операции се поддържат от специфични структури за данни за вход и изход. Основните елементи са Push, Pull Request, Pull Response, Feedback и Acknowledgment.



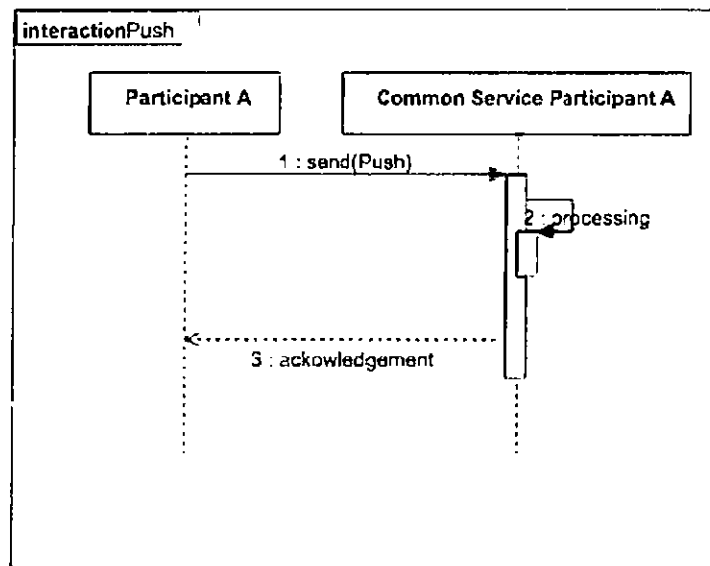
20
185

Изпълнителят ще реализира собствена услуга CISEMessageService, която ще съдържа следните операции:

Push

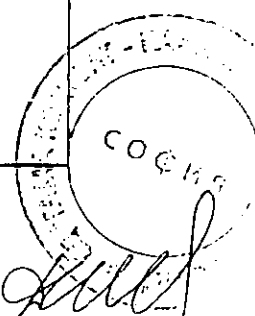
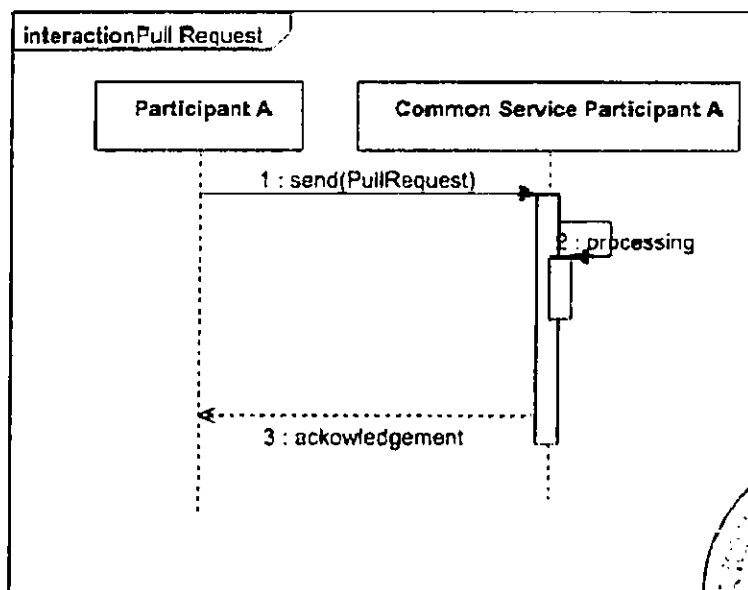
При тази операция адаптерът ще изпрати съобщение Push и ще получи потвърждение, което показва, че съобщението е изпратено до мрежата EUCISE (потребители).

На по-късен етап, могат да бъдат получени други асинхронни потвърждения, което ще показва, че съобщението е доставено на крайните потребители.



Pull

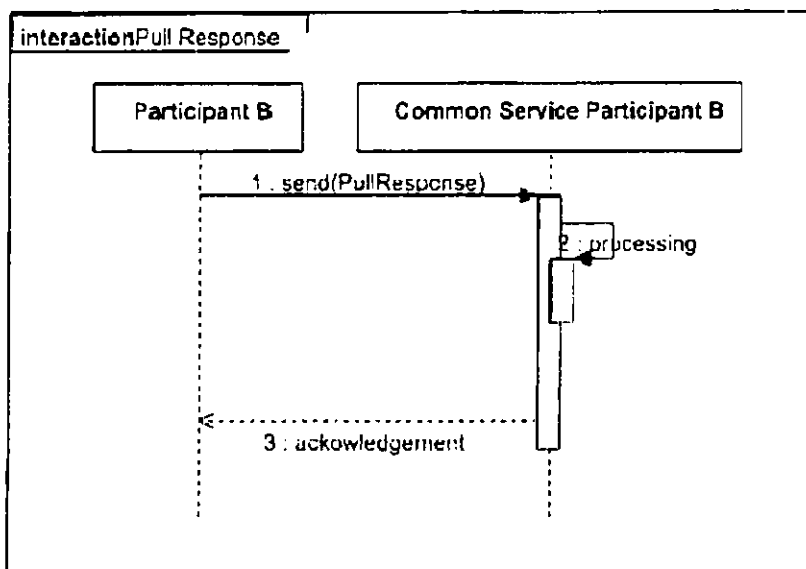
Операциите Pull request и Pull response се случват в два различни момента във времето, тъй като операцията Pull е асинхронна, резултатите от нея ще настъпят по-късно във формата на Pull response.



Handwritten signatures and scribbles at the bottom of the page.

Адаптерът ще изпрати съобщение за Pull request и ще получи потвърждение, показващо, че съобщението е изпратено до мрежата EUCISE.

На по-късен етап, могат да бъдат получени други асинхронни потвърждения, което ще показва, че съобщението е доставено.

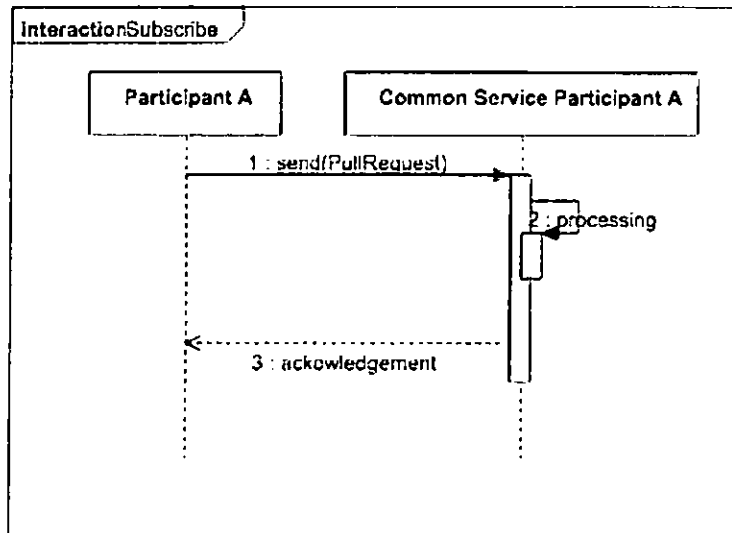


На по-късен етап, други адаптери в мрежата, които са получили съобщение за Pull Request ще подготвят отговор и ще изпратят съобщение за Pull response до заявителя.

Subscribe

Моделът "Subscribe/Unsubscribe" се състои от три независими комуникационни процеса: един за абонамент (обработван с операция Pull) и друг за публикуване (обработван с операцията Push).

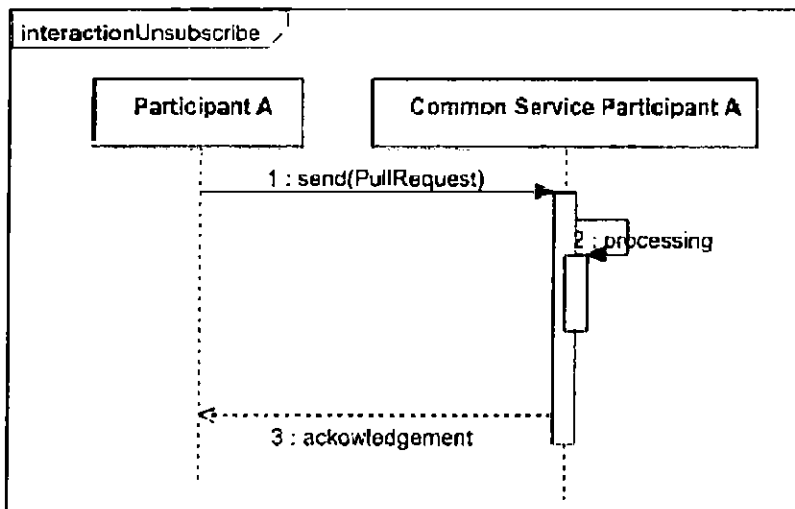
- **Subscribe:** Потребителят на CISE се абонира за информация от доставчика на CISE, като използва операцията Pull. В този случай съобщението PullRequest ще показва, че това е процес на абонамент (PullType = абониране). Доставчикът на услуга в мрежата на EUCISE2020 ще регистрира потребителя от мрежата и ще изпрати идентификационен номер за корелация (correlationID) в съобщението PullResponse.
- **Publish:** Когато доставчикът на услуга в мрежата на EUCISE2020 иска да предостави информация (напр. Списък с обекти за данни), Gateway на доставчика трябва да провери списъка с абонати (потребители на CISE). След това за всеки абонат доставчикът ще изпраща съобщение Push с предоставения преди това идентификационен номер за корелация - correlation ID.
- **Unsubscribe:** Абонаментът може да продължи за период от време, посочен в параметър от вида ServiceCapability (SubscriptionDuration) или докато потребител CISE не се абонира за използване на съобщението PullRequest (PullType = unsubscribe).



[Handwritten signature]

За прекратяване на абонамент участникът изпраща заявка Pull Request от типа Unsubscribe.

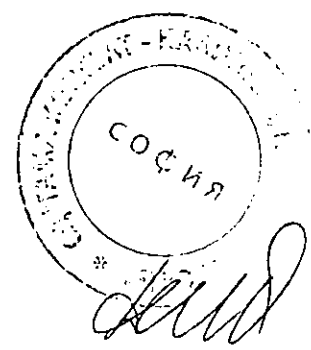
[Handwritten signature]

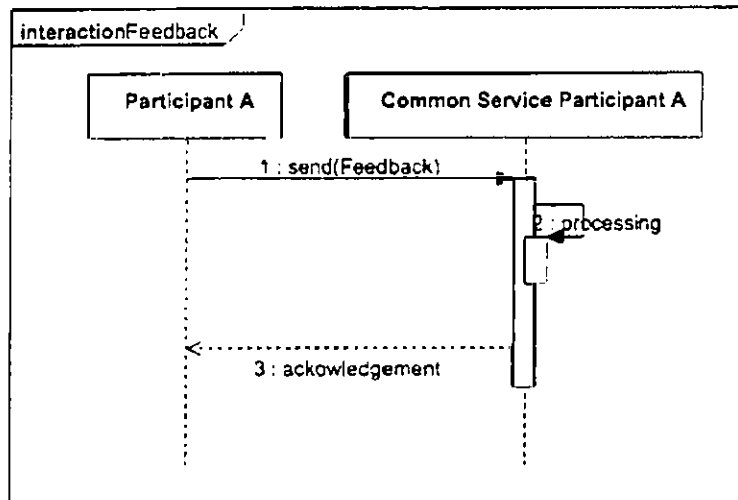


Feedback

Изпращането на обратна информация ще се задейства от участник, който желае да изпрати допълнителна информация относно предишно изпратено или получено съобщение.

[Handwritten signature]





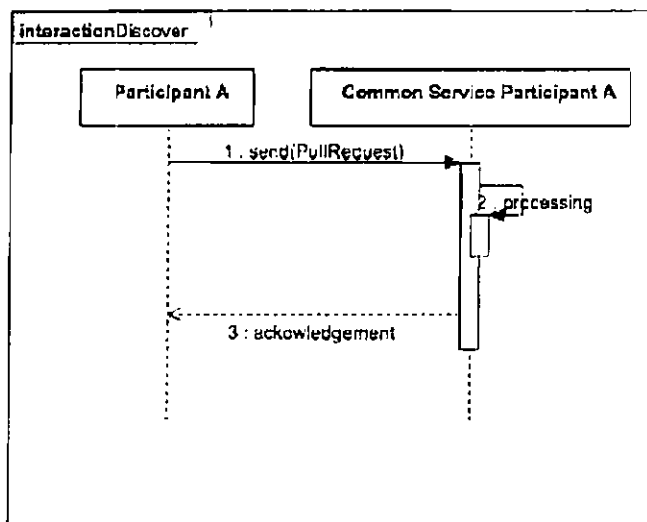
[Handwritten signature]

[Handwritten signature]

Discover

Операцията discover ще се използва от участник за намиране на други Legacy системи на други участници в мрежата EUCISE2020.

Тъй като тази операция се изпълнява само между адаптера и неговия Gateway, операцията е синхронна, в смисъл, че адаптерът ще изпрати съобщение Pull Request, с посочен тип на заявката Discover и ще получи в потвърждението евентуално откритите услуги.

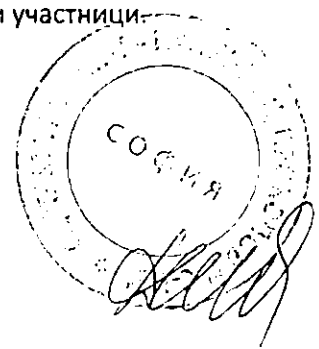


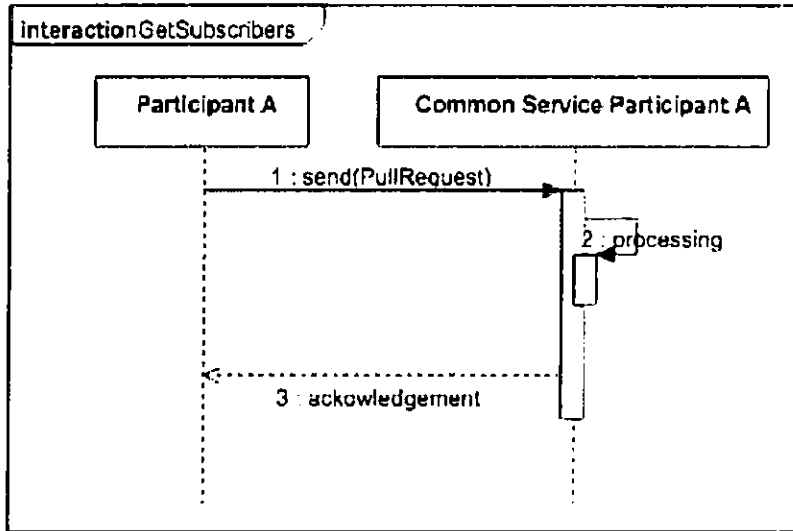
Get subscribers

Участник може да поиска списък с получатели, които са се абонирали за услугите му.

Адаптерът ще изпрати съобщение Pull Request, с посочен тип на заявката Get Subscribers и ще получи в потвърждението евентуалните абонираните за неговите услуги други участници.

[Handwritten signature]

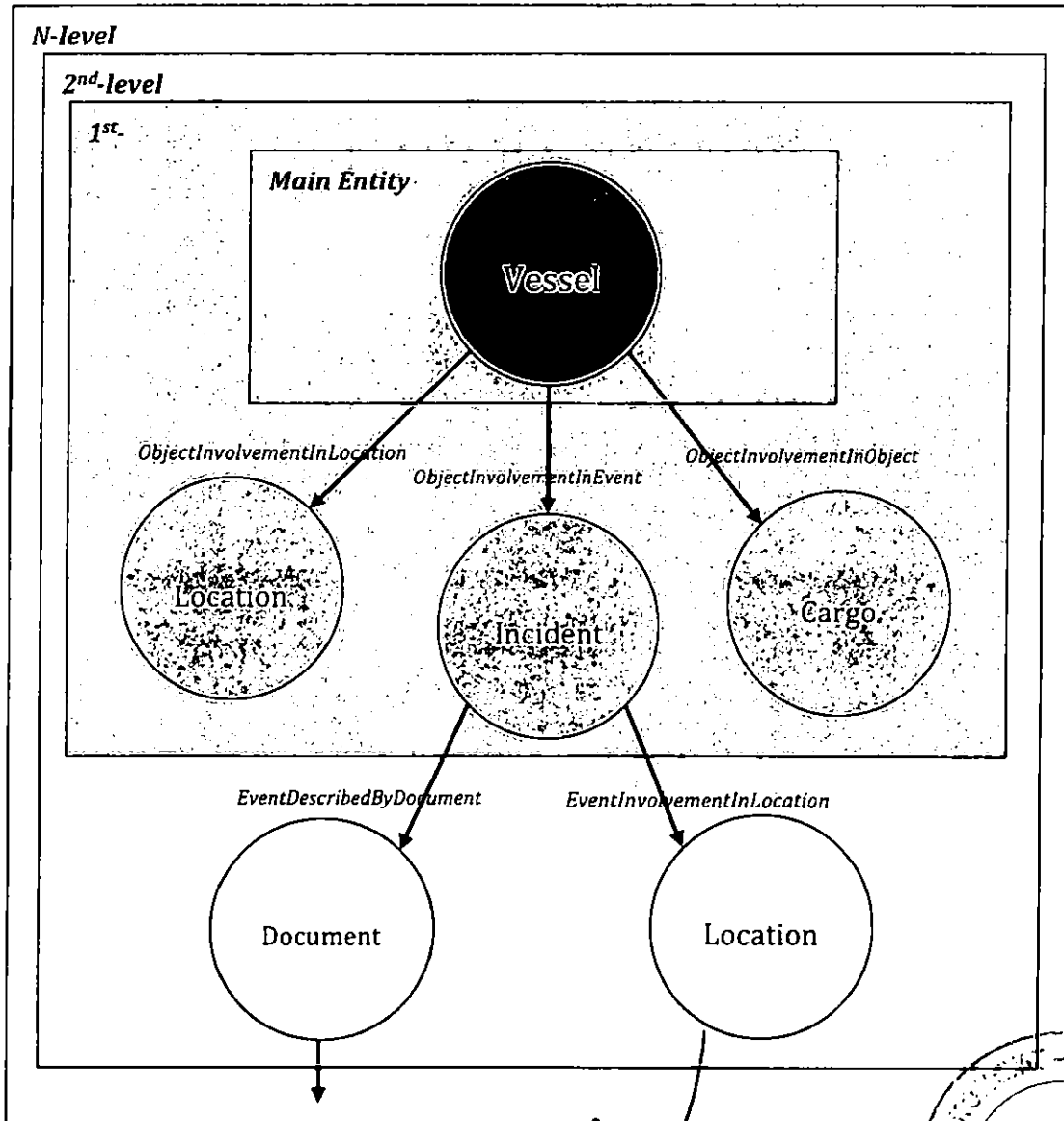




[Handwritten signature]

[Handwritten signature]

Следващата диаграма дава ясна картина на предвидения реляционен модел:



[Handwritten signature]

[Handwritten signature]



3.3. Възможности за оптимизация на вътрешносистемните процеси и информационни потоци

Националната система VTМIS съхранява информация, която може да бъде споделена с други Legacy системи чрез мрежата EUCISE2020, т.е. всяка една Legacy система от мрежата EUCISE2020 може да предоставя или получава информация от други Legacy системи чрез мрежата EUCISE2020.

Една Legacy система обменя информация с друга Legacy система чрез EUCISE2020 адаптер, предмет на разработка, който може да преобразува комуникационния протокол на VTМIS в EUCISE2020, т.е. адаптерът ще позволи на VTМIS да предостави и използва информационни услуги EUCISE2020 в съответствие с модела на услугата EUCISE2020.

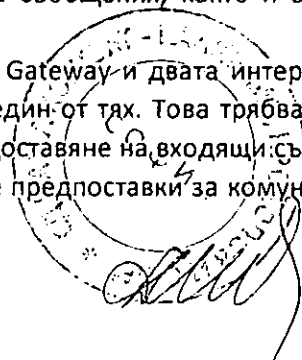
С разработването на EUCISE2020 адаптера ще се автоматизират и подобрят процедурите за предоставяне на данни между Националната Legacy система VTМIS и други Legacy системи в мрежата EUCISE2020 като се имплементира бизнес логиката на услугата CISEMessageService.

Ще се автоматизират процесите по:

- изпращане и получаване на съобщения с искания за предоставяне на информация между Националната Legacy система VTМIS, EUCISE2020 портала и други Legacy системи;
- предоставяне на данни от VTМIS и към потребителите ѝ.

3.4. Ограничения и предположения

- AC 1 За да е възможно потребителите на VTМIS да получават данни от EUCISE2020 Gateway/Node и/или други системи от мрежата EUCISE2020 е необходимо да бъдат разработени и интерфейси за получаване на данни. Разработката на тези интерфейси са извън обхвата на настоящата обществена поръчка.
- AC 2 Презентирането на данните, получени от други системи чрез потребителски интерфейс е извън обхвата на настоящата поръчка.
- AC 3 Настоящата разработка на Национален EUCISE2020 адаптер ще включва реализирането на възможност за връзка на адаптера към функциониращ конектор на VTМIS, който ще осигурява предаването на данните към самата VTМIS.
- AC 4 За целите на разработката на EUCISE2020 адаптера е необходимо Възложителят да осигури:
- Node описание;
 - Gateway описание;
 - Достъп до EUCISE2020 Gateway/Node;
 - Симулатор;
 - Модел на данните на VTМIS;
 - Спецификация на интерфейсите.
- AC 5 Общите услуги (Common services) се предлагат чрез един общ интерфейс за уеб услуга, който има една операция, поддържаща всеки модел за обмен на съобщения, както и всеки тип оперативна услуга.
- Уеб услугата се предлага в два протокола, SOAP и REST, а в Gateway и двата интерфейса са достъпни като входни точки, в адаптера може да се избере един от тях. Това трябва да бъде конфигурирано в Service Manager. За да позволи правилното доставяне на входящи съобщения към адаптера, Възложителят трябва да осигури необходимите предпоставки за комуникация с



разработчиците на EUCISE2020 Gateway/Node, към който трябва да бъде свързан разработвания адаптер, предмет на обществената поръчка.

4. Нефункционални характеристики на EUCISE2020 Адаптер

В таблицата по-долу са представени нефункционалните изисквания, на които ще отговаря Националният EUCISE2020 Адаптер.

№ по ред	Изискване
<p>NFREQ.1.</p>	<p>Услуги за сигурност на приложенията</p> <p>Националният адаптер ще осигурява най-малко следните услуги за сигурност на приложенията ще бъдат достъпни (във фонов режим или на интерфейса Човек-Машина на Националната система VTMISS):</p> <ul style="list-style-type: none"> • Услуга за валидиране на потребител; • Инфраструктура на Публичен ключ (PKI): <ul style="list-style-type: none"> ○ Всяко съобщение, рутирано в EUCISE2020 мрежата ще бъде подписано от тази услуга преди изпращането му; ○ Всяко съобщение, получено от Националния EUCISE2020 адаптер ще бъде верифицирано от тази услуга; <p>Услуги за проверка на правилата. След получаване на заявка за обект/атрибут/информация от Националната система, EUCISE2020 адаптера ще провери разрешението за изпълнение на заявката чрез тази услуга като изпрати следното искане за детайли: потребител от Националната система, оперативна цел, доставчик на услугата.</p>
<p>NFREQ.2.</p>	<p>Услуги за сътрудничество</p> <p>Най-малко следните услуги за сътрудничество ще бъдат достъпни (във фонов режим или на интерфейса Човек-Машина на Националната система VTMISS):</p> <ul style="list-style-type: none"> • Услуга за електронна поща Email (изпращане/получаване/получатели); • Услуга за трансфер на файлове - Услуга за управление на трансфера на файлове сред EUCISE2020 участниците (качване/сваляне/изтриване); • Услуга за споделени документи (списък/създаване/изтриване/преименуване); • Услуга за споделен календар – споделяне на календари и събития (списък/създаване/споделяне на събития); • Услуга за видео и аудио конференция (изпращане/получаване); • Услуга за мигновени съобщения (Instant Message) (изпращане/получаване/получатели); • Услуга за Бяла дъска (White Board).
<p>NFREQ.3.</p>	<p>Мрежови услуги и услуги за сигурна комуникация</p> <p>Следните мрежови услуги и услуги за сигурна комуникация ще бъдат достъпни (във фонов режим или на интерфейса Човек-Машина на Националната система):</p> <ul style="list-style-type: none"> • Търсене на услуги в регистъра на услугите в EUCISE2020 мрежата; • Откриване на информация – търсене на информация от определен тип в регистъра

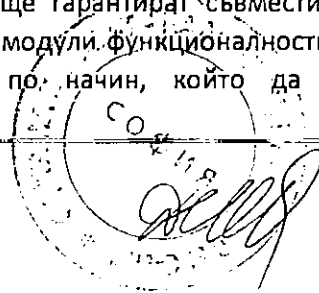
[Handwritten signatures]

[Official stamp and signature]

№ по ред	Изискване
	<p>на услугите;</p> <ul style="list-style-type: none"> • Комуникационни услуги – рутиране на съобщения/заявки в EUCISE2020 мрежата за призоваване на услуги (Общи, Специални или Иновативни) предлагани от други EUCISE2020 участници, използвайки Модела на данни и услуги EUCISE2020 за обмен на информация.
NFREQ.4.	<p>Внедряване на Общи услуги (Common services) на EUCISE2020 Gateway/Node</p> <ol style="list-style-type: none"> 1. Националния EUCISE2020 адаптер ще използва услугите на EUCISE2020 Node C; 2. Националната система VTMISS, чрез EUCISE2020 адаптора, ще бъде едновременно потребител и доставчик на данни, в зависимост от конкретния случай и избрания модел за обмяна на съобщения. 3. Начинът на реализация за обмен на съобщения е описан по-подробно в т. 3.2.2. <i>Обмен на съобщения</i> от настоящото Приложение 3.
NFREQ.5.	<p>Авторски права и изходен код</p> <ol style="list-style-type: none"> 1. Всички компютърни програми, които ще се разработват за реализиране на Адаптора, ще отговарят на критериите и изискванията за софтуер с отворен код; 2. Всички авторски и сродни права върху произведения, обект на закрила на Закона за авторското право и сродните му права, включително, но не само, компютърните програми, техният изходен програмен код, структурата и дизайнът на интерфейсите и базите данни, чието разработване е включено в предмета на поръчката, възникват за Възложителя в пълен обем без ограничения в използването, изменението и разпространението им и представляват произведения, създадени по поръчка на Възложителя съгласно чл. 42, ал. 1 от Закона за авторското право и сродните му права; 3. Всички доработки и/или новосъздадени софтуерни модули в рамките на настоящата поръчка, стават собственост на възложителя след окончателното им приемане. 4. Изпълнителят ще предаде пълния комплект програмни (сorc) кодове на разработените или модифицирани от него компоненти на единната информационно-комуникационна среда, придружени със съответната техническа документация. 5. Изпълнителят гарантира, че услугите, във вида на тяхното предоставяне, не накърняват права на интелектуалната собственост на трети страни. 6. Адаптерът ще включва и доставка и инсталиране на необходимите за неговото нормално функциониране лицензи и поддръжка;
NFREQ.6.	<p>Системна и приложна архитектура</p> <ol style="list-style-type: none"> 1. Адаптерът ще бъде реализиран като разпределена модулна информационна система. 2. Адаптерът трябва да бъде реализиран със стандартни технологии и да поддържа общоприети комуникационни стандарти, които ще гарантират съвместимост на Адаптера с бъдещи разработки. Съществуващите модули функционалности трябва да бъдат рефакторирани и/или надградени по начин, който да осигури изпълнението на настоящето изискване;

Handwritten signature

Handwritten signature



№ по ред	Изискване
	<ol style="list-style-type: none"> 3. Бизнес процесите и услугите ще бъдат проектирани колкото се може по-независимо с цел по-лесно надграждане, разширяване и обслужване. Адаптерът ще бъде максимално параметризиран и да позволява настройка и промяна на параметрите през служебен (администраторски) потребителски интерфейс; 4. При разработката, тестването и внедряването на Адаптера Изпълнителят ще прилага наложени се архитектурни (SOA, MVC или еквивалентни) модели и дизайн-шаблони, както и принципите на обектноориентирания подход за разработка на софтуерни приложения; 5. Адаптерът ще бъде реализиран със софтуерна архитектура, ориентирана към услуги - Service Oriented Architecture (SOA); 6. Взаимодействията между отделните модули в Адаптера и интеграциите с външни информационни системи ще се реализират и опишат под формата на уеб-услуги (Web Services), които ще са достъпни за ползване от други системи в държавната администрация, а за определени услуги – и за гражданите и бизнеса; 7. За всеки от отделните модули/функционалности на Адаптера ще се реализират и опишат приложни програмни интерфейси – Application Programming Interfaces (API). Приложните програмни интерфейси трябва да са достъпни и за интеграция на нови модули и други вътрешни или външни системи; 8. Приложните програмни интерфейси и информационните обекти задължително ще поддържат атрибут за версия; 9. Версията на програмните интерфейси, представени чрез уеб-услуги, ще версията по един или няколко от следните начини: <ol style="list-style-type: none"> a. Като част от URL-а b. Като GET параметър c. Като HTTP header (Асепт или друг) 10. При разработването на Адаптера ще се предвидят възможни промени, продиктувани от непрекъснато променящата се нормативна, бизнес и технологична среда. Адаптерът ще бъде разработен като гъвкава и лесно адаптивна ИС, като отчита законодателни, административни, структурни или организационни промени, водещи до промени в работните процеси; 11. Изпълнителят ще осигури механизми за реализиране на бъдещи промени в Адаптера без промяна на съществуващия програмен код. 12. Архитектурата на Адаптера и всички софтуерни компоненти (системни и приложни) ще бъдат така подбрани и/или разработени, че да осигуряват работоспособност и отказоустойчивост на Адаптера, както и недискриминационно инсталиране (без различни условия за инсталиране върху физическа и виртуална среда) и опериране в продуктивен режим, върху виртуална инфраструктура, съответно върху Държавния хибриден частен облак (ДХЧО); 13. Изпълнителят ще проектира, подготви, инсталира и конфигурира като минимум следните среди за Адаптера: тестова, стейджинг, продуктивна; 14. Адапторът ще бъде разгърнат върху съответните среди (тестова за вътрешни нужди, тестова за външни нужди, стейджинг и продуктивна); 15. Всеки обект в системата ще има уникален идентификатор.
NFREQ.7.	<p>Изграждане и поддръжка на множество среди</p> <p>Изпълнителят ще изгради и да поддържа минимум следните логически разделени среди на:</p>

[Handwritten signature]

[Circular stamp]
 29/1
 194

№ по ред	Изискване
	<p>собствена инфраструктура:</p> <ul style="list-style-type: none"> ○ Development; ○ Staging ○ Production.
<p>NFREQ.8.</p>	<p>Процес на разработка, тестване и разгръщане</p> <p>В случай че върху част от компонентите, нужни за компилация, има авторски права, те могат да бъдат или в отделно хранилище с подходящия за това лиценз или за тях ще бъде предоставен заместващ „mock up“ компонент, така че да не се нарушава компилацията на проекта.</p> <p>За всеки един разработван компонент Изпълнителят ще покрие следните изисквания за гарантиране на качеството на извършваната разработка и на крайния продукт:</p> <ul style="list-style-type: none"> ○ Документиране на Адаптора в изходния код, минимум на нивс процедура/функция/клас; ○ Покритие на минимум 50% от изходния код с функционални тестове; ○ Използване на continuous integration практики; ○ Използване на dependency management. <p>Във всеки един компонент на Адаптера, който се build-ва и подготвя за инсталация (deployment), ще присъстват следните реквизити:</p> <ul style="list-style-type: none"> ○ Дата и час на build; ○ Място/среда на build; ○ Потребител извършил/стартирал build процеса; ○ Идентификатор на ревизията от кодовото хранилище на компонента, срещу която се извършва build-ът.
<p>NFREQ.9.</p>	<p>Контрол на натоварването и защита от DoS/DDoS атаки</p> <ol style="list-style-type: none"> 1. Адаптерът ще поддържа на приложно ниво "Rate Limiting" и/или "Throttling" на заявки от един и същ клиентски адрес, както към страниците с уеб-съдържание, така и по отношение на заявките към приложните програмни интерфейси, достъпни публично или служебно като уеб-услуги (Web Services) и служебни интерфейси. 2. Адаптерът ще позволява конфигуриране от страна на администраторите на лимитите за отделни страници, уеб-услуги и ресурси, които се достъпват с отделен URL/URI. 3. Адаптерът ще поддържа възможност за конфигуриране на различни лимити за конкретни автентикирани потребители и ще предоставя възможност за генериране на справки и статистики за броя заявки по ресурси и услуги.
<p>NFREQ.10</p>	<p>Кохерентно кеширане на данни и заявки</p> <p>Като минимум разпределен кохерентен кеш ще се предвиди при:</p> <ul style="list-style-type: none"> ○ Извличане на информация от номенклатури и атомични данни за статус и актуално състояние на партии от регистри в информационните системи; ○ Извличане на информация от предефинирани периодични справки;

[Handwritten signature]

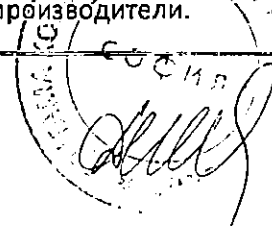
[Handwritten signature]

№ по ред	Изискване
	<ul style="list-style-type: none"> ○ Информация от лога на транзакциите при достъп с електронно-ИД до дадена услуга; ○ Информация за извършените плащания; ○ Други, които са идентифицирани на етап бизнес и системен анализ. <p>От кеша ще бъдат изключени прикачени файлове и големи по обем резултати от справки.</p>
NFREQ.11	<p>Бързодействие</p> <p>Адаптерът ще осигуряват висока производителност и минимално време за отговор на заявки.</p> <p>Ще бъдат създадени тестове за натоварване.</p>
NFREQ.12	<p>Използване на HTTP/2</p> <p>С оглед намаляване на служебния трафик, времената за отговор и натоварването на сървърите ще се използва HTTP/2 протокол при предоставяне на публични потребителски интерфейси с включени като минимум следните възможности:</p> <ul style="list-style-type: none"> • Включена header compression; • Използване на brotli алгоритъм за компресия; • Включен HTTP pipelining; • HTTP/2 Server push, приоритизиращ специфични компоненти, изграждащи страниците (CSS, JavaScript файлове и др.); • Публичните потребителски интерфейси ще поддържат адаптивен избор на TLS cipher suites според вида на процесорната архитектура на клиентското устройство - AES-GCM за x86 работни станции и преносими компютри (с налични AES-NI CPU разширения), и ChaCha20/Poly1305 за мобилни устройства (основно базирани на ARM процесори); • Ако клиентският браузър/клиент не поддържа HTTP/2, ще бъде предвиден fall-back механизъм към HTTP/1.1. Тази възможност ще може лесно да се реконфигурира в бъдеще и да отпадне, когато браузърите/клиентите, неподдържащи HTTP/2, станат незначителен процент.
NFREQ.13	<p>Подписване на документите</p> <ol style="list-style-type: none"> 1. При реализацията на електронно подписване с всички видове електронен подпис ще се подписва сигурен хеш-ключ, генериран на базата на образа/съдържанието, а не да се подписва цялото съдържание. 2. Минимално допустимият алгоритъм за хеширане, който ще се използва при електронно подписване, е SHA-256. В случаите, в които не се подписва уеб съдържание (например документи, файлове и др.), ще се реализира поточно хеширане, като се избягва зареждането на цялото съдържание в оперативната памет. 3. Адаптерът ще поддържа подписване с електронни подписи, издадени от Доставчици на доверителни услуги в ЕС, които отговарят на изискванията за унифициран профил на електронните подписи, съгласно подзаконовите правила към Регламент ЕС 910/2014, които влизат в сила и са задължителни от 1 януари 2017

№ по ред	Изискване
	<p>Г.;</p> <p>4. Ще бъдат анализирани техническите възможности за реализиране на подписване на електронни изявления и документи без използване на Java аplet и без да се изисква от потребителите да инсталират Java Runtime, като по този начин се осигури максимална съвместимост на процеса на подписване с всички съвременни браузъри.</p>
<p>NFREQ.14</p>	<p>Качество и сигурност на програмните продукти и приложенията</p> <p>Ще бъде спазено покритие на изходния код с тестове – над 60%, документиране на изходния код, използване на среда за непрекъсната интеграция (Continuous Integration), възможност за компилиране и пакетиране на продукта с една команда, възможност за инсталиране на нова версия на сървъра с една команда, система за управление на зависимостите (Dependency Management).</p>
<p>NFREQ.15</p>	<p>Информационна сигурност и интегритет на данните</p> <ol style="list-style-type: none"> 1. Ще бъде предвидена система за ежедневно създаване на резервни копия на данните, които да се съхраняват извън инфраструктурата на системата; 2. Всички уебстраници (вътрешни и публично достъпни в Интернет) ще бъдат достъпни единствено и само през протокол HTTPS. Криптирането ще се базира на сигурен сертификат с валидирана идентичност (Verified Identity), позволяващ задължително прилагане на TLS 1.2, който е издаден от удостоверяващ орган, разпознаван от най-често използваните браузъри (Microsoft Internet Explorer, Google Chrome, Mozilla Firefox); 3. Ще бъдат извършени тестове за сигурност на всички уебстраници, като минимум чрез автоматизираните средства на SSL Labs за изпитване на сървърна сигурност (https://www.ssllabs.com/ssltest/). Като временна мярка за съвместимост настройките на уебсървърите и Reverse Proxy сървърите трябва да бъдат балансирани така, че приложенията да позволяват използване и на клиентски браузъри, поддържащи по-стария протокол TLS 1.1. Това изключение от общите изисквания за информационна сигурност не се прилага за достъпа на служебни потребители от държавната администрация, услуги, които имат служебен достъп до ресурси на Адаптора; 4. При разгръщането на всички уеб услуги (Web Services) ще се използва единствено протокол HTTPS със задължително прилагане на минимум TLS 1.2; 5. Програмният код ще включва методи за автоматична санитизация на въвежданите данни и потребителски действия за защита от злонамерени атаки, като минимум SQL инжекции, XSS атаки и други познати методи за атаки, и ще отговаря, където е необходимо, на Наредбата за оперативна съвместимост и информационна сигурност.
<p>NFREQ.16</p>	<p>Използваемост</p> <p>Функционалностите на потребителския интерфейс на Адаптера ще бъдат независими от използваните от потребителите интернет браузъри и устройства, при условие че последните са версии в период на поддръжка от съответните производители.</p>

Ваня

[Handwritten signature]

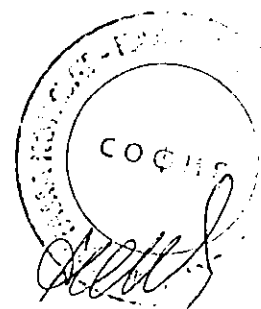


№ по ред	Изискване
NFREQ.17	<p data-bbox="335 280 933 309">Дизайн на бази данни и взаимодействие с тях</p> <p data-bbox="335 336 1516 403">При използване на база данни ще бъдат следвани добрите практики за дизайн и взаимодействие с базата данни, в т.ч.:</p> <ul data-bbox="422 436 1516 1388" style="list-style-type: none"> • дизайнът на схемата на базата данни (ако има такава) ще бъде с максимално ниво на нормализация, освен ако това не би навредило сериозно на производителността; • базата данни ще може да оперира в клъстер; в определени случаи ще бъде използван т.нар. sharding; • имената на таблиците и колоните ще следват унифицирана конвенция; • ще бъдат създадени индекси по определени колони, така че да се оптимизират най-често използваните заявки; създаването на индекс ще е мотивирано и подкрепено със замервания; • връзките между таблици ще са дефинирани чрез foreign key; • периодично ще бъде правен анализ на заявките, включително чрез EXPLAIN (при SQL бази данни), и да бъдат предприети мерки за оптимизиране на бавните такива; • задължително ще се използват транзакции, като нивото на изолация трябва да бъде мотивирано в предадената документация; • при операции върху много записи (batch) ще се избягват дългопродължаващи транзакции; • заявките ще бъдат ограничени в броя записи, които връщат; • при използване на ORM или на друг слой на абстракция между приложението и базата данни, ще се минимизира броят на излишните заявки (т.нар. n+1 selects проблем); • при използване на нерелациона база данни ще се използват по-бързи и компактни протоколи за комуникация, ако такива са достъпни.

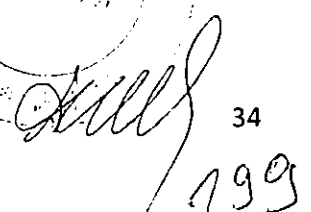
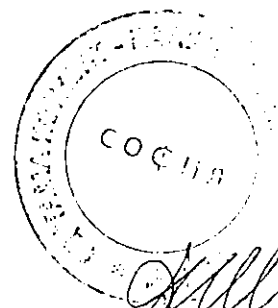
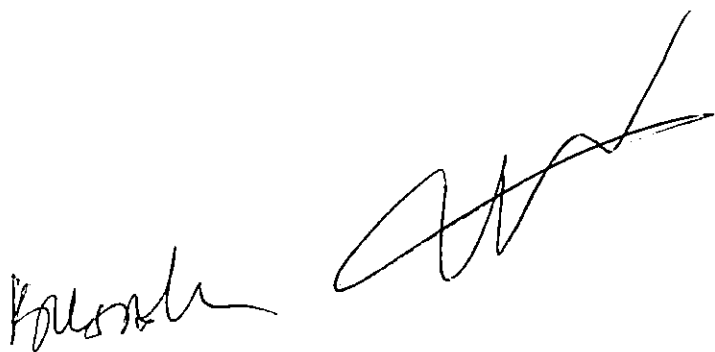
5. Мотивация за предложеното технологично решение

- Технологичното решение отговаря на всички функционални изисквания;
- Технологичното решение отговаря на всички допълнителни/нефункционални изисквания;
- Технологичното решение е съобразено с концепцията и архитектурата на EUCISE2020 и отговаря на всички изисквания, произтичащи от това;
- Технологичното решение отговаря на нормативните изисквания;
- Технологичното решение е базирано на световно-утвърдени и популярни технологии;
- Използва се утвърден базов софтуер с отворен код, което предполага ценова ефективност при експлоатацията;
- Предложена е гъвкава архитектура, която позволява:
 - Надграждане на адаптера;
 - Лесно интегриране на други системи към адаптера;
 - Разширяване на модела на обменните данни;
 - Добавяне на нови видове трансформация на данните;

Кьосал



- Възможност за изграждане на класифициран адаптер на базата на настоящата разработка;
- Решението предоставя възможност за скалиране при повишаване на натоварването и осигуряване на устойчивост.



34
1999

- Ще изпълнява възложените му задачи за осигуряване на качеството;
- Ще сигнализира за дефекти и грешки, намерени при изпълнение на собствените си задачи, включително когато не са пряко свързани с текущата задача.

Представители на Възложителя ще имат възможност да вземат участие в процесите за осигуряване на качеството.

Екипът на Възложителя ще:

- Участва в обучения;
- Участва в планиране и провеждане на тестове;
- Координира работа с разработчика на EUCISE2020 Gateway/Node;
- Координира работата с организацията, отговорна за VTMISS.

3.4. Процедури за контрол

Съществуват няколко метода, които ще бъдат използвани при осигуряването на проекта и материалите по него в съответствие с подходящите стандарти за качество:

- Вътрешни прегледи на проект (review) - Това са работни сесии на екипа по проекта, в които екипът преглежда всички материали по конкретна фаза преди да определи методологията за официален преглед. Прегледът се извършва от ръководителя на проекта и ръководителя на разработката.
- Пробно изпълнение - Това са сесии на групово работа, в които тестовият екип проверява материалите като използва предварително определени сценарии, презентации, въпроси и отговори, и дори мисловни сесии (brainstorming), ако е подходящо. Извършва се от експерта по контрол на качество и внедряване.
- Инспекции - Това е съвместен преглед на материали от представители на Възложителя и Изпълнителя (изпълняващия екип) за целите на проверка и приемане.

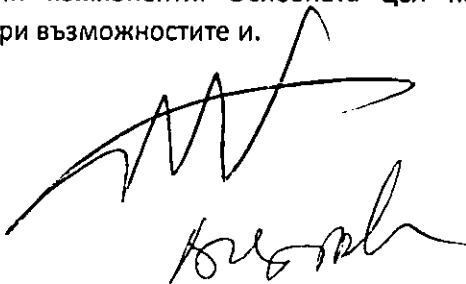
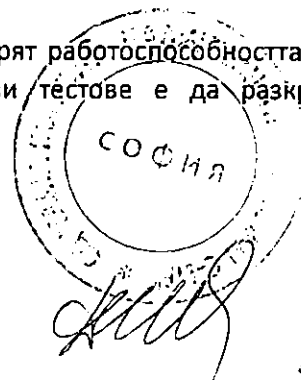
3.5. Стратегия за тестване

Тестването е етап на изпълнение, по време на който разработения продукт се проверява за съответствие с техническата и функционална спецификация на проекта. Тестването се съотнася с процеса по контрол на качеството. Тестването се изпълнява паралелно или след процеса по разработка. Задължителен вход за процеса по тестване са резултатите от процеса по разработка. Тестването е тясно свързано с процеса по осигуряване на качеството и се изпълнява едновременно/интегрирано с процесите по анализ, планиране, проектиране, разработване и внедряване. Това осигурява превантивното откриване и ранно отстраняване на грешки.

Тестването ще бъде проведено в три фази:

- Фаза1: Първичен тест – включва проверка на разработените модули и функционалности, изпълнени върху тестова подсистема (среда);
- Фаза 2: Вторичен тест – включва проверка на разработените модули и функционалности, изпълнени върху продукционна и резервна подсистема, след успешно приключване на първичния тест ;
- Фаза 3: Тестове за натоварване и надеждност – включва тестове на Системата, който гарантират кохерентната работа на съвкупността от компоненти като тестове за натоварване, тестове за непрекъснатост на работата, тестове за възстановяване на предишна стабилна работеща версия на Системата;

Стратегията за тестване се състои от серия различни тестове, които ще проверят работоспособността на системата, включвайки всички нейни компоненти. Основната цел на тези тестове е да разкрие ограниченията на системата и да измери възможностите и.

Подготовката на учебните материали и провеждането на обучението също са предмет на процедури и стандарти за качество по проекта.

3.3. Роли и отговорности

Ръководителят на проекта ще бъде отговорен за общото качество на проекта от страна на Изпълнителя.

Ръководител на проекта ще бъде отговорен за:

- Определяне на стратегия за осигуряване на качеството;
- Установяване на процедури за осигуряване на качеството;
- Определяне на задачи и отговорности на членовете на екипа свързани с изпълнението на процедурите по осигуряване на качеството;
- Наблюдение на процедурите по осигуряване на качеството и извършване на коригиращи действия, при необходимост.



Експерт по контрол на качеството:

- Участва в определянето на стратегия и процедури за осигуряване на качеството;
- Организира изпълнението на процедурите и стратегията за осигуряване на качеството;
- Разработва подробен тестов план.



Всеки ръководител на технически екип:

- Ще бъде отговорен за спазването на установените в проекта стандарти и процедури за осигуряване на качеството в екипа, който ръководи;
- Ще сигнализира на Ръководителя на проекта за потенциални слабости в качеството на разработвания в екипа и другите екипи продукт и ще дава мотивирани предложения за подобряване на стандартите за качество.

Всеки член на екипа:

- Ще бъде отговорен за прилагането на стандартите за качество на проекта при изпълнението на своята работа;
- Ще изпълнява възложените му задачи за осигуряване на качеството;
- Ще сигнализира за дефекти и грешки, намерени при изпълнение на собствените си задачи, включително когато не са пряко свързани с текущата задача.

Експерт по контрол на качеството:

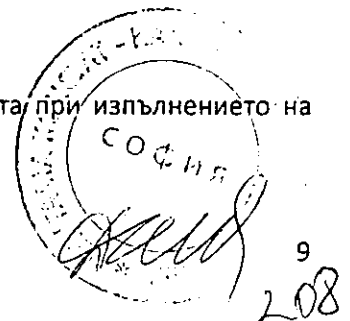
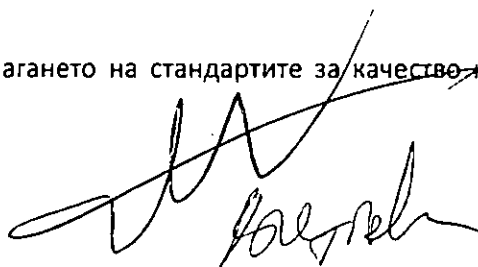
- Участва в определянето на стратегия и процедури за осигуряване на качеството;
- Организира изпълнението на процедурите и стратегията за осигуряване на качеството;
- Разработва подробен тестов план.

Всеки ръководител на технически екип:

- Ще бъде отговорен за спазването на установените в проекта стандарти и процедури за осигуряване на качеството в екипа, който ръководи;
- Ще сигнализира на Ръководителя на проекта за потенциални слабости в качеството на разработвания в екипа и другите екипи продукт и ще дава мотивирани предложения за подобряване на стандартите за качество.

Всеки член на екипа:

- Ще бъде отговорен за прилагането на стандартите за качество на проекта при изпълнението на своята работа;



Осигуряването на качеството включва управление на качеството на всички базови продукти, компоненти на системата и процеси свързани с изграждането, включително тези на управление и мониторинг и контрол.

Критериите за определяне на качеството се определя от заинтересованите лица и основно потребителите на системата. Те са свързани и с разходите за прилагане на необходимите изисквания, които зависят както от наличните бюджети, потребителите, така и от самия продукт, тъй като определено ниво на качеството е свойствено за точно определени категории продукти.

Качеството на продукта може да бъде разглеждано от два аспекта:

- 1) Какви характеристики притежава;
- 2) До каква степен тези характеристики са реализирани.

Притежаваните характеристики се задават при изготвянето на функционалната спецификация на проекта. Притежаваните характеристики са ограничени от обхвата на договора, от разполагаемото време и от наличните ресурси. Предмет на тестването е да установи до каква степен тези характеристики са осъществени. Поради това то се изпълнява след завършване на разработката. При процес на бърза разработка, то се изпълнява паралелно с разработката, в края на всяка итерация, но е възможно да се окаже, че изисква значително повече ресурси спрямо посветеното тестване, в края на проекта. При този похват обаче се спестява време, тъй като голяма част от дейността се изпълнява паралелно на разработката. Частта от дейността, която не може да се изпълни паралелно е осъществяването на приемните тестове. Приемните тестове се осъществяват на базата на предварително изготвени и съгласувани с Възложителя приемни сценарии. Те имат за цел да покажат, че системата може да бъде въведена в експлоатация.

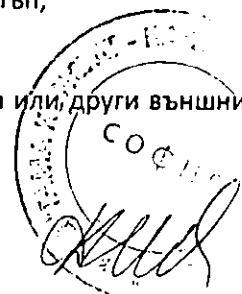
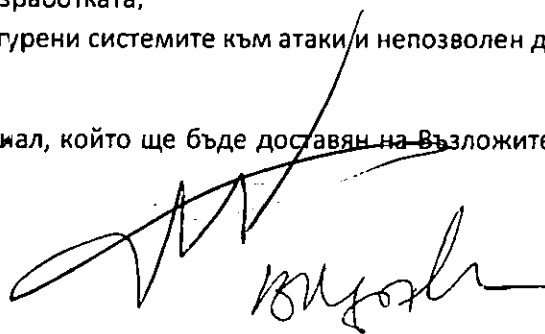
Като част от проектния план ще бъде разработен План за осигуряване на качеството. След като Планът за осигуряване на качеството бъде разработен и приет от Възложителя, той ще бъде водещият документ за организиране на процесите по управление на качеството в проекта.

Всеки етап в жизнения цикъл на проекта ще се отрази като артефакт, даващ основа за следващия етап. Чрез контрола на качеството тези артефакти ще бъдат проверявани преди да са използвани в следваща фаза, което ще гарантира верността ѝ. Те също ще бъдат верифицирани съобразно техническата спецификация.

Системата ще бъде оценявана за:

- Коректност/Надеждност – степен, в която системите отговарят на документираните изисквания и спецификации;
- Ефикасност – колко е ефективно използването на софтуера съобразно използваните системни ресурси и каква е скоростта на изпълнение;
- Използваемост – до колко е лесно за потребителя да усвои работата със системите и да възприема резултатите то нея;
- Възможност за поддръжка – до колко е лесно внасянето на промени в системите и документацията и с цел актуализиране и отстраняване на грешки;
- Възможност за тестване – до колко е лесно и възможно тестването на системите в развойна среда;
- Преносимост – до колко е лесно пренасянето на системите върху друга хардуерна или софтуерна среда;
- Възможност за повторна употреба – до колко е лесно и възможно използването на софтуера или компоненти от него при разработка на бъдещи системи и до колко е било възможно използването на готови компоненти при разработката;
- Цялостност – до колко са осигурени системите към атаки и непозволен достъп;
- и др.

Освен това, всеки документ и материал, който ще бъде доставян на Възложителя или други външни лица подлежи на контрол на качеството.



Изпълнителят провежда вътрешни одити на системата за управление в съответствие с изискванията на ISO 9001:2015, ISO 14001:2015, OHSAS 18001:2007, AQAP 2110:2006 и ISO 27001:2013.

Целите на вътрешните одити са:

- да се увери ръководството на фирмата, че системата за управление функционира и отговаря на изискванията на ISO 9001:2015, ISO 14001:2015, OHSAS 18001:2007, AQAP 2110:2006 и ISO 27001:2013;
- да се определи дали системата за управление функционира и се поддържа ефикасно;
- да се определят коригиращи действия, когато системата за управление е неефективна или когато има несъответствия.

Вътрешните одити се планират и организират от представителя на ръководството.

Вътрешните одити се провеждат от одиторски екип от специално определени и квалифицирани за целта кадри, които не са свързани пряко с одитираното звено или процес. Констатираните несъответствия и предвидените коригиращи действия за отстраняването им се документират. Върху коригиращите действия се упражнява контрол за изпълнение и ефективност.

Отчети за вътрешните одити и резултатите от тях се представят на ръководството.

Вътрешните одити се провеждат по годишен план и планове за конкретни одити.

В обосновани случаи се провеждат извънредни одити на системата за управление.

- **ISO: 27001** – този стандарт се отнася към управлението на сигурността на информацията и има за цел да осигури целостта и поверителността на информацията, непрекъсваемост на работата и повишено внимание към всеки детайл от дейността на фирмата.

Основни цели по осигуряване на сигурността на информацията на на Изпълнителя са:

- определяне на рисковете за сигурността на информацията, собствениците на тези рискове и защитата им от вътрешни, външни, предумишлени и случайни заплахи;
- определяне на критериите за оценяване на рисковете, както и нивото на приемлив риск, спрямо които се оценява вероятността от възникване на заплахи и тежестта на тяхното въздействие за активите на фирмата;

Методологията за осигуряване на качеството, която експертите по осигуряване на качеството при Изпълнителя използват са базирани на световно признатите практики на Международния борд за софтуер, тестване и качество (ISTQB).

3.2. Стратегия за осигуряване на качеството

Стратегията за осигуряване на качеството определя дейностите, които ще осигурят, че контролът на качеството се прилага ефективно и механизмите за контрол на качеството са адекватно планирани, графициите са спазвани, преглеждани и актуализирани.

Осигуряване на качеството се отнася до систематично измерване, сравнение със стандарти, мониторинг на процесите и свързаните с тях дейности за осъществяване на обратна връзка, която спомага за предотвратяването на грешки. Това контрастира с процеса по контрол на качеството, който се фокусира върху изхода на процесите.

Осигуряването на качеството се основава на два базови принципа:

- 1) продуктът трябва да може да изпълнява предназначението си;
- 2) откритите грешки трябва да се отстраняват възможно най-бързо.



Качеството е характеристика, която се изразява в способността за задоволяване установени или скрити нужди. Управлението на качеството включва изисквания и процеси, които осигуряват успешното изпълнение на проекта. Всички дейности по управлението на проекта и крайният резултат се включват в мероприятията по осигуряване на качеството. Такива са политиката по качество, обстоятелствата, отговорностите и тяхното прилагане чрез планиране на качеството (quality planning QP), контрол по качеството (quality control QC) и осигуряване на качеството (quality assurance QA).

- **Планиране (Quality Planning)** – определя кои стандарти по качеството са приложими за проекта и как те да бъдат постигнати;
- **Осигуряване на качеството (Quality Assurance)** – цялостна оценка на изпълнението на задачите по проекта, извършвана периодично и даваща сигурност, че са постигнати съответните стандарти за качество;
- **Контрол на качеството (Quality Control)** – наблюдение на специфични резултати в хода на изпълнение на проекта и определяне дали те съответстват на съответните стандарти за качество, както и намиране на пътища за елиминиране на причините за незадоволително изпълнение.

Коригиращи действия ще бъдат прилагани в съответствие със същността и степента на отклоненията от стандартите. Къде те да бъдат насочени и какви мерки ще се предприемат, ще бъде предмет на процедурата по контрол на промените. Изпълнителят е сертифициран по системата за управление на качеството ISO: 20 000, ISO: 9001 и ISO: 27001 и ще прилага тези стандарти при работата по проекта.

Следва кратко описание на стандартите за осигуряване на качеството:

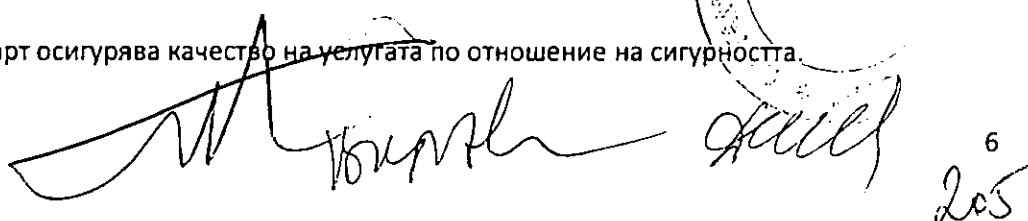
➤ **ISO: 20 000-1/2011** – този стандарт се отнася към управлението на услуги. Изпълнителят има действаща Система за управление на услугите, в съответствие с този стандарт. По този начин Изпълнителят гарантира способността за управление на услуги в контекста на бизнеса на Дружеството и изискванията на клиентите:

- ◆ Да разработи политика, цели и планове за управление на услуги;
- ◆ Да оповести важността за изпълнение на целите по управление на услугата;
- ◆ Да осигури постоянното подобряване на системата за управление на услуги;
- ◆ Да се определят изискванията и удовлетвореността на клиентите;
- ◆ Да се определи член от ръководството (ресорен управител), който да отговаря за координирането и управлението на всички услуги;
- ◆ Да определи и осигури ресурси за планирането, внедряването, наблюдението, прегледа и подобряването на предоставянето на услуги и тяхното управление;
- ◆ Да определи изискванията за образование, квалификация и опит на персонала и възможностите за тяхното подобряване и развитие;
- ◆ Да анализира рисковете, свързани с дейността на Изпълнителя, както и рисковете за изпълнение на услугите;
- ◆ Да оценява системата за управление на услуги, чрез прегледи от ръководството, за да се гарантира че тя е подходяща, адекватна и ефикасна.

Тя е изградена и функционира в структурните звена, ангажирани с управлението на услугите, включени в обхвата на системата:

- Сервизна дирекция;
- Бизнес операции;
- Софтуерна къща.

➤ **ISO: 9001** – този стандарт осигурява качество на услугата по отношение на сигурността.



Handwritten signature and circular stamp. The stamp contains the text 'СОБИ' and '2011'. Below the signature, the number '6' and '205' are written.

- Отстраняване на дефекти в Адаптера;
- Извършване на диагностика на регистриран проблем с цел осигуряване на правилното функциониране на системите и модулите;
- Възстановяването на системата и данните при евентуален срив на системата, както и коригирането им вследствие на грешки в системата;
- Актуализация на документацията на системата в резултат на извършени действия в рамките на поддръжката;
- Консултация за разрешаване на проблеми по предложената конфигурация на средата (операционна система, база данни, middleware, хардуер и мрежи), използвана от приложението, включително промени в конфигурацията на софтуерната инфраструктура на мястото на инсталация;
- По време на изпълнение на дейността по осигуряване на работоспособността на Адаптера ще осигурим използването на уеб базирана система за регистриране и проследяване на инцидентите, която да се ползва съвместно от представители на Изпълнителя и Възложителя.

Предлаганата услуга НЕ включва разработка на нови функционалности.

- Разработка на нова функционалност;
- Помощ на крайните потребители на системата;

2.2. Срок

Изпълнителят ще осигури поддръжка по работоспособността на Адаптера и осигурената ИКТ инфраструктура по време и за целите на демонстрационното тестване.

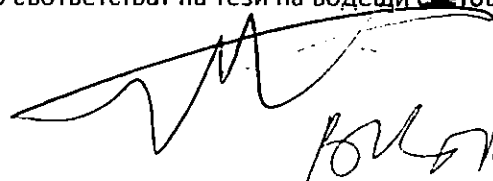


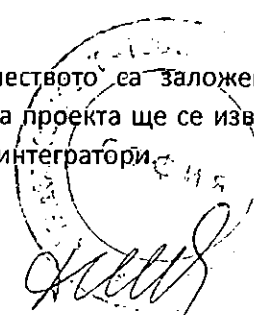
3. Начин на изпълнение на интеграционни тестове и подход за управление на качеството

Следва предложението ни за осигуряване на качество и провеждане на тестовете за интеграция на EUCISE2020 Адаптера.

Ключови елементи от предложението	Препратка
Мерки за осигуряване на качеството	Т. 3.1 - „Стандарти и добри практики, които ще бъдат използвани“ Т. 3.2 – „Стратегия за осигуряване на качеството“
Отговорности на участващите в процеса	Т. 3.3 – „Роли и отговорности“
Действия и процедури, които ще бъдат извършени при осигуряване на качеството и провеждане на тестовете за интеграция на EUCISE2020 Адаптера, така че да се гарантира ефективното изпълнение на услугата	Т. 3.4 – „Процедури за контрол“ Т. 3.5.1 – „Видове тестове“ Т. 3.5.4 – „Проследяване на дефектите“ Т. 3.5.2 – „Разработване и използване на успешни тестови сценарии“ Т. 3.6 – „Azure виртуални машини EUCISE2020“ Т. 4.3.3 – „Управление на проблеми“
Обосновка за приложимостта и полезността на предложеното решение при изпълнението на поръчката	Т. 5 – „Обосновка за приложимостта и полезността на предложеното решение“

3.1. Стандарти и добри практики, които ще бъдат използвани

Използването на основни принципи от стандартите за управление на качеството са заложили при изпълнението на настоящата поръчка и гарантират, че дейностите в рамките на проекта ще се извършват по критерии, които напълно съответстват на тези на водещи световни системни интегратори.

1. Въведение

В настоящия документ сме представили информация за начина на осигуряване на работоспособност на EUCISE2020 Адаптера за периода на тестовата обмяна на данни. Описани са ефективни процедури, конкретни мерки и конкретни дейности, които ще прилага за осигуряване работоспособността на Адаптера за периода на договора, които гарантират степента на постигане на целите при съпоставяне на действителните и очакваните резултати от изпълнението. Дефинирани и категоризирани са инцидентите, които могат да се появяват в процеса на тестова обмяна на данни. Посочени са отговорностите на участващите в процеса, в т.ч. и тези на Възложителя.

В настоящото Приложение 4 представяме визията ни за предоставяне на услуга по поддръжка, включваща тестване на интеграцията и отстраняване на евентуални недостатъци, открити при тестовете на Националния EUCISE2020 Адаптер по време на изпълнение на **Дейност 2 Тестване и съпровождане на адаптера** от проекта.

Описани са организацията, дейностите и процедурите обуславящи начина на изпълнение на услугата, които ще бъдат прилагани при осигуряване на работоспособността на EUCISE2020 Адаптера за периода на тестовата обмяна на данни в мрежата на EUCISE2020.

Към настоящото предложение са включени описания на действията и начините на изпълнение на задачите, които планираме да приложим, с което ще се осигури напълно покриване на очакваните резултати, като по този начин ще се постигне висока ефективност при провеждане на фазата на демонстрационно тестване.

2. Обхват и срок на предоставяне на услугата

2.1. Обхват

В обхвата на услугата ще се включват всички необходими дейности за извършване на тестове и поддръжка на работоспособността на Адаптера. При необходимост, по време на тестовия период, ще бъдат осъществявани дейности по осигуряване на експлоатационната годност на системата и нейното ефективно използване от Възложителя в случай, че настъпят явни отклонения от нормалните експлоатационни характеристики, заложи в спецификациите.

При изменение на софтуерен компонент, част от EUCISE2020 Адаптера, Изпълнителят ще прилага изискванията за тестване и проверка – подходът за извършване на тестовете за интеграция и осигуряване на качеството на разработения продукт е описан в *т. 3 Начин на изпълнение на интеграционни тестове и подход за управление на качеството* от настоящото приложение.

За приемане на нова версия на софтуер, Изпълнителят ще е изпълнил:

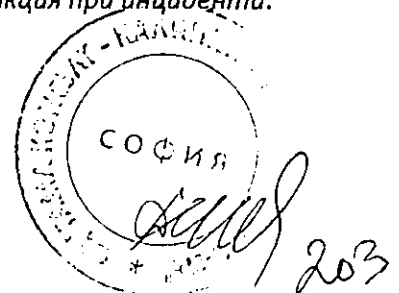
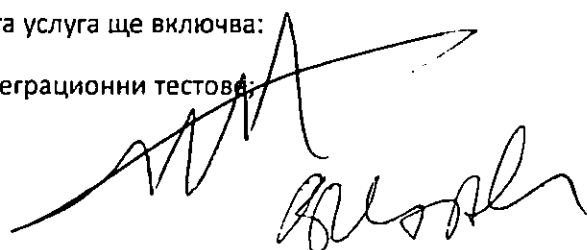
- Успешно тестване и проверка в тестова и/или симулаторна среда;
- Успешно прилагане на новата версия в засегнатата от дефекта среда;
- Успешно обновяване на прилежащата документация.

Изпълнителят ще проведе всички тестове, които ще гарантират безпроблемното функциониране на Адаптера.

Услугите по изпълнение на дейностите по отстраняване на евентуални недостатъци, открити при тестовете и осигуряването на поддръжка на работоспособността на Адаптера ще са налични съгласно описаните в *т. 4.1 Категоризация на инцидентите. Приоритети, достъпност и време за реакция при инциденти.*

Обхватът на предоставяната услуга ще включва:

- Извършване на интеграционни тестове;

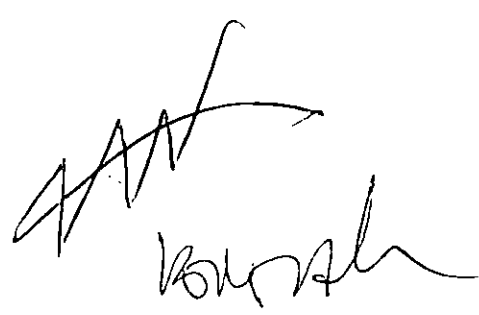
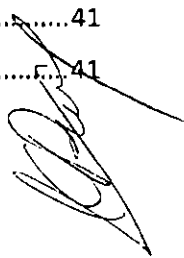


4.5. Система за автоматизиран мониторинг Zabbix 3.2 38

4.6. Предпоставки за успешна поддръжка от страна на Възложителя 41

4.7. Очаквани резултати от изпълнението на дейностите по отстраняване на евентуални
недостатъци, открити при тестовете 41

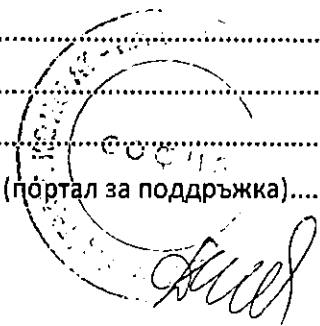
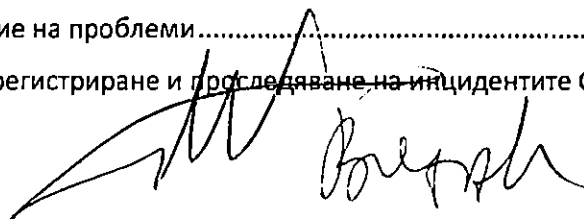
5. Обосновка за приложимостта и полезността на предложеното решение 41



3
202

Съдържание

1. Въведение.....	4
2. Обхват и срок на предоставяне на услугата	4
2.1. Обхват	4
2.2. Срок.....	5
3. Начин на изпълнение на интеграционни тестове и подход за управление на качеството.....	5
3.1. Стандарти и добри практики, които ще бъдат използвани	5
3.2. Стратегия за осигуряване на качеството.....	7
3.3. Роли и отговорности	9
3.4. Процедури за контрол.....	10
3.5. Стратегия за тестване.....	10
3.5.1. Видове тестове	12
3.5.2. Входни и изходни критерии.....	13
3.5.2.1. Критерии за започване на тестовия процес	13
3.5.2.2. Критерии за преустановяване на тестването	13
3.5.2.3. Критерии за подновяване на тестването	13
3.5.2.4. Критерии за спиране на тестовия процес.....	13
3.5.2.5. Критерии за приемане.....	14
3.5.3. Разработване и използване на успешни тестови сценарии.....	14
3.5.4. Проследяване на дефектите	15
3.6. Azure виртуални машини EUCISE2020.....	16
3.7. Очаквани резултати от изпълнението на интеграционните тестове.....	20
4. Начин на изпълнение на дейностите по отстраняване на евентуални недостатъци, открити при тестовите. Подход за управление на услугата по поддръжка	20
4.1. Категоризация на инцидентите. Приоритети, достъпност и време за реакция при инциденти	21
4.2. Организация при предоставяне на услугата.....	22
4.2.1. Нива на поддръжка.....	22
4.2.2. Отговорности и задължения на екипа за поддръжка.....	23
4.2.3. Канали за комуникация	24
4.2.4. Работно време.....	25
4.3. Методика за обработка на инциденти	25
4.3.1. Видове дейности за отстраняване на инциденти	25
4.3.2. Процедура за обработка на инциденти.....	25
4.3.3. Управление на проблеми.....	29
4.4. Система за регистриране и проследяване на инцидентите СПРИ (портал за поддръжка).....	35

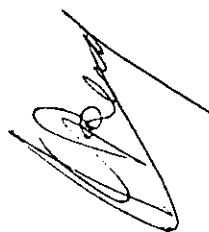


към


ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

за участие в открита процедура за възлагане
на обществена поръчка с предмет:

**„Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в
съответствие с инженеринговите услуги, поддържащи прилагането на
EUCISE2020 мрежата и съпровождащи услуги по време на
демонстрационната фаза на обмен на данни“**



Предложение за поддръжка
работоспособността на Адаптера за
периода на тестова обмяна на данни



2018 г.



200

Тестването започва след етапа на разработка на тестваема част от завършени функционални изисквания за съответния и компонент и ще се планира като дейност, която е част от общото изпълнение на проекта и преди приемането на Системата. Процесът по тестване ще бъде итеративен и интегриран с цикъла за софтуерна разработка. Тестовите ще бъдат планирани на базата на одобрените изисквания.

Ще бъде прилагана методологията, която разчита на повтарящи се много къси цикли за разработка (итерации). В итеративния подход се приоритизира реализацията на базата на риска, нуждите на крайните потребители и важността за архитектурата, след което се преминава към по-малките компоненти на системата. Този подход ще се използва през целия последващ цикъл на тестване и внедряване. По този начин тестваният екип получава модули за тестване постепенно и има достатъчно време за тестване преди приключването на проекта. Всяка версия дава възможност за оценка на качеството на крайния продукт. Това също така дава възможност на екипа по разработка да извършва малки промени, когато разходите за прилагането им са ниски. Това намалява риска и увеличава качеството.

Планирането и подготовката на тестовите се извършва преди разработката, веднага след поставянето на изискванията. Още във фазата на проектиране за всички функционални и някои нефункционални изисквания се определят тестовите сценарии, с които те трябва да бъдат покрити. Изготвя се дизайн на това как ще бъде тестван продукта и неговите компоненти. Тестовите се разработват преди да бъдат разработени самите компоненти. При планирането на тестовите се прилага принципа на Парето известен още като „правило 80/20“. Тестовите трябва да се приоритизират и фокуса да бъде върху най-приоритетните причини, които пораждаат множеството проблеми. Тестовите на функционалните и нефункционалните изисквания се групират в тестови сценарии. Изпълнението на тестовите сценарии може да бъде автоматизирано или неавтоматизирано. Решение какъв тест да бъде извършен се взема в етапа на планиране на тестването, като решението е базирано на технологичните особености на конкретната функционалност. В хода на самата разработка процента на реализираните и преминалите тестове постепенно нараства. В края на итерацията тестовите за итерацията трябва да са проведени и да са преминали успешно. В края на разработката абсолютно всички тестове трябва да са проведени и да са преминали успешно.

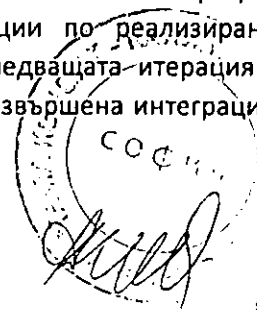
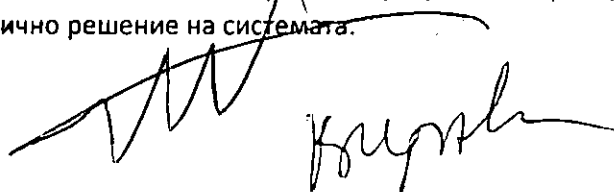
Възприети са различни видове тестове на софтуерните продукти. В зависимост от конкретния случай се използват различни техники за тестване. Техниките на тестване са в съответствие с избраните технологии, използвани за разработка. Изборът на това кой конкретен вид тест да бъде използван зависи от конкретния казус (модул, функционалност и др.). Резултатите от тестовите и докладите за дефекти се регистрират, за да може да се извършва контролна функция.

Приемането от Възложителя ще се извършва, при приключване на разработката на пълната функционалност на всички компоненти на Системата, описани в спецификацията на софтуерните изисквания от изпълнявания етап на поръчката. Резултатите от изпълнението ще бъдат нанесени в съответния протокол.

При разработката ще се извършват следните дейности, свързани с тестването:

- Разработка на тестови сценарии за проверка на реализацията на системата;
- Тестване на разработената функционалност по съответния компонент;
- Документиране на резултатите от проведените тестове.

Изпълнението на дейностите по разработка трябва да бъдат планирани, както по елементи, така и по дефинирани периодични итерации, включващи разработка и тестване на отделните елементи и компоненти. Тестване и представяне на резултатите на Възложителя ще се извършва за всяка итерация, в оперативен порядък. В случай, че Възложителят има изисквания за корекции по реализираната функционалност, те ще бъдат отразени от Изпълнителя до представяне на следващата итерация по разработка на съответния компонент. На база извършената разработка ще бъде извършена интеграция и тестване в цялостното технологично решение на системата.

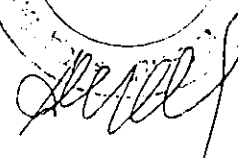
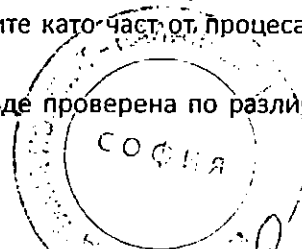
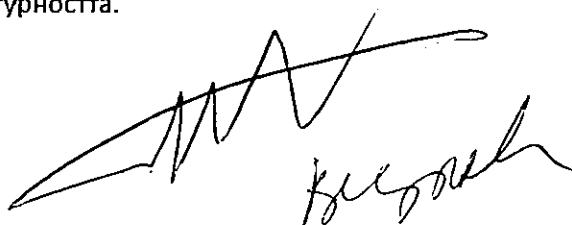


3.5.1. Видове тестове

Тестването е етап на изпълнение, по време на който разработения продукт се проверява за съответствие с функционалната спецификация и проекта. Тестването се съотнася с процеса по контрол на качеството. Тестването се изпълнява паралелно или след процеса по разработка. Задължителен вход за процеса по тестване са резултатите от процеса по разработка. Тестването е тясно свързано с процеса по осигуряване на качеството. Процесът по осигуряване на качеството се изпълнява преди и паралелно на всички останали процеси, включително и този на контрол на качеството. Той се изпълнява едновременно/интегрирано с процесите по анализ, планиране, проектиране, разработване, тестване и внедряване. Това осигурява превантивното откриване и ранно отстраняване на грешки.

Ще бъдат приложени следните видове тестове, за които ще бъдат разработвани тестови случаи:

- Функционално тестване - Системите ще бъдат проверени, за да се потвърди, че те покриват функционалните изисквания на Възложителя, описани в техническата спецификация и съответстват на подробната спецификация, съгласувана с Възложителя.
- Системни тестове - Системните тестове ще бъдат изпълнявани, когато всяка от системите е функционална. Итеративният жизнен цикъл позволява системното тестване да се изпълни възможно най-бързо – веднага след като е завършено някакво подмножество от функционалност. Целта е да бъдат проверени всички функционални елементи на системите. Системните тестове ще се извършват върху симулаторна и тестова среди. Възложителят не трябва да приема софтуер, за чието тестване резултатът от системен тест е частично успешен или неуспешен.
- Интеграционни тестове - Интеграционните тестове ще бъдат изпълнявани, за да се осигури, че компонентите в изпълнението на системите работят коректно след като са комбинирани, за да изпълняват определена функционалност. Целта е компонентите или множествата от компоненти да бъдат проверени за незавършеност или грешки при реализацията. Най-често това са грешки в интерфейсите между компонентите.
- Тестове за производителност - Тези тестове ще проверят дали бързодействието на системата отговаря на очакванията на потребителите. Тестът трябва да докаже приемлива производителност в смисъла на изпълнение и време за отговор. Производителността ще бъде тествана с близки до реалните обеми данни.
- Тестове под натоварване – При тези тестове системата ще бъде поставена под изкуствено натоварване, което симулира високо реално натоварване. Изпълняват се в разпределена среда, от множество машини. Това се прави посредством виртуализирани сървъри, които симулират натоварване от множество потребители.
- Стрес тестове – При тези тестове системата ще бъде поставена под изкуствено натоварване, което симулира екстремално натоварване. Изпълняват се в разпределена среда, от множество машини. Това се прави посредством виртуализирани сървъри, които симулират натоварване от множество потребители, като броят на потребителите се увеличава до отказ на системата.
- Приемни тестове - Приемните тестове ще бъдат последното тестово действие преди внедряване на софтуера. Целта на приемните тестове е да се провери дали софтуерът е готов и може да бъде използван от крайните потребители за изпълнение на техните функции и задачи, за което продуктът е създаден.
- Тестване на модулите - Тестването на модулите е фокусирано върху проверката на най-малкият елемент от софтуера, възможен за тестване. Тестването на модули ще бъде приложено по време на разработката, за да удостовери, че потоците на данните и интерфейса са покрити и функционират според очакванията. Разработчиците ще изпълняват тестове на модулите като част от процеса на разработка. Извършват се от програмиста.
- Тестове за сигурност на системата - при тези тестове, системата ще бъде проверена по различни аспекти засягащи ИТ сигурността.



Приемните тестове ще се извършват съвместно с екип на Възложителя. Всички останали видове тестове са предмет на вътрешни процедури на Изпълнителя и се изпълняват от негов екип. Всички тестове, с изключение на тестването на модулите се извършват от експерта по контрол и осигуряване на качеството. Тестването на модулите се извършва от програмистите.

Някои материали по проекта са предмет на проектиране. Те ще бъдат тествани за:

- **Удовлетворяване на изискванията** - Тези тестове ще докажат, че разработените проекти или процедури отговарят на техническите изисквания.
- **Изпълнимост** - Тестовите за изпълнимост изследват възможността за изпълнение на софтуерния продукт по начина, по който е проектиран.
- **Ефективност** - Целта на тестовите за ефективност е да се докаже, че изпълнението на конкретният дизайн е най-ефективното решение, като се съобразят стойността, риска и ползите.

Документът, съдържащ резултатите от тестовите ще бъде разработен на базата на спецификацията на изискванията и проекта. На тяхна база ще бъде изготвен документ за провеждане на тестови сценарии, който ще бъде съгласуван с Възложителя. Документа с тестовите сценарии ще бъде обновяван при възникване на промени в останалите спецификации. В този документ ще бъде приложен и протокол за отразяване на резултатите от тестовите. По време на провеждането на окончателните тестове, резултатите от изпълнението ще бъдат нанесени в съответния протокол.

3.5.2. Входни и изходни критерии

3.5.2.1. Критерии за започване на тестовия процес

Критерий за започване на тестовия процес е всички необходими документи по проекта да са изготвени от Изпълнителя и одобрени от Възложителя.

3.5.2.2. Критерии за преустановяване на тестването

Ако се открият дефекти, които сериозно се отразяват на напредъка на тест, Ръководител тестване може да избере да спре тестването. Критерии, които ще оправдаят преустановяване на тестването са:

- Хардуера/софтуера, необходими за тестването не са налични;
- Разработената функционалност има един или повече критични дефекти (бъгове), които сериозно възпрепятства или ограничава тест прогреса;
- Ресурсите определени за тестване не са на разположение, когато са необходими на тестерите;
- Постъпило е искане за промяна, което засяга функционалностите, които са застъпени в настоящата итерация.

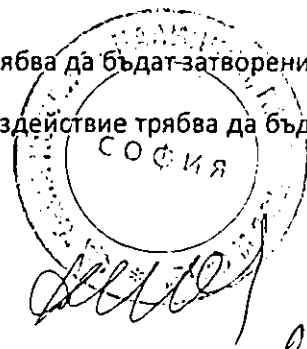
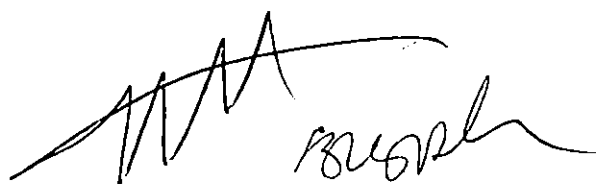
3.5.2.3. Критерии за подновяване на тестването

Ако тестването е преустановено, подновяването му ще се случи, след като причините предизвикали спирането са отстранени. Когато критичен дефект е причината за спирането, отстраняването му трябва да бъде проверено.

3.5.2.4. Критерии за спиране на тестовия процес

При покриването на следните критерии се счита, че тестовия процес е преминал успешно:

- 100% от планираните тестови сценарии трябва да са изпълнени;
- 100% от всички документирани критични програмни грешки и дефекти трябва да бъдат затворени;
- 100% от всички документирани програмни грешки и дефекти с високо въздействие трябва да бъдат затворени;



- Фаза 1 : Първичен тест е изпълнен успешно;
- Фаза 2 : Вторичен тест е изпълнен успешно;
- Фаза 3: Тестове за натоварване и надеждност са изпълнени успешно.

3.5.2.5. Критерии за приемане

Системата и нейните компоненти ще се приемат в експлоатация когато:

- Отговарят на изискванията на настоящата техническа спецификация и на утвърдената от Възложителя методология за изпълнение на дейностите;
- Успешно са преминали всички дефинирани тестове;
- Не произвежда неправилни резултати (и не изпада в състояние да не произведе резултат) при правилни входни данни;
- Не нарушава целостта си и целостта на съхраняваната информация в следствие на некоректни входни данни, програмни или други грешки;
- Не изпада в недетерминирани състояния ("блокира", "заспива") в следствие на некоректни входни данни, програмни и други грешки или продължителна работа;
- Няма логически грешки;
- Няма грешки или съществен спад на производителността, проявяващи се по време на претоварване, увеличаване на капацитета на базата от данни или автоматични действия по архивиране, индексирание и т.н.;
- Няма грешки, зависещи от настъпването и взаимодействието на асинхронно възникващи събития, както и от забавянето на отговора/реакцията на други приложения;
- Няма грешки, проявяващи се след системен срив или системно аварийно възстановяване след изключителни събития (напр. отпадане на захранването или апаратна повреда);
- Съществуват показания, че информационната система може да обработи големи обеми данни без значителен спад на производителността;
- Произвежда резултат в очакваното време за отговор;
- Успешно са изпълнени Фаза 1 : Първичен тест, Фаза 2 : Вторичен тест и Фаза 3: Тестове за натоварване и надеждност.

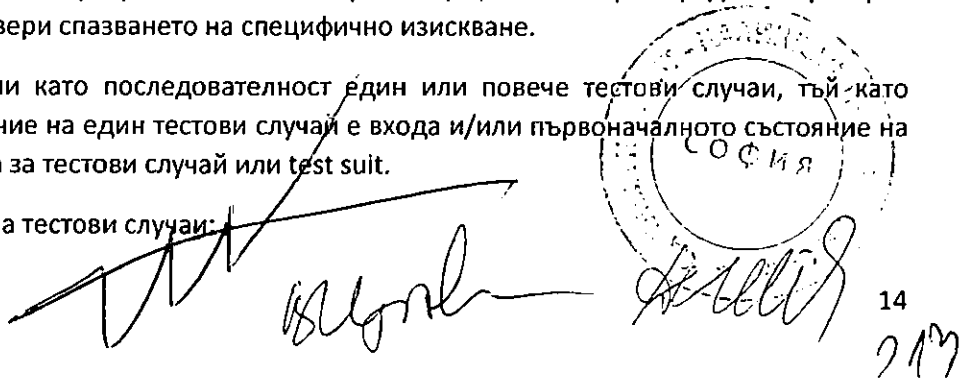
3.5.3. Разработване и използване на успешни тестови сценарии

Предвижда се разработването и изпълнението на пълен набор от тестови сценарии, покриващи в най-голяма степен функционалните и допълнителни изисквания към системите.

Тестовият случай е документация, която определя стойностите за въвеждане, очакваните резултати и необходимите предпоставки за изпълнение на теста. Описват се набор от тестови входове. Условието за изпълнение и очакваните резултати са разработени за конкретна цел, като например да се проиграе частичен програмен път или се провери спазването на специфично изискване.

Теста се определя като: Изпълнени като последователност един или повече тестови случаи, тъй като резултатите и/или крайното състояние на един тестови случаи е входа и/или първоначалното състояние на следващия. Думата тест се използва за тестови случаи или test suit.

Цели за писането и изпълнението на тестови случаи:



14
217

- 3) Намиране на дефекти в софтуерните продукти
- 4) Потвърждение, че софтуерът отговаря на изискванията на крайните потребители
- 5) Подобрене на качеството на софтуера
- 6) Минимизиране на разходите за обслужване и поддръжка на софтуера
- 7) Избягване на рискове при разгръщане на приложението
- 8) Гарантиране на съответствие с процесите
- 9) Помощ при взимане на решение за доставка на софтуера на поръчителя

Документацията за тестовите случаи е също така част от тестовите резултати. Чрез четене на тази документация заинтересованите страни могат да получат представа за качеството на написаните тестови случаи и ефективността на тези тестове. Заинтересованите страни могат също така да предоставят входни данни за текущия набор от тестови случаи, както и да предложат липсващи тестови случаи, ако има такива.

Всеки тестови случай има определени характеристики, които са:

- 1) Трябва да бъде точен и да тества това, което е предназначен да тества.
- 2) Не е задължително да бъдат включени стъпки в него.
- 3) Трябва да може да се използва повторно.
- 4) Трябва да се извършва съгласно изискванията.
- 5) Трябва да бъде съобразен с разпоредби.
- 6) Трябва да бъде независим, т.е. трябва да може да се изпълни в произволен ред без никаква зависимост от други.
- 7) Трябва да бъде прост и ясен, всеки тестер трябва да бъде в състояние да го разбере, когато го прочетете.

На базата на спецификациите на софтуерните продукти ще бъдат разработени тестови сценарии. За всяко изискване от спецификациите ще има един или повече тестови сценарии, успешното изпълнение на които доказва съответствието на продуктите със спецификациите. За целите на осигуряване на съответствие на крайния продукт със спецификациите ще бъдат изготвени и проведени тестови сценарии, успешното изпълнение на които доказва съответствието на продукта със заданието. Резултатите от проведените тестове ще бъдат документирани в Configuration Management системата.

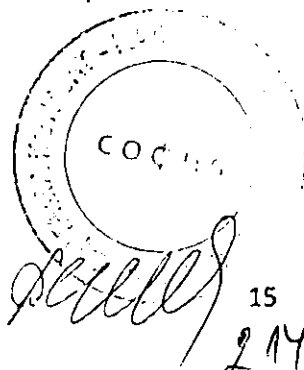
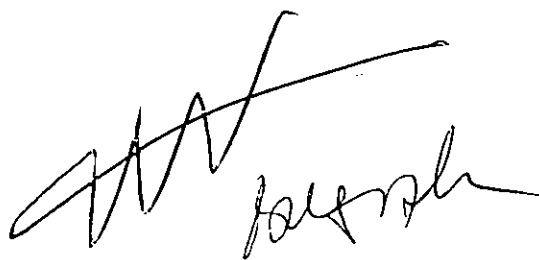
Тестовите сценарии се разработват от експерта по контрол и осигуряване на качеството.

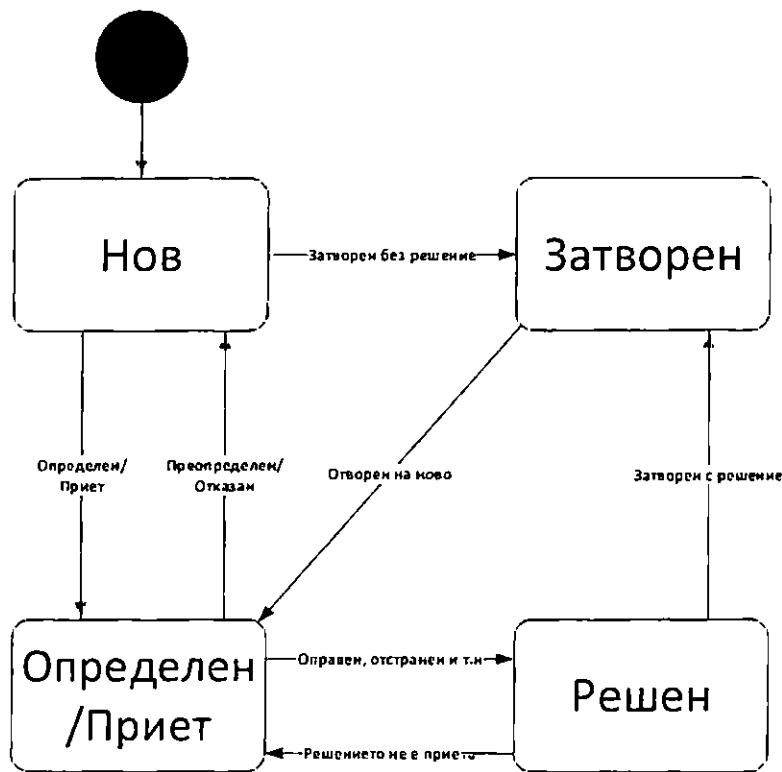
3.5.4. Проследяване на дефектите

Проследяването на дефектите е процес на намиране на дефекти (несъответствия) в даден продукт (чрез проверка, тестване или запис на обратна връзка от клиенти), и създаване на нови версии на продукта, в който тези дефекти са отстранени. Проследяването на дефектите е важно в софтуерното инженерство, тъй като сложните софтуерни системи обикновено имат десетки, стотици или хиляди дефекти - управлението, оценката и приоритизирането на тези дефекти е трудна задача. Системите за проследяване на дефектите са компютъризирани база от данни, които съхраняват информацията за дефектите и помагат на хората да ги управляват.

Прилаганите процедури за управление и отстраняване на установени несъответствия и дефекти ще бъдат базирани на ITIL.

Чрез системата за проследяване на дефекти, екипът следи състоянието на регистрираните дефекти. Следва се концепцията дефектите да бъдат прегледани преди да бъдат възложени задачи за отстраняването им. Те също така се преглеждат преди да бъдат маркирани като коригирани.





Състояния на несъответствията и преходи между тях

Целта на предотвратяването на дефектите и несъответствията е след като те бъдат открити и отстранени да се осигури те да не се появяват в следващите итеративни цикли.

Предотвратяването на дефектите може да бъде постигнато чрез изпълнение на следните стъпки:

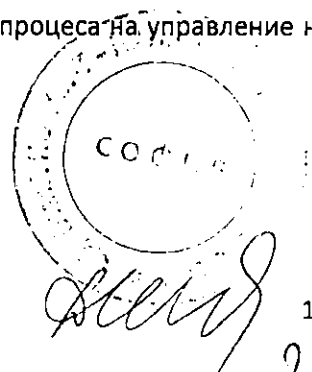
- 1) Събиране на данни за дефектите и несъответствията и провеждане на периодични прегледи с използване на историята за проведените тестове: тези данни трябва да се използват за разделяне и класифициране на дефектите по първопричините за тяхното появяване.
- 2) Идентифициране на стратегии за подобряване;
- 3) Ескалиране на въпроси към висшия мениджмънт или клиента, когато е необходимо;
- 4) Да се изготви план за действие, който обръща внимание на повтарящите се дефекти и подобряване на процеса на разработка. Планът трябва да се преразглежда редовно за ефективност и промяна, когато се докаже, че заложените мерки са неефективни.
- 5) Извършване на периодични проверки, за да се провери, че плановете се спазват;
- 6) Редовно да се представят доклади на дефектите, обикновено подредени по продължителност на съществуването. Ако дефектът е стар и въздействието му е голямо, това трябва да породи безпокойство и да се вземат съответните мерки;
- 7) Класифициране на дефекти в категории като критични дефекти, функционални дефекти, както и козметични дефекти.
- 8) Проследяването и предотвратяването на дефектите е част от процеса на управление на софтуерната конфигурация.

Проследяването и предотвратяването на дефектите и несъответствията е част от процеса на управление на софтуерната конфигурация.

В процедурата за проследяване на дефектите участва целия екип.

3.6. Azure виртуални машини EUCISE2020

[Handwritten signatures]



За нуждите на проекта ще се използват виртуалните машини, които ще бъдат реализирани в Microsoft Azure.

Azure е една от водещите платформи за облачни услуги, предлагана от Microsoft. Позволява голям набор от услуги и функционалности - виртуални машини, база данни, изграждане на мрежи, създаване на хранилища, архивиране на ресурси, активна директория, приложения и много други. Важна характеристика на платформата е, че функционира с всяка операционна система и широк спектър от програмни езици.

Microsoft Azure позволява лесно да се мащабират приложения във всякакви размери. Той е напълно автоматизирана платформа за самообслужване, която позволява предоставянето на ресурси в рамките на минути. Може гъвкаво да се увеличава и намалява мащаба на ресурсите спрямо нужди. Microsoft Azure е на разположение в множество центрове за данни по целия свят.

Microsoft Azure е проектирана за достъп до всички услуги, машини и приложения, без да се налага да се притеснявате за прекъсвания в достъпността или загуба на данни. Системата Azure осигурява 99,95% месечно SLA, непрекъснат мониторинг на състоянието на услугите и постоянна техническа 24 часова помощ.

Достъпът до виртуалните машини ще се реализира с изграждане на VPN тунел, като клиентът трябва да осигури устройство от негова страна, върху което да бъде терминиран постоянен VPN. Трафикът до облачната услуга и обратно е криптиран. Повече информация за сигурността може да се прочете тук: <https://azure.microsoft.com/en-us/overview/trusted-cloud/> , <https://azure.microsoft.com/en-us/blog/microsoft-azure-leads-the-industry-in-iso-certifications/> , <https://docs.microsoft.com/en-us/azure/security/azure-network-security> , <https://docs.microsoft.com/en-us/azure/security/azure-security> .

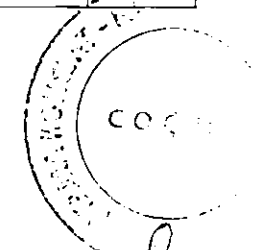
В следващите таблици са описани необходимите виртуални машини

Таблица 1 Изисквания за адаптора

Hostname	Service Domain	Services	Изисквания			Предложени Azure VMs			Съответствие
			CPU	RAM (GB)	Disk (GB)	CPU	RAM (GB)	Disk (GB)	
UCISE2020-omm01	Слой на приложението	Сървър на приложението	2	4	64	2	4	64	ДА
UCISE2020-omm02	Слой на приложението	Сървър на приложението	2	4	64	2	4	64	ДА
UCISE2020-m01	Симулатор	Сървър на приложението Симулатор EGN mode	6	16	64	8	16	64	Надвисява
бщо			10	24	192	12	24	192	Надвисява

Таблица 2 Изисквания за възела

[Handwritten signature]



[Handwritten signature]
17
216

Област на услугите	Услуги	Изисквания			Предложени Azure VMs			Съответствие	Инсталирани продукти
		CPU	RAM (GB)	Disk (GB)	CPU	RAM (GB)	Disk (GB)		
Слой на приложението	Сървър на приложението Общи услуги Услуги на ядрото	6	16	128	8	16	128	Надвишава	Oracle jdk 1.8.0 Pacemaker 1.1.16 GlusterFS 3.x Wildfly 10.1.0.Final
Слой на приложението	Сървър на приложението Vessel Fusion услуги	2	8	96	2	8	128	Надвишава	Oracle jdk 1.8.0 Pacemaker 1.1.16 GlusterFS 3.x Wildfly 10.1.0.Final
Слой на приложението	Мрежови услуги Комуникационни услуги Услуги на приложението	4	8	52	4	8	64	Надвишава	Apache HTTPD 2.x Apache ActiveMQ Artemis 1.5.x Consul 0.7.5 Pacemaker 1.1.16 GlusterFS 3.x
Слой на приложението	Административна конзола	2	8	96	2	8	128	Надвишава	Oracle jdk 1.8.0 Oracle jdk 1.7.0 JBoss 7.1 Liferay 6.2 GlusterFS 3.x
Разпределена мрежа за данни	Сървър на приложението Машина за синтезиране на данни База данни	8	32	96	8	32	128	Надвишава	Oracle JDK 1.8.0 Oracle Coherence
База данни	PostgreSQL	2	8	382	2	8	512	Надвишава	PostgreSQL 9.6 Pacemaker 1.1.16 GlusterFS 3.x
Представяне	Компонента за наблюдение (Light Client) GIS Light Client услуги на приложението	6	16	232	8	16	256	Надвишава	Oracle Jdk jdk1.8.0_121 Postgresql 9.4.4 Postgis: postgis-bundle-pg94 2.1.7-1 Wildfly 10.0.0.Final WMS server (Geo Server or WorldWind)
Мрежови услуги	DNS, LDAP, NTP	2	2	96	2	4	128	Надвишава	Oracle jdk 1.8.0 OpenLDAP 2.4 Pacemaker 1.1.16 GlusterFS 3.x

Съвместни услуги	Съвместни услуги, NFS	2	4	544	2	4	1024	Надвишава	Oracle jdk 1.8.0 Apache OpenMeetings 3.2 AfterLogic WebMail Lite 7.7 Apache HTTPD 2.x PHP engine 5.x MySQL 5.x Pacemaker 1.1.16 GlusterFS 3.x
Услуги за одитиране	Услуги за одитиране	2	4	96	2	4	128	Надвишава	Oracle jdk 1.8.0 WildFly 10.1.0.Final Active MQ 5.14.4 Nagios x.x
Инфраструктурни услуги	Routing/Security	2	2	10	2	4	32	Надвишава	VyOS
Сертификат за авторизация	CA	2	4	64	2	4	64	ДА	Oracle jdk 1.8.0 OpenSSL 1.x EJBCA 6.x
Total		40	112	2020	44	116	2720	Надвишава	

Таблица 3 Изисквания за симулатора

Област на услугите	Услуги	Изисквания			Предложени Azure VMs			Съответствие	Инсталирани продукти
		CPU	RAM (GB)	Disk (GB)	CPU	RAM (GB)	Disk (GB)		

[Handwritten signature]

[Circular stamp with text 'СОФТ' and handwritten signature]

Област на услугите	Услуги	Изисквания			Предложени Azure VMs			Съответств	Инсталирани продукти
Симулатор	Сървър на приложението Симулатор в LSA режим	2	8	96	2	8	128	Надвигва	Oracle jdk 1.8.0 Apache Tomcat 8.5.11 Simulator webapp SoapUI 5.3.0 JMeter Vessel Simulator Asterix datablock generator Pacemaker 1.1.16 GlusterFS 3.x

3.7. Очаквани резултати от изпълнението на интеграционните тестове

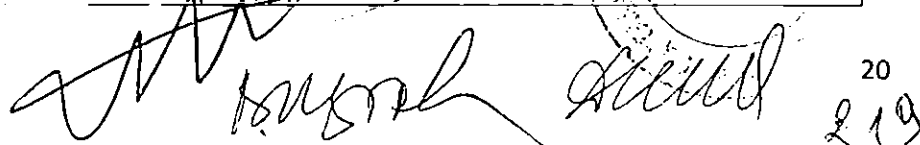
В резултат от изпълнението на дейността Изпълнителят ще предостави следните резултати:

- Протоколи за изпълнени тестове
- Протоколи за извършени услуги за техническа асистенция

4. Начин на изпълнение на дейностите по отстраняване на евентуални недостатъци, открити при тестовете. Подход за управление на услугата по поддръжка

Следва предложение за предоставяне на услуга по осигуряване на работоспособността и отстраняване на евентуални недостатъци, открити при тестовете на EUCISE2020 Адаптера. Дефинирани и категоризирани са инцидентите, които могат да се появяват в процеса на тестова обмяна на данни. Описан и обоснован е начина на организация, действия и процедури, които Изпълнителят ще извърши при осъществяване на услугата, така че да се гарантира висока степен на ефективност на разработеното решение спрямо изискваните параметри и качество на услугата.

Ключови елементи от предложението	Препратка
Начин на организация	Т. 4.1 - „Категоризация на инцидентите. Приоритети, достъпност и време за реакция при инциденти“ Т. 4.2 - „Организация при предоставяне на услугата“ Т. 4.4 - „Система за регистриране и проследяване на инцидентите (портал за поддръжка)“ Т. 4.5 - „Система за автоматизиран мониторинг Zabbix 3.2“ Т. 4.6 - „Предпоставки за успешна поддръжка от страна на Възложителя“
Действия и процедури, които ще бъдат извършени при осъществяване на поддръжката на EUCISE2020	Т. 4.3.1 - „Видове дейности за отстраняване на инциденти“



Адаптера, така че да се гарантира ефективното изпълнение на услугата	Т. 4.3.2 – „Процедура за обработка на инциденти“ Т. 4.3.3 – „Управление на проблеми“
Обосновка за приложимостта и полезността на предложеното решение при изпълнението на поръчката	Т. 5 – „Обосновка за приложимостта и полезността на предложеното решение“

4.1. Категоризация на инцидентите. Приоритети, достъпност и време за реакция при инциденти

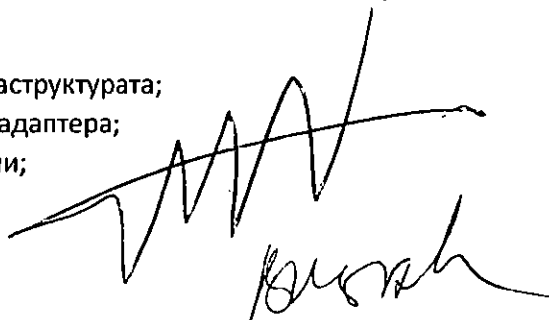
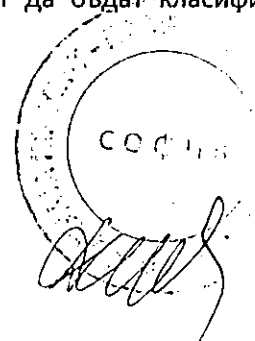
Приоритетите на инцидентите и проблемите се определят от Възложителя в зависимост от влиянието им върху работата му.

В таблицата по-долу е представена категоризацията на инцидентите и проблемите спрямо приоритета, дефинирана е достъпността на услугата и времето за реакция, които ще бъдат осигурени:

Приоритет	Описание	Достъпност на услугата	Време за реакция
Висок	Въздействие върху работните процеси вследствие нарушаване на основните функционалности на структуроопределящите компоненти, водещо до пълната системна неработоспособност; Системата е напълно неработоспособна или работата ѝ предизвиква невъзстановимо записване на грешни данни.	В рамките на работното време от 09:00 часа до 17:30 часа без официалните празници и почивните дни.	До 4 часа в рамките на работния ден, считано от уведомяването от страна на Възложителя.
Среден	Въздействие върху работните процеси вследствие частично нарушаване на функционалностите на структуроопределящите компоненти, водещо до частична системна неработоспособност; Част от съпътстващата функционалност на системата е неработоспособна, например справки и др.	В рамките на работното време от 09:00 часа до 17:30 часа без официалните празници и почивните дни.	В рамките на следващия работен ден, считано от уведомяването от страна на Възложителя.
Нисък	Въздействие върху работните процеси вследствие нарушаване на функционалности на неструктуроопределящите компоненти, водещо до частична системна неработоспособност; Системата е работоспособна, но забелязаният недостатък предизвиква неудобство и затруднение при работа.	В рамките на работното време от 09:00 часа до 17:30 часа без официалните празници и почивните дни.	До 3 работни дни, считано от уведомяването от страна на Възложителя.

В зависимост от областта на възникване, инцидентите и проблемите могат да бъдат класифицирани основно в следните групи:

- Проблеми в ИТ инфраструктурата;
- Програмни грешки в адаптера;
- Грешки в базата данни;

- Програмни грешки в тестовия клиент;
- Грешки при връзка с EUCISE2020 GW/Node;
- Грешки при дефинирането на модела на данните;
- Др., идентифицирани по време на провеждане на тестването на Адаптера.

4.2. Организация при предоставяне на услугата

4.2.1. Нива на поддръжка

Системата за поддръжка се базира на три нива:

- Ниво 1 – Поддръжка на място, Текущи задачи по поддръжката, Централизирана поддръжка за клиента, Системна конфигурация и управление – Изпълнява се от екип на Възложителя;
- Ниво 2 – Казуси, които не могат да бъдат решени на по-ниско ниво на поддръжка – Изпълнява се от звеното за поддръжка на Изпълнителя;
- Ниво 3 – Казуси, които изискват промени в елементи на системата – Изпълнява се от инженерни екипи на Изпълнителя.

Първото ниво на поддръжка се осъществява от администраторите на ИС и ключовите потребители на Възложителя. При невъзможност проблемът да бъде решен на първо ниво, лицата за контакт при Възложителя ескалират проблема към второ ниво на поддръжка.

Първо ниво поддръжка е определено както следва:

Обучени служители на Възложителя, изпълняващи ролята на първо ниво на поддръжка ще приемат и регистрират проблеми със системата в портала за поддръжка. При невъзможност да се справят с проблема, първо ниво ще уведоми второ ниво на поддръжка за инцидента чрез имейл или телефон, ако регистрирания проблем е от критично естество. Първо ниво на поддръжка ще бъде инструктирано от екипа по телефон, чрез e-mail или портала за поддръжка, описан в т. 4.4 Система за регистриране и проследяване на инцидентите (портал за поддръжка) от настоящето Приложение № 4. Във всеки един от гореописаните случаи на връзка - имейл или телефон, описанието на инцидента, статуса, както решението му ще бъдат въведени в системата за регистрация на инциденти.

Обученият персонал, изпълняващ функциите на първо ниво на поддръжка ще предоставя следните услуги:

- Помощ и обучение за потребителите по отношение на общи знания за системата и използването на документацията.
- Експлоатиране на системата.
- Базови дейности по поддръжка. Възложителят ще поддържа системата в добро състояние, както и ще следи за изпълнението на превантивни дейности и дейности по ежедневна поддръжка. Възложителят ще поддържа журнал за Инцидентите в системата (дата, час, инцидент, причина, възстановителни дейности и др.)
- Изпращане на квалифицирано описание на проблема (последователност от събития, причина и др.) чрез текущия запис за инцидента.
- Помощ при отстраняването на проблема чрез отдалечен достъп или дейности на място, при необходимост и извън нормално установеното работно време.
- Инсталиране и тестване на пакети с отстранени грешки.
- Адаптиране на физически параметри (слова и файлове за инициализиране)



Handwritten signatures and initials at the bottom of the page.

Първо ниво поддръжка включва обработване на:

- Потребителски инциденти;
- Потребителски грешки;
- Потребителски въпроси;
- Оперативно наблюдение и администриране на системата;

Второ ниво поддръжка е определено както следва:

ИЗПЪЛНИТЕЛЯТ ще предостави второ ниво на поддръжка. Второто ниво оказва помощ на първото ниво на поддръжка по телефон, електронна поща или чрез портала СПРИ (описан в т 4.4 Система за регистриране и проследяване на инцидентите (портал за поддръжка) от настоящето Приложение № 4). В случай, че инцидентът не може да бъде разрешен от първото ниво за поддръжка се уведомява второто ниво за инцидента и то го анализира. При необходимост от по-подробна информация относно проблема, второто ниво може да изиска от първото ниво необходимата информация.

В случай на необходимост, екипите от второ ниво на поддръжка могат да ескалират инцидент към третото ниво.

Трето ниво на поддръжка:

Ролята на трето ниво на поддръжка се изпълнява от инженерните звена при Изпълнителя: Софтуерна поддръжка, Сервиз и т.н.

Трето ниво на поддръжка включва следните видове дейности:

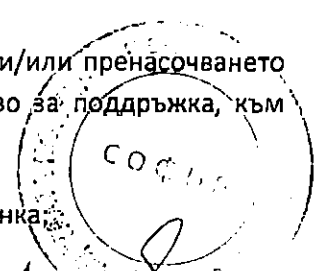
- Отстраняване на дефекти в софтуера;
- Отстраняване на дефекти в дизайна на системата;
- Промени в техническата инфраструктура;
- Разрешаване на инциденти, които не могат да бъдат решени на по-ниски нива.

При невъзможност казусът да бъде разрешен на нивото, на което е поставен, той се ескалира на следващото по-горно ниво, т.е. Ниво 1 ескалира казусите към Ниво 2, а Ниво 2 ескалира към Ниво 3.

4.2.2. Отговорности и задължения на участниците в процеса

Екипът за поддръжка има следните задължения:

- Приемане на информация за инциденти със софтуера, както и приемане на информация за осъществяване на обратна връзка с Възложителя;
- Регистриране и проследяване на Инциденти и заявки за промени;
- Информирание на Възложителя относно статуса на заявката и постигнатия напредък на работата по нея;
- Първоначална оценка на регистрирания проблем, опит за разрешаване му и/или пренасочването на проблема, в случай, че не може да бъде разрешен от съответното ниво за поддръжка, към следващо такова, съгласно договорените условия;
- Управление на жизнения цикъл на заявката, включително приключване и оценка.



- Предоставяне на справки на управляващия персонал;
- Идентифициране на проблеми;
- Идентифициране и оценка на необходимостта от обучение на клиента;
- Приключване на Инциденти след получено потвърждение от страна на заявителя, че инцидента е разрешен.

Отговорностите за поддръжка на системата ще бъдат разпределени между Изпълнителя и Възложителя. Изпълнителя ще отговаря за второ и трето ниво на поддръжка, а Възложителят - за първо ниво. Предварително подбрани и обучени от Изпълнителя специалисти на Възложителя, ще поемат функциите на първо ниво на поддръжка.

Непосредственото привличане на служители на Възложителя в работата по първо ниво на поддръжка се препоръчва от най-добрите световни практики, включително ITIL (IT Infrastructure Library) стандарта. Този подход има следните предимства:

- Повишаване капацитета на Възложителя да експлоатира и поддържа системата в след-гаранционния период;
- Подпомагане на Възложителя да оценява по-добре своите бизнес нужди и изисквания;
- Подпомагане внедряването на единна практика - отговори на въпроси, свързани с функционалност на системата и помощ при прилагане на процедурите за работа на Възложителя;
- Подпомагане на Възложителя да оцени влиянието на инцидентите и да установява приоритети за разрешаването им;
- Създаде единна точка на контакт с потребителите - това ще елиминира възможността една и съща заявка да се регистрира и при Възложителя и при Изпълнителя, и необходимостта от уточняване и съгласуване между засегнатите страни;
- Подпомогне наблюдението на натовареността на служителите на Възложителя и идентифицирането на необходимостта от обучение на потребителите и поддържащия системата персонал.

Екипът на Възложителя има следното участие:

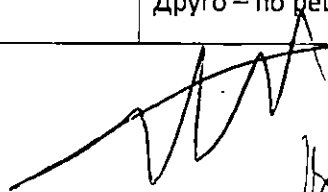


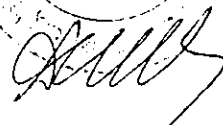
- Уведомява Изпълнителя в случай на установени грешки;
- Координира комуникация с трети страни по осъществяване на поддръжката.

4.2.3. Канали за комуникация

Всеки Инцидент свързан с проблем във софтуера или искане за промяна, докладвани от Възложителя, тяхната история и решения ще бъдат регистрирани.

Комуникационни канали:

Първо ниво на поддръжка (Управлява се от Възложителя)	Ел. поща: предоставя се от Възложителя
	Телефон: предоставя се от Възложителя
	Друго – по решение на Възложителя

Второ и трето ниво на поддръжка (Управлява се от Изпълнителя)	Ел. Поща
	Телефон
	Портал за поддръжка

Всички Инциденти трябва да бъдат предадени от първо на второ ниво за поддръжка писмено, като се използва портала за поддръжка, включително с прикачени всички необходими файлове и информация за анализ на грешката.

Ескалиране на инциденти към второ ниво на поддръжка:

Всички приоритети	Всички инциденти се подават от страна на екипа от първо ниво на поддръжка чрез портал за поддръжка.
Приоритет Висок	Допълнително, за тези Инциденти се очаква предупредително обаждане по телефон в рамките на редовното работно време. Единствено за тази цел е предоставен номер за връзка.

4.2.4. Работно време

Обработката на инциденти се извършва в рамките на редовното работно време в интервала от 9:00 до 17:30 ч. през официалните работни дни в Република България.

При инциденти със статус „Висок“, заявление за инцидент се приема на телефон.

4.3. Методика за обработка на инциденти

4.3.1. Видове дейности за отстраняване на инциденти

За осъществяване на поддръжката ще се прилагат следните дейности:

- Дистанционна техническа помощ. При това Възложителят трябва да е осигурил персонал за извършване на поддръжката от първо ниво, който да контактува с екипа за поддръжка на второ ниво.
- Online техническа помощ (задължително преди това действие е пробвано проблемът да се реши чрез дистанционна техническа помощ). Възложителят трябва да е осигурил персонал за извършване на поддръжката от първо ниво, който да контактува с екипа за поддръжка на изпълнителя и online връзка до сървъра, при който е възникнал проблема.
- Техническа помощ на място. След като са изчерпани вариантите за дистанционна и online техническа помощ може да се пристъпи към техническа помощ на място, при която екип на Изпълнителя посещава мястото при Възложителя, където е възникнал проблема.
- Ремонт/подмяна на дефектирали компоненти на системата.
- Предложения за профилактични дейности по поддръжката. Екипът за поддръжка на Изпълнителя може да прави такива предложения пред централния екип за поддръжка на Възложителя.

4.3.2. Процедура за обработка на инциденти

[Handwritten signature]

[Circular stamp: ССС И С]
[Handwritten signature]

НАЧАЛО

ПОСТЪПВАНЕ НА ИНЦИДЕНТ

Инцидент може да постъпи по един от следните начини:

Наличност на услугата	Възможни начини на подаване на заявка
8 часа x 5 дни	- на портала за поддръжка - на електронна поща - по телефон

Оторизиран представител на Възложителя, Оператор

ОТОРИЗАЦИЯ:

1. Преди да се пристъпи към обработката на нова заявка, Операторът се уверява, че лицето, което изпраща сервизната заявка, е оторизирано от Възложителя.
2. Ако той/тя не е в регистъра с договори с оторизирани служители, Операторът се свързва с основното лице за контакт от страна на Възложителя и изисква потвърждение за валидността на заявката и добавяне на нов контакт в регистъра с оторизирани лица.

Отговорник: Оператор

РЕГИСТРИРАНЕ НА ИНЦИДЕНТА

Регистриране на повредата при установено отклонение, съобразно Процедура за генериране на отчети и разпространение на информацията или подадена заявка за сервизна повреда от оторизирани представители на Възложителя

Събиране на детайлна информация за системата и характера на дефекта.

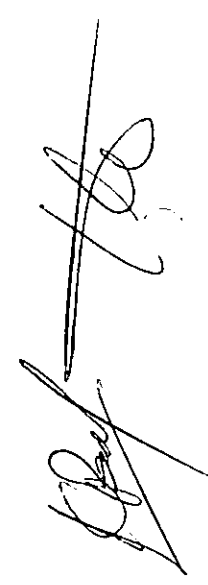
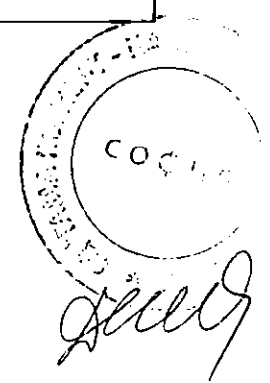
Уточняване обстоятелствата за инцидента, анализиране на признаците на проблемите и данните, ползвайки база данни с Известни грешки;

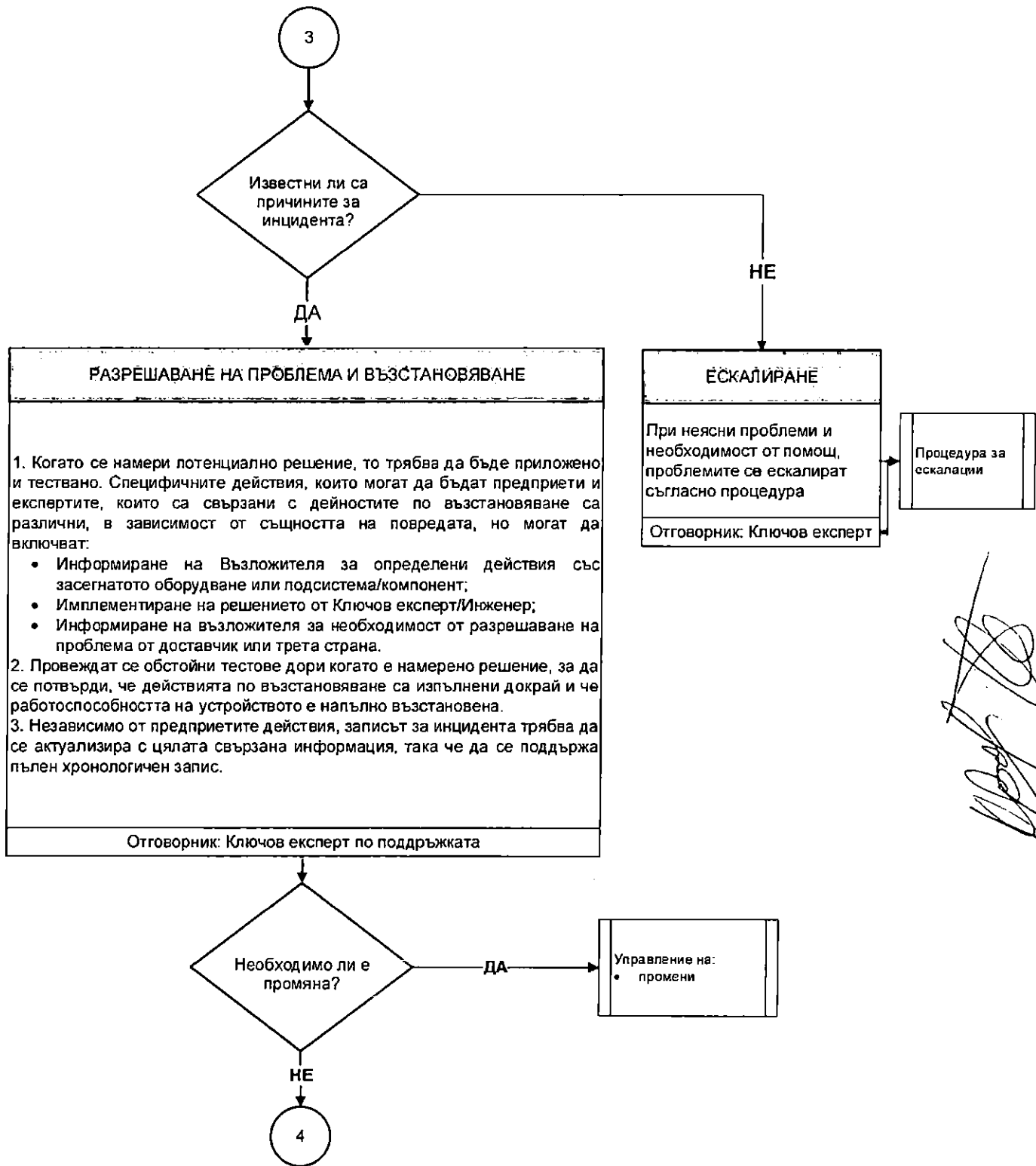
При необходимост инцидентът се ескалира към следващо ниво;

Подаване данни за инцидента в системата на управление на инциденти и заявки, разполагаща с модул за измерване на ниво на обслужване (SLA – Service Level Agreement)

Отговорник: Оператор

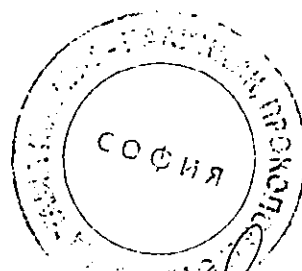
1



[Handwritten signature]

[Handwritten signature]



[Handwritten signature]

27
126

4

ЗАТВАРЯНЕ НА ИНЦИДЕНТА

1. Преди затваряне (разрешаване) на инцидента, трябва да се получи потвърждение, че случаят е разрешен и признаците на проблема не се проявяват вече.

2. Разрешаването на инцидента се описва в системите за управление на задачите и СМВД и се изпраща уведомление към Възложителя, че случаят е разрешен.

Отговорници: Ключови експерти, Координатор на екипите в двата центъра



Управление на:

- проблеми
- промени

ДОПЪЛВАНЕ НА БАЗАТА ДАННИ СМВД ЗА ИЗВЕСТНИ ГРЕШКИ

1. В случай че инцидентът не може да бъде свързан с известна грешка, то Ключовият експерт, отговорен за обновяване на базата, допълва новите данни, като описва максимално детайлно инцидента и причините, които са го породили.

2. Ключовият експерт може да получи данни от:

- Оператори/Първо ниво на поддръжка;
- 3-то ниво поддръжка;
- Възложител.

Отговорник: Ключов експерт

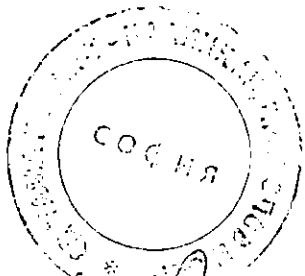
АНАЛИЗ НА УПРАВЛЕНИЕТО НА ИНЦИДЕНТИ (КРІ)

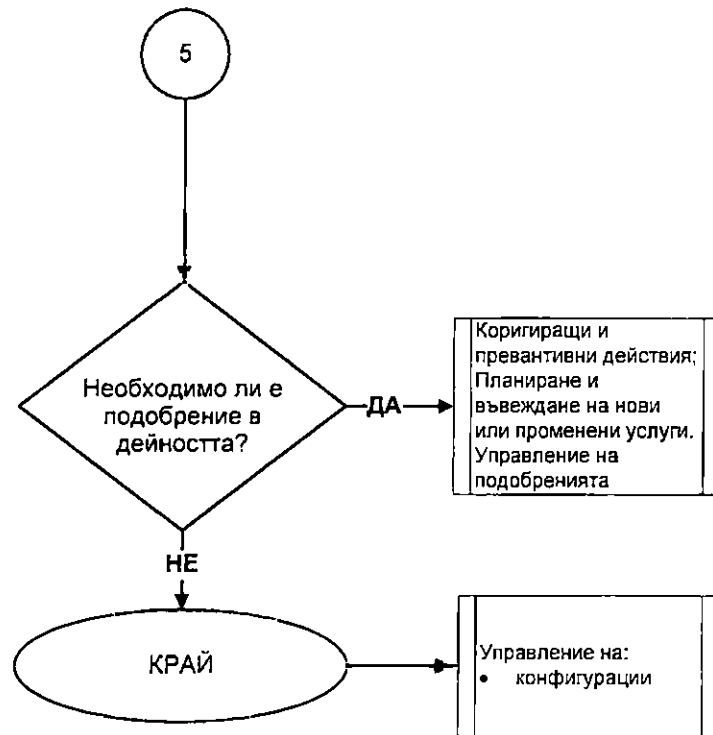
Периодично, съгласно Процедурата за генериране на отчети и разпространение на информацията се извеждат специфични справки и отчети, като се проследяват тенденции за следните параметри:

- повишени нива на инциденти;
- често повтарящи се видове;
- очертаващи се насоки – повтарящи се инциденти, проблеми, известни грешки;
- недостиг на ресурси – персонал, обучение, документация;
- несъответствия спрямо стандарти, политики, законодателство;
- несъответствия спрямо SLA;
- и др.

Отговорник: Ръководител на екипи, Координатор на екипи

5





4.3.3. Управление на проблеми

Проблем е неизвестната причина за поява на един или повече инциденти.

Отговорности и роли при управлението на проблеми:

Ръководител на процеса *Управление на проблеми* е Ръководителят на проекта.

Ролите, свързани с Управление на проблемите, са дефинирани в контекста на управленските функции и не съответстват с организационни длъжностни позиции. Важно е да се знае, че това са по-скоро роли, а не длъжностни характеристики.

Ролите свързани с процеса са:

Ръководител управление на проблеми – Ръководител на проблем за конкретен случай е инженерът, който управлява Инцидента, довел до този проблем, който е установил, че съществува проблем.

Ръководител управление на проблеми е отговорен за разрешаването на проблема през целия процес. Тази роля координира всички дейности по Управление на проблема и има специфични отговорности, като:

- Поддържането на история за основните стъпки по изолиране, възстановяване и решение на проблема
- Превръщането на проблема в Известна грешки
- Генериране на Заявка за промяна (RFC)
- Идентифициране, документиране и записване на алтернативно решение
- Осигуряване на ресурси за оказване на поддръжка
- Съгласуване с всички групи за разрешаване на проблеми с цел бързо решение на проблема в рамките на договорените параметри от договора
- Осигуряване на проактивни анализи на данни за идентифициране на потенциални прекъсвания и отпадания
- Формално затваряне на всички Форми за регистриране на проблем.

Вълчан

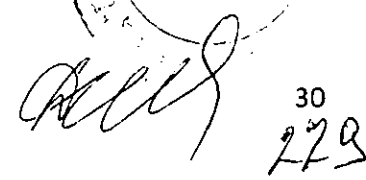
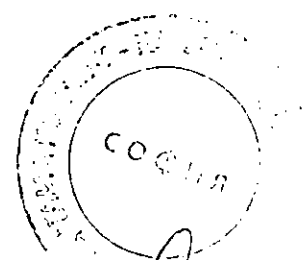
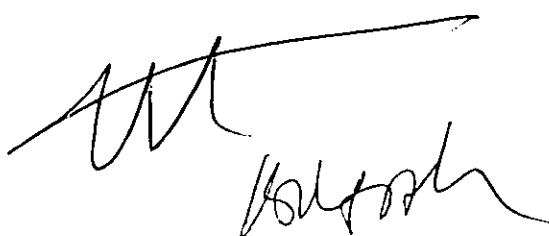
СОФИА
29
228

Група за разрешаване на проблеми – разрешаването на проблем се поема от една или повече технически групи инженери и/или доставчик или партньор – под координацията на Ръководител проблеми. Специфичните отговорности включват:

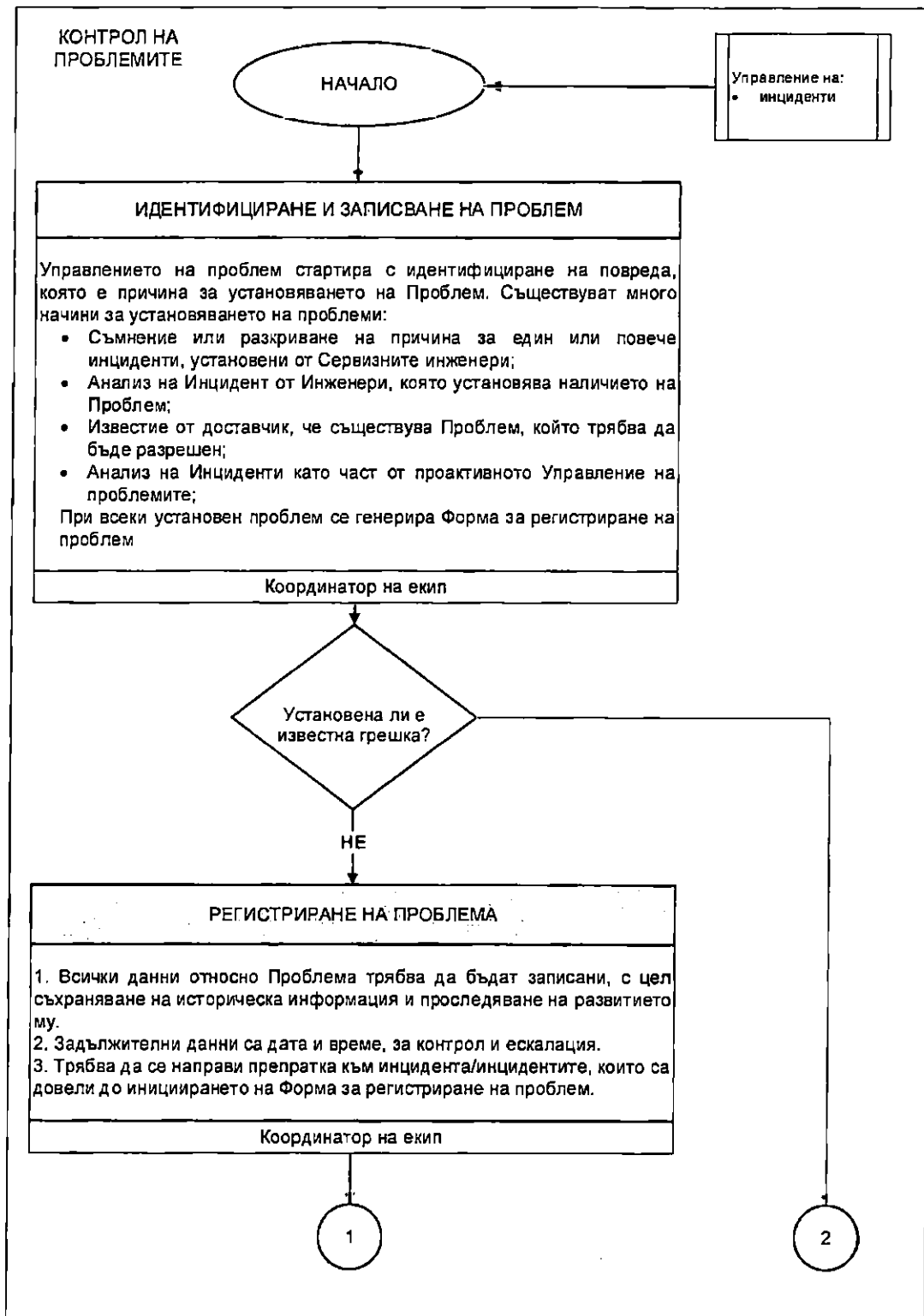
- Анализ на проблем за идентифициране на причините
- Осигуряване на информация и документация за проблемите
- Препоръка за постоянно решение (я)
- Съветване на служителите, занимаващи се с Управление на инцидентите, относно най-добрите алтернативни решения на Инциденти, свързани с неразрешени проблеми/ известни грешки.

Контролът на проблемите представлява систематичен анализ на Инцидентите, които са повлияли на наличността и работоспособността на компонентите, с цел да се определят Проблемите, които да се проучат, да се поставят приоритети в процеса по идентифициране на проблеми и диагностика на причините.

Етапът *Контрол на грешките* от процеса по Управление на Проблем се занимава с управление на Известните грешки до момента на тяхното успешно разрешаване.



30
229



[Handwritten signature]

[Handwritten signature]

[Handwritten signature]

1

КАТЕГОРИЗИРАНЕ НА ПРОБЛЕМА

1. Проблемите са категоризират по сходен начин, като Инцидентите

Координатор на екип

ИДЕНТИФИЦИРАНЕ НА НЕОБХОДИМИТЕ УМЕНИЯ И НАЗНАЧАВАНЕ НА РЕСУРСИ

1. Определят се необходимите знания и умения за откриване на причините за Проблема.
2. Назначава се ресурс със съответната компетенция и се сформира група за решаване на проблема.

Ръководител на екип

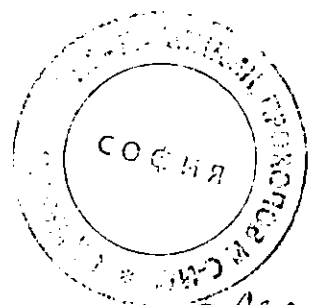
ПРОУЧВАНЕ НА ПРОБЛЕМ И ДИАГНОСТИЦИРАНЕ

Извършва се проучване с цел установяване причината за възникването на Проблема. Дейностите включват:

- Потвърждаване на съществуващи Алтернативни решения и/или определяне на нови такива
- Проучване на причините за възникването на Проблем
- Документиране на причините

Координатор, Ключови експерти

2



Handwritten signature and number 231

2

КОНТРОЛ НА ГРЕШКИТЕ

ИДЕНТИФИКАЦИЯ И ДОКУМЕНТИРАНЕ НА ГРЕШКИТЕ

1. Известна грешка се определя, когато се установи повредата, която причинява или най-вероятно е причина за Инцидент.
2. Веднага след като диагностицирането приключи или е намерено алтернативно решение, Известната грешка трябва да бъде документирана в базата данни с Известни грешки, за да може при бъдещи такива грешки по-бързо да се възстанови услугата.
3. Известната грешка съдържа:
 - точна информация за повредата;
 - някои или всички симптоми, които са известни;
 - алтернативното решение или постоянното решение, предприети за възстановяване на услугата и/или разрешаване на Проблем.
4. Важно е всички Инциденти, свързани с Известна грешка да бъдат документиран заедно, което дава информация за честотата на проблема, което от своя страна да окаже значение за намирането на по-постоянно решение от наличното алтернативно такова.

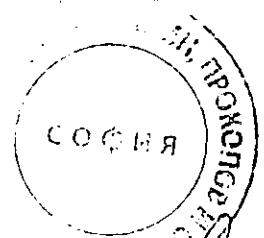
Координатор, Ключови експерти

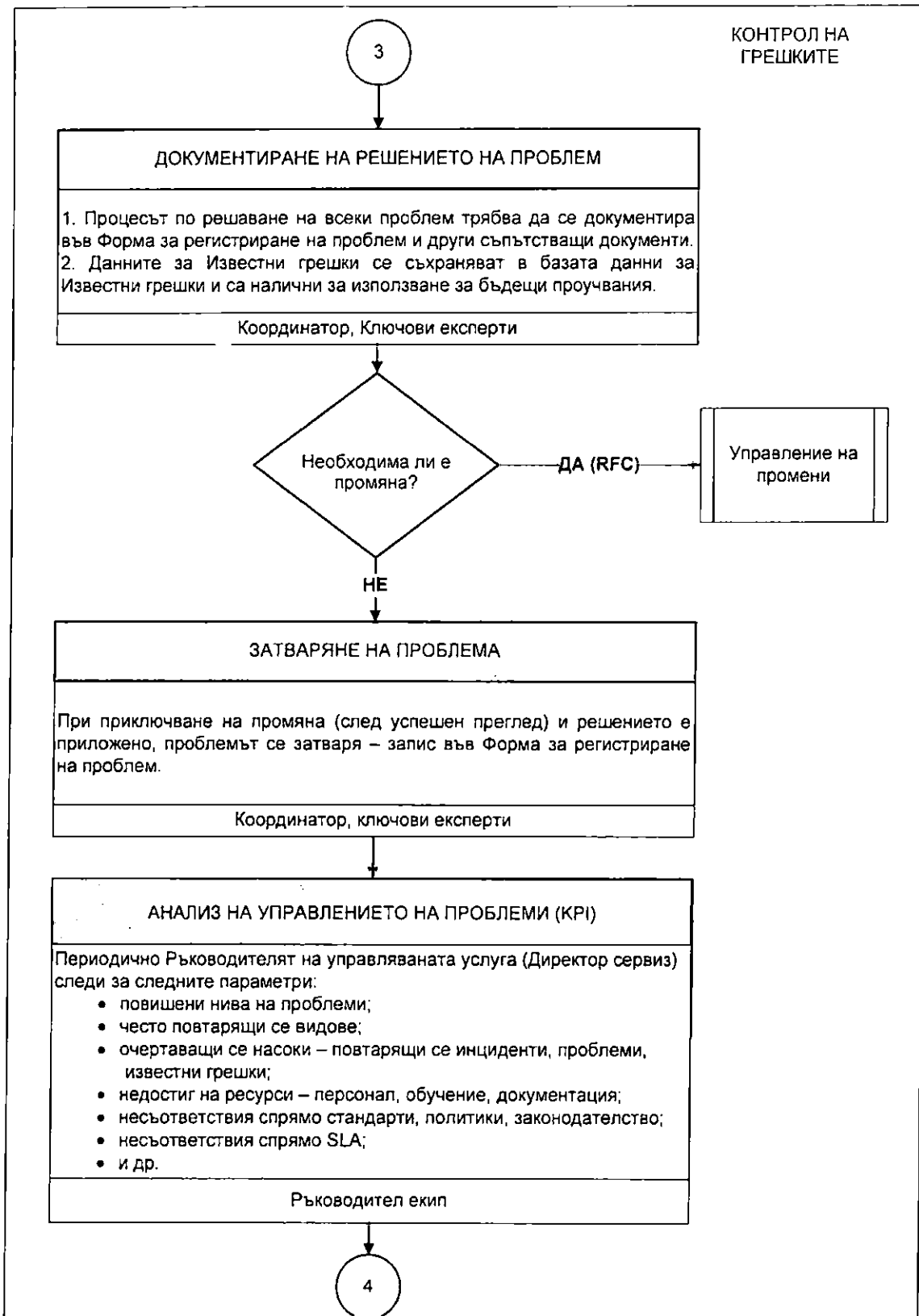
ОЦЕНКА НА ГРЕШКА

1. Извършване на първоначална оценка на средствата за решаване на Грешката заедно с други специалисти.
2. Финалните етапи от решаването на Проблем:
 - анализ на влиянието;
 - детайлна оценка на действията, които да бъдат предприети;
3. Промяна на част от грешката и тестване на промяна са под контрола на Управление на промени.
4. При оценка на грешка може да бъде установено дали има уязвимост в защитата, като при наличието на такава, Проблемът получава по-висок приоритет и се предприемат съответните действия за отстраняване на пробива в допустимия приемлив срок.

Координатор, Ключови експерти

3



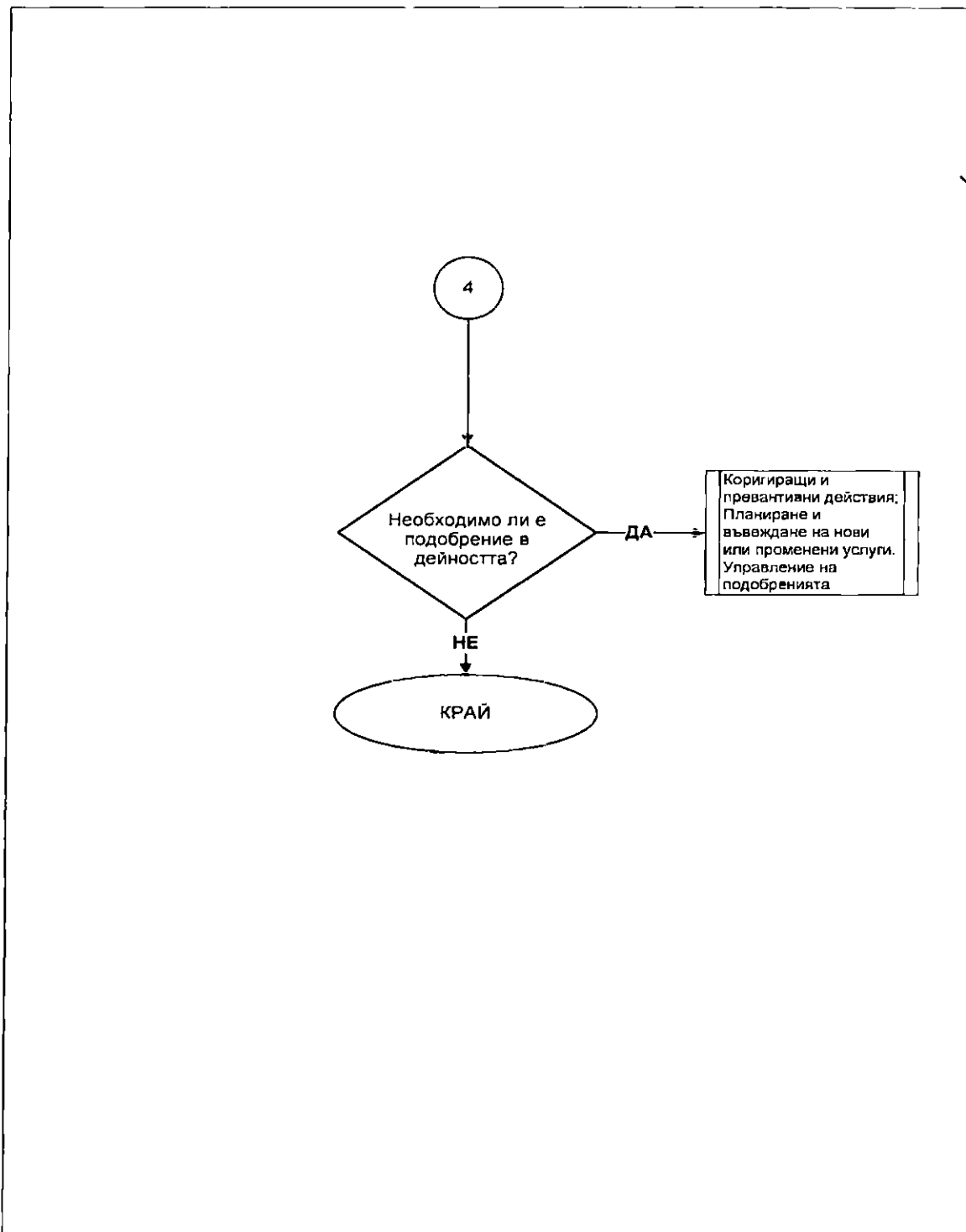


[Handwritten signature]

[Handwritten signature]





[Handwritten signature] 233



4.4. Система за регистриране и проследяване на инцидентите СПРИ (портал за поддръжка)

По време на гаранционния период, Изпълнителят ще осигури използването на уеб базирана система за регистриране и проследяване на дефектите, която да се ползва съвместно от представители на Изпълнителя и Възложителя.

След окончателното приемане на системата започва фазата на гаранционно обслужване. За по-ефективно управление по време на гаранционния период ще бъде предложена организация и начин на работа, базирани на Система за проследяване и регистриране на инциденти.

Ползите от внедряването и прилагането на портал за поддръжка са:

- Предоставяне на единна точка за контакт с Възложителя;
- Улесняване на възстановяването на работата със системата с цел минимално негативно влияние върху работата на Възложителя;
- Наличие на информация за възникналите инциденти и подпомагане на изграждането на капацитет при Възложителя за разрешаване на най-леките случаи;
- Наличие на информация от второ ниво поддръжка за начина на разрешаване на инциденти и при следващото им проявление служители на Възложителя да имат знанията за разрешаването им още на първо ниво на поддръжка;
- Възможност за проследяване и подпомагане на установяването на обща практика за работа със системата в рамките на цялата организация;
- Проследяване и анализиране на нуждите от обучение на персонала, ползващ системата;
- Възможност за регистриране на коментари - ще има възможност да се добавят всякакви коментари относно процеса на разрешаване на проблема;
- Възможност за ескалация на инцидента - когато проблема остане неразрешен дълго време или изисква незабавно решение ще бъде възможно инцидентът да бъде ескалиран с по-висок приоритет;
- Възможност за обратна връзка с крайния потребител.

Чрез портала за поддръжка ще се отговаря на всички запитвания свързани със софтуерни проблеми на ИС.

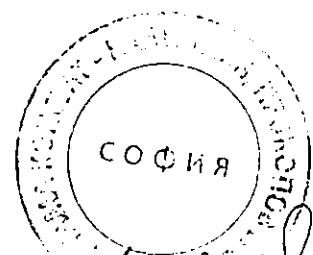
Целите на портала за поддръжка са да предостави единна точка за контакт за крайните потребители, което ще улесни значително процеса за приемане и регистриране на инцидентите със системата. Регистрацията на инцидентите се извършва от първо ниво на поддръжка. Инциденти, които не могат да бъдат разрешени бързо от първо ниво на поддръжка ще бъдат пренасочвани към групата за второ ниво на поддръжка за по-задълбочено диагностициране и разрешаването им. Докато трае разрешаването на проблема, ролята на СПРИ ще бъде да държи Възложителя информиран, относно статуса на неговата заявка или да му помогне с временно решение, което да позволи продължаването на работата. Информиранието на крайния потребител, относно статуса на заявката е задължение на първо ниво на поддръжка. Задължение на второ ниво на поддръжка е да информира първо ниво на поддръжка за актуалния статус на докладваните инциденти.



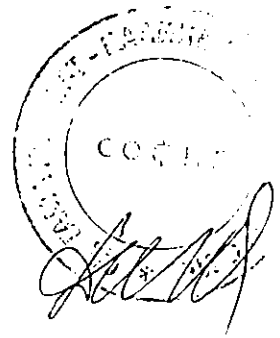
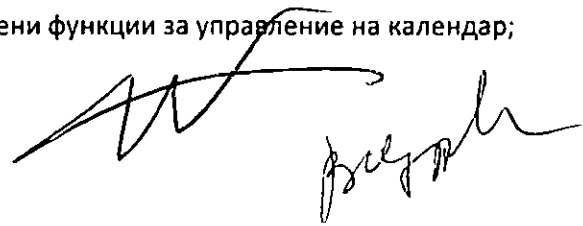
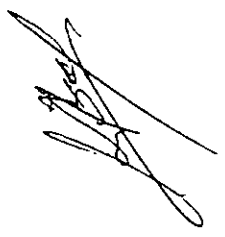
Система за регистриране и проследяване на инцидентите (СПРИ) ще бъде реализирана на базата на OTRS (Open Technology Real Services) – гъвкав програмен продукт за управление. OTRS е един от най-успешните и дълготрайни проекти в областта на управлението на услуги и поддръжка в световен мащаб.

OTRS предлага следните основни функционалности:

- Регистриране на заявки за инциденти (Ticket):
 - Канали за постъпване на инциденти:
 - Потребителски портал;
 - Електронна поща;



- Телефон;
- Факс;
- Регистрираните инциденти се появяват в т. нар. „Team-Postboxes“ – опашки;
- Управление на заявки (Ticket Management):
 - Приоритизация на насочване на заявки;
 - Предаване на заявки и последващи действия по тях;
 - Каталог на услугите;
 - OTRS::ITSM CMDB – база данни за управление на конфигурациите;
 - OTRS::ITSM управление на конфигурации;
 - Разделяне и обединяване на заявки;
 - Групова обработка на заявки;
 - Свързване на заявки;
 - Конфигурируеми нотификация по заявки;
 - Шаблони за стандартни текстове;
 - Подписи;
 - Бележки;
 - Прикачени файлове;
- Управление на база знания и самообслужване:
 - Често задавани въпроси (FAQ) / база знания;
 - Портал за потребители;
 - Анкети;
 - Информационен център за потребители;
- Автоматизация и процеси:
 - Автоматични нотификации;
 - Индивидуални информационни полета в заявките;
 - Функционалност за взаимовръзки (Master/Slave);
 - OTRS управление на процеси;
 - OTRS::ITSM управление на промени;
 - Шаблони за процеси;
- Управление на времето:
 - Разширени функции за управление на календар;



- Задаване на работно време;
- Отчитане на времето;
- Времена за разрешаване и напомняне;
- Управление на нивото на обслужване;
- Ескалации;
- Интеграция:
 - Интерфейс с широко приложение чрез SOAP+REST;
 - Системен мониторинг;
 - Стандартни възможности за интеграция със SAP, baramundi, BMC и др.
 - XSLT съответствие.
- Отчетност:
 - Генериране на статистики;
 - Експорт в CSV/PDF;
 - Визуализация на контролни табла (Dashboards);
- И други.

Система за регистриране и проследяване на инцидентите ще бъде публикувана и ще се използва на интернет-страницата на Изпълнителя.

4.5. Система за автоматизиран мониторинг Zabbix 3.2

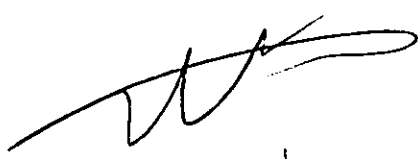
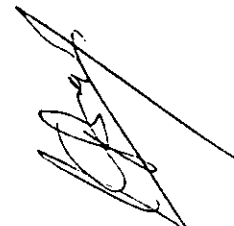
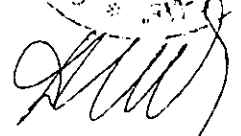
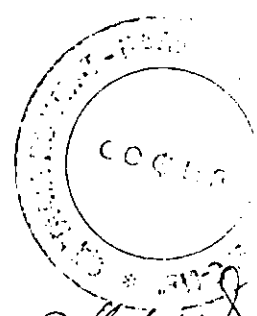
За осигуряване работоспособността на Националния EUCISE2020 Adapter, Изпълнителят ще използва специализирано решение за извършване на мониторинг на инсталираните софтуерни приложения на базата на системата **Zabbix 3.2**.

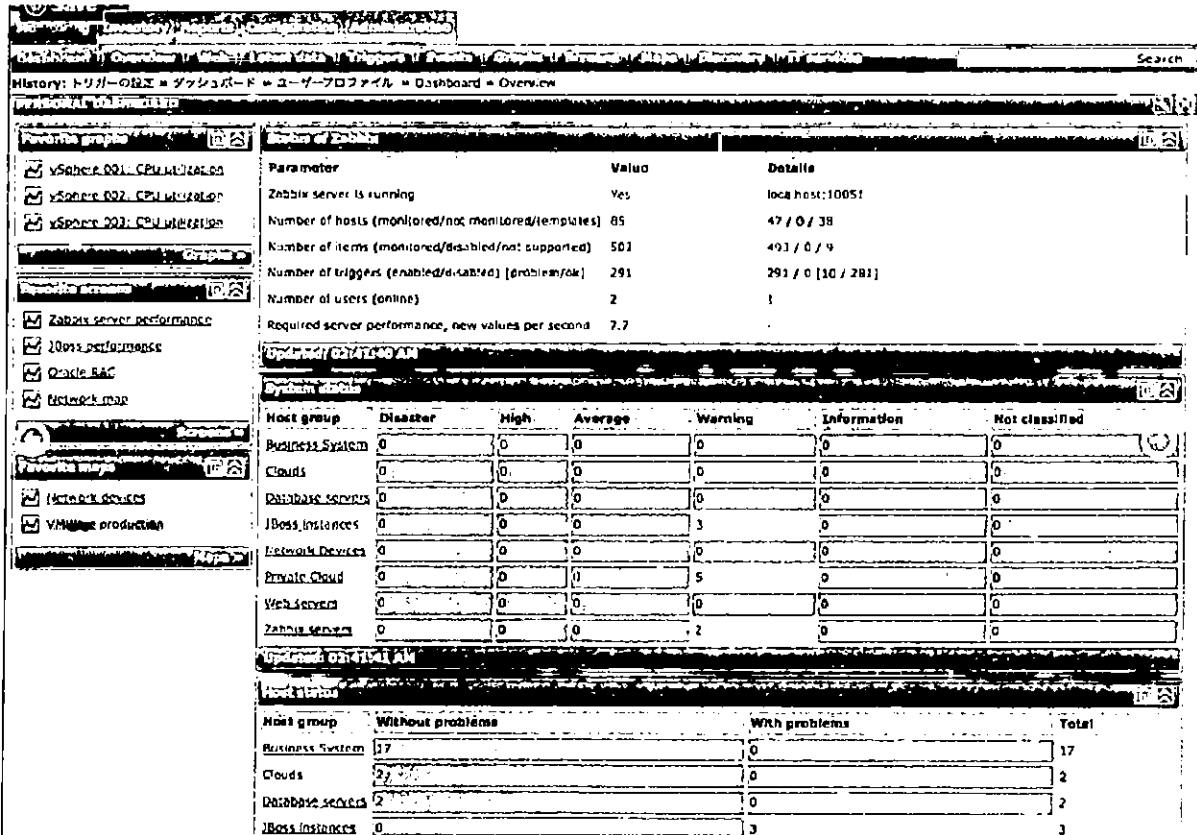
Zabbix 3.2 е мрежови инструмент. Това е програма, която е предназначена да предложи възможности за мониторинг на мрежа, като е специално създадена за ползване в големи и малки предприятия. Основната му функция е да контролира IT инфраструктурни компоненти и да събира данни от мрежови устройства и виртуални машини и услуги. Освен визуализиране на данните, **Zabbix 3.2** също е в състояние да предоставя карти, графики и екрани, които съдържат цялата информация, от която има нужда, за пълна яснота с нивото на производителност на мрежата.

Системата притежава следните функционалности:

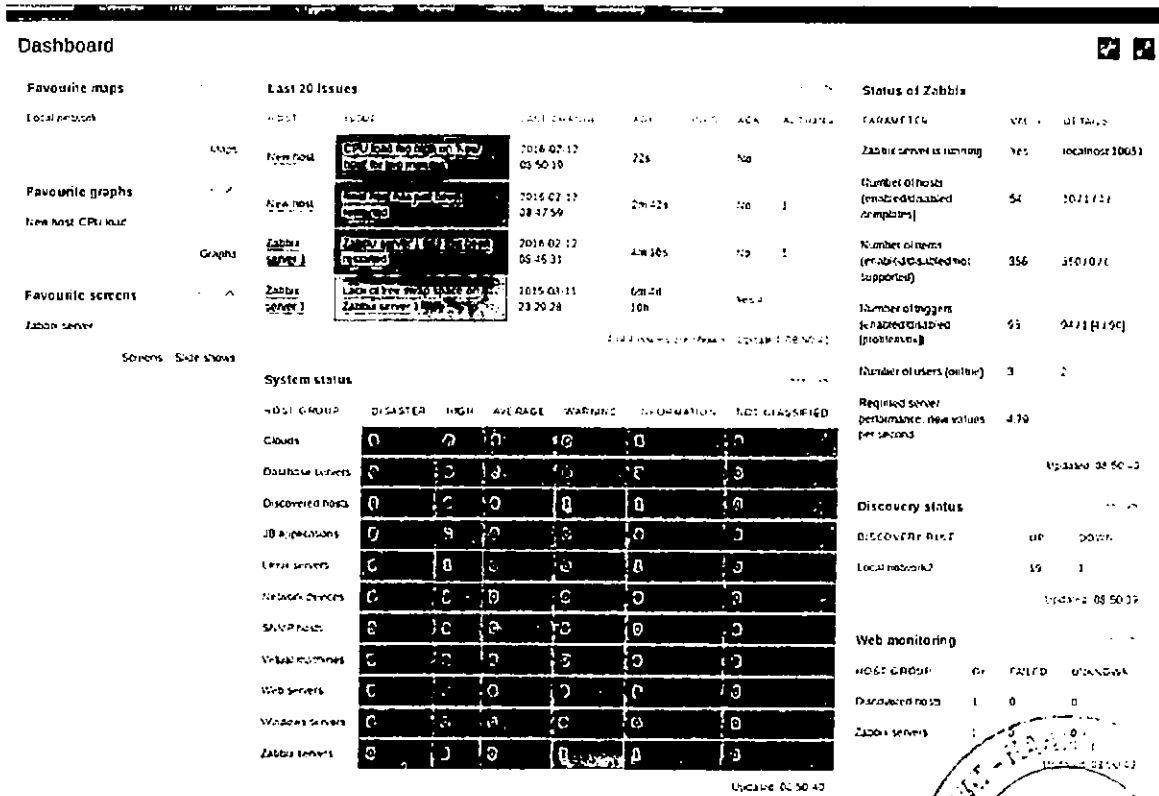
- наблюдение на мрежови системи, програмни модули и комуникации (SMTP, POP3, HTTP, NNTP, ICMP, SNMP, FTP, SSH);
- наблюдение на сървъри (напр. processor load, disk usage, system logs) за различни операционни системи, вкл. предложените, чрез модули за наблюдение (monitoring agents);
- Наблюдение на работни станции (наличност, достъпност, processor load, disk usage, system logs) за различни операционни системи, вкл. предложените, чрез модули за наблюдение (monitoring agents);
- Репорт за наличност, възможност за дефиниране на "maitanance mode"

- отдалечено наблюдение чрез криптирани връзки (remote monitoring through ssh/ssl encrypted tunnels);
- способност за дефиниране на допълнителни проверки и наблюдения чрез различни средства и езици; автоматични уведомления чрез и-мейл, sms и др.;
- автоматична ротация на логовете;
- веб интерфейс за наблюдение на текущия статус, логове и др.

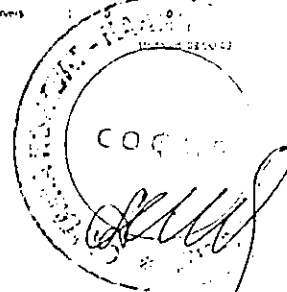


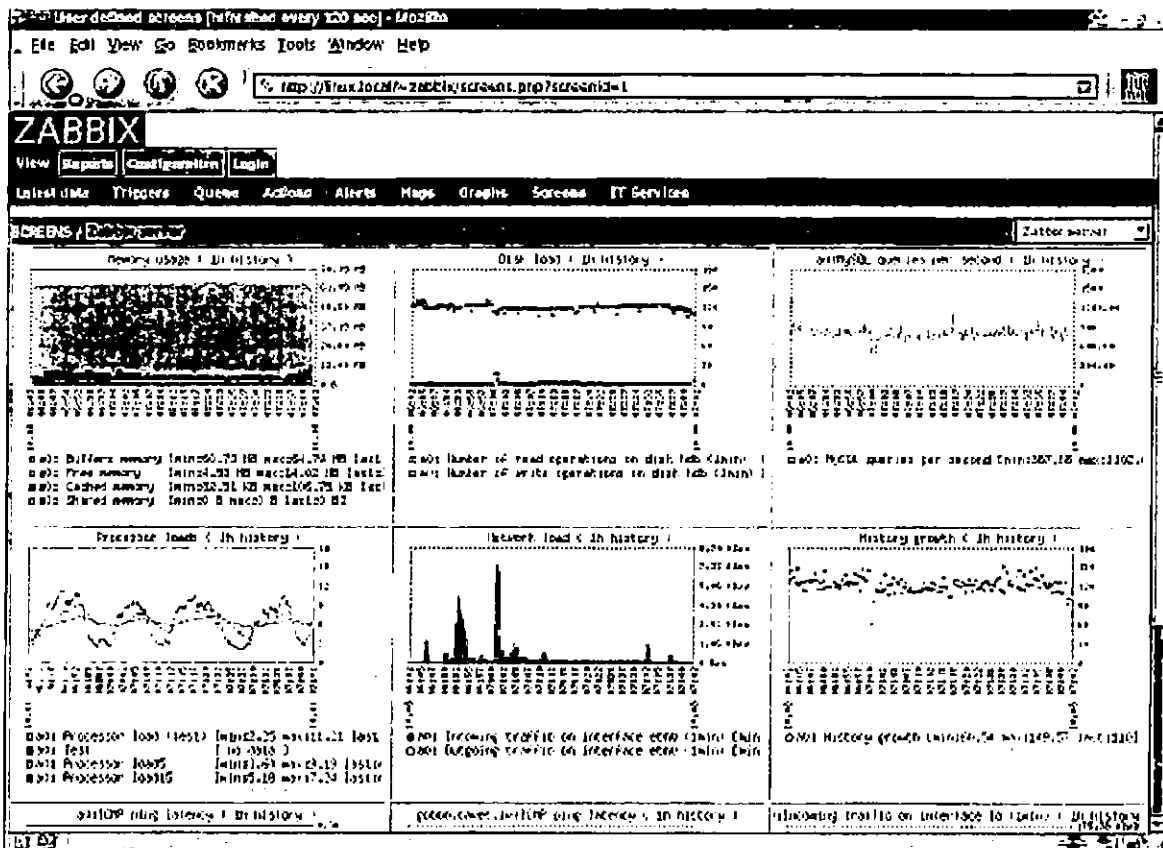
Фиг.: Пример Zabbix



Фиг.: Пример Zabbix

Handwritten signature





[Handwritten signatures and scribbles on the right side of the page]

Фиг. : Пример Zabbix

4.6. Предпоставки за успешна поддръжка от страна на Възложителя

- Възложителят осигурява нужните персонал и организация за осъществяване на първо ниво на поддръжка. За целта е необходимо служителите на Възложителя да са преминали обучение за работа с Адаптера.
- Възложителят е отговорен за оперативното наблюдение и администриране на системата.
- Взаимодействията с екипите на трети страни - външни доставчици на софтуерни или инфраструктурни компоненти от общата архитектура и услуги, по силата на съществуващи договори на Възложителя, както и на бъдещи такива се координират и контролират от Възложителя.

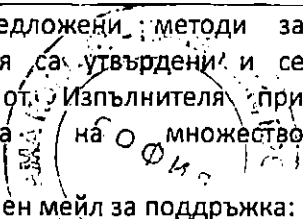
4.7. Очаквани резултати от изпълнението на дейностите по отстраняване на евентуални недостатъци, открити при тестовите

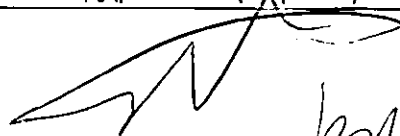
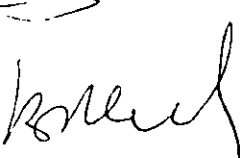
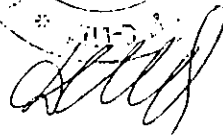
- Актуализиран софтуер за национален адаптер за EUCISE2020 в следствие на коригирани на установени при тестването недостатъци
- Програмен код на актуализирания софтуер за Национален адаптер за EUCISE2020
- Актуализирани документи – спецификации, ръководства и др.

5. Обосновка за приложимостта и полезността на предложеното решение

[Handwritten signature]



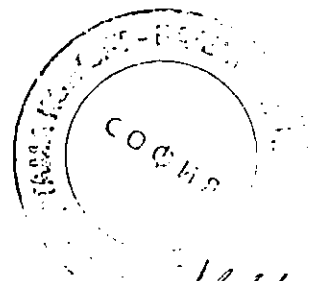
Област	Полезност	Приложимост
Начин на организация	Предложеният начин на организация на поддръжката следва най-добрите доказани световни практики в областта – ITIL (IT Infrastructure Library). Инцидентите се разрешават възможно най-бързо и на оптималното ниво на поддръжка. Стремешт е към непрекъснато подобряване на системата чрез трайно отстраняване на причините за инцидентите. Включването на екип на Възложителя в първото ниво на поддръжка осигурява бърза реакция и разбиране на бизнес-проблемите, а също и дава по-добро разбиране на Възложителя за функционирането на собствената му система. Включването на инженерните звена, които са проектирали и реализирали системата в третото ниво на поддръжка осигурява възможно най-компетентна реакция.	Предложеният начин на организация изцяло съпада с организацията на поддръжката при Изпълнителя и с възможностите и нуждите на Възложителя. Първо ниво на поддръжка се извършва от определени ключови потребители и ИТ екипа на Възложителя. Второ ниво на поддръжка се извършва от звеното за поддръжка на Изпълнителя. Трето ниво на поддръжка се извършва от инженерните звена на Изпълнителя.
Дейности	Дистанционното разрешаване на инцидентите осигурява по-кратки времена за реакция и отстраняване на дефектите. Проактивните действия по подобряване на системата намаляват броя и сложността на инцидентите. Поддържането на база данни с известни грешки/проблеми улеснява по-бързото разрешаване на инцидентите на по-ниско ниво на поддръжка.	Преди която и да е част от системата да бъде приета, Изпълнителят ще демонстрира с помощта на изчерпателен набор от изпитвания и операции, че всички части от софтуера функционират задоволително и в съответствие с предоставената техническа документация. В тази връзка Изпълнителят ще предостави тестове както за отделните компоненти, така и за системата като цяло. Предложените дейности са утвърдена практика при изпълнението на поддръжки. Дейностите са технически и ресурсно обезпечени. Няма нормативни пречки за осъществяване на дейностите.
Процедури	Предложените процедури са част от сертифицираната и действаща при Изпълнителя система за управление на услугите ISO 20 000. Процедурите са базирани на ITIL.	Процедурите са действащи и интегрирани в организацията на Изпълнителя.
Комуникация	Предложили сме на практика всички приложими методи за комуникация: портал за поддръжка, електронна поща и телефон. От страна на изпълнителя е организирана единна точка за контакт – бюро за поддръжка (helpdesk). В	Всички предложени методи за комуникация са утвърдени и се използват от Изпълнителя при поддръжката на множество клиенти: <ul style="list-style-type: none"> • Единен мейл за поддръжка: 




42
241

Област	Полезност	Приложимост
	<p>системата за регистриране и проследяване на инцидентите работи функционално, която автоматично регистрира съобщения за инциденти, получени по електронна поща. Всички заявки за инциденти ще бъдат регистрирани, включително и получените по телефона. Това осигурява проследимост и пълнота на данните за поддръжката.</p> <p>Последователното ескалиране между нивата за поддръжка осигурява правилното изпълнение на процесите по поддръжката.</p>	<p>support@stemo.bg;</p> <ul style="list-style-type: none"> • Поддържа национална телефонна линия за поддръжка: 0700 17 978; • Използва световно-утвърден софтуер за регистриране и проследяване на инциденти – OTRS.

[Handwritten signature]

[Handwritten signature]



[Handwritten signature]

[Handwritten signature]

ДЕКЛАРАЦИЯ за конфиденциалност по чл.102, ал.1 и 2 от ЗОП

Долуподписаният Деян Прокопов, с л.к. № издадена на
(собствено, бащино и фамилно име)

от: с постоянен адрес: На осн. Чл.2, ал.1 от ЗЗЗТ
в качеството си на Управител (длъжност) на Гама Консулт - Калинкин, Прокопов и с-ие
СД (наименование на участника) със седалище и адрес на управление гр.София п.к.1574, бул.
Шипченски проход 63, ет. 2, вписано в Търговския регистър с ЕИК 121562456, тел.:
02/447-6000, факс: 02/447-6099, e-mail: gammaconsult@gammaconsult.com и адрес за
кореспонденция гр.София п.к.1574, бул. Шипченски проход 63, ет. 2;

ДЕКЛАРИРАМ, ЧЕ:

Във връзка с участието ни в публично състезание за възлагане на обществена поръчка с предмет: „Дизайн, разработване, внедряване и верификация на EUCISE2020 Адаптер в съответствие с инженеринговите услуги, поддържащи прилагането на EUCISE2020 мрежата и съпровождащи услуги по време на демонстрационната фаза на обмен на данни“:

1. Информацията, съдържаща се в Техническо предложение (посочва се конкретната част/части) на офертата ни, да се счита за конфиденциална, тъй като съдържа търговска тайна.

2. Не бихме желали информацията по т.1 да бъде разкривана от възложителя, освен в предвидените от закона случаи.

Известна ми е отговорността по чл.313 от Наказателния кодекс на Република България за неверни данни.

Дата: 09/03/2018г

Подпис:

.....

Деян Прокопов

(име и фамилия на законния представител на участника или надлежно упълномощено лице)



Указания към попълване на Образец № 2:

1. Декларацията за конфиденциалност по Образец № 2 не е задължителна част и се прилага по преценка на участника и при наличието на основания за това към Техническото предложение.

2. Образец № 2 се подписва от законния представител на участника или надлежно упълномощено лице.

3. Ако участникът е обединение, Образец № 2 се представя от името на обединението участник и се подписва от партньора, който представлява

 EUCISE2020 has received funding from the European Union's seventh framework programme under grant agreement no: 608385



Handwritten signature

243

обединението за целите на обществената поръчка или надлежно упълномощено лице.

4. Участниците не могат да се позовават на конфиденциалност по отношение на ценовите предложения от офертите им.

