



EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND
INVESTING IN YOUR FUTURE!



GOVERNMENT OF ROMANIA



GOVERNMENT OF BULGARIA



Образец № 3

ДО
МИНИСТЪРСТВО НА ТРАНСПОРТА,
ИНФОРМАЦИОННИТЕ ТЕХНОЛОГИИ И СЪОБЩЕНИЯТА

ПРЕДЛОЖЕНИЕ ЗА ИЗПЪЛНЕНИЕ

на открита процедура за възлагане на обществена поръчка с предмет:
**„Доставка на интегрирана система за корабна инспекция – софтуерен пакет
с електронна база данни и оборудване“**

От СТЕМО ООД, седалище и адрес на управление: гр. Габрово 5300, ул. „Николаевска“ 48, адрес за кореспонденция: гр. София 1407, бул. „Черни връх“ 51Б, телефон: +359 2 816 2300, факс: +359 2 816 2303, e-mail sf.office@stemo.bg ЕИК 817080126, обслужваща банка: Уникредит Булбанк, IBAN BG48UNCR76301053899623, BIC: UNCRBGSF, лице за контакти Александър *Ил. осн. Чл. 2, ал. 1 от 33/17* Ризов, длъжност: акаунт мениджър корпоративни продажби и пълномощник на управителя на СТЕМО ООД, телефон / факс: телефон: +359 2 816 2300, факс: +359 2 816 2303

Участникът се представлява от: Христо Ковачев, Стефан Морев и Георги Тихов – управители и Александър Ризов - пълномощник

УВАЖЕМИ ГОСПОДИН ИЗПЪЛНИТЕЛЕН ДИРЕКТОР,

1. Заявявам, че представляваното от мен дружество/участник желае да участва, при обявените условия, при възлагане на обществена поръчка с горепосочения предмет.

2. Направените в настоящата оферта предложения и поетите с нея задължения са валидни за срок от 4 (четири) месеца, считано от крайния срок за получаване на офертите.

3. Изпълнението на обществената поръчка ще извършим при спазване на приложимите за обекта на същата технически изисквания и параметри, заложиени в Техническата спецификация и документацията за обществена поръчка, действащите в Европейския съюз и Република България норми, стандарти, споразумения, конвенции, директиви и др. изисквания, регламентиращи дейността, а именно:

СТЕМО ООД предлага изпълнение на поръчката, което включва:

- Предложение за технологично решение и функционалности – Приложение 1
- Предложение за организация за изпълнение и управление на поръчката – Приложение 2



000004



[Handwritten signature]



- Предложение за управление на риска при изпълнение на поръчката – Приложение 3
- Подробен график, в който следва да се конкретизират сроковете за изпълнение на всяка дейност от настоящата поръчка – Приложение 4
- Таблица за съответствие на предлаганата техника – Приложение 5

ДЕКЛАРИРАМ, ЧЕ:

- Предложението за изпълнение на Дейност 1 и Дейност 2 от поръчката отговаря в пълен обем на изискванията на Техническата спецификация.
- Представяваният от мен участник приема и се съгласява безусловно с клаузите на проекта на договор за обществена поръчка с предмет: „Доставка на интегрирана система за корабна инспекция – софтуерен пакет с електронна база данни и оборудване”
- Срокът на валидност на подадената от нас оферта е 4 (четири) месеца, считано от датата, определена в обявлението за краен срок за получаване на офертите.
- Действителен собственик по смисъла на чл. 6, ал. 2 от Закона за мерките срещу изпирането на пари във връзка с чл. 3, ал. 5 от Правилника за прилагане на Закона за мерките срещу изпирането на пари на горепосоченото юридическо лице са следните физически лица:

1. Христо Ковачев, ЕГН: *На осн. Чл. 2, ал. 1 от ЗЗПН*, лична карта № *На осн. Чл. 2, ал. 1 от ЗЗПН* издадена на *На осн. Чл. 2, ал. 1 от ЗЗПН* адрес: *На осн. Чл. 2, ал. 1 от ЗЗПН*

2. Стефан Морозов, ЕГН: *На осн. Чл. 2, ал. 1 от ЗЗПН*, лична карта № *На осн. Чл. 2, ал. 1 от ЗЗПН* издадена на *На осн. Чл. 2, ал. 1 от ЗЗПН* адрес: *На осн. Чл. 2, ал. 1 от ЗЗПН*

3. Георги Тихов, ЕГН: *На осн. Чл. 2, ал. 1 от ЗЗПН* лична карта № *На осн. Чл. 2, ал. 1 от ЗЗПН* издадена на *На осн. Чл. 2, ал. 1 от ЗЗПН* адрес: *На осн. Чл. 2, ал. 1 от ЗЗПН*

4. Иван Божинев, ЕГН: *На осн. Чл. 2, ал. 1 от ЗЗПН* лична карта: *На осн. Чл. 2, ал. 1 от ЗЗПН* издадена на *На осн. Чл. 2, ал. 1 от ЗЗПН* адрес: *На осн. Чл. 2, ал. 1 от ЗЗПН*

Известна ми е наказателната отговорност по чл. 313 от Наказателния кодекс за деклариране на неверни обстоятелства.

II. Дейности по изпълнение.

Декларираме, че ще изпълним следните дейности от предмета на обществената поръчка – доставка на интегрирана система за корабна инспекция – софтуерен пакет с електронна база данни и оборудване, както следва:

Дейност 1: Доставка, адаптиране и внедряване на ИИС за инспекция на кораби
Съгласно Приложения от 1 до 5 от настоящото техническо предложение

Дейност 2: Доставка, инсталиране и конфигуриране на ИКТ инфраструктура за ИИС за инспекция на кораби

Съгласно Приложения от 1 до 5 от настоящото техническо предложение.



1. Всички елементи от доставката ще бъдат нови и неупотребявани, притежаващи валиден сертификат за произход и качество или техен еквивалент.
2. Гаранционният срок на доставените от нас Интегрирана система за инспекция на кораби и оборудване започва да тече от датата на фактурата за извършената от нас доставка.
3. В случай че доставен от нас софтуерен пакет или оборудване имат дефект, който пречи за нормалната им експлоатация, се задължаваме да подменим и монтираме същите с нови, изцяло за собствена сметка.
4. Гаранционният срок за доставяните от нас софтуерен пакет и оборудване е:

No.	Артикул	Гаранционен срок
1	Интергрирана система за корабна инспекция	24 месеца
2	Блейд шаси	36 месеца
3	Блейд сървър	36 месеца
4	Споделен сторидж	36 месеца
5	Непрекъсваемо токоподаващо захранване	24 месеца
6	Шкаф за техника	24 месеца
7	Тауър сървър	36 месеца
8	Видеоконферентен терминал	36 месеца
9	Дисплей	24 месеца
10	Мултифункционално устройство	24 месеца
11	Преносим компютър	36 месеца
12	Охладителна система	36 месеца

Посочения гаранционен срок започва да тече от датата на подписване на двустранен протокол за извършената дейност.

5. В случай че в срока на гаранцията възникне повреда, дължаща се на лошо качество на доставения софтуерен пакет или оборудване, се задължаваме да отстраним същата изцяло за собствена сметка.

6. При несъответствие на посочените в офертата с цифри стойности и тяхното изписване с думи, обвързващо за нас е предложението, посочено с думи.

Дата
Наименование на участника
Име и фамилия на представителя на участника
Длъжност *На осн. Чл.2.ал.1 от ЗЗЛП*

25.06.2018 г.
СТЕМО ООД
Георги Тихов
Управител на СТЕМО ООД

Подпис



към

ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

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

**„Доставка на интегрирана система за корабна инспекция –
софтуерен пакет с електронна база данни и оборудване“**

Предложение за технологично решение и
функционалности

Съдържание

1. Въведение.....	4
2. Архитектура на системата	4
2.1. Принципи и добри практики за изграждане и внедряване на системата.....	4
2.2. Софтуерна архитектура	6
2.2.1. Диаграма	6
2.2.2. Описание на елементите и взаимовръзките	8
2.2.3. Подход за интеграция с външни системи	12
2.3. Технологии и средства за разработка.....	12
2.4. Системна архитектура	12
2.5. Базов софтуер.....	19
2.5.1. Изисквания към СУБД	19
2.5.2. Изисквания към сървъра за приложения.....	29
2.6. Обосновка за предложената архитектура.....	38
2.6.1. Относно софтуерната архитектура.....	38
2.6.2. Относно системната архитектура.....	40
2.6.3. Относно базовия софтуер	42
3. Общи нефункционални характеристики по НОИИСРЕАУ.....	45
4. Функционални характеристики.....	65
4.1. Диаграма на функционалностите.....	65
4.2. Основни функционалности	66
4.2.1. Общи функционални изисквания към ИС за корабни инспекции	66
4.2.2. Модул Обединен списък на корабите.....	69
4.2.3. Модул Корабни инспекции и формалности	70
4.3. Модул Управление и статистически данни	76
4.4. Обмен на данни с други системи	77
4.5. Допълнителни функционалности.....	78
4.5.1. Допълнителна функционалност 1: Доклад от извършена проверка.....	78
4.5.2. Допълнителна функционалност 2: Рисков профил на кораба	80
4.5.3. Допълнителна функционалност 3: Уведомяване за лимбирания	81
5. Подход за бизнес анализ.....	83
5.1. Методика за бизнес анализ	83
5.2. Планиране на анализа.....	87

5.3.	Извличане и документиране на изискванията	88
5.4.	Инструменти, които ще се използват за извършване на функционално и техническо специфициране	93
5.4.1.	Моделиране с UML.....	94
5.4.2.	Софтуер за моделиране на UML – Sparx Systems Enterprise Architect	103
5.5.	Съдържание на Спецификация на изискванията.....	104
5.6.	Проследяване на изисквания	105
5.7.	Обосновка за предложените подход и инструменти за бизнес анализ.....	106
5.8.	Подход за прилагане на continuous integration практики.....	106
5.9.	Dependency management инструмент.....	110
6.	Гаранционна поддръжка.....	111



1. Въведение


Настоящото приложение представя визията на Кандидата за реализиране на функционалните и технически характеристики на системата. В етапа на дизайн те ще бъдат прецизирани, детайлизирани и съгласувани с Възложителя.


2. Архитектура на системата

2.1. Принципи и добри практики за изграждане и внедряване на системата

При проектиране на системата ще бъдат анализирани всички възможности за дефиниране на обобщен и конфигурируем модел на бизнес процесите. Това ще позволи информационната система да бъде разработена като гъвкава и лесно адаптивна към законодателни, административни, структурни или организационни промени, водещи до промени в работните процеси.

Системата ще бъде проектирана и разработена като отделни модули, обменящи бизнес информация през Уеб сървиси. Всяка от услугите ще бъде дефинирана така, че да отговаря на следните условия:

- Всяка услуга изпълнява някаква бизнес дейност с определен резултат;
 - Всяка услуга е самостоятелна;
 - Начинът на реализация на бизнес логиката във всяка услуга е недостъпен за консуматорите ѝ;
 - При реализацията си всяка услуга може да ползва други базови услуги.
- 



По този начин ще бъде възможно лесното надграждане на системата с допълнителни модули, бъдещи разширения и интеграции с други системи.

За всяка от интеграциите с външни регистри и при наличие на предоставени от тяхна страна Уеб сървиси за интеграция, обменът на данни през тях ще бъде предпочетен с цел постигане на автоматизиран обмен без необходимост от ръчен процес по синхронизация и обмен.

Комуникацията с румънския компонент на системата ще се осъществява през Уеб услуги. По този начин ще може да се постигне максимално автоматизиран трансфер на информация между двата компонента. От ключово значение за постигане на ефективен обмен ще бъде съвместното проектиране на протокола за обмен и дизайна на обменяните бизнес обекти.

Обменът през Уеб услуги с всички външни системи (регистри, румънски компонент и други) ще бъде реализиран по защитен HTTPS протокол като криптирането ще се базира на сигурен сертификат с валидирана идентичност.

При изграждане на системата ще бъдат следвани всички добри практики за обектноориентирано програмиране и проектиране и използване на релационни бази от данни.

За реализация на системата ще бъдат използвани актуални и съвременни технологии, софтуерни компоненти и библиотеки, в максимална степен покриващи изискванията за производителност и сигурност.

Всички модули на системата ще бъдат разработени така, че да осигуряват надеждна защита от злонамерени опити за достъп до данните с цел неправомерно ползване, промяна, повреждане или унищожаване.

В системата ще бъдат разработени и заложиени механизми за контрол на натоварването и защита от DoS/DDoS атаки. На приложно ниво ще бъде имплементиран механизъм за "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.

Системата ще позволява конфигуриране от страна на администраторите на лимитите за отделни страници, уеб-услуги и ресурси, които се достъпват с отделен URL/URI на ниво конкретни автентикирани потребители (напр. системи на други администрации). Ще бъде предоставена възможност за генериране на справки и статистики за броя заявки по ресурси и услуги и за изпращане на подходящи нотификации при достигане на определени параметри на натоварването.

Архитектурата и всички софтуерни компоненти (системни и приложни) ще бъдат подбрани и/или разработени така, че:

- да осигуряват работоспособност и отказоустойчивост на Системата;
- да бъдат съобразени с реално очакваното натоварване и да осигуряват необходимото бързодействие на Системата;
- да бъдат платформено независими;
- да гарантират недискриминационно инсталиране, т.е. без различни условия за инсталиране върху физическа и виртуална инфраструктура.

Техническите изисквания на Възложителя дефинират широк обхват на проекта, без да поставят ограничения на Изпълнителя за концепцията за реализация на системата за корабна инспекция и предоставят възможност да се оферира хардуерна инфраструктура, която в максимална степен е съобразена със системните изискванията на разработвания софтуер.

Основен принцип при проектирането на хардуерната платформа е осигуряването на защита на направените от Възложителя инвестиции, чрез избор на оборудване и системен софтуер от производители, които са дългогодишни лидери в областта, с което се елиминира риска от прекратяване на поддръжката на даден продукт или изчезването му от продуктовото портфолио.

Архитектурата на виртуализационната среда предоставя гъвкавост и лесно управление на ресурсите, като същевременно осигурява възможност за бъдещо преместване на вече разработения софтуер върху други хардуерни ресурси. Това гарантира на Възложителя независимост от направения избор на хардуерна платформа, защото не го обвързва с решението на конкретен производител и позволява да се гарантира свободната конкуренция.

При дизайна на предложеното решение са взети предвид принципите на отвореност и модулност и в максимална степен се осигуряват предимствата от използването на софтуер с отворен код.

Избраните хардуерни продукти използват световно-признати стандарти и протоколи, съчетали добрите практики и десетки години опит и оптимизации, което осигурява отлична съвместимост с други решения на пазара и възможности за бъдещо разширение в синхрон с нуждите на Възложителя.

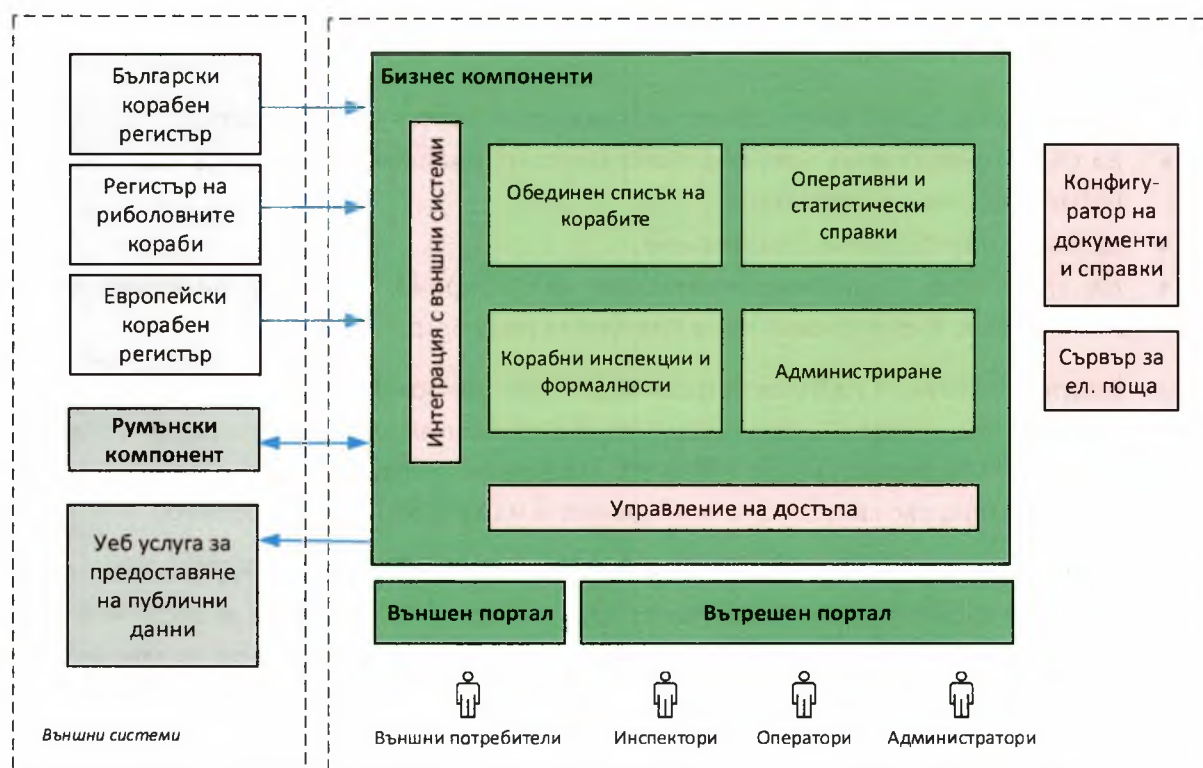
2.2. Софтуерна архитектура

В настоящият раздел е представено аргументирано описание на визията за софтуерна архитектура на системата, нейните компоненти и тяхната взаимовръзка. Посочени са мотивите за избора на предложената архитектура, обяснена е логиката на всяка от връзките и необходимостта от изграждането ѝ. Посочени са предимствата на архитектурата от гледна точка на производителност и ефективност на системата.

Софтуерната архитектура ще бъде прецизирана, детайлизирана и съгласувана с Възложителя в етапа на анализ и проектиране на системата.

2.2.1. Диаграма

На фигурата по-долу е представена софтуерната архитектура на Системата.



Системата ще бъде изградена така, че да отговаря на изискванията, представени в таблицата по-долу:

№ по ред	Изискване
NFREQ.1.	ИИС ще бъде изградена като централизирана Уеб базирана система, която ще обслужва както контролните органи, извършващи инспекции - служители на РМА и ИАМА, така и външни потребители - собственици и капитани на кораби.

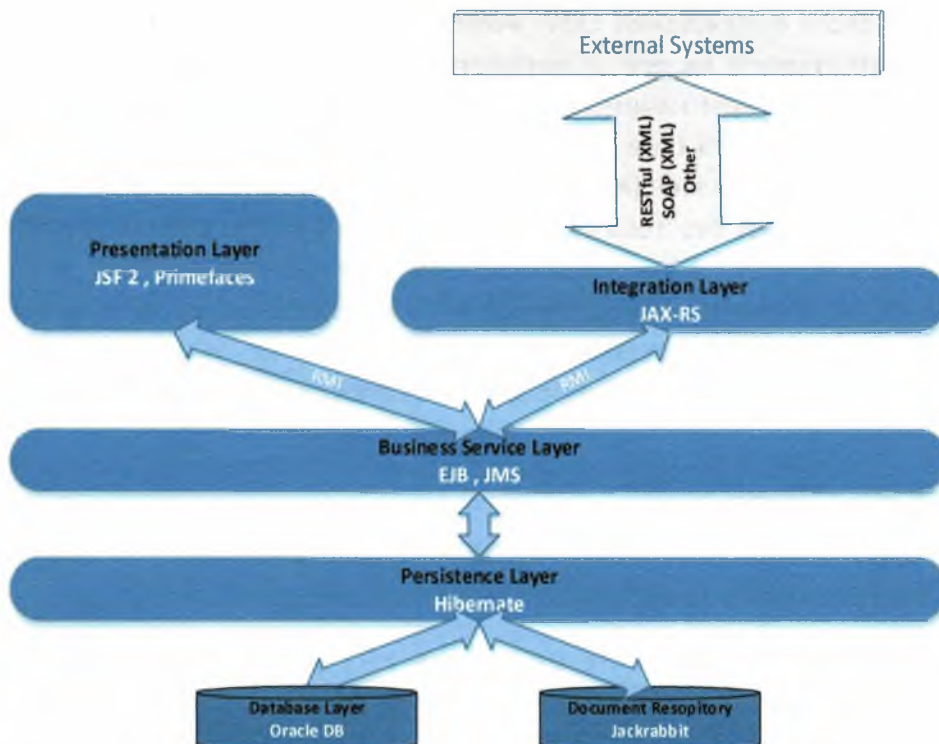
№ по ред	Изискване
NFREQ.2.	Външните потребители (собственици на кораби, капитани и други) ще имат достъп до предвидените за тяхната работа компоненти и функционалности на системата през външен портал.
NFREQ.3.	Всички вътрешни потребители на системата ще достъпват функционалността през вътрешен портал.
NFREQ.4.	<p>Потребителският интерфейс на всички модули на системата ще бъдат достъпвани посредством стандартен уеб браузер:</p> <ul style="list-style-type: none"> • Internet Explorer последна актуална версия; • Google Chrome последна актуална версия; • Mozilla Firefox последна актуална версия.
NFREQ.5.	Правата за достъп в системата ще бъдат базирани на роли, които се дават на потребители и/или групи потребители. Всеки потребител ще има достъп до компонентите и данните, за които има дадени права.
NFREQ.6.	<p>Идентификацията в системата се осъществява по някой от следните начини:</p> <ul style="list-style-type: none"> • с потребителско име и парола • през активна директория • с електронен подпис • чрез националната схема за електронна идентификация.
NFREQ.7.	<p>Системата ще поддържа и системен журнал за всички потребителски действия, както и версия на предишното състояние на данните, които той е променил. Като минимум в системния журнал при всеки запис ще се съхраняват следните атрибути:</p> <ul style="list-style-type: none"> • дата/час на действието; • модул на системата, в който се извършва действието; • действие; • обект, над който е извършено действието; • допълнителна информация; <p>IP адрес и браузър на потребителя.</p>
NFREQ.8.	<p>Системата за корабни инспекции ще съдържа следните обособени модули:</p> <ul style="list-style-type: none"> • Обобщен списък на корабите; • Корабни инспекции и формалности; • Оперативни и статистически справки; <p>Администриране.</p>
NFREQ.9.	Системата ще позволява потребители с определени права да променят шаблони на документи, съдържание на нотификации и да дефинират модели на оперативни и статистически справки без необходимост от допълнителна разработка.

№ по ред	Изискване
NFREQ.10.	Интеграциите с външни регистри (Български корабен регистър, Регистър на риболовните кораби и Европейски корабен регистър) ще бъдат реализирани преимуществено посредством уеб услуги, ако има предоставени такива или посредством импорт от структурирани файлове.
NFREQ.11.	В етапа на анализ ще бъдат дефинирани и в последствие разработени набор уеб услуги за комуникация с румънския компонент на системата. Интеграцията със системата от румънска страна при възможност ще бъде проектирана и координирана с екипа, разработващ тяхната част от система.
NFREQ.12.	За предоставяне на информация към външни потребители ще бъде разработена уеб услуга с публичен достъп. Данните предоставяни чрез тази услуга могат да бъдат само публични и ще бъдат прецизирани по време на етапа на анализ.
NFREQ.13.	Системата ще бъде интегрирана със сървъра за електронна поща на Възложителя за изпращане на нотификации към потребителите.

2.2.2. Описание на елементите и взаимовръзките

№ по ред	Изискване
NFREQ.14.	Системата за корабни инспекции ще бъде изградена на модулен принцип. Всеки от модулите на системата ще бъде изграден като многослойна Java EE базирана архитектура.
NFREQ.15.	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.

Софтуерната архитектура е представена на следната схема:



Database Layer

В базата данни се съхраняват всички структурирани бизнес и системни данни за оперативните, справочни и административни модули на системата. Всеки обект в системата има собствен уникален идентификатор. За всеки обект се пази история на промените като не се допуска физическо изтриване на данни.

Описание на реализацията:

Database слоя ще бъде реализиран с релационна база данни Oracle Database Standard Edition. Избраната база данни е с високата производителност, надеждност и сигурни механизми за защита на данните и възстановяване след срив.

Document Repository

Document Repository е слой, който ще съхранява и предоставя за ползване неструктурираната част от информацията в системата – документи, графики, изображения и други.

Данните могат да бъдат съхранявани йерархично с възможност за пълнотекстово търсене и поддържане на версии на документите.

Описание на реализацията:

Document Repository слоят ще бъде реализиран чрез инструмент с отворен код Jackrabbit на Apache. Проектът на Apache е напълно съвместим с избраната базова технология на системата като реализира изцяло дефинираният програмен интерфейс (API) за поддържане на хранилище за данни (JCR, специфицирано в JSR 170 и JSR 283).




Persistence Layer

Persistence слой е междинен слой между базата данни и Business Service слоя 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 слоя е по-доброто капсулиране на логиката на базата данни в един слой.


Описание на реализацията:

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

- 
- Посредством заложените механизми за кеширане в Hibernate се постига по-добра производителност при работа с базата данни;
 - Hibernate притежава вграден механизъм за поддържане на версии на данните и одит информация;
 - Hibernate 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 на всеки от модулите на системата ще бъде имплементирана съответно:

- 
- бизнес логиката на обобщения списък на корабите;
 - бизнес логиката на модула за корабни инспекции;
 - бизнес логиката на модула за предоставяне на статистически данни;
 - бизнес логиката по управление на интеграцията с румънския компонент;
 - бизнес логиката по управление на интеграциите с другите външни системи - Български корабен регистър, Регистъра на риболовните кораби, Европейски регистър на корабите (EHDB);
 - бизнес логиката по управление на потребителския достъп;
 - бизнес логиката на модула за управление на системата – конфигуриране и администриране;

Като минимум бизнес логиката обхваща процесите по прилагане на правилата по валидации и изчисления, свързани с данните и реализира правилното следване на дефинираните бизнес процеси. Всяко обръщение към този слой гарантира, че заявената услуга ще бъде изпълнена успешно чрез управление на транзакции, осигуряващи цялостно и консистентно имплементиране на бизнес логиката, независимо от броя и вида на паралелните извиквания.

При проектиране на системата ще се цели максимално параметризиране и възможност за настройка и промяна на параметрите през служебен (администраторски) потребителски интерфейс.

Описание на реализацията:

За реализация на Business Service слой на системата ще се използват стандартни Java EE библиотеки като EJB (Enterprise Java Beans), JMS (Java Message Service) и други. Комуникацията с Integration слоя и Service слоя ще се осъществява чрез Java RMI (Remote Method Invocation). Комуникацията между Business Service слоевете на различните модули на системата ще се осъществява чрез Java RMI (Remote Method Invocation) или Restfull уеб услуги, ако съответната услуга е необходимо да бъде дефинирана и като публична и достъпна за бъдещи интеграции и разширения на системата.

Integration Layer

Чрез Integration слой ще се осъществява комуникацията с:

- румънския компонент;
- Български корабен регистър;
- Регистъра на риболовните кораби;
- Европейски регистър на корабите (EHDB);
- Електронна поща

Integration слой ще бъде проектиран на модулен принцип като интеграцията с всеки външен източник (система) ще бъде обособена като отделен модул. Модулният принцип за изграждане на този слой ще позволи по-лесната поддръжка и управление на всяка от интеграциите с външни системи. По този начин действия като подмяната на модул с нова версия, спиране, пускане, рестартиране при установен проблем, преконфигуриране и др. няма да засяга нормалната работа на останалите модули, осъществяващи интеграции с външни системи.


Описание на реализацията:

За реализация на интеграцията с румънския компонент ще се използват RESTful уеб услуги, имплементирани с помощта на стандартната Java EE библиотека JAX-RS (Java API for RESTful Web Services).

За реализация на всяка от интеграциите с корабните регистри ще се използват RESTful или SOAP уеб услуги в зависимост от реализираните технически възможности за обмен със съответната външна система. В случай, че съответната външна система не предоставя автоматизиран интерфейс за обмен чрез стандартни уеб услуги, ще бъде разработен механизъм за импорт на данни през структурирани файлове.

Presentation Layer

За изграждане на Presentation слой ще бъдат използвани JSF 2 (JavaServer Faces) и Primefaces. JSF 2 е 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 е фреймуърк с



повече от 100 визуални компоненти за JSF 2 с отворен код и не изисква допълнително лицензиране.

2.2.3. Подход за интеграция с външни системи

За всяка от предвидените в системата интеграции ще бъде ползвана преимуществено възможността за напълно автоматизиран обмен през уеб услуги.

В случай, че съответната външна система не предоставя възможност за обмен през Уеб услуги, ще бъде разработен и приложен импорт през структурирани файлове от потребител със специални права.

При проектиране на комуникацията с румънския компонент на системата като механизъм за обмен ще бъде заложено разработването на RESTful сървиси, обменящи информация в XML формат. Реализацията на бизнес логиката на системата и правилата на протокола за обмен между двете системи ще цели максимално автоматизиран трансфер на информация по възможност в реално време. Предвид на това, че двете системи (българска и румънска) ще бъдат проектирани и изградени паралелно, съвместното проектиране на интеграцията ще бъде от ключово значение за постигане на ефективен обмен.

2.3. Технологии и средства за разработка



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

За приложен сървър ще бъде използва Oracle WebLogic Server Standard Edition.

За база данни ще бъде използван Oracle Database Standard Edition, поради високата си производителност, надеждност, сигурни механизми за защита на данните, репликация с гарантирано време за синхронизация, възстановяване след срив без загуба на данни.

Сървърната част от системата ще работи във виртуални машини, използващи VMware ESXi хипервайзори.

Операционните системи работещи във виртуални машини ще бъдат Linux базирани. Предлагаме да се използва CentOS Linux.

Свободните компоненти, които ще бъдат използвани за разработката на системата са JDK SE Development Kit 8.

2.4. Системна архитектура

При дизайна на хардуерната архитектура, са взети под внимание всички най-добри практики при изграждане на подобен тип решения. Решението се базира на доказалите своята надеждност в практиката сървъри на Hewlett Packard Enterprises, а именно HPE Proliant BL460 Gen9. За споделен дисков масив ще се ползва HPE MSA 2050 SAN Storage.

Подробно описание на системната конфигурацията с отделните ѝ компоненти, техните параметри и функционални характеристики, можете да намерите в документа „Таблица за съответствие“. Изображенията от фигурите, в настоящия документ са примерни и целят да представят хардуера и софтуера, които ще бъдат доставени, на концептуално ниво. Външният вид на оборудването може да се различава от това на схемите.

Handwritten mark

Разположението на техниката в сървърния шкаф е съобразено с евентуалните нужди от бъдещо разширение на системата, като е оставено достатъчно място, за да може да се монтира допълнително оборудване, без да е необходимо разместването на текущото.

Илюстративна схема на монтаж на оборудването в сървърния шкаф е представена на следващата фигура.



Примерно разположение на сървърите в шкафа

Мрежовата свързаност на решението към мрежата на Възложителя се осъществява посредством два 1Gb интерфейса, като в случай на нужда може да се увеличи до запълването на всички портове на комутатора HPE Virtual Connect Flex-10/10D Module. Мрежовата свързаност на портовете за управление на оборудването е препоръчително да бъде изолирана в отделни комутатори, а при липсата на такива в отделен VLAN.

Свързаността между сървърите в шасито ще се осъществява посредством два 10Gb интерфейса на всеки сървър чрез вградените в шасито връзки. По този начин, основния трафик между отделните компоненти на решението ще остане изолиран вътре в самото шаси.

Илюстративна схема на външната LAN свързаността на решението:

Handwritten signature

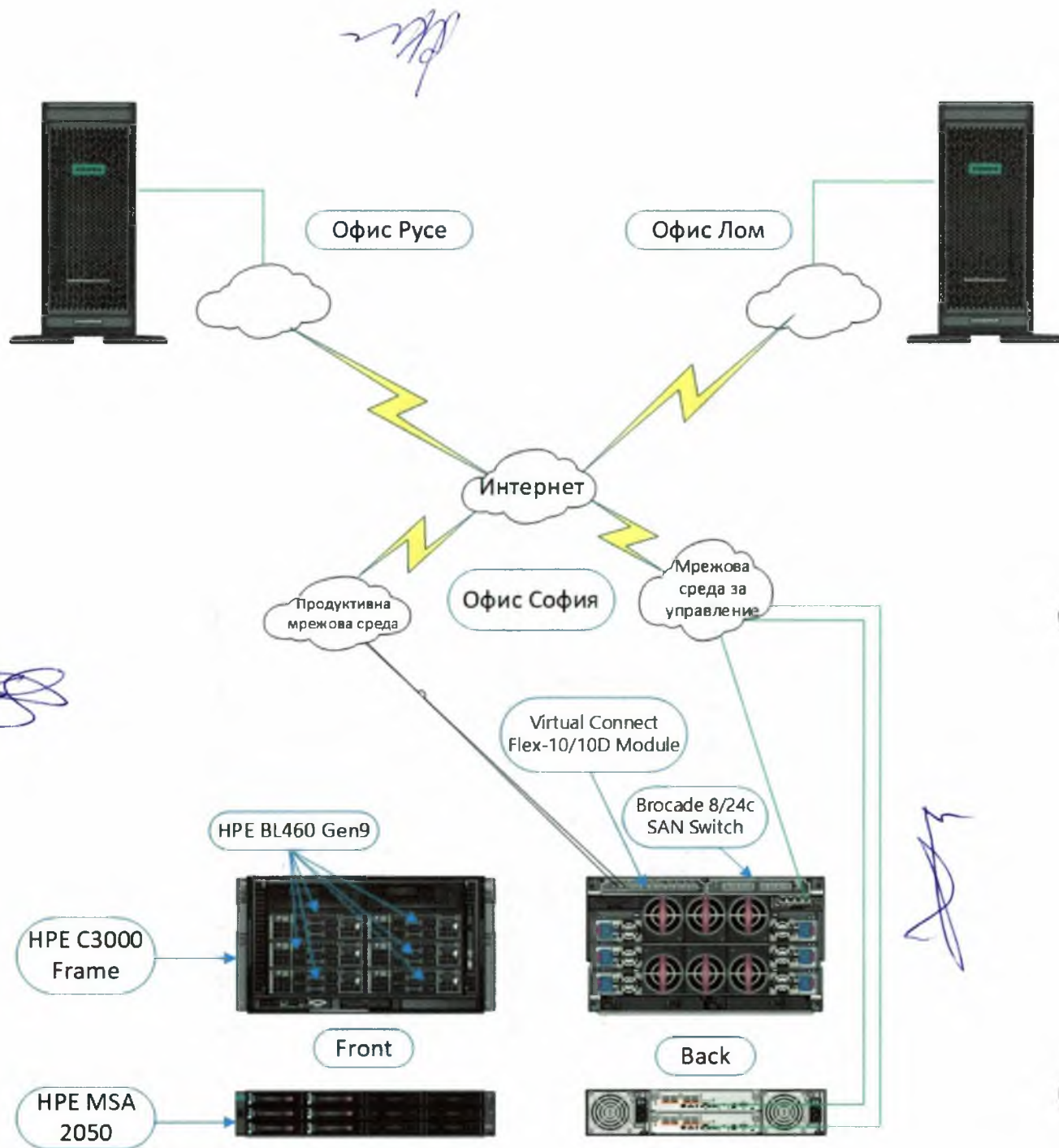


Схема на мрежовата свързаност

- 1Gb
- 1Gb
- ⚡ Интернет свързаност между офисите

SAN свързаност:

[Handwritten signature]

Свързаността на споделения дисков масив и сървърите се осъществява посредством четири 8Gb SFP връзки, по две от всеки контролер. SAN комутаторът ще бъде конфигуриран съобразно добрите практики на производителя.

Сървърите ще се свързват към SAN комутатора посредством два порта със скорост 8GB през вътрешните за шасито връзки.

Илюстративна схема на SAN свързаността на решението:

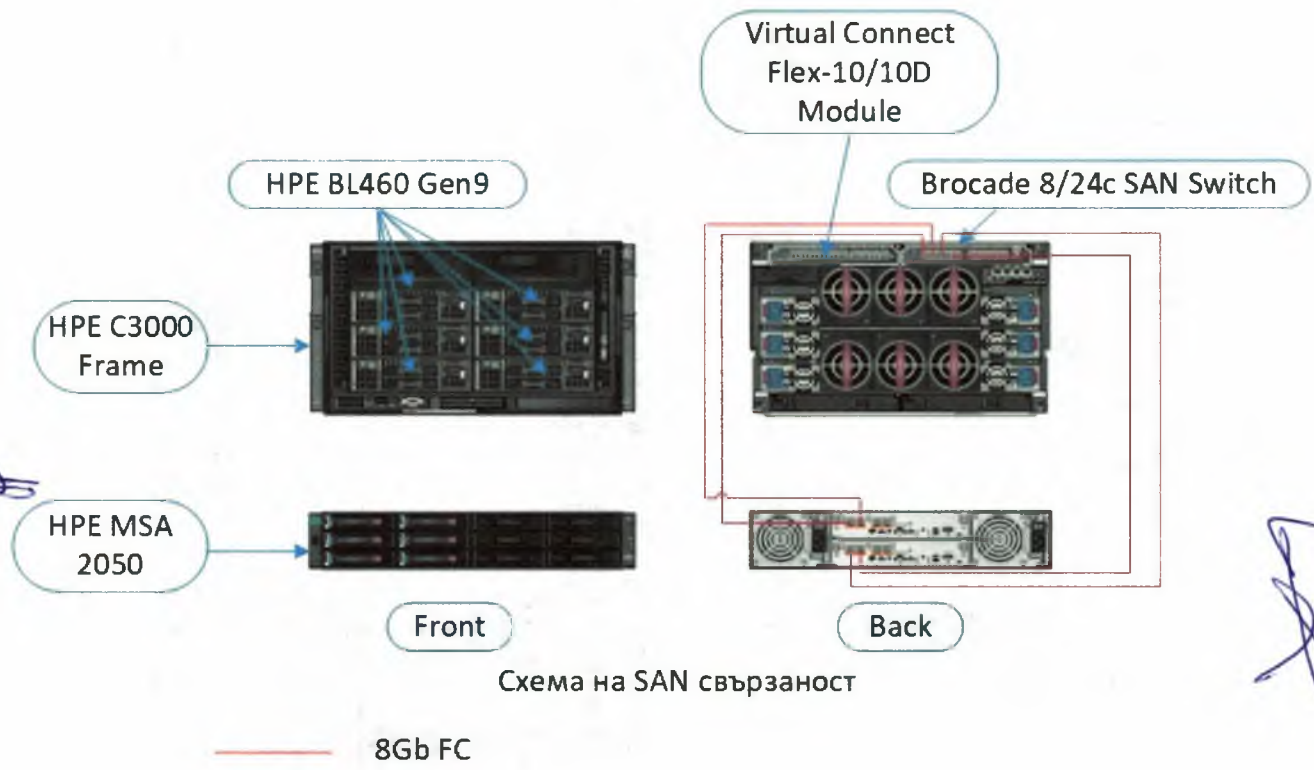


Схема на SAN свързаност

За осигуряване на достатъчно време за правилно спиране на сървърите при отпадане на захранването поради авария в електропреносната мрежа, е предвиден UPS APC Smart-UPS SRT 5000VA. Съгласно добрите практики, единият кръг на захранването ще бъде свързан към непрекъсваемото захранване (UPS APC), а другия директно в електропреносната мрежа.

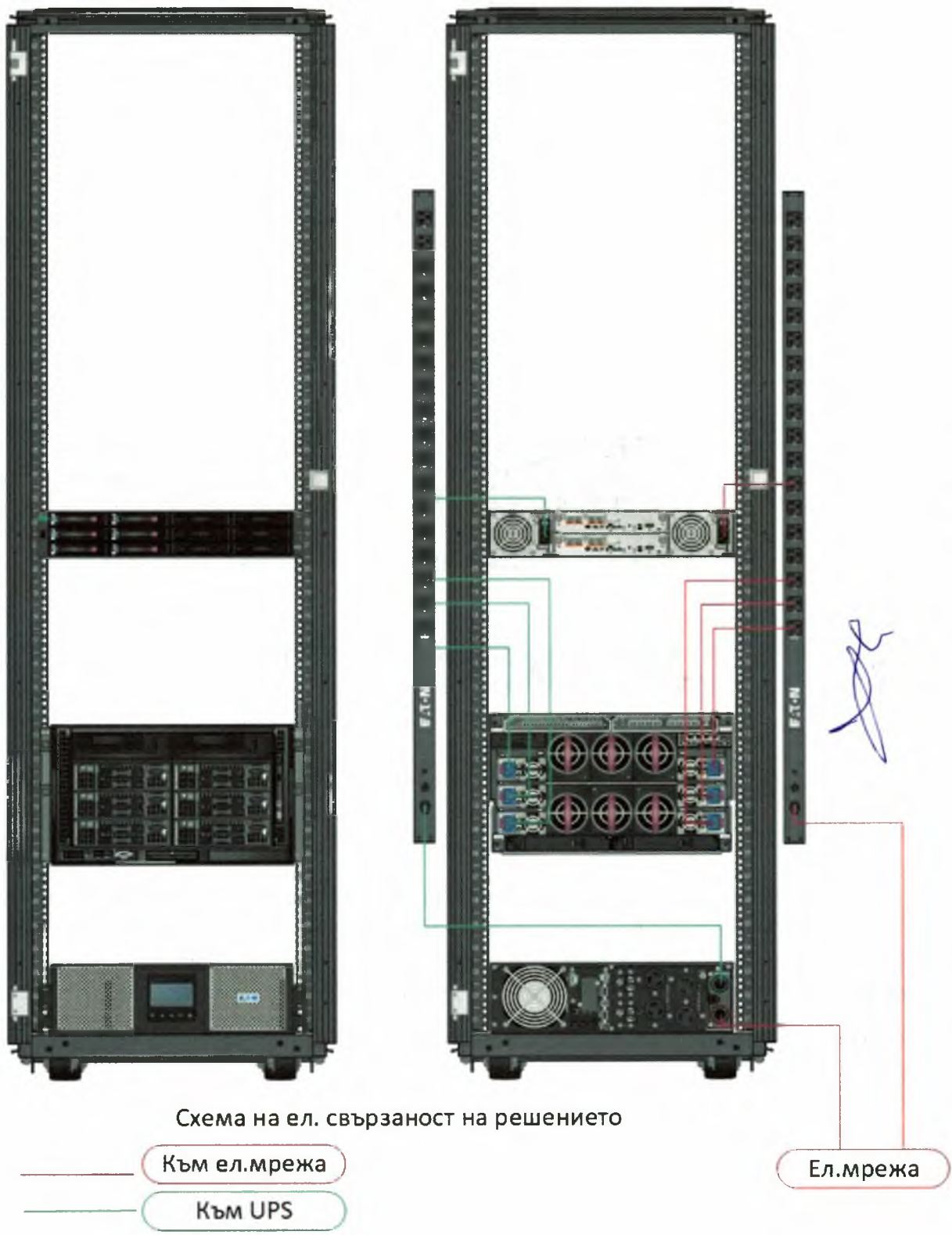
Илюстративна схема на свързаността към електропреносната мрежа на решението е представена на следната фигура.

[Handwritten signature]

0000-0

Handwritten signature

Handwritten signature

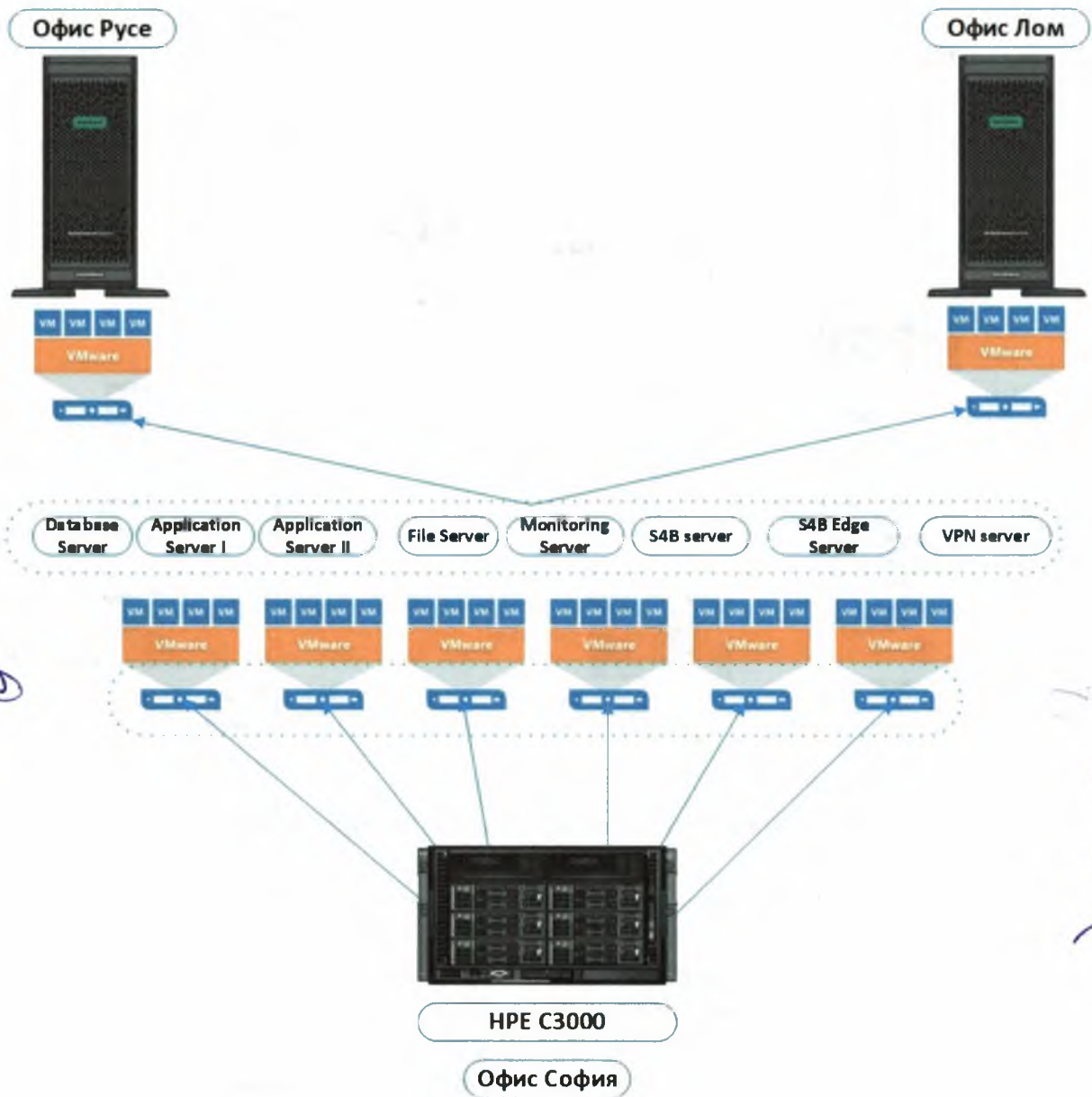


Handwritten signature

За по-голяма устойчивост на решението сървърите ще бъдат виртуализирани, чрез използването на платформата VMWare ESXi. По този начин в случай на проблем с някой от физическите сървъри, лесно ще може да бъдат стартирани услугите, които той предлага, на някой от другите сървъри в шасито. Тази концепция осигурява резервираност на системата. Предвидено е критичната информация за системата да бъде репликирана ежедневно на

Handwritten signature

сървърите в отдалечените локации в Русе и Лом, за да може да се гарантира запазване на данните в случай на проблем с основната локация.



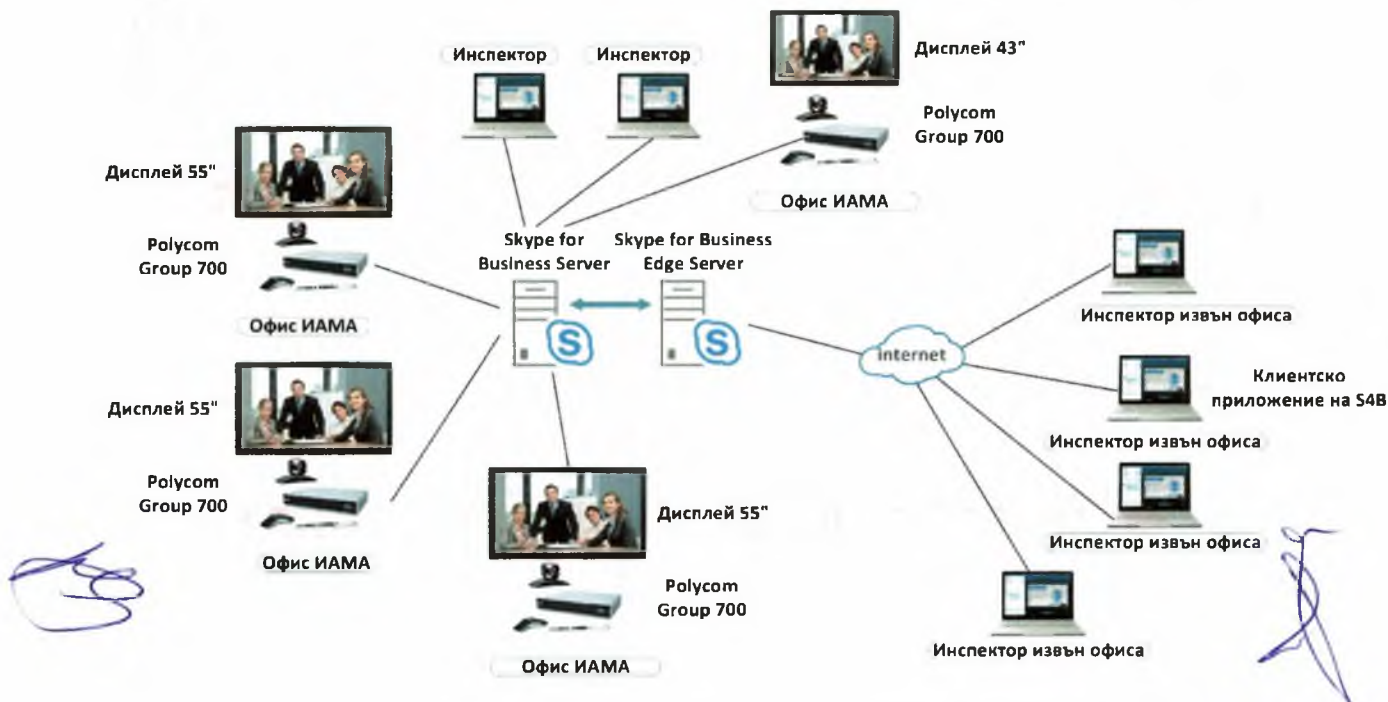
За целите на проекта ще бъдат инсталирани следните виртуални машини:

1. Database server – сървър за база данни
2. Application server I – сървър за приложения
3. Application server II – сървър за приложения
4. Monitoring server – Служи за наблюдение на системата и изпращане на съобщения в случай на проблеми
5. File server - Ще се използва за споделяне на общи файлове
6. Skype for Business server (S4B) – осигурява видеоконферентна свързаност
7. Skype for Business (S4B) Edge server – предоставя възможност на отдалечени потребители на системата да провеждат видеоконферентни разговори

8. VPN server – осигурява сигурната репликация на данни между централната локация и офисите в Русе и Лом.

Предвидена е изграждането на система за висококачествени видео комуникации, която ще предостави възможност за ефективна комуникация както между партньорите по проекта, така и вътрешно между служителите на ИАМА.

На следната фигура е представена платформата за видеоконферентна свързаност.



Платформа за видеоконферентна свързаност

Избраната платформа за видеоконференции Microsoft Skype for Business ще бъде инсталирана върху две виртуални машини, за които са осигурени всички необходими софтуерни лицензи и хардуерни ресурси. В офисите на организацията ще бъдат инсталирани четири хардуерни терминала Polycom RealPresence Group 700, които са сертифицирани, че по дизайн имат осигурена пълна съвместимост с платформата Skype for Business. Предвидени са шест клиентски приложения на Skype for Business за видеоконферентни разговори, които ще бъдат инсталирани на мобилни компютри, предназначени за инспекторите.

Видеоконферентните терминали в офисите на ИАМА, както и инспекторите, които се намират в тези офиси ще се свързват към Skype for Business сървъра през вътрешната мрежа на организацията. Сървърът Skype for Business Edge позволява провеждането на видео разговори с потребители, които се свързват през интернет. Такива са мобилните компютри на инспекторите, които се намират извън офисите на организацията. Това гарантира гъвкавост на работния процес и възможност за провеждане на видео разговори от мястото на събитието, спестявайки ценно време за придвижване между локациите.

Предвидена е доставка на офис оборудване, което се състои от осем мултифункционални устройства HP LaserJet Pro MFP M426fdn и седем мобилни компютри HP EliteBook 755 G4. В сървърното помещение, където ще бъде инсталирано сървърното шаси HP C3000, ще бъде инсталирана специализирана климатична система Daikin Comfora 24000BTU.

2.5. Базов софтуер


2.5.1. Изисквания към СУБД

Базата данни ще отговаря на изискванията на Възложителя, представени в техническата спецификация на поръчката и ще има следните характеристики:



- Ще осигурява възможност за разработка и развой на релационната база от данни
- Ще съдържа аналитични функции
- Ще осигурява надеждност на данните
- Ще осигурява механизъм за одит във времето на потребителските действия
- Ще използва/поддържа GUI (графичен потребителски интерфейс) инструмент за управление и оптимални настройки на базата от данни.

По отношение на възможността за разработка и развой на релационната база от данни, Системата ще отговаря на изискванията, представени в таблицата по-долу:

№ по ред	Изискване
NFREQ.16.	<p>Ще поддържа неограничен брой потребители за актуализация, корекция, преглед и визуализация</p> <p><i>Описание на реализацията:</i></p> <p>Предлаганото решение с Oracle Database не ограничава броя на потребителите за актуализация, корекция, преглед и визуализация на данните, съхранявани и управлявани в базата от данни. Лицензът за правото на ползване е с метрика Процесор и на практика позволява Oracle Database да се ползва в приложни конфигурации с неограничен брой потребители, стига хардуерните машини да имат нужния ресурс за целта.</p>
NFREQ.17.	<p>Ще поддържа всички стандартни релационни типове данни, а също и типове данни за съхраняване на JSON документи, XML сървиси, текст, изображения, аудио и видео данни, сензор данни, RDF графи и географски данни</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Database е релационна СУБД, която поддържа стандартни за този тип бази данни типове данни като числови NUMBER и стрингови данни CHAR, VARCHAR2, данни тип дата-час DATE, данни-файлове BLOB, които са вградени SQL типове данни. JSON документи могат да се съхранят в колони от таблици в Oracle Database като VARCHAR2, CLOB, или BLOB, което позволява работата с JSON документите по начин, по който се ползват и стандартните типове данни, т.е. механизмите на обработка на данни са идентични, следователно ефективни в</p>

№ по ред	Изискване
	<p>производителност. Oracle Database поддържа JSON нативно с функции за реляционна база данни, включително транзакции, индексирание, декларативно запитване и изгледи.</p> <p><i>За повече информация:</i> Overview of JSON in Oracle Database</p> <p>Oracle XML DB е набор от технологии вградени в Oracle Database, свързани с високоефективното управление на XML данни: съхранение, генериране, достъп, търсене, валидиране, трансформиране, разработване и индексирание. Oracle Database съхранява XML и URI данни, които се обработват чрез XMLType функции по обръщение, запитване и извличане, участващи в XPath изрази. Таблицы и изгледи могат да съдържат XMLType. Когато се съхраняват XML данни в таблиците, могат да се съхраняват в CLOB или в реляционен обект. Запитванията и DML заявките са идентични и независят от типа съхранение.</p> <p><i>За повече информация:</i> Oracle Database SQL Language Reference XMLType</p> <p>Вградените в Oracle Database LOB типове данни BLOB, CLOB и NCLOB (съхранени вътрешно) и BFILE (съхранени външно) могат да съхраняват големи и неструктурирани данни като текстови, графични, видео и пространствени данни.</p> <p>Освен това, Oracle Database предоставя вграден геопространствен тип Locator данни – пространствен вектор (SDO_GEOMETRY), за да позволи описание на геометрии като точки, линии и полигони.</p> <p><i>За повече информация:</i> Spatial and Graph Developer's Guide - Oracle Locator</p>
<p>NFREQ.18.</p>	<p>Ще поддържа езици за извикване и изпълнение на заявки към данните: REST сървиси за достъп до таблици в базата и JSON обекти, XQuery за XML обекти, SPARQL за RDF графи, Java API и стандартни SQL заявки</p> <p><i>Описание на реализацията:</i></p> <p>Oracle предоставя Oracle REST Data Services (ORDS), които улесняват разработването на модерни REST интерфейси за реляционни данни в Oracle Database, Oracle Database 12c JSON Document Store и Oracle NoSQL Database. Java приложението мапира HTTP (S) действията (GET, POST, PUT, DELETE и т.н.) със съответните транзакции в базата данни и връща резултата, в JSON формат. ORDS се поддържа в Weblogic, Tomcat, Glassfish и също като самостоятелно приложение, изпълняващо Jetty в режим на вграждане.</p> <p>Oracle XML DB поддържа най-новата версия на спецификацията на езика Xquery - препоръката на W3C XQuery 1.0. Можете да използвате XQuery, за да заявите XML данни, независимо къде се намират, дали са съхранени в таблици на базата данни, достъпни чрез Web Services или създадени по друг начин в движение.</p>

№ по ред	Изискване
	<p><i>За повече информация:</i> Oracle XML DB: Best Practices to Get Optimal Performance out of XML Queries.</p> <p>Oracle Locator функционалността в Oracle Database осигурява бързо пространствено индексирание с R-tree и пространствени заявки, като отдалеченост от геометриите, най-близко разположение, буферирание и прочие. Географските данни и данните за местоположението се манипулират, като се използват същите семантики, прилагани за типовете CHAR, DATE или INTEGER, които са познати на всички потребители на SQL. Oracle Database Locator функцията е основа за GIS SPARQL базирани разработки и интерпретация на RDF графи.</p> <p><i>За повече информация:</i> Spatial and Graph Developer's Guide - Oracle Locator</p> <p>Oracle предоставя цялостно Java решение за създаване, внедряване и управление на Java приложения - програмни интерфейси от страна на клиента и инструменти за разработка на Java и JVM, интегриран в Oracle Database. В допълнение към Oracle JVM, Java програмната среда осигурява: Java съхранени процедури, тясно интегрирани с PL/SQL; програмни интерфейси JDBC и SQLJ за достъп до SQL данни; и инструменти и скриптове, които подпомагат разработването, зареждането и управлението на Java класове.</p> <p><i>За повече информация:</i> Database Java Developer's Guide</p>
NFREQ.19.	<p>Ще дава възможност да се дефинират сложни типове данни от стандартните такива, за да може да се борава с цял обект като данна</p> <p><i>Описание на реализацията:</i></p> <p>Oracle е релационна система за управление на бази данни на база обект (ORDBMS), което означава, че потребителите могат да дефинират допълнителни видове данни - уточняващи структурата на данните и начините за опериране върху тях - и да използват тези обектни типове в релационния модел. Този подход добавя стойност към данните, съхранявани в база данни и разработчиците на приложения работят със сложни данни като изображения, аудио и видео. Обектите в базата съхраняват структурирани бизнес данни в естествената им форма, като приложенията могат да боравят с обекта и го извличат директно от базата, а не съставно по неговите елементи. Така, приложенията, които управляват и споделят данните, трябва да съдържат само логиката на приложението, а не логиката на данните – нещо, което е характерно за обектно-ориентираните езици на програмиране.</p> <p><i>За повече информация:</i> Introduction to Oracle Objects</p>
NFREQ.20.	<p>Освен релационен SQL модел да поддържа и не релационен non-SQL модел на съхранение на JSON документи</p>

№ по ред	Изискване
	<p><i>Описание на реализацията:</i></p> <p>Oracle Database предоставя поддръжка за съхраняване, индексиране и запитване на JSON документи в базата данни и позволява разработчиците да създават директно приложения, базирани на документи. Освен съхранение на JSON документи, Oracle Database изпълнява ACID транзакции през тези JSON документи - SQL заявки до JSON документи, за целите на отчети и анализи в приложенията за съхраняване на документи, както и проста SQL интеграция между JSON документи, релационни данни и други типове данни.</p> <p><i>За повече информация:</i> Overview of JSON in Oracle Database</p>
<p>NFREQ.21.</p> 	<p>Ще поддържа всички XML стандарти включително XML, SQL/XML and XSLT, XQuery, Namespaces, Document Object Model</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Database поддържа XML стандарт, като обхваща двата модела за данни SQL и XML по оперативно съвместим начин в SQL/XML модел. Oracle XML DB включва XSLT процесор, предназначен за XSL трансформации в базата данни. XQuery и XPath Full Text (XQFT) и Document Object Model (DOM) стандарти се поддържат от Oracle XML DB 12c. Предоставя набор от предефинирани namespaces и prefixes, които могат да се ползват с XQuery в Oracle XML DB.</p> <p><i>За повече информация:</i> XML DB Developer's Guide</p>
<p>NFREQ.22.</p> 	<p>Ще съхранява и изпълнява логика, разработена на собствен език за програмиране, както и на езиците JAVA и .NET</p> <p><i>Описание на реализацията:</i></p> <p>Oracle JVM, като част от Oracle Database позволява съхранение на Java процедури. Запазените процедури са Java методи, публикувани в SQL и съхранявани в базата данни за обща употреба. Когато се извиква от клиентските приложения, съхранената процедура може да приема аргументи, да препраща към класове Java и да връща резултати от Java. В резултат на това, при извикването на метода през trigger или SQL заявка, време на изпълнение е кратко.</p> <p><i>За повече информация:</i> Database Java Developer's Guide</p> <p>Предлаганият с Oracle Database, Oracle Data Provider за .NET класове и приложни програмни интерфейси (API) предоставят достъп до Oracle Database от .NET приложението и от .NET съхранени процедури и функции. Разработчиците на приложения могат да записват съхранени процедури и функции, използвайки всеки съвместим с .NET език, като C # и VB.NET, и да използват тези .NET съхранени процедури в базата данни по същия начин, както другите PL/SQL или Java съхранени процедури. .Net съхраняваните процедури могат да бъдат</p>

№ по ред	Изискване
	<p>извиквани от PL/SQL пакети, процедури, функции и задействия; от SQL изразите или от всяко място, където може да се извика PL / SQL процедура или функция.</p> <p>За повече информация: Database Extensions for .NET Developer's Guide</p>
<p>NFREQ.23.</p>	<p>Ще има връзка с външни приложения по WebDAV, HTTP и FTP интерфейси</p> <p><i>Описание на реализацията:</i></p> <p>Database File System (DBFS) е функция на Oracle Database, която улеснява потребителите да имат достъп и да управляват файлове, съхранявани в базата данни. Достъпът на файловете по HTTP / HTTPS, FTP и WebDAV е на основата на възможностите за работа с файлове в базата, предоставяни от Oracle XML DB. Oracle Database12c въвежда HTTP / HTTPS, FTP и WebDAV достъп до DBFS чрез виртуалната директория "/" dbfs" в хранилището на XML DB.</p> <p>За повече информация: Database SecureFiles and Large Objects Developer's Guide</p>
<p>NFREQ.24.</p>	<p>Ще предоставя BLOB тип съхранение и управление на съдържание</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Database съхранява и обработва документи, изображения и медийно съдържание, съхранявани в файлове чрез реализации, базирани на "показалец". За съхранение на неструктурирани данни в таблици се ползват Binary Large Objects или BLOBs контейнери. Oracle Database съхранява и управлява големи обекти в BLOB, CLOB, NCLOB и BFILE на големи блокове от неструктурирани данни (като текст, графични изображения, видеоклипове и звукови вълни) в двоичен или символен формат. Може да съхранява неструктурирани данни с компресия, криптиране или дедупликация с големина до 128 терабайта.</p> <p>За повече информация: Database Concepts: Content Management</p>

По отношение на аналитичните функции, базата данни ще отговаря на следните изисквания:

<p>NFREQ.25.</p>	<p>Ще притежава вградени възможности за анализ на данните и да предоставя функции за ранкиране, агрегиране и статистики, както и възможност за моделиране, откриване на патърн съответствия, работа с приближения и хипотези в ранкиране и разпределяне</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Database 12c има вградени аналитични функции като:</p> <ul style="list-style-type: none"> - Enhanced reporting – SQL функции за windowing, lag-lead и агрегатните функции (SUM, AVERAGE, COUNT, MAX, MIN и други); Pivot операции (PIVOT clause); Уплътняване на данни за отчитане и изчисления на динамични редове (partitioned outer joint с PARTITION BY clause ще резултира в уплътнен набор от
------------------	---

[Handwritten signature]

данни, върху който ще могат да се приложат аналитични функции за усреднени стойности и суми и техните промени.

- Advanced aggregations като: Rollup - изчислява няколко нива на междинните суми в определена група от величини; Cube - изчислява междинните суми за всички възможни комбинации от група от размери и изчислява общата сума; Grouping - помощи за идентифициране на кои редове в резултат се генерират от операция по натрупване или куб; Grouping set - набор от дефинирани от потребителя групи, генерирани като част от комплекта резултати.
- Statistical analysis: Oracle Database предоставя богата библиотека от статистически функции, включително поддръжка за Линейна алгебра; Линейна регресия; Статистически агрегати за тестове на хипотези.
- Pattern matching: Oracle Database предоставя SQL синтаксис за съвпадение на шаблони. Усвоявайки Perl за работа с изрази чрез SQL се дефинират набор от правила за шаблони в последователност (потоци) от редове (чрез PATTERN clause на MATCH_RECOGNIZE construct). Тази нова способност за inter-row търсене на шаблон допълва вече съществуващите възможности за търсене на шаблоните в рамките на символните низове на един запис.
- Spreadsheet-like data modeling: SQL MODEL клаузата въвежда ново ниво на мощ и гъвкавост при SQL изчисленията. С MODEL клаузата може да се създаде многомерен масив от резултати от заявки и след това да се приложат формули (наречени правила) в този масив, за да се изчислят нови стойности. Правилата могат да варират от основни аритметични до рекурсивни уравнения. За някои приложения MODEL клаузата може да замени електронни таблици тип spreadsheet.

За повече информация: [Analytical SQL in Oracle Database 12c](#)

[Handwritten mark]

[Handwritten mark]

NFREQ.26.

Ще поддържа метаинформация за таблиците с данни във вид на изгледи от изпълнение на конкретен SELECT, чрез който да се извличат данните. Тези изгледи трябва да е възможно да се създават върху повече от една таблица и да се съхраняват като материализиран изглед в базата от данни

Описание на реализацията:

Материализирания изглед (Materialized View) в Oracle Database е обект от базата данни, която съдържа резултат от изпълнено запитване. Когато се създава обект - materialized view, Oracle Database създава вътрешна таблица с поне един индекс и един базов изглед, които разглеждаме като метаинформация, асоциирана с записа на резултата от изпълнението на даден SELECT.

За повече информация: [Create Materialized View in Oracle Database](#)

NFREQ.27.

Ще поддържа мета информация за таблиците с данни, които са в модел тип звезда, във вид на димензионни изгледи, които да извършват автоматична агрегация по дименсиите и изчисление на фактите, обект на SELECT заявката

[Handwritten signature]

Handwritten mark at the top of the page.

	<p><i>Описание на реализацията:</i></p> <p>SQL MODEL клаузата позволява да се създадат многомерни масиви, като преобразувате колоните на заявката в три групи: дялове, дименсии и измервани стойности. MODEL клаузата дава възможност да се зададат правила за манипулиране на измерваните стойности в клетките от многоизмерния масив, дефиниран от дяловете и дименсиите. Правилата достъпват и актуализират стойностите на колоните чрез директно определяне на стойностите на величините. Референтните правила, водят до един много лесен за четене такъв модел. Правилата са кратки и гъвкави и могат да използват wild карти и повтаряеми структури за максимална изразителност. Oracle Database оценява правилата по ефективен начин, паралелизира изчисляването на моделите, когато е възможно, и осигурява безпроблемна интеграция на клаузата MODEL с други SQL клаузи. Така клаузата MODEL е мащабируем и управляем начин за изчисляване на бизнес модели в базата данни.</p> <p>За повече информация: Database Data Warehousing Guide - SQL for Modeling</p>
--	--

По отношение на надеждността на данните, СУБД ще отговаря на следните изисквания:

<p>NFREQ.28.</p>	<p>Ще позволява онлайн преместване на файлове и данни от сторидж масив на сторидж масив, без да се спира работа с данните</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Automatic Storage Manager (ASM) е volume manager и file system за файлове на Oracle Database. С разгръщането на Oracle Database в ASM, може да се добавят или премахват сторидж дискове от група дискове, докато база данни продължава да получава достъп до файловете от групата дискове. Когато се добавят или премахват дискове от група дискове, Oracle ASM автоматично преразпределя съдържанието на файла и елиминира необходимостта от престой (downtime) при преразпределение на съдържанието между наличното пространство. Така, с добавяне на дискове от втори сторидж масив и отнемане на дискове от първоначалния сторидж масив, се мигрират онлайн файловете на базата от сторидж масив на сторидж масив.</p> <p>За повече информация: Automatic Storage Management Administrator's Guide</p>
<p>NFREQ.29.</p>	<p>Ще позволява онлайн увеличаване на заделения за целите на работата й сторидж ресурс</p> <p><i>Описание на реализацията:</i></p> <p>Oracle ASM използва дискови групи за съхраняване на файлове с данни; дисковата група на Oracle ASM е колекция от физически дискове, които Oracle ASM управлява като единица. В рамките на една дискова група Oracle ASM излага интерфейс - файлова система за файлове на Oracle Database. Съдържанието на файловете, които се съхраняват в дискова група, се разпределя равномерно, за да се премахнат</p>

Handwritten mark on the left side of the table.

Handwritten mark at the bottom of the page.

	<p>горещите точки и да се осигури равномерно натоварване на дисковете. Ефективността е сравнима с ефективността на raw devices. Боравенето с дискови групи позволява онлайн разширение на пространството на базата данни, заделено на сториджа без престой (downtime) или забавяне на I/O.</p> <p>За повече информация: Automatic Storage Management Administrator's Guide</p>
NFREQ.30.	<p>Ще поддържа и предоставя инструменти за онлайн архивиране на базата данни</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Recovery Manager (RMAN) е клиент-инструмент на Oracle Database, който изпълнява задачи за архивиране и възстановяване на вашите бази данни и автоматизира администрирането на вашите стратегии за архивиране. Това опростява архивирането и възстановяването на файловете на базата данни. За разлика от други инструменти за архив, RMAN не изисква офлайн режим на Oracle Database, защото знае формата на блоковете данни. RMAN гарантира, че няма да архивира блокове, които в момента се променят и не са подходящи за архив на момента. По време на архив RMAN сесията към Oracle Database чете всеки блок данни и проверява дали е счупен чрез сравняване на хедъра на блока и съдържанието му. Ако хедъра съдържа метаинформация, различна от съдържанието му, то блокът се отчита като фрактуриран за архив, като в последствие RMAN сесията отново преглежда блока. Ако се открие същата аномалия, блокът се счита за трайно корумпиран. Също така, RMAN не е необходимо да замразява контролната точка на хедъра на файла с данни, състоящ се от много блокове с данни, защото знае реда, в който се четат блоковете, което му позволява да запише известен добро копие на файла.</p> <p>За повече информация: Database Backup and Recovery User's Guide - About Online Backups and Backup Mode</p>
NFREQ.31.	<p>Ще поддържа механизъм за онлайн възстановяване на базата данни</p> <p><i>Описание на реализацията:</i></p> <p>Oracle RMAN позволява да се възстанови една или повече таблици или таблични дялове в определен момент от времето, без да се отразяват на останалите обекти на базата данни, т.е. онлайн. Можете да използвате резервни копия, създадени по-рано, за възстановяване на таблици и дялове на таблици до определен момент.</p> <p>Възстановяването на таблици и таблични дялове от архивиране на RMAN е полезно в следните сценарии:</p> <ul style="list-style-type: none">- Трябва да бъдат възстановени много малък брой таблици до определен момент от времето, тогава това е най-ефективното решение, защото премества всички обекти в пространството за таблици до определен момент от време.- Трябва да бъдат възстановени таблици, които са били логически повредени или са били изхвърлени и изчистени.

MD

- Трябва да се възстановят загубени данни, след като DDL (Data Definition Language) операция промени структурата на таблиците.

Oracle RMAN позволява да се възстанови и отделен блок с данни или отделни блокове данни в рамките на файл с данни. Това е полезно за проблеми с физическата корупция, които включват малък, известен брой блокове на диска. Загубата на данни на ниво блок обикновено е резултат от случайни входно-изходни грешки, които не причиняват широко разпространение на корупция на памет от диска.

В случаи, при които степента на загуба или корупция на данните е неизвестна и целият файл с данни изисква възстановяване, възстановяването на файловете с данни остава най-доброто решение.

За повече информация: [Database Backup and Recovery User's Guide - Recovering Tables and Table Partitions from RMAN Backups](#)
[Database Backup and Recovery User's Guide - Performing Block Media Recovery](#)

Други изисквания, на които ще отговаря СУБД:

NFREQ.32. Ще предоставя механизъм за одит във времето на потребителските действия с данните в таблиците

Описание на реализацията:

Oracle Database предоставя вграден одит механизъм за отделни действия, като например типа изпълнена SQL операция или комбинации от данни, които могат да включват потребителско име, приложение, време и прочие.

Може да се конфигурира одит за успешни и/или неуспешни дейности, да включва или изключва конкретни потребители. В допълнение към одита на стандартните SQL DDL/DML дейности, които предоставя базата данни, одитът може да включва дейности от Oracle Recovery Manager, Oracle Data Pump, и Oracle SQL*Loader (Direct Path)

Одитът в Oracle Database е активиран по подразбиране. Всички записи от одита се записват в единен формат на обща одит-трактовка и се предоставят чрез изгледа UNIFIED_AUDIT_TRAIL. Тези записи се намират в схемата AUDSYS. Записите от одита се запазват по подразбиране в SYSAUX пространството за таблици. Oracle препоръчва да конфигурирате отделно пространство за таблици за одит-трактовката, за да не се натоварва SYSAUX пространството.

Oracle Database генерира одиторски записи по време или след фазата на изпълнение на одитираните SQL изрази. Oracle Database индивидуално одитира SQL заявките в PL/SQL програмни елементи, когато се изпълняват такива.

Честотата на запис на одита на диск може да се контролира. Ако записите са в режим „стои опашка“, тогава записите от одита се записват периодично в RAM паметта,

MD

0000032

2008

	<p>ползвана от Oracle Database, а не на диск незабавно, което дава свобода на режима на работа с одит.</p> <p>Одитът е ефективен метод за налагане на строг вътрешен контрол на работа с данните, в съответствие с нормативните изисквания по сигурността.</p> <p>За повече информация: Database Security Guide - Monitoring Database Activity with Auditing</p>
NFREQ.33.	<p>Ще използва/поддържа GUI (графичен потребителски интерфейс) инструмент за управление и оптимални настройки на базата от данни</p> <p><i>Описание на реализацията:</i></p> <p>Oracle Enterprise Manager Database Express, наричан още EM Express, е уеб базиран инструмент за управление на предлаганата Oracle Database 12c. Вграден вътре в сървъра на базата данни, той предлага поддръжка за основни административни задачи като конфигурация, съхранение и управление на потребителите:</p> <ul style="list-style-type: none">- Конфигурация: Четирите основни функции, налични в менюто "Конфигурация", са Параметри на Инициализация, Конфигурация на Паметта, Използване на Характеристики на Базата данни и Свойства на Базата данни. Администраторите на база данни могат да преглеждат и задават параметри за инициализация за един или повече копия на базата данни. Те могат също така да определят как е конфигурирана паметта, и по-специално, количеството памет, разпределена на SGA и PGA, върховите процеси, които консумират памет, и дали се използват или не някои от функциите за автоматично управление на паметта. В допълнение, данните на съветника за паметта също се показват на страницата <p>Съхранение: Като част от администрацията за съхранение, DBA могат да управляват регистрационни файлове, архивни дневници и таблични пространства (да гледат разход на място, да създават и / или да променят пространства за таблици). Освен това на DBA се дава известна видимост в съдържанието на контролния файл. DBA могат също така да видят Undo Advisor, както и да отменят статистически данни, като например да отменят обема на генериране и да отменят използването на пространството.</p> <p>Сигурност: DBA могат да използват функциите, налични в менюто "Сигурност", за да управляват потребителите, ролите и профилите, като създават, променят или предоставят и отнемат привилегии от тях.</p> <p>За повече информация: Oracle Database 12c: EM Database Express</p>
NFREQ.34.	<p>За пълно съответствие и добро поведение на бъдещата система е предимство, ако предлаганите СУБД и сървър за приложения са от един и същи доставчик</p> <p><i>Описание на реализацията:</i></p>

Handwritten marks on the left margin, including a large blue scribble and a smaller one below it.

Handwritten signature or mark at the bottom center of the page.

	Предлаганите СУБД и сървър за приложения са от един и същи доставчик, което гарантира пълно съответствие на технологиите, стабилност и добро поведение на бъдещата система, намаляване на риска от дефекти в софтуерната архитектура, финализиране на проекта навреме и минимализиране на риска от скъпа и сложна за подобрения в конфигурацията последваща поддръжка.
--	--


Изпълнителят е трябва да включи в своята оферта и да осигури в рамките на изпълнението на проекта необходимите лицензи за СУБД. В своето предложение Участникът следва да опише по вид и брой осигуряваните от него лицензи.

2.5.2. Изисквания към сървъра за приложения

Сървърът за приложение ще отговаря на следните характеристики:

- Ще осигурява поддръжка на стандартите:
 - Java EE 7 поддръжка и пълна съвместимост;
 - JDK 8 и Server JRE 8 сертификация;
 - Web Services стандарти;
 - Стандарти за сигурност
- Ще осигурява възможност за управление и мониториране на данните
- Ще осигурява възможност за интеграция със други СУБД;

По отношение на стандартите, предложената от кандидата СУБД ще поддържа:

 <p>NFREQ.35.</p>	<p>Java EE 7 поддръжка и пълна съвместимост</p> <p><i>Описание на реализацията:</i></p> <p><i>Описание на реализацията:</i></p> <p>Предложеният сървър за приложения Oracle WebLogic Server 12c (12.2.1) е Java EE-базиран приложен сървър, който напълно поддържа и е напълно съвместим с Java Platform Enterprise Edition версия 7.0 (т.нар. Java EE 7).</p> <p>Пълната имплементация на Java EE 7 от WebLogic Server предоставя стандартен набор от приложни програмни интерфейси (API) за създаване на дистрибутирани Java приложения, които достъпват голямо разнообразие от услуги като бази данни и Уеб услуги за интеграция и обмяна на данни с вътрешни и външни корпоративни системи.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none"> • Oracle WebLogic Server Java EE 7 Support: <ul style="list-style-type: none"> - https://docs.oracle.com/middleware/12213/wls/NOTES/whatsnew.htm#GUID-BCB7D3C3-6091-4D8D-87E7-3807C0E93F2B • Introduction to Oracle WebLogic Server: <ul style="list-style-type: none"> https://docs.oracle.com/middleware/12213/wls/INTRO/intro.htm#INTRO123
<p>NFREQ.36.</p>	<p>JDK 8 и Server JRE 8 сертификация</p>

Handwritten signature

	<p>Oracle WebLogic Server 12c е сертифициран за използване с JDK 8. Поддържаните клиенти на WebLogic Server 12c са сертифицирани за използване с JDK 7 и JDK 8. В допълнение, WebLogic Server също е сертифициран за използване с Server JRE 8, предназначен за "server-side" приложения. Server JRE прави разгръщането и внедряването на приложения по-бързо и по-лесно, като включва само най-често изискваните функции и компоненти от JRE и JDK.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none">• Oracle WebLogic Server JDK 8 and Server JRE 8 Certification: https://docs.oracle.com/middleware/12213/wls/NOTES/whatsnew.htm#NOTES405
<p>NFREQ.37.</p> <p><i>Handwritten mark</i></p> <p><i>Handwritten mark</i></p>	<p>Web Services стандарти, като JAX-WS, JAX-RS, JAX-RPC</p> <p><i>Описание на реализацията:</i></p> <p>В програмния Java EE модел, поддържан от Oracle WebLogic Server, се включва и работна рамка за изграждане и внедряване на Уеб услуги наречена WebLogic Web Services. WebLogic Web Services е разпределена среда, която позволява да се интегрират хетерогенни корпоративни приложения и да се излагат бизнес функции, и която поддържа различни функции и стандарти, като поддържаните типове Web Services са: Java API for XML-Based Web Services (JAX-WS), Java API for RESTful Web Services (JAX-RS) и Java API for XML-Based RPC (JAX-RPC).</p> <p>В допълнение, WebLogic Web Services предоставят общ набор от функции, които са достъпни за други системи в дадена мрежа и могат да бъдат използвани като компонент на разпределени Уеб базирани приложения.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none">• Introducing Oracle WebLogic Web Services: https://docs.oracle.com/middleware/12213/wls/WSOVR/weblogic-web-service-intro.htm#WSOVR109• Features and Standards Supported by WebLogic Web Services: https://docs.oracle.com/middleware/12213/wls/NOTES/whatsnew.htm#GUID-461DC56C-8436-4F0C-A81D-A36C71611B32
<p>NFREQ.38.</p>	<p>Стандарти за сигурност, като JAAS, JASPIC, JACC, JCE, JSSE, SSL, SSO, X.509, LDAP, TLS, SAML, XACML, PKI, RSA, Kerberos</p> <p><i>Описание на реализацията:</i></p> <p>Oracle WebLogic Server включва архитектура за сигурност и защита, която осигурява уникална и сигурна основа за корпоративни приложения. Чрез функциите за сигурност в WebLogic Server, предложеното решение се възползва от всеобхватна и гъвкава инфраструктура за сигурност, предназначена да посрещне предизвикателствата на сигурността при предоставянето на приложения в интернет пространството.</p> <p>Архитектурата за сигурност на WebLogic Server е отворената и гъвкава, въвежда разширен дизайн за сигурност на сървъри на приложения, както и предлага и</p>

Handwritten signature

поддържа широк набор от функционалности и разпространени стандарти за сигурност, включително JAAS, JASPIC, JACC, JCE, JSSE, SSL, SSO, X.509, LDAP, TLS, SAML, XACML, PKI, RSA, Kerberos и др.

За повече информация:

- Overview of the WebLogic Security Service:
<https://docs.oracle.com/middleware/12213/wls/SCOVR/model.htm#SCOVR116>
- Understanding WebLogic Server Security:
<https://docs.oracle.com/middleware/12213/wls/INTRO/security.htm#INTRO232>
- WebLogic Server Standards Support:
<https://docs.oracle.com/middleware/12213/wls/NOTES/whatsnew.htm#NOTES309>
- Secure Oracle WebLogic Server:
<https://docs.oracle.com/middleware/12213/wls/wls-secure.htm#>

По отношение на управлението, предложената от кандидата СУБД ще отговаря на следните изисквания:

NFREQ.39.

Ще предоставя единна рамка за централизирано управление и наблюдение (включително конфигуриране, администриране, диагностика, отстраняване на проблеми, настройка, сигурност, анализи)

Описание на реализацията:

Oracle WebLogic Server предоставя единна централизирана рамка за осъществяване на пълния жизнен цикъл на приложния сървър и разгърнатите на него приложения и ресурси. Рамката се характеризира с богат вграден инструментариум за управление, наблюдение, диагностика и сигурност, с цел ефикасни и финансово изгодни ИТ операции. Системната администрация на WebLogic Server включва широк спектър от задачи, като създаване на домейни на WebLogic Server, внедряване на приложения, мигриране на домейни от среда за разработка до производствени среди, наблюдение и управление на производителността на системата за изпълнение, конфигуриране и управление на сигурността за приложения и системни ресурси, отстраняване на проблеми и др. WebLogic Server включва и услуга за мониторинг и диагностика, която позволява да се създават, събират, анализират и архивират диагностични данни, генерирани от работещ потребителски модел на сървъра и приложенията, които са разгърнати на него. Тези диагностични данни осигуряват проицателност върху производителността на изпълняваните от WebLogic Server потребителски модули и внедрени приложения, което помага да се изолират и диагностицират недостатъците и затрудненията при изпълнение, когато те се появят. За да се увеличи оперативната ефективност на сървъра, се предлагат и вградени инструменти за автоматизация на управлението и наблюдението на сървъра и приложенията. Единната рамка за централизирано управление и наблюдение на WebLogic Server предлага гъвкавост и избор за опериране на средата, посредством включеният богат инструментариум: уеб-базиран графичен потребителски

Handwritten signature

интерфейс; скриптов език за автоматизиране; набор от RESTful интерфейси; SNMP; съветници за конфигуриране; и помощни програми на командния ред.

За повече информация:

- [Understanding Oracle WebLogic Server:](https://docs.oracle.com/middleware/12213/wls/INTRO/toc.htm)
<https://docs.oracle.com/middleware/12213/wls/INTRO/toc.htm>

(Специално секциите:

- *1 Introduction*
- *2 System Administration*
- *3 Overview of Administration Consoles*
- *10 Understanding WebLogic Server Security*
- *13 Monitoring, Diagnosing, and Troubleshooting*

NFREQ.40.

Единната рамка включва рамка за сигурност за управление и наблюдение на автентичността, упълномощаването, одита, самоличността и доверието

Описание на реализацията:

Единната рамка за централизирано управление и наблюдение на WebLogic Server включва и изчерпателен инструментариум за сигурност, с които се осъществяват основни концепции за сигурност в WebLogic Server, като удостоверяване/автентикация, оторизация/упълномощаване, идентичност/самоличност, доверие и одит.

За повече информация:

- [Administering Security for Oracle WebLogic Server:](https://docs.oracle.com/middleware/12213/wls/SECMG/toc.htm)
<https://docs.oracle.com/middleware/12213/wls/SECMG/toc.htm>
- [Secure Oracle WebLogic Server:](https://docs.oracle.com/middleware/12213/wls/wls-secure.htm#)
<https://docs.oracle.com/middleware/12213/wls/wls-secure.htm#>

NFREQ.41.

Ще приоритизира работните процеси въз основа на определени правила и реални статистически данни за ефективността на текущото време

Описание на реализацията:

Посредством вградената си функционалност, наречена Work Manger, WebLogic Server позволява приоритизиране на работните процеси съгласно определени правила и на база реални данни. WebLogic Server дава възможност да се определи как дадено приложение дава приоритет на изпълнението на работата си. Въз основа на дефинирани правила и чрез наблюдение на ефективността по време на изпълнение, WebLogic Server може да оптимизира производителността на дадено приложение и да поддържа зададените споразуменията на ниво услуга (service-level agreements). Вие определяте правилата и ограниченията за вашето дадено приложение, като дефинирате Work Manger и го прилагате или глобално към домейн на WebLogic Server, или към специфичен компонент на приложението.

За повече информация:

- [Using Work Managers to Optimize Scheduled Work:](#)

Handwritten marks: an arrow pointing up and a signature.

Handwritten signature

<p>NFREQ.42.</p>	<p>https://docs.oracle.com/middleware/12213/wls/CNFGD/self_tuned.htm#CNFGD113</p> <p>Ще поддържа версионирание на приложенията: внедряване на нова версия на актуализирано приложение заедно с по-стара версия на същото приложение</p> <p><i>Описание на реализацията:</i></p> <p>С цел надеждно и сигурно разгръщане и актуализиране на приложенията, WebLogic Server предлага гъвкави стратегии и подход за версионирание внедряване на опреснения и ремонти на приложения. Една от тях е стратегията за пренасочване на производството, която включва разгръщане на нова версия на актуализирано приложение заедно с по-стара версия на е същото приложение. WebLogic Server автоматично управлява клиентските връзки така, че само новите заявки на клиенти се насочват към новата версия. Клиентите, които вече са свързани с приложението по време на преразпределението, продължават да използват по-старата версия на приложението, докато не завършат работата си, в който момент WebLogic Server автоматично оттегля по-старото приложение и насочва следващите, вече нови заявки, към новата версия.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none"> • Redeploying Applications in a Production Environment: https://docs.oracle.com/middleware/12213/wls/DEPGD/redeploy.htm#DEPGD259
<p>NFREQ.43.</p>	<p>Ще поддържа паралелно разгръщане на приложения и модули</p> <p><i>Описание на реализацията:</i></p> <p>За случаи на използване, включващи внедряване на няколко приложения или внедряване на едно приложение с няколко модула, WebLogic Server предлага функционалност за паралелното внедряване на приложения и модули, което подобрява времето за тяхното разгръщане и стартиране.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none"> • Enabling Parallel Deployment for Applications and Modules: https://docs.oracle.com/middleware/12213/wls/DEPGD/managing.htm#GUID-E78D971B-1CA7-4BCC-A3DE-3AF59D481B8D
<p>NFREQ.44.</p>	<p>Да предоставя механизъм за предпазване от претоварване, както и възможност за откриване, избягване и отстраняване на причините за претоварване</p> <p><i>Описание на реализацията:</i></p> <p>WebLogic Server има вградени функции за защита от претоварване, които помагат за откриване, избягване и възстановяване от условия на претоварване. Тези функции предотвратяват отрицателните последици, които произтичат от продължаващото приемане на заявки, когато капацитетът на системата е достигнат, които последици могат да понижават производителността и стабилността на приложенията.</p> <p><i>За повече информация:</i></p>

Handwritten signature

- **Avoiding and Managing Overload:**
<https://docs.oracle.com/middleware/12213/wls/CNFGD/overload.htm#CNFGD146>

NFREQ.45.

Ще предоставя механизъм за сегрегация на мрежовия трафик

Описание на реализацията:

WebLogic Server позволява да се управлява мрежовия трафик, свързан с приложенията, по различни начини и да се конфигурира средата така, че да отговаря на разнообразните изисквания на приложенията и крайните потребители. WebLogic Server предоставя вграден механизъм за управление и сегрегация на мрежовия трафик, който позволява:

- Определяне на мрежовите интерфейси (NIC) и портовете, които да се ползват от определен(и) сървър(и) за различните видове мрежов трафик
- Избор на конкретни мрежови протоколи и изисквания за сигурност
- Определяне периода/времетраенето на определен мрежови канал и съобщенията преминаващи през него
- Ограничаване на размера на съобщенията преминаващи през определен мрежови канал

За повече информация:

- **Configuring Network Resources:**
<https://docs.oracle.com/middleware/12213/wls/CNFGD/network.htm#CNFGD156>

NFREQ.46.

Ще предоставя устойчивост и надеждност на JMS и JDBC ресурси

Описание на реализацията:

WebLogic Server предоставя вградено високоефективно решение за устойчивост на подсистеми и ресурси на сървъра, включително на JMS и JDBC ресурси.

По отношение на JMS, WebLogic функцията Store and Forward предоставя устойчивост на JMS съобщенията, като те могат да се съхраняват постоянно или за определен период от време в база данни или във файлова система. Например, едно съобщение, преди да бъде изпратено до адресанта, се съхранява в бази данни или файл, като съхранението продължава, докато съобщението не бъде изпратено и потвърдено като получено от адресанта. След успешно получаване, съобщението може да бъде преместено в архивно хранилище. По този начин се гарантира, че всяко JMS съобщение е устойчиво и надеждно – или ще бъде изпратено до адресанта в рамките на зададените брой опити и времетраене, или ще бъде декларирано като невъзможно за изпращане и съхранено за последващи обработка и действия.

По отношение на JDBC, WebLogic предоставя богат набор от функционалност и механизми за резервираност и преодоляване на грешки (failover) и за балансиране на натоварването (load balancing) между възлите на високостъпна система за бази данни, в посока устойчивост и надеждност на връзките към бази данни. Механизмите са автоматизирани, като е възможно да бъдат задействани и ръчно. Пример за устойчивост и надеждност на връзка към бази данни е следният: даден

Handwritten mark

Handwritten mark

Handwritten signature

Handwritten signature at the top of the page.

JDBC ресурс е конфигуриран така, че да сочи към два резервирани възела на една база данни или към две различни, но синхронизирани бази данни. При отпадане на единия възел, или едната база данни, връзката автоматично е пренасочена към другия наличен възел или база данни. По същия начин се осъществява и сценария за балансиране на натоварването между два възела или две бази данни. Конфигурацията на JDBC ресурса се съхранява като метаданни във файл и база данни, така че JDBC ресурса може лесно да бъде възстановен, преместен и дистрибутиран в една или повече WebLogic сървърни конфигурации.

За повече информация:

- Overview of JMS and Oracle WebLogic Server:
<https://docs.oracle.com/middleware/12213/wls/JMSAD/overview.htm#JMSAD-GUID-167A4438-4828-4E71-B3F9-7885F5422CB7>
- WebLogic Server Messaging:
<https://docs.oracle.com/middleware/12213/wls/INTRO/jms.htm#INTRO224>
- Administering the WebLogic Persistent Store:
<https://docs.oracle.com/middleware/12213/wls/STORE/toc.htm>
- JDBC Data Sources:
<https://docs.oracle.com/middleware/12213/wls/NOTES/whatsnew.htm#NOTES115>
- WebLogic Server Data Sources
<https://docs.oracle.com/middleware/12213/wls/INTRO/jdbc.htm#INTRO217>

NFREQ.47.

Ще предоставя вграден механизъм за самонастройка

Описание на реализацията:

WebLogic Server дава приоритет на работата и разпределя работните нишки въз основа на модел за изпълнение, който взема в предвид параметрите дефинирани от администратора, както и действителната производителност и пропускателна способност на изпълнението.

За да се определи приоритет на изпълнението на работата на дадено приложение, както и на сървъра като цяло, WebLogic Server предоставя работен мениджър наречен Work Manager. За да се справи с управлението на нишките и да извърши самонастройка, WebLogic Server имплементира Work Manager по подразбиране. Този Work Manager се използва от дадено приложение, когато в дескрипторите за внедряване на приложението не са посочени други работни мениджъри.

За повече информация:

- Using Work Managers to Optimize Scheduled Work:
https://docs.oracle.com/middleware/12213/wls/CNFGD/self_tuned.htm#CNFGD113
- Self-Tuning Thread Pool
https://docs.oracle.com/middleware/12213/wls/CNFGD/self_tuned.htm#CNFGD354

NFREQ.48.

Ще притежава официални резултати от тестове за производителност и продуктивност от реномирани упълномощени инстанции

Описание на реализацията:

Handwritten signature at the bottom of the page.

	<p>WebLogic Server има доказани дългогодишна репутация и присъствие като лидер в официални Benchmark резултати. Като пример, по-долу са приложени връзки към SPECjEnterprise2010 – Benchmark който измерва цялостната производителност за Java Enterprise Edition (Java EE) сървъри на приложения и разширява обхвата на показателя SPECjAppServer2004, както и към официални Benchmark публикации от производителя на WebLogic Server.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none">• SPECjEnterprise2010 Results: http://www.spec.org/jEnterprise2010/results/ http://www.spec.org/jEnterprise2010/results/res2017q3/• Oracle Benchmark Results: http://www.oracle.com/us/solutions/benchmark/apps-benchmark/benchmark-results-063877.html#SPECj
--	--

По отношение на възможностите за интеграция с други СУБД, предлаганата от Кандидата такава ще отговаря на следните изисквания:

	<p>NFREQ.49. JDBC data source за достъп до базата данни и управление на връзката с базата данни</p> <p><i>Описание на реализацията:</i></p> <p>В WebLogic Server връзката и достъпа до бази данни се реализира посредством конфигуриране на източниците на данни в Java Database Connectivity (JDBC) ресурс, наречен JDBC Data Source. Този ресурс се насочва и разгръща на една или повече сървърни инстанции в даден WebLogic домейн. WebLogic Server предоставя голяма гъвкавост при интеграцията със СУБД, като предлага различни типове JDBC Data Sources, съгласно архитектурата на базата данни и желаната нужда.</p> <p><i>За повече информация:</i></p> <ul style="list-style-type: none">• WebLogic Server Data Sources: https://docs.oracle.com/middleware/12213/wls/INTRO/jdbc.htm#INTRO215• Administering JDBC Data Sources for Oracle WebLogic Server: https://docs.oracle.com/middleware/12213/wls/JDBCA/toc.htm
	<p>NFREQ.50. Да осигурява поддръжка на: JDBC драйвери за връзка с широко разпространени СУБД, X/Open XA (XA) стандарта, Distributed Relational Database Architecture (DRDA) стандарта и two-phase commit protocol (2PC) протокола</p> <p><i>Описание на реализацията:</i></p> <p>Една от основните характеристики на WebLogic Server е управлението на транзакциите - сървърът предоставя пълна инфраструктура, която гарантира интегритета на транзакциите.</p> <p>В инсталационния си пакет WebLogic Server предлага широк набор от JDBC драйвери за връзка с различни СУБД, като например Oracle Database, MySQL, DB2, Informix,</p>

Handwritten signature

Microsoft SQL Server, Sybase и др.. В допълнение, WebLogic Server поддържа интеграция с JDBC драйвери от производителя на базата данни, както и със сертифицирани JDBC драйвери трети страни. WebLogic Server може да участва в различни сценарии за дистрибутирани транзакции координирани от трети страни или от самия WebLogic Server, като за целта напълно поддържа X/Open XA (XA), Distributed Relational Database Architecture (DRDA) и two-phase commit protocol (2PC), както и предоставя богат инструментариум за конфигуриране, координиране и управление на дистрибутирани транзакции.

За повече информация:

- Using JDBC Drivers with WebLogic Server:
https://docs.oracle.com/middleware/12213/wls/JDBCA/third_party_drivers.htm#JDBCA699
- Selecting a JDBC Driver:
https://docs.oracle.com/middleware/12213/wls/JDBCA/jdbc_datasources.htm#JDBCA140
- Developing JTA Applications for Oracle WebLogic Server
<https://docs.oracle.com/middleware/12213/wls/WLJTA/toc.htm>
- Administering JDBC Data Sources for Oracle WebLogic Server:
<https://docs.oracle.com/middleware/12213/wls/JDBCA/toc.htm>

NFREQ.51. Ще притежава устойчивост на транзакции

Описание на реализацията:

WebLogic Server предоставя вградено високоефективно решение за съхранение на подсистеми и ресурси, които изискват устойчивост, наречено WebLogic Persistent Store. Устойчивостта на транзакциите се реализира чрез съхраняване на информация за координираните транзакции (JTA Transaction Log, за кратко TLOG) във база данни или файлова система. За да се улесни възстановяването след срив, WebLogic Server предоставя услугата за възстановяване на транзакции, която автоматично се опитва да възстанови транзакциите при стартиране на системата. При (re)стартиране на системата, услугата за възстановяване на транзакции преглежда всички записи на регистрите на транзакциите (TLOG) за непълни транзакции и ги изпълнява съгласно типа транзакция и зададените и политики.

За повече информация:

- WebLogic Persistent Store:
<https://docs.oracle.com/middleware/12213/wls/STORE/overview.htm#STORE221>
- Using a JDBC TLog Store:
<https://docs.oracle.com/middleware/12213/wls/STORE/jdbc.htm#STORE291>
- Managing Transactions:
<https://docs.oracle.com/middleware/12213/wls/WLJTA/trxman.htm#WLJTA173>

NFREQ.52. Ще предоставя механизъм за ,failover' и ,load balancing' на връзките с бази данни (СУБД)

Описание на реализацията:

Handwritten signature

Handwritten signature

За осигуряване на ,failover' и ,load balancing' на връзките с бази данни, WebLogic Server предоставя вграден механизъм наречен JDBC Multi Data Sources. Това е абстракция на група генерични data sources, която е свързана с контекста на JNDI или локалното приложение, като нейното конфигуриране, управление и наблюдение се извършва от единната централизирана рамка за управление и наблюдение на WebLogic Server. За осигуряване на ,failover' и ,load balancing' конкретно на връзките с Oracle бази данни и Oracle RAC, WebLogic Server предоставя специализиран (за Oracle) вграден механизъм наречен Active GridLink Data Sources. В допълнение, с цел по-добро оползотворяване и разпределение на натоварването на връзките с бази данни, WebLogic Server предоставя още един вграден механизъм наречен Shared Pooling Data Sources, който позволява споделяне на връзките към бази данни от множество data sources.

За повече информация:

- [Configuring JDBC Multi Data Sources:](https://docs.oracle.com/middleware/12213/wls/JDBCA/jdbc_multidatasources.htm#JDBCA209)

https://docs.oracle.com/middleware/12213/wls/JDBCA/jdbc_multidatasources.htm#JDBCA209

- [Using Active GridLink Data Sources:](https://docs.oracle.com/middleware/12213/wls/JDBCA/gridlink_datasources.htm#JDBCA373)

https://docs.oracle.com/middleware/12213/wls/JDBCA/gridlink_datasources.htm#JDBCA373

- [Using Shared Pooling Data Sources:](https://docs.oracle.com/middleware/12213/wls/JDBCA/shared_pooling_ds.htm#JDBCA-GUID-39F03601-A999-4F57-85CB-6A157378E68B)

https://docs.oracle.com/middleware/12213/wls/JDBCA/shared_pooling_ds.htm#JDBCA-GUID-39F03601-A999-4F57-85CB-6A157378E68B

Handwritten mark

NFREQ.53. За пълно съответствие и добро поведение на бъдещата система е предимство, ако предлаганите СУБД и приложен Java EE сървър са от един и същи доставчик

Описание на реализацията:

WebLogic Server предоставя богата функционалност за връзка към всички известни СУБД, но специално за връзка към Oracle СУБД, WebLogic Server предоставя разширен специализиран инструментариум и функционалност, които значително увеличават производителността, работоспособността и надеждността, както на приложния сървър, така и на СУБД, опростяват операциите по администриране и поддръжка, намаляват риска от грешки и несъответствия и благоприятстват успешното внедряване и доброто поведение на системата.

За повече информация:

- [Using Oracle Databases with WebLogic Server:](https://docs.oracle.com/middleware/12213/wls/JDBCA/Using_OracleDatabase.htm#JDBCA-GUID-42495BA0-B1FA-4450-89C4-C3CCC1410B43)

https://docs.oracle.com/middleware/12213/wls/JDBCA/Using_OracleDatabase.htm#JDBCA-GUID-42495BA0-B1FA-4450-89C4-C3CCC1410B43

Handwritten mark

2.6. Обосновка за предложената архитектура

2.6.1. Относно софтуерната архитектура

Предложената софтуерна архитектура за реализация на системата гарантира, че:

Handwritten signature

- Технологичното решение отговаря на всички функционални изисквания;
- Технологичното решение отговаря на всички нефункционални изисквания;
- Технологичното решение е съобразено с концепцията и архитектурата на предвидената за разработка система, целяща да осигури оптимален и хармонизиран надзор в областта на корабоплаването в смесения румънско- български участък на р.Дунав от компетентните органи на България и Румъния;
- Технологичното решение отговаря на нормативните изисквания;
- Технологичното решение е базирано на световно утвърдени, актуални и перспективни технологии;
- За реализация на решението се предвижда използване на утвърден базов софтуер, отговарящ на всички заложен изисквания.

В предложената софтуерна архитектура е спазен т. нар. принцип SOLID (Single responsibility, Open-closed, Liskov substitution, Interface segregation and Dependency inversion):

S	Принцип на конкретна отговорност (Single responsibility principle) Компонентите трябва да имат конкретна, единична отговорност. В случая всяка функционална група на системата е обособена в конкретен компонент (модул), съобразно заложените изисквания.
O	Принцип отворени/затворени (Open/closed principle) Компонентите трябва да бъдат отворени за разширяване и затворени за промяна. Реализацията чрез отделни взаимосвързани компоненти (модули), осигурява изпълнението на този принцип.
L	Принцип на Лисков за заместването (Liskov substitution principle) Компонентите трябва да бъдат заменими с инстанции от техни подтипове без да се променя коректността на програмата. Това се постига чрез използване на стандартни интерфейси за комуникация между отделните компоненти. Конкретно това се използва при възможността за добавяне на инстанции с цел скалируемост на системата.
I	Принцип за разделяне на интерфейсите (Interface segregation principle) Много специфични за даден клиент интерфейси са по-добри от един интерфейс с общо предназначение. Затова е предвидено реализирането на множество специфични услуги за вътрешна и външна комуникация, групирани в независими модули.
D	Принцип за инверсия на зависимостите (Dependency inversion principle one) Нещата трябва да зависят от абстракциите, а не от конкретиката. Използването на утвърдени стандарти и стандартни интерфейси дава възможност за използване на различни компоненти, които ги реализират.

Благодарение на предложената модулна и ориентирана към услуги архитектура (SOA), която е съобразена със SOLID принципите и използването на утвърдени стандарти и технологии, системата ще предоставя възможност за бъдещо разширение, надграждане и интеграция с


други системи. Доразвиването, добавяне на нови модули и функционалности и интеграции с други системи ще бъде възможно без необходимост от промени във вече разработените, освен с цел надграждане и разширение.

Предложеното технологично решение предоставя възможност за скалиране при повишаване на натоварването без необходимост от промени в изходния код на системата и разработка на допълнителни компоненти. Системата ще позволява разширяване, резервиране и балансиране на натоварването между множество инстанции на сървъри с еднаква роля.

2.6.2. Относно системната архитектура

обосновка за избора на блейд шаси HPE BladeSystem c3000 Enclosure, блейд сървъри HPE ProLiant BL460c Gen9, дисков масив HPE MSA 2050 и тауър сървъри HPE ProLiant ML350 Gen10

Проектирането на хардуерната обезпеченост на интегрираната система за корабна инспекция е съпътствано от детайлен анализ на пазара на модулни сървърни системи и решения за дълготрайно съхранение на данни.


 Hewlett Packard Enterprise е категоричен дългогодишен лидер в областта според проучвания на водещите агенции за технологични изследвания Gartner Inc., IDC (International Data Corporation) и Frost & Sullivan. Решенията на компанията се отличават с високо ниво на иновации и постоянно внедряване на подобрения както в хардуерно, така и в софтуерно отношение.

Избраните продукти блейд шаси HPE BladeSystem c3000 Enclosure, блейд сървъри HPE ProLiant BL460c, дисков масив HPE MSA 2050 и тауър сървъри HPE ProLiant ML350 Gen10 отдавна присъстват в портфолиото на компанията и актуалните им версии са резултат от постоянната еволюция на продуктите в технологичен аспект.


Конкретните модели са съобразени с изискванията на Възложителя и са оразмерени, така че да позволяват оптимална работа за разработения софтуер при ниски нива на консумирана електрическа енергия.

Определящ фактор при селектирането на подходящ хардуер и системен софтуер е възможността за лесно управление и администриране на доставеното решение. Интегрирания мениджмънт интерфейс на блейд шасито позволява всички блейд сървъри да се управляват централизирано от едно място, което улеснява задачите по системното администриране.

Предимство на оферираното решение е високото ниво на интеграция между големия брой системни компоненти от един производител, които по дизайн са разработени да бъдат съвместими помежду си, което обуславя високоефективната им работа заедно – какъвто е случаят с блейд шасито, сървърите в него, дисковия масив и тауър сървърите.

 Многото опции за скалиране на параметрите и функциите на избраните продукти от портфолиото на HPE, предоставя възможности за бъдещо разширение на системата, в синхрон с нуждите на Възложителя, което гарантира защита на направените инвестиции.

На база на направеното проучване на пазара и точното оразмеряване на отделните компоненти са подбрани продуктите блейд шаси HPE BladeSystem c3000 Enclosure, блейд сървъри HPE



ProLiant BL460c, дисков масив HPE MSA 2050 и тауър сървъри HPE ProLiant ML350 Gen10, което е гаранция, че предложеното решение за хардуерна платформа, е гъвкаво, ще обезпечи нуждите за сигурна и надеждна работа и ще осигури непрекъснатост на бизнес процесите.

обосновка за избора на виртуализационната платформа VMware ESXi

Предвид високо-отговорните задачи, свързани с ефективното управление на хардуерните ресурси, които са съсредоточени във виртуализационния слой на предложеното решение, изборът на подходяща виртуализационна платформа е от ключово значение за успеха на проекта.

Компанията VMware Inc. от години доминира на пазара за софтуер за виртуализация и определя тенденциите в тази област. Решенията на компанията са на първо място в класациите и изследванията, провеждани от реномираните изследователски агенции Gartner Inc. и IDC и са високо оценени, защото внедряват върхови технологии, позволяващи гъвкавост и оптимално използване на възможностите на използваната хардуерна инфраструктура.

Платформата VMware ESXi е определена за най-подходяща за реализация на виртуализация в проекта след подробен преглед на широкия спектър от решения, налични в портфолиото на VMware Inc., и детайлен анализ на изискванията за системни ресурси, които ще са необходими за оптималната работа на софтуера на интегрираната система за корабни инспекции.

VMware ESXi осигурява чудесен баланс между налични функционалности и възможност за оптимизиране на разходите за виртуализационна инфраструктура.

Продуктът представлява хипервайзор от тип 1, специално разработен, така че да се инсталира директно на физическите машини, без да изисква наличие на операционна система. VMware ESXi осигурява директен достъп до хардуерните ресурси и техния лесен контрол по начин, който е много по-ефективен от своите конкуренти. Сред предимствата на VMware ESXi са:

- осигуряване на изключително високо ниво на утилизация на използваните хардуерни ресурси
- повишена производителност на виртуализационните функции
- централизирано управление
- намаляване на капиталовите и оперативните разходи
- използване на минимално количество хардуерни ресурси за работата на хипервайзора, което спестява средства

Необходимостта от надеждност и гъвкавост определя избора на VMware ESXi като сигурна виртуализационна платформа, която ще осигури оптимална работа на предложеното решение.

обосновка за избора на видеоконферентна платформа Microsoft Skype for Business и видеоконферентни терминали Polycom RealPresence Group 700

Изборът на платформа за провеждане на видеоконферентни разговори предопределя качеството на комуникацията между корабните инспектори и служителите на ИАМА, и възможностите за пълноценно общуване между екипи, разположени в отдалечени локации.

Handwritten signature

Предвид факта, че по време на видео разговорите, може да се споделя чувствителна информация, нивото на сигурност на решението е определящ фактор.

Компанията Polycom Inc. от десетилетия е сред лидерите на пазара на видеоконферентни системи и често получава признанието както на изследователските агенции Gartner Inc. и IDC, така и на милиони потребители в цял свят. Microsoft е световният лидер в разработката на софтуерни технологии и от 2009 година насам работи в тясно сътрудничество с Polycom Inc. за разработване на продукта си за работна колаборация Skype for Business.

Изборът на предложената платформа Skype for Business в комбинация с хардуерните терминали Polycom RealPresence Group 700, е резултат от проучване, взело предвид високите очаквания към качеството на видеоконферентните връзки и необходимостта от сигурност, която не позволява компромиси.

Microsoft Skype for Business on premise е най-добрият софтуер за споделена работна среда на пазара, предоставящ богата палитра от функционални възможности за видео разговори, аудио обаждания, изпращане на текстови съобщения, получаване в реално време на информация за състоянието на потребителя (presence). Microsoft Skype for Business ще бъде платформата, която ще обезпечи провеждането на висококачествени разговори с много участници.

Handwritten mark

Терминалите Polycom RealPresence Group 700 по дизайн са разработени за оптимална работа в Skype for Business среда и ще осигурят надеждна защитена комуникация с офисите на ИАМА в София, Русе и Лом, както и с офиса на партньорската организация в Констанца, Румъния. Продуктът е представител на високия сегмент от портфолиото на Polycom Inc., защото внедрява върхови технологии, в пълно съответствие с популярните в цял свят стандарти и протоколи.

Оферираните софтуерни клиенти Skype for Business и хардуерни терминали Polycom RealPresence Group 700, които използват платформата Microsoft Skype for Business осигуряват огромни възможности за скалиране на мащаба на решението и интегрирането на други участници във видео комуникацията.

2.6.3. Относно базовия софтуер

обосновка за избора на Oracle Database релационна база от данни

Handwritten mark

Oracle е водещият доставчик на бази данни за глобалните индустрии според ИТ анализаторите. Използваната технология на съхранение и обработка на данните, предоставена от Oracle Database, я правят изключително функционална и в следствие полезна за редица корпорации и институции, които се занимават ежедневно с транзакционни обработки, изискващи производителност и надеждност при съхранение на данни.

Ефективността в обработките и съхранението на данните е основната характеристика, която разработчиците търсят при избора на система за бази данни. Oracle Database е известна с това, че поддържа технически софтуерни възможности, с които автоматично управлява и оползотворява хардуерния ресурс – памет, CPU, диск, на машините в полза на производителността на системите.

Handwritten signature

Handwritten signature

Oracle Database работи еднакво ефективно и интегрирано, както в Windows OS, така и в Linux OS, и в UNIX базирани среди. Това дава свобода на бъдеща миграция на системата между различни OS среди и осигурява спазване на принципите на свободна и лоялна конкуренция.

Във вертикална и хоризонталната скалируемост Oracle Database е водеща. За разлика от всички други релационни бази от данни, които са предимно скалируеми във вертикала, Oracle Database поддържа работа с ресурса на обща клъстърна платформа от HW, където множество сървъри се комбинират, за да образуват клъстер за една и съща база данни. Така че, при необходимост от допълнителни ресурси, възложителя може да добави сървър като втори клъстерен нод, като използва технологията Oracle Real Application Cluster. Това значително увеличава степента на мащабируемост.

В скалируемостта на дисковите дялове Oracle Database също е водеща. При необходимост могат да се добавят дискови дялове към Oracle Database, която да ребалансира блоковете с данни върху тях автоматично, като поддържа едновременно с това копие (mirroring) на блоковете на две или на три места, чрез технологията Oracle Automatic Storage Management част от Oracle Database.

Oracle Database има гъвкава архитектура, която я прави предпочитана, защото тя позволява много лесно да се организират различни приложения във високонадежден, клъстерен режим на работа. Работата на Oracle Database върху обща клъстърна платформа от HW е надеждна дори и ако някои от сървърите, участници в клъстера откаже работа. Oracle Database не спира достъпа на потребителите към базата данни и не губи промените от транзакциите, изпълнявани от потребителските сесии към отказалия сървър. Работата на Oracle Database в клъстърна среда е изключително сигурна и високонадеждна с гаранция за целостност на данните, променени от транзакциите от отпадналия сървър. При това, новите потребителски сесии се разпределят на оцелелия сървър напълно автоматично и базата от данни не прекъсва работа.

Oracle Database предоставя предпочитана технология за бази от данни, с оглед на поддръжка и елиминирание на спирането на работа за целите на профилактика и ъпгрейд.

При подновяване на дискове или дискови масиви, в рамките на процедурата, както и за в бъдеще, технологията Oracle Automatic Storage Management, като част от Oracle Database, позволява ребалансиране на данните между наличните и новодобавени логически дискови дялове, като това става докато Oracle Database работи с потребителски сесии и нормално обработва данни. При добавянето на всеки един нов дял ще има достатъчно място и след като се ребалансират данните, вече е възможно отнемане на „стар“ дял и последващо ново автоматично ребалансиране. И системата е в готовност за ново добавяне на дисков дял – един или повече. По този начин е възможно да се прехвърлят всички данни автоматично от дисков масив на дисков масив, в онлайн режим на работа на Oracle Database с приложението, без прекъсване на работата на системата.

Oracle Database позволява на администраторите безпрецедентна гъвкавост да променят табличните физически атрибути и да трансформират структурата на данните и таблиците, като същевременно позволяват на потребителите пълен достъп до базата данни. Тази способност ще даде надеждност на системата в работна среда и ще позволи лесно, безопасно и бързо надстройване на приложението в следващи фази на проекта. С оглед дейността и важността на

разработваната система, тази функционалност на Oracle Database релационната база от данни е положителна.

обосновка за избора на Oracle WebLogic Server пред друг сървър за приложения

Предимства на Oracle WebLogic Server:

- Вградена интеграция с Oracle Database - WebLogic Server се интегрира уникално с Oracle Database посредством набор от функционалности и характеристики изобретени специално за Oracle Database, които осъществяват много по-бързо изпълнение на транзакциите, осигуряват по-надеждни и по-сигурни връзки към базата данни, по-устойчиви и консистентни транзакции, по-добро управление и наблюдение на връзките и транзакциите, което пряко влияе на производителността и надеждността на системата, както и на удовлетвореността на крайните потребители
 - Облекчаване на ИТ операциите и съкращаване времето за привеждане на системата в продукция - конфигурирането на тези функционалности и на самите връзки към Oracle Database става посредством графични интерфейси с визуални 'помощници'
 - Видимост в изпълнението на транзакциите и бързо диагностициране на проблеми
 - Защита на данните благодарение на вградената функционалност/механизми за сигурност на WebLogic Server към Oracle Database
 - Намален непринуден престой на системата благодарение на резервираността на връзките (failover) и равномерното разпределение на натоварването (load balancing) към СУБД
- Предложените СУБД и сървър за приложения са от един и същи доставчик, което дава следните предимства:
 - Сервизните заявки към поддръжката на доставчика се комуникират и обслужват от една точка - по-лесно, по-бързо и по-ефективно управление на инцидентите и поддръжката и отстраняването на проблеми
 - Системата е в съответствие - издаването на опреснения и подобрения на СУБД и сървърът за приложения са от една точка - двупосочно съгласувани, стандартизирани и в съответствие; рискът от неподдръжани функционалности и разлика във версиите м/у СУБД и сървърът за приложения е елиминиран
 - WebLogic Server отговаря 100% на изисквания към сървъра за приложения от точка '8.1.2.2.3'
 - WebLogic Server е от доказан доставчик, сървърът е дългогодишен лидер в тестовете за производителност, има множество референции от широкото му използване в световен мащаб от голям брой компании от различни индустрии, включително в публичният сектор в България, което значително намалява риска от неуспешното изпълнение на проекта поради неконсистентен продукт в системата.

В обобщение, Oracle WebLogic Server и Oracle Database, ползвани в заедно в решението, гарантира по-добра работоспособност и ефективност на системата (производителност, надеждност, защита на данните, занижен риск от несъответствия и непринуден престой, ниска крайна цена за притежание), което допринася за успешното изпълнение на проекта.

Предложените продукти Oracle Database и Oracle WebLogic Server ползвани в комбинация в решението биха дали на системата по-добри работоспособност, производителност,

надеждност и ефективност, както и по-добра крайна цена за притежание, което ще допринесе за успешното изпълнение на проекта.

3. Общи нефункционални характеристики по НОИИСРЕАУ

В настоящата точка сме описали визията на Кандидата за реализиране на нефункционалните характеристики на системата. Реализацията е в тясна връзка с предложената в т. 2.2 на настоящото приложение софтуерна архитектура.

В таблицата по-долу са представени нефункционалните изисквания, на които ще отговаря Системата:

№ по ред	Изискване
NFREQ.54.	<p>Електронна идентификация на потребителите</p> <ol style="list-style-type: none">1. Електронната идентификация на всички потребители ще бъде реализирана в съответствие с изискванията на Регламент ЕС 910/2014 и Закона за електронната идентификация;2. Ще бъде реализирана интеграция с националната схема за електронна идентификация съгласно изискванията на Закона за електронната идентификация и действащите нормативни правила за оперативна съвместимост. За целта подсистемата за автентикация и оторизация на потребителите ще поддържа интеграция с външен доставчик на идентичност - в случая с Центъра за електронна идентификация към Държавна агенция „Електронно управление“. Реализацията на интеграцията ще бъде осъществена по стандартни протоколи SAML 2.0 и/или OpenID Connect;3. Системата ще поддържа и стандартен подход за регистрация на потребители с потребителско име и парола - за потребители, които нямат издадени удостоверения за електронна идентичност, и за потребители, които желаят да продължат да използват електронни административни услуги с КЕП;4. Процесът по регистрация на потребители ще бъде максимално опростен и бърз и ще включва следните специфични стъпки:<ul style="list-style-type: none">○ Визуализиране на информацията относно стъпките по регистрация и информацията във връзка с процеса за потвърждаване на регистрацията и активиране на потребителския профил. Съвети към потребителите за проверка на настройките на имейл клиентите, свързани с блокиране на спам, и съвети за включване на домейна на Възложителя в "бял списък";○ Избор на потребителско име с контекстна валидация на полетата (in-line validation), включително и за избраното потребителско име;

№ по ред	Изискване
	<ul style="list-style-type: none">○ Избор на парола с контекстна валидация на полето (in-line validation) и визуализиране на сложността на паролата като "слаба", "нормална" и "силна";○ Реализиране на функционалност за потвърждение и активиране на регистрацията чрез изпращане на съобщение до регистрирания имейл адрес на потребителя с хипер-линк, с еднократно генериран токън с ограничена времева валидност за потвърждение на регистрацията. Възможност за последващо препращане на имейла за потвърждение, в случай че е бил блокиран от системата на потребителя. <p>5. При реализиране на вход в Системата с удостоверение за електронна идентичност, по Националната схема за електронна идентификация, Системата ще използва потребителския профил, създаден в Системата за електронна идентификация, чрез интерфейси и по протоколи съгласно подзаконовата нормативна уредба към Закона за електронната идентификация. В случай че даден потребител има регистриран потребителски профил в Системата, който е създаден преди въвеждането на Националната схема за електронна идентификация, Системата ще предлага на потребителя възможност за "сливане" на профилите и асоцииране на локалния профил с този от Националната система за електронна идентификация. Допустимо е Системата да поддържа и допълнителни данни и метаданни за потребителите, но само такива, които не са включени като реквизити в централизирания профил на потребителя в Системата за електронна идентификация.</p> <p>6. Системата ще се съобразява с предпочитанията на потребителите, дефинирани в потребителските им профили в Системата за електронна идентификация, по отношение на предпочитаните комуникационни канали и канали за получаване на нотификации.</p>
NFREQ.55.	Отворени данни <ul style="list-style-type: none">1. Ще бъде разработен и внедрен онлайн интерфейс за свободен публичен автоматизиран достъп до документите, информацията и данните в Системата (наричани заедно „данните“). Интерфейсът ще осигурява достъп до данните в машинночетим, отворен формат, съгласно всички изисквания на Директива 2013/37/ЕС за повторна употреба на информацията в общественния сектор и на Закона за достъп до обществена информация;2. Ще бъде предвидена разработката и внедряването на отворени онлайн интерфейси и практически механизми, които да улеснят търсенето и достъпа до данни, които са на разположение за повторна употреба, като

№ по ред	Изискване
	<p>например списъци с основни документи и съответните метаданни, достъпни онлайн и в машинночетим формат, както и интеграция с Портала за отворени данни http://opendata.government.bg, който съдържа връзки и метаданни за списъците с материали, съгласно изискванията на Закона за достъп до обществена информация (ЗДОИ);</p> <p>3. Ще се разработи и ще се поддържа актуално публично описание на всички служебни и отворени интерфейси, отворените формати за данни, заедно с историята на промените в тях, в структуриран машинночетим формат;</p> <p>4. Ще се разработят процеси по предоставяне на данни в отворен, машинночетим формат заедно със съответните метаданни. Форматите и метаданните ще съответстват на официалните отворени стандарти.</p>
NFREQ.56.	<p>Формиране на изгледи</p> <p>Ще бъде предоставена възможност на потребителите на Системата да достъпват различна информация чрез филтриране, пренареждане и агрегиране на данните. Резултатът ще се представя чрез:</p> <ul style="list-style-type: none">• Визуализиране на таблици;• Графична визуализация на екран;• Разпечатване на хартиен носител;• Експорт на данни в един или в няколко от изброените формати – ODF, Excel, PDF, HTML, TXT, XML, CSV.
NFREQ.57.	<p>Администриране на Системата</p> <p>Системата ще осигури функционалности за администриране на потребителите и правата за достъп.</p> <p><i>Начин на реализация:</i></p> <p>За всяка роля за достъп ще може да се дефинират съответстващи права за достъп до потребителския интерфейс и вътрешните услуги на системата.</p> <p>Този модул ще предоставя възможност за поддържане, преглеждане, създаване и редактиране на акаунтите на отделните потребители посредством реализиране на гъвкава функционалност за управление на потребителските им роли и права на достъп. Модулът ще бъде достъпен само за администраторите на системата.</p> <p>Потребителите ще достъпват приложението след идентификация с потребителско име и парола. За потребителско име потребителите ще могат да посочват и електронната си поща. Главният администратор на приложението ще може по всяко време да създаде нов потребител или да деактивира акаунт на потребител и да преустанови достъпа му до приложението.</p> <p>Правата за достъп до ресурсите и възможните действия с тях ще се определят на база на конфигурирани потребителски роли. Потребител с роля администратор</p>

№ по ред	Изискване
	<p>(или съответните права) ще може да създава и конфигурира потребителски групи (роли). В една група могат да участват един или няколко потребители. На всяка потребителска група (роля) ще може да бъде променяна конфигурацията. При редактиране на правата на дадена роля (група), едновременно ще се променят правата на всички участници в тази група. На всеки потребител ще може да се назначава една, повече от една роли или права за достъп. Един потребител ще може да бъде участник в много потребителски групи. Той ще наследява правата от всички групи, в които участва. Администраторът на системата освен да променя правата ще може да забранява и/или спира достъпа на потребителя върху ресурсите на системата.</p> <p>В зависимост от нуждите на Възложителя, начина на администриране на потребителите ще бъде уточнен, при изпълнение на проекта на етап Планиране и бизнес анализ.</p>
<p>NFREQ.58.</p>	<p>Авторски права и изходен код</p> <ol style="list-style-type: none"> 1. Всички компоненти на Системата, които ще бъдат разработени 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. Приложимите и допустими лицензи за софтуер с отворен код са: <ul style="list-style-type: none"> ○ GPL (General Public License) 3.0 ○ LGPL (Lesser General Public License) ○ AGPL (Affero General Public License) ○ Apache License 2.0 ○ New BSD license ○ MIT License ○ Mozilla Public License 2.0 4. Изходният код (Source Code), разработван по проекта, както и цялата техническа документация ще бъдат публично достъпни онлайн като софтуер с отворен код от първия ден на разработка чрез използване на система за контрол на версиите и хранилището по чл. 7в, т. 18 от ЗЕУ; 5. Всички доработки и/или новосъздадени софтуерни модули в рамките на настоящата поръчка, ще станат собственост на възложителя след окончателното им приемане.

№ по ред	Изискване
NFREQ.59.	<p data-bbox="548 21 689 150"><i>[Handwritten signature]</i></p> <p data-bbox="357 226 848 260">Системна и приложна архитектура</p> <ol data-bbox="404 291 1480 1875" style="list-style-type: none">1. Системата ще бъде реализирана като разпределена модулна информационна система. Системата ще бъде реализирана със стандартни технологии и ще поддържа общоприети комуникационни стандарти, които ще гарантират съвместимост на Системата с бъдещи разработки. Съществуващите модули функционалности ще бъдат рефакторирани и/или надградени по начин, който ще осигури изпълнението на настоящето изискване;2. Бизнес процесите и услугите ще бъдат проектирани колкото се може по-независимо с цел по-лесно надграждане, разширяване и обслужване. Системата ще бъде максимално параметризирана и ще позволява настройка и промяна на параметрите през служебен (администраторски) потребителски интерфейс;3. При разработката, тестването и внедряването на Системата ще прилагаме наложили се архитектурни (SOA, MVC или еквивалентни) модели и дизайн-шаблони, както и принципите на обектноориентирания подход за разработка на софтуерни приложения;4. Системата ще бъде реализирана със софтуерна архитектура, ориентирана към услуги - Service Oriented Architecture (SOA);5. Взаимодействията между отделните модули в Системата 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; За всеки от отделните модули/функционалности на Системата ще се реализират и опишат приложни програмни интерфейси – Application Programming Interfaces (API). Приложните програмни интерфейси ще бъдат достъпни и за интеграция на нови модули и други вътрешни или външни системи;6. Приложните програмни интерфейси и информационните обекти задължително ще поддържат атрибут за версия;7. Версията на програмните интерфейси, представени чрез уеб-услуги, ще поддържа версията по един или няколко от следните начини:<ul data-bbox="501 1627 1042 1757" style="list-style-type: none">○ Като част от URL-а;○ Като GET параметър;○ Като HTTP header (Асепт или друг);8. За всеки отделен приложен програмен интерфейс ще бъде разработен софтуерен комплект за интеграция (SDK) на поне две от популярните развойни платформи (.NET, Java, PHP); <p data-bbox="548 1908 721 2102"><i>[Handwritten signature]</i></p>

№ по ред	Изискване
	<p>9. Системата ще осигурява възможности за разширяване, резервиране и балансиране на натоварването между множество инстанции на сървъри с еднаква роля;</p> <p>10. При разработването на Системата ще се предвидят възможни промени, продиктувани от непрекъснато променящата се нормативна, бизнес и технологична среда. Основно изискване се явява необходимостта информационната система да бъде разработена като гъвкава и лесно адаптивна, като отчита законодателни, административни, структурни или организационни промени, водещи до промени в работните процеси;</p> <p>11. Изпълнителят ще осигури механизми за реализиране на бъдещи промени в Системата без промяна на съществуващия програмен код. Когато това не е възможно, времето за промяна, компилиране и пускане в експлоатация ще е сведено до минимум. Бъдещото развитие на Системата ще се налага във връзка с промени в правната рамка, промени в модела на работа на потребителите, промени във външни системи, интегрирани със Системата, отстраняване на констатирани проблеми, промени в модела на обслужване и др. Такива промени ще се извършват през целия период на експлоатация на Системата, включително и по време на гаранционния период;</p> <p>12. Архитектурата на Системата и всички софтуерни компоненти (системни и приложни) ще бъдат така подбрани и/или разработени, че ще осигуряват работоспособност и отказоустойчивост на Системата, 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);</p> <p>13. Ще проектираме, подготвим, инсталираме и конфигурираме като минимум следните среди за Системата: тестова, стейджинг, продуктивна;</p> <p>14. Системата ще бъде разгърната върху съответните среди (тестова за вътрешни нужди, тестова за външни нужди, стейджинг и продуктивна);</p> <p>15. Тестовата среда за външни нужди ще бъде създадена и поддържана като "Sandbox", така че да е достъпна за използване и извършване на интеграционни тестове от разработчици на информационни системи, включително такива, изпълняващи дейности за други администрации или бизнеса, с цел по-лесно и устойчиво интегриране на съществуващи и бъдещи информационни системи. Тестовата среда за външни нужди ще бъде напълно отделна от останалите среди и нейното използване няма да влияе по никакъв начин на нормалната работа на останалите среди или да</p>

000000
000055

№ по ред	Изискване
	<p>създава каквито и да било рискове за информационната сигурност и защитата на личните данни;</p> <p>16. По-подробно описание на добрите практики, които ще прилагаме по отношение на всеки аспект от системната и приложната архитектура на Системата са описани в <i>т. 2 Архитектура на системата</i>;</p> <p>17. Ще бъде създаден административен интерфейс, чрез който ще може да бъде извършвана конфигурацията на софтуера;</p> <p>18. Всеки обект в системата ще има уникален идентификатор;</p> <p>19. Записите в регистрите няма да подлежат на изтриване или на промяна, а всяко изтриване или промяна ще представлява нов запис.</p> <p>По-подробно системната архитектура с описание на елементите е представена в <i>т. 2 Архитектура на системата</i> от настоящия документ.</p>
NFREQ.60.	<p>Повторно използване (преизползване) на ресурси и готови разработки</p> <p>Проектът ще преизползва налични публично достъпни инструменти, библиотеки и платформи с отворен код.</p> <p>За реализацията на Системата ще се използват в максимална степен софтуерни библиотеки и продукти с отворен код.</p> <p><i>Начин на реализация:</i></p> <p>Подход за избор на отворени имплементации и продукти</p> <p>За реализацията на дадена техническа функционалност съществуват множество отворени алтернативни проекти, които могат да се използват в настоящата Система.</p> <p>Поради факта че настоящият проект ще се изпълнява по обществена поръчка, той ще бъде отворен по смисъла на ЗЕУ и ще отговаря на следните критерии:</p> <ul style="list-style-type: none">• За разработката му ще се използва система за управление на версиите на кода и ще е наличен механизъм за съобщаване на несъответствия и приемане на допълнения;• Ще има разработена техническа документация за актуалната стабилна версия;• Ще има повече от един активен програмист, работещ по развитието на Системата;• Ще има възможност за предоставяне на комерсиална поддръжка;• Няма да има намаляваща от година на година активност;• Ще има разработени unit tests с code coverage над 50%, а проектът ще използва Continuous Integration (CI) подходи – build bots, unit tests run, регулярно използване на статични/динамични анализатори на кода и др.

Plus


№ по ред	Изискване
	<ul style="list-style-type: none">• По възможност проектите да са подкрепени от организации с идеална цел, държавни или комерсиални организации. <p>Ако бъдем избрани за Изпълнител, ще спазим препоръките за преизползването на проекти, финансирани със средства на Европейския съюз, както и на такива, в които имаме активни разработчици. Използването на closed source и на инструменти, библиотеки, продукти и системи с платен лиценз ще става за наша сметка, като това ще бъде допустимо за случаите, когато липсва подходяща свободна алтернатива с необходимата функционалност или тя не отговаря на горните условия.</p> <p>Ако бъдем избрани за изпълнител ще осигурим поддръжка от комерсиална организация, развиваща основните отворени продукти, които ще бъдат използвани като минимум за операционните системи и софтуерните продукти за управление на базите данни.</p> <p>Подход за работа с външните софтуерни ресурси</p> <p>При използването на свободни имплементации на софтуерни библиотеки ще се организира копие (fork) на съответното хранилище в общото хранилище за проекти с отворен код, финансирани с публични средства в България (към момента https://github.com/governmentbg). При използване на свободните библиотеки компоненти ще задаваме за "upstream repo" хранилищата в областта governmentbg, като задължително ще се реферира използваната версия/commit identifier.</p> <p>Когато се налага промяна в изходния код на използван софтуерен компонент, промените ще се извършват във fork хранилището на governmentbg в съответствие с изискванията на основния проект. Ако бъдем избрани за изпълнител ще извършим необходимите действия за включване на направените промени в основния проект чрез "pull requests" и извършване на необходимите изисквани от разработчиците на основния проект промени до приемането им. Тези дейности ще бъдат извършвани по време на целия проект.</p> <p>При установяване на наличие на нови версии на използваните проекти ще се извърши анализ на влиянието върху настоящата система. В случаите, при които се оптимизира използвана функционалност, отстраняват се пропуски в сигурността, стабилността или бързодействието, новата версия ще се извлича и използва след успешното изпълнение на интеграционните тестове.</p>
NFREQ.61.	<p>Изграждане и поддръжка на множество среди</p> <p>Ако бъдем избрани за изпълнител ще изградим и да поддържаме минимум следните логически разделени среди:</p> <ul style="list-style-type: none">○ Development;

↑

[Signature]

[Signature]


Handwritten signature

№ по ред	Изискване
	<p>Чрез Development средата ще се осигурява работата по разработката, усъвършенстването и развитието на Системата. В тази среда ще са налични и допълнителните софтуерни системи и инсталации, необходими за управление на разработката – continuous integration средства, системи за автоматизирано тестване и др.</p> <ul style="list-style-type: none">○ Staging; <p>Чрез Staging средата ще се извършват тестове преди разгръщане на нова версия от Development средата върху Production средата. В нея ще се извършват всички интеграционни тестове, както и тестовете за натоварване.</p> <ul style="list-style-type: none">○ Production. <p>Това е средата, която е публично достъпна за реална експлоатация и интеграция със съответните външни системи и услуги.</p> <p>Управлението на средите ще става чрез автоматизирана система за провизиране и разгръщане на системните компоненти. При необходимост от страна на Възложителя Изпълнителят ще съдейства за изграждането на нови системни среди.</p>
NFREQ.62.	<p>Процес на разработка, тестване и разгръщане</p> <p>Процесите, свързани с развитието на Системата ще гарантират висока прозрачност и възможност за обществен контрол над всички разработки по проекта. Изграждането на доверие в гражданите и в бизнеса налага радикално по-висока публичност и прозрачност чрез отворена разработка и публикуването на системите компоненти под отворен лиценз от самото начало на разработката. По този начин гражданите биха могли да съдействат в процесите по развитие и тестване на разработките през целия им жизнен цикъл.</p> <p>Всички софтуерни приложения, системи, подсистеми, библиотеки и компоненти, които са необходими за реализацията на Системата ще бъдат разработвани като софтуер с отворен код и ще бъдат достъпни в публично хранилище. Към настоящия момент ще се използва общото хранилище за проекти с отворен код, финансирани с публични средства в България (към момента https://github.com/governmentbg).</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>Ако бъдем избрани за Изпълнител, ще извършим 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</p>

Handwritten scribble

Handwritten signature

Handwritten signature

№ по ред	Изискване
	<p>възможност за активно участие от страна на гражданите и събиране на информация с цел обратна връзка за функционалната реализация на разработеното приложение. Това ще включва следните средства за изпълнение на изискването:</p> <ul style="list-style-type: none">• Анкетни карти и въпросници;• Предоставяне на достъп до публичната част посредством тестови акаунт;• Форма за обратна връзка. <p>За всеки един разработван компонент, ако бъде избран за Изпълнител, Кандидатът ще покрие следните изисквания за гарантиране на качеството на извършваната разработка и на крайния продукт:</p> <ul style="list-style-type: none">○ Документиране на Системата в изходния код, минимум на ниво 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 Системата, 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.
NFREQ.63.	<p>Контрол на натоварването и защита от DoS/DDoS атаки</p> <ol style="list-style-type: none">1. Системата 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. Системата ще позволява конфигуриране от страна на администраторите на лимитите за отделни страници, уеб-услуги и ресурси, които се достъпват с отделен URL/URI.3. Системата 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.

Handwritten signature

Handwritten signature

№ по ред	Изискване
NFREQ.64.	<p>Кохерентно кеширане на данни и заявки</p> <p>Като минимум разпределен кохерентен кеш ще се предвиди при:</p> <ol style="list-style-type: none"> 1. Извличане на информация от номенклатури и атомични данни за статус и актуално състояние на партии от регистри в информационните системи; 2. Извличане на информация от предефинирани периодични справки; 3. Информация от лога на транзакциите при достъп с електронно-ИД до дадена услуга; 4. Информация за извършените плащания; 5. 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.65.	<p>Бързодействие</p> <p>Системата ще бъде проектирана така, че да осигурява висока производителност и минимално време за отговор на заявки.</p> <p>Ще бъдат създадени тестове за натоварване.</p>
NFREQ.66.	<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); • В случаите когато клиентският браузър/клиент не поддържа HTTP/2, ще бъде предвиден fall-back механизъм към HTTP/1.1. Тази възможност ще може лесно да се преконфигурира в бъдеще и да отпадне, когато браузърите/клиентите, неподдържащи HTTP/2, станат незначителен процент.
NFREQ.67.	<p>Подписване на документите</p>

Handwritten mark at the top center of the page.

№ по ред	Изискване
	<p>В случай, че по време на изпълнение на поръчката, на етап бизнес анализ, Изпълнителят идентифицира нужда за реализиране на функционалност за електронно подписване на документи, то тя ще отговаря на следните изисквания:</p> <ul style="list-style-type: none">▪ 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.▪ 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 ще се избягва зареждането на цялото съдържание в оперативната памет.▪ Ще поддържа подписване на електронни изявления и електронни документи и 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 г.;▪ 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 ще се осигури максимална съвместимост на процеса на подписване с всички съвременни браузъри. Такава реализация може да бъде осъществена чрез:<ul style="list-style-type: none">○ използване на стандартни компоненти с отворен код, отговарящи на горните условия, които са разработени по други проекти на държавната администрация и са достъпни в хранилището, поддържано от Държавна агенция „Електронно управление“ – при наличие на такива компоненти в хранилището те ще се преизползват и само ще бъдат интегрирани в Системата;○ използване на плъгин-модули с отворен код, достъпни за най-разпространените браузъри (Browser Plug-ins), които са адаптирани и поддържат унифицираните профили на електронните подписи, издавани от ДДУ в ЕС, и съответните драйвери за крайни устройства за четене на сигурни носители или по стандартизиран в националната нормативна уредба протокол за подписване извън браузъра;○ чрез интеграция с услуги за отдалечено подписване, предлагани от доставчици на доверителни услуги в ЕС.
NFREQ.68.	Качество и сигурност на програмните продукти и приложенията

Handwritten mark on the left side of the table.

Handwritten mark at the bottom left of the page.

Handwritten mark at the bottom center of the page.

Handwritten signature at the top of the page.

№ по ред	Изискване
	<ul style="list-style-type: none">▪ Ще бъде предвидено спазването на добри практики на софтуерната разработка – 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);▪ Публичните модули, които ще предоставят информация и електронни услуги в Интернет, ще отговарят на актуалните уебстандарты за визуализиране на съдържание.
NFREQ.69.	Информационна сигурност и интегритет на данните <ul style="list-style-type: none">▪ Съхранението на пароли на администратори, на вътрешни и външни потребители и на акаунти за достъп на системи няма да бъде в явен вид. Всички пароли ще бъдат защитени с подходящи сигурни алгоритми (напр. BCrypt, PBKDF2, scrypt (RFC 7914) за съхранение на пароли и където е възможно, ще се използва и прозрачно криптиране на данните в СУБД със сертификати (transparent data-at-rest encryption);▪ Ще бъде предвидена система за ежедневно създаване на резервни копия на данните, които да се съхраняват извън инфраструктурата на системата;▪ Няма да се използват Self-Signed сертификати за публични услуги;▪ 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). Ежегодното издаване и подновяване на сертификата ще бъде включено като разходи и дейности в гаранционната поддръжка за целия срок на поддръжката;▪ 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/). За нуждите на автентикация с КЕП ще се предвиди имплементирането на обратен прокси сървър (Reverse Proxy) с балансиране на натоварването, който да препраща клиентските сертификати към вътрешните приложни сървъри с нестандартно поле (дефинирано в процеса на разработка на Системата) в HTTP Header-а. Схемата за проксиране на заявките ще бъде защитена от Spoofing;▪ Като временна мярка за съвместимост настройките на уебсървърите и Reverse Proxy сървърите ще бъдат балансирани така, че Системата ще позволява използване и на клиентски браузъри, поддържащи по-стария протокол TLS

Handwritten signature on the left margin.

Handwritten signature on the left margin.

Handwritten signature at the bottom of the page.

№ по ред	Изискване
	<p>1.1. Това изключение от общите изисквания за информационна сигурност няма да се прилага за достъпа на служебни потребители от държавната администрация и доставчици на обществени услуги, които имат служебен достъп до ресурси на Системата;</p> <ul style="list-style-type: none">▪ При разгръщането на всички уебслужби (Web Services) ще се използва единствено протокол HTTPS със задължително прилагане на минимум TLS 1.2;▪ 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;▪ При проектирането и разработката на компонентите на Системата и при подготовката и разгръщането на средите ще се спазват последните актуални препоръки на OWASP (Open Web Application Security Project);▪ Ще бъде изграден модул за проследимост на действия и събития в Системата. За всяко действие (добавяне, изтриване, модификация, четене) ще съдържа следните атрибути:<ul style="list-style-type: none">○ Уникален номер;○ Точно време на възникване на събитието;○ Вид (номенклатура от идентификатори за вид събитие);○ Данни за информационна система, където е възникнало събитието;○ Име или идентификатор на компонент в информационната система, регистрирал събитието;○ Приоритет;○ Описание на събитието;○ Данни за събитието.▪ Астрономическото време за удостоверяване настъпването на факти с правно или техническо значение се отчита с точност до година, дата, час, минута, секунда и при технологична необходимост - милисекунда, изписани в съответствие със стандарта БДС ISO 8601:2006;▪ Астрономическото време за удостоверяване настъпването на факти с правно значение и на такива, за които се изисква противопоставимост, ще бъде удостоверявано с електронен времеви печат по смисъла на Глава III, Раздел 6 от Регламент ЕС 910/2014. Ще бъде реализирана функционалност за получаване на точно астрономическо време, отговарящо на горните условия, и от доставчик на доверителни услуги или от държавен орган, осигуряващ такава услуга, отговаряща на изискванията на RFC 3161;▪ Ще бъдат проведени тестове за проникване (penetration tests), с които ще се идентифицират и коригират слаби места в сигурността на Системата.

№ по ред	Изискване
NFREQ.70.	<p>Изисквания за използваемост и достъпност</p> <ul style="list-style-type: none"> ▪ Всички ресурси ще са достъпни чрез GET заявка на уникален адрес (URL). Няма да се използва POST за достигане до формуляр за подаване на заявление, за генериране на справка и други; ▪ Функционалностите на потребителския интерфейс на Системата ще бъдат независими от използваните от потребителите интернет браузъри и устройства, при условие че последните са версии в период на поддръжка от съответните производители. Ще бъде осигурена възможност за ползване на публичните модули на приложимите услуги през мобилни устройства – таблети и смарт-телефони, чрез оптимизация на потребителските интерфейси за мобилни устройства (Responsive Design); ▪ Няма да се използва Капча (Captcha) като механизъм за ограничаване на достъпа до документи и/или услуги. Алтернативно, Системата ще поддържа "Rate Limiting" и/или "Throttling". Captcha ще се използва единствено при идентифицирани много последователни опити от предполагаем „бот“; ▪ Ще бъде осигурен бърз и лесен достъп до електронните услуги и те ще бъдат промотирани с подходящи навигационни елементи на публичната интернет страница – банери, елементи от главното меню и др.; ▪ Публичните уеб страници на Системата ще бъдат проектирани и оптимизирани за ефективно и бързо индексирание от търсещи машини с цел популяризиране сред потребителите и по-добра откриваемост при търсене по ключови думи и фрази. При разработката на страниците и при изготвяне на автоматизираните процедури за разгръщане на нова версия на Системата ще се използват инструменти за минимизиране и оптимизация на размера на изходния код (HTML, JavaScript и пр.) с оглед намаляване обема на файловете и по-бързо зареждане на страниците; ▪ Няма да се използва HTML Frames, за да не се пречи на оптимизациите за търсещи машини; <ul style="list-style-type: none"> ○ При разработката на публични уеббазирани страници ще се използват и ще се реализира поддръжка на: <ul style="list-style-type: none"> - Стандартните семантични елементи на HTML5 (HTML Semantic Elements); - JSON-LD 1.0 (http://www.w3.org/TR/json-ld/); - Open Graph Protocol (http://ogp.me) за осигуряване на поддръжка за качествено споделяне на ресурси в социални мрежи и мобилни приложения; ▪ В екранните форми на Системата ще се използват потребителски бутони с унифициран размер и лесни за разбиране текстове в еднакъв стил.

№ по ред	Изискване
	<ul style="list-style-type: none">▪ Всички текстови елементи от потребителския интерфейс ще бъдат визуализирани с шрифтове, които са подходящи за изобразяване на екран и които осигуряват максимална съвместимост и еднакво възпроизвеждане под различни клиентски операционни системи и браузъри. Няма да се използват серифни шрифтове (Serif).▪ Полета, опции от менюта и командни бутони, които не са разрешени конкретно за ролята на влезлия в системата потребител, няма да са достъпни за този потребител. Това не отменя необходимостта от ограничаване на достъпа до бизнес логиката на приложението чрез декларативен или програмен подход.▪ Всяка екранна форма ще има наименование, което ще се изписва в горната част на екранната форма. Наименованията ще подсказват на потребителя какво е предназначението на формата.▪ Всички търсения ще са нечувствителни към малки и главни букви.▪ Полетата за пароли задължително ще различават малки и главни букви.▪ Полетата за потребителски имена ще позволяват използване на имейл адреси като потребителско име, включително ще допускат всички символи, регламентирани в RFC 1123, за наименоуването на хостове;▪ Главните и малките букви на въвежданите данни ще се запазват непроменени, няма да се допуска Системата да променя капитализацията на данните, въвеждани от потребителите.▪ Системата ще позволява въвеждане на данни, съдържащи както български, така и символи на официалните езици на ЕС.▪ Наименованията на полетата ще са достатъчно описателни, като максимално се доближават до характера на съдържащите се в тях данни.▪ Системата ще поддържа прекъсване на потребителски сесии при липса на активност. Времето ще може да се променя от администратора на системата без промяна в изходния код. Настройките за време за прекъсване на неактивни сесии ще се включват и възможността администраторите да дефинират стилизирана страница с информативно съобщение, към която Системата ще пренасочва автоматично браузърите на потребителите в случай на прекъсната сесия;▪ Дългите списъци с резултати ще се разделят на номерирани страници с подходящи навигационни елементи за преминаване към предишна, следваща, първа и последна страница, към конкретна страница. Навигационните елементи ще са логически обособени и свързани със съответния списък и ще се визуализират в началото и в края на HTML контейнера, съдържащ списъка;

№ по ред	Изискване
	<ul style="list-style-type: none">▪ За големите йерархически категоризации ще се предвиди възможност за навигация по нива или чрез отложено зареждане (lazy load).
NFREQ.71.	Интернационализация <ul style="list-style-type: none">▪ Системата ще може да съхранява и едновременно ще визуализира данни и съдържание, което е въведено/генерирано на различни езици;▪ Всички софтуерни компоненти на Системата, използваните софтуерни библиотеки и развойни комплекти, приложните сървъри и сървърите за управление на бази данни, елементите от потребителския интерфейс, програмно-приложните интерфейси, уебслужбите и др. ще поддържат стандартно и ще са конфигурирани изрично за спазване на минимум Unicode 5.2 стандарт при съхранението и обработката на текстови данни, съответно ще се използва само UTF-8 кодиране на текстовите данни.▪ Всички публично достъпни потребителски интерфейси ще поддържат многоезичност, като минимум български и английски език.▪ Публичната част на Системата ще бъде разработена и ще включва набори с текстове на минимум два официални езика в ЕС, а именно български и английски език.▪ Версиите на съдържанието на съответните езици ще включват всички текстове, които се визуализират във всички елементи на потребителския интерфейс, справките, генерираните от системата електронни документи, съобщения, нотификации, имейл съобщения, номенклатурите и таксономиите и др. Данните, които се съхраняват в Системата само на български език, се изписват/визуализират на български език;▪ Публичната част на Системата ще позволява превключване между работните езици на потребителския интерфейс в реално време от профила на потребителя и от подходящ, видим и лесно достъпен навигационен елемент в горната част на всяка страница, който включва не само текст, но и подходяща интернационална икона за съответния език;▪ При визуализация на числа ще се използва разделител за хиляди (интервал).▪ При визуализация на дати и точно време в елементи от потребителския интерфейс в генерирани справки или в електронни документи всички формати за дата и час ще са съобразени с изчисления от потребителя език/локация в настройките на неговия профил:<ul style="list-style-type: none">○ За България стандартният формат е „DD.MM.YYYY HH:MM:SS”, като наличието на време към датата е в зависимост от вида на визуализираната информация и бизнес-смисъла от показването на точно време;○ Системата ще поддържа и всички формати съгласно ISO БДС 8601:2006.

Handwritten mark at the top of the page.

№ по ред	Изискване
NFREQ.72.	<p data-bbox="351 224 994 257">Използваемост на потребителския интерфейс</p> <ul style="list-style-type: none"><li data-bbox="351 290 1467 716">▪ Електронните форми за подаване на заявления и за обявяване на обстоятелства ще бъдат реализирани с AJAX или с аналогична технология, като по този начин се гарантират следните функционалности:<ul style="list-style-type: none"><li data-bbox="448 421 1467 497">○ Контекстна валидация на въвежданите данни на ниво "поле" от форма и контекстни съобщения за грешка/невалидни данни в реално време;<li data-bbox="448 508 1467 716">○ Възможност за избор на стойности от номенклатури чрез търсене в списък по част от дума (autocomplete) и визуализиране на записи, отговарящи на въведеното до момента, без да е необходимо пълните номенклатури да са заредени в браузъра на клиента и потребителят да скорлира дълги списъци с повече от 10 стойности;<li data-bbox="351 727 1467 891">▪ В електронните форми ще бъде реализирана валидация на въвежданите от потребителите данни на ниво "поле" (in-line validation). Валидацията ще се извършва в реално време на сървъра, като при успешна валидация данните от съответното поле следва да бъдат запазени от сървъра;<li data-bbox="351 901 1467 1196">▪ Системата трябва да гарантира, че въведените, валидираните и запазените от сървъра данни остават достъпни за потребителите дори за процеси, които не са приключили, така че при волно, неволно или автоматично прекъсване на потребителската сесия поради изтичане на периода за допустима липса на активност потребителят да може да продължи съответния процес след повторно влизане в системата, без да загуби въведените до момента данни и прикачените до момента електронни документи;<li data-bbox="351 1207 1467 1546">▪ Трябва да бъде реализирана възможност за добавяне и редактиране от страна на администраторите на Системата, без да са необходими промени в изходния код, на контекстна помощна информация за:<ul style="list-style-type: none"><li data-bbox="448 1338 1467 1415">○ всяка електронна форма или стъпка от процес, за която има отделен екран/форма;<li data-bbox="448 1426 1467 1502">○ всяка група полета за въвеждане на данни (в случаите, в които определени полета от формата са групирани тематично);<li data-bbox="448 1513 1467 1546">○ всяко отделно поле за въвеждане на данни;<li data-bbox="351 1557 1467 1720">▪ Трябва да бъде разработена контекстна помощна информация за всички процеси, екрани и електронни форми, включително ясни указания за попълване и разяснения за особеностите при попълване на различните групи полета или на отделни полета;<li data-bbox="351 1731 1467 1939">▪ Контекстната помощна информация, указанията към потребителите и информативните текстове за всяка електронна административна услуга не трябва да съдържат акроними, имена и референции към нормативни документи, които са въведени като обикновен текст (plain-text). Всички акроними, референции към нормативни документи, формуляри, изисквания

Handwritten mark on the left margin.

Handwritten mark on the left margin.

Handwritten mark at the bottom of the page.

Handwritten signature

№ по ред	Изискване
	<p>и др. трябва да бъдат разработени като хипервръзки към съответните актуални версии на нормативни документи и/или към съответния речник/списък с акроними и термини;</p> <ul style="list-style-type: none">▪ Достъпът на потребителя до контекстната помощна информация трябва да бъде реализиран по унифициран и консистентен начин чрез подходящи навигационни елементи, като например чрез подходящо разположени микро бутони с икони, разположени до/пред/след етикета на съответния елемент, за който се отнася контекстната помощ, или чрез обработка на "Mouse Hover/Mouse Over" събития;▪ При проектирането и реализацията на потребителския интерфейс трябва да се отчете, че той трябва да бъде еднакво използваем и от мобилни устройства (напр. таблети), които не разполагат с мишка, но имат чувствителни на допир екрани.▪ Потребителският интерфейс следва да бъде достъпен за хора с увреждания съгласно изискванията на чл. 48, ал. 5 от ЗОП.
NFREQ.73.	<p>Изисквания за използваемост в случаи на прекъснати бизнес процеси</p> <ul style="list-style-type: none">▪ Системата ще осигури възможност за съхраняване на данните по започнал процес/процедура, текущия им статус и всички въведени данни и прикачени документи дори ако потребителят е прекъснал волно или неволно потребителската си сесия;▪ При вход в системата потребителят ще получава прегледна и ясна нотификация, че има започнати, но недовършени/неизпратени/неподписани документи, и да бъде подканен да отвори модула за преглед на историята на транзакциите;▪ Модулът за преглед на историята на транзакциите ще поддържа следните функционалности:<ul style="list-style-type: none">○ Ще визуализира списък с историята на въведените данни;○ Ще предлага видни и лесни за използване от потребителите контроли/инструменти:<ul style="list-style-type: none">- за филтриране на списъка (от дата до дата, за предефинирани периоди, като "последния един месец", "последната една година");- сортиране на списъка по всяка от колоните, без това да премахва текущия филтър;- свободно търсене по ключови думи по всички колони в списъка и метаданните на прикачените/свързаните документи със заявленията, което да води до динамично филтриране на списъка.

Handwritten signature

Handwritten signature

flex

№ по ред	Изискване
NFREQ.74.	<p data-bbox="352 222 603 254">Системен журнал</p> <p data-bbox="352 286 1466 394">Изгражданото решение ще осигурява проследимост на действията на всеки потребител (одит), както и версия на предишното състояние на данните, които той е променил в резултат на своите действия (системен журнал).</p> <p data-bbox="352 426 1466 502">Атрибутите, които ще се запазват при всеки запис, ще включват като минимум следните данни:</p> <ul data-bbox="399 534 1183 782" style="list-style-type: none">▪ дата/час на действието;▪ модул на системата, в който се извършва действието;▪ действие;▪ обект, над който е извършено действието;▪ допълнителна информация;▪ IP адрес и браузър на потребителя. <p data-bbox="352 793 1466 901">Размерът на журнала на потребителските действия нараства по време на работа на всяка система, което налага по-различното му третиране от гледна точка на организация на базата данни:</p> <ul data-bbox="399 933 1466 1524" style="list-style-type: none">▪ по време на работа на Системата потребителският журнал ще се записва в специализиран компонент, който поддържа много бързо добавяне на записи;▪ специална фоновая задача ще акумулира записаните данни и ще ги организира в отделна специално предвидена за целта база данни, отделна от работната база данни на Системата;▪ данните в специализираната база данни ще се архивират и изчистват, като в специализираната база данни ще бъде достъпна информация за не повече от 2 месеца назад; при необходимост от информация за предишен период администраторът на Приложението ще трябва първо да възстанови архивните данни;▪ ще бъде предоставен достъп до системния журнал на органите на реда чрез потребителски или програмен интерфейс; за достъпа ще се изисква електронна идентификация.

↑

→

flex

Handwritten mark

№ по ред	Изискване
NFREQ.75.	<p data-bbox="357 232 1008 265">Дизайн на бази данни и взаимодействие с тях</p> <p data-bbox="357 293 1475 362">При използване на базата данни ще бъдат следвани добрите практики за дизайн и взаимодействие с нея, в т.ч.:</p> <ul data-bbox="404 396 1475 1078" style="list-style-type: none"><li data-bbox="404 396 1475 513">▪ дизайнът на схемата на базата данни ще бъде с максимално ниво на нормализация, освен ако това не би навредило сериозно на производителността;<li data-bbox="404 526 1475 595">▪ базата данни ще може да оперира в клъстър; в определени случаи следва да бъде използван т.нар. sharding;<li data-bbox="404 607 1475 640">▪ имената на таблиците и колоните ще следват унифицирана конвенция;<li data-bbox="404 653 1475 769">▪ ще бъдат създадени индекси по определени колони, така че да се оптимизират най-често използваните заявки; създаването на индекс ще бъде мотивирано и подкрепено със замервания;<li data-bbox="404 782 1475 814">▪ връзките между таблици ще са дефинирани чрез foreign key;<li data-bbox="404 827 1475 896">▪ при операции върху много записи (batch) ще се избягват дългопродължаващи транзакции;<li data-bbox="404 909 1475 942">▪ заявките ще бъдат ограничени в броя записи, които връщат;<li data-bbox="404 955 1475 1071">▪ 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).

Handwritten mark

4. Функционални характеристики

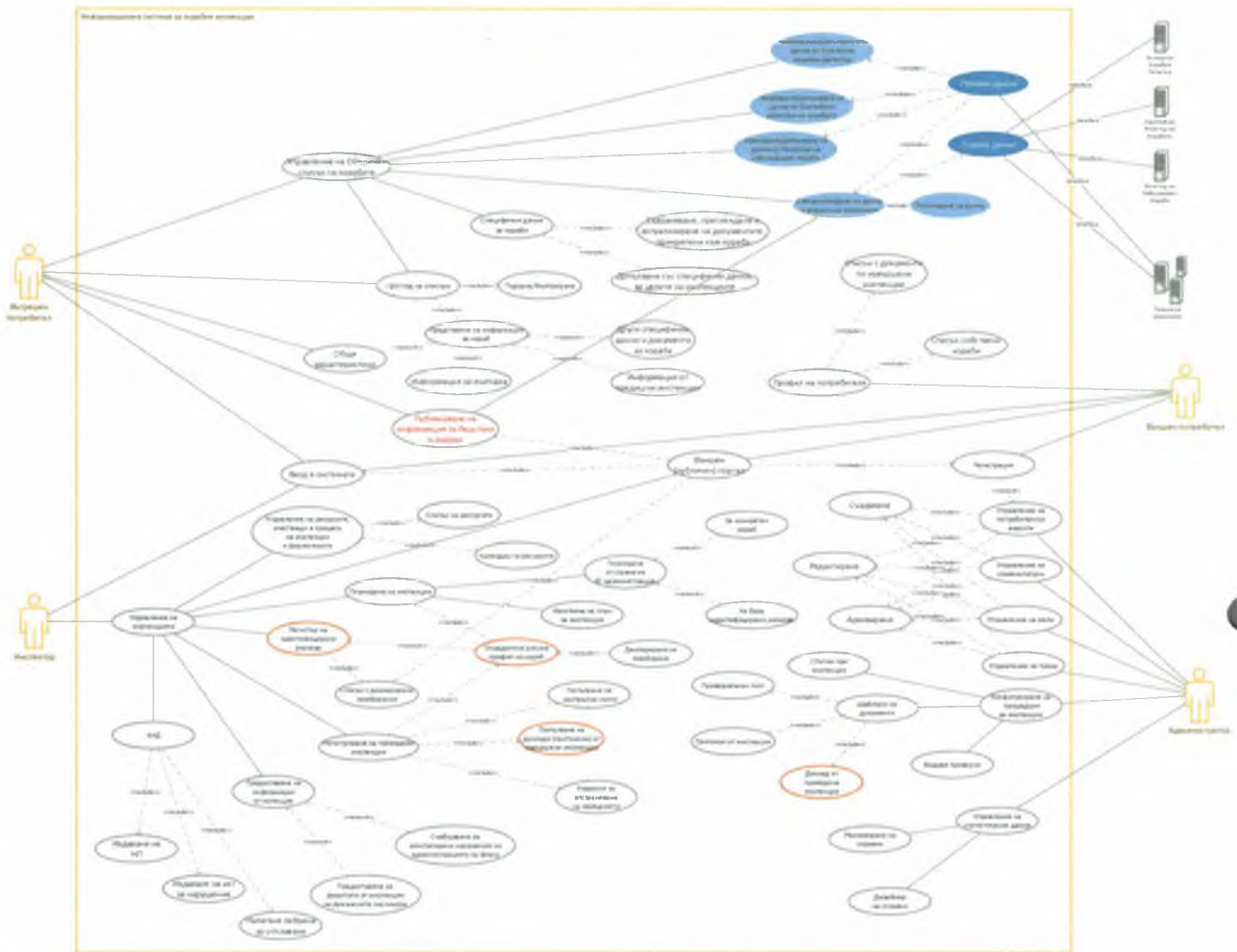
Следва описание на визията на Кандидата за реализиране на функционалните характеристики на системата. В етапа на дизайн функционалните и технически характеристики на системата ще бъдат прецизирани, детайлизирани и съгласувани с Възложителя.

4.1. Диаграма на функционалностите

На фигурата по-долу сме представили диаграма на случаите на употреба, за всеки един от модулите, заложен в обхвата на Системата.

Handwritten signature

Sketch



Handwritten mark

4.2. Основни функционалности

4.2.1. Общи функционални изисквания към ИС за корабни инспекции

Системата ще бъде изградена от външен и вътрешен портал и ще бъде реализирана на модулен принцип (виж т.2.2 - Софтуерна архитектура).

Изисквания към външния портал:

Handwritten mark

Handwritten signature

000071

№ по ред	Изискване
FREQ.1.	Потребителите, достъпващи портала ще са: <ul style="list-style-type: none">• Регистрирани – корабособственици и капитани на кораби;• Нерегистрирани – достъпват само публичната информация на портала.
FREQ.2.	За нерегистрираните потребители външния портал ще служи те да могат да извършват регистрация.
FREQ.3.	Регистрацията на външните потребители ще се активира от администраторите на ИИС в модул Управление и статистически данни.
FREQ.4.	За регистрираните потребители външния портал ще служи за вход в ИИС с цел да: <ul style="list-style-type: none">• управляват своите профили;• да се информират за издадени актове за нарушение и влезли в сила наказателни постановления;• да получават информация за предстоящи проверки, или да проследяват статуса на конкретна проверка.
FREQ.5.	В профила си, всеки регистриран потребител ще може да: <ul style="list-style-type: none">• Преглежда списък на собствените си кораби;• Преглежда списък с проведени и регистрирани в ИИС проверки на собствените си кораби и резултатите от тях;• Уведомяване съответната Администрация за предстоящи лимбирания – предложено като допълнителна функционалност (виж т. 4.5.3. Допълнителна функционалност 3: Уведомяване за лимбирания).
FREQ.6.	Външният портал ще съдържа секция с публикувана информация относно настъпили бедствия и аварии. Публикуването ще се извършва в административния панел на портала от потребител с необходимите за това права.

Изисквания към Вътрешния портал:

Вътрешният портал е основният компонент на ИИС. Той ще отговаря на изискванията в таблицата по-долу:


Handwritten signature

№ по ред	Изискване
FREQ.7.	<p>Вътрешния портал ще бъде изграден от следните модули:</p> <ul style="list-style-type: none">➤ Модул Корабни инспекции и формалности;➤ Модул Обединен списък на корабите;➤ Модул Управление и статистически данни.
FREQ.8.	<p>Достъп до вътрешния портал ще имат само регистрирани потребители, служители на Възложителя.</p>
FREQ.9.	<p>Вътрешният портал ще служи за:</p> <ul style="list-style-type: none">• Поддържане на актуална информация в ИИС относно процеса по извършване на проверки;• Управление на административно-наказателната дейност;
FREQ.10.	<p>Потребителите на вътрешния портал ще бъдат разделени в следните потребителски роли:</p> <ul style="list-style-type: none">• Оператори на ИС за инспекция на кораби;• Ключови потребители;• Инспектори – отразяват резултатите от извършените дейности, свързани с инспекция на кораби;• Администратори на системата.
FREQ.11.	<p>Потребителите с роля Оператор ще могат да преглеждат и да актуализират ръчно информацията, постъпила в обединения списък на корабите от регистрите, румънския компонент на ИИС – а функционалност (виж т. 4.5.3. Допълнителна функционалност 3: Уведомяване за лимбирания)</p>
FREQ.12.	<p>Ключовите потребители ще достъпват график на планираните инспекции. Модулът ще им предостави възможност за изготвяне на график, отразяващ дейността на Инспекторите. Графикът ще съдържа информация за броя и вида на планираните инспекции, ресурсната обезпеченост в процеса на извършване на проверки, натовареността на инспекторите.</p>
FREQ.13.	<p>Потребители с роля Инспектор ще въвеждат информация, свързана с извършените проверки, в т.ч.:</p> <ul style="list-style-type: none">• Проверовъчни листи;• Доклад от извършена проверка – предложено е като допълнителна функционалност (виж т. 4.5.1. Допълнителна функционалност 1: Доклад от извършена проверка);• Актове за установено нарушение;• Наказателни постановления.

Handwritten mark

Handwritten mark

Handwritten signature



FREQ.14.	Потребители с роля Администратор ще отговарят за правилното функциониране на ИИС и ще имат възможност за управление на: <ul style="list-style-type: none">• Потребителски профили и акаунти;• Роли и Права за достъп;• Сигурността на ИИС.
-----------------	--

4.2.2. Модул Обединен списък на корабите

4.2.2.1. Описание

Всяка от Администрациите – българска и румънска, докладва за извършените от нея съгласно Споразумението проверки и резултатите от тях, чрез Интегрираната система за проверка на кораби.

Чрез модула ще се реализират функционалности по събиране и обобщаване на данни по отношение на плавателните съдове, подлежащи на проверки, необходими за провеждане на инспекциите от страна на отговорните за това български и румънски органи. Тази информация се съдържа в регистри, поддържани от различни информационни системи и за целта ще бъде реализирана интеграция на разработваната система, предмет на настоящата обществена поръчка с регистри, на базата на уеб услуги или чрез импортиране на данни (виж т. 2.2.3 Подход за интеграция с външни системи) и ще обезпечават процеса по приемане на данни и събирането им в обединен списък на корабите. По този начин ще се осигури възможност за постигане на цялостност и пълнота на данните.

Предвидено е да се реализира връзка със следните регистри:

- Български корабен регистър;
- Европейски регистър на корабите;
- Регистър на риболовните кораби.

4.2.2.2. Начин на реализация

Следва описание на функционалните изисквания към модула:



Handwritten signature

№ по ред	Изискване
FREQ.15.	<p>Модулът ще осигурява възможност за зареждане/допълване на данни от:</p> <ul style="list-style-type: none">• Български корабен регистър• Европейски регистър на корабите (EHDB)• Регистър на риболовните кораби <p>Ще бъде реализирано на базата на уеб услуги или чрез импортиране на данни импортиране на данни. Форматът на файловете ще бъде уточнен по време на изпълнението на проекта в етап Анализ и проектиране.</p>
FREQ.16.	<p>Ще бъде реализирано двупосочно синхронизиране на корабните регистри между българския и румънския компонент, както и допълнителна информация за корабите.</p>
FREQ.17.	<p>Ще бъде предоставена възможност за поддържане на специфични данни за кораби, реализирана на базата на:</p> <ul style="list-style-type: none">• Допълване на данните в регистъра със специфична за целите на инспекциите данни• Съхраняване, преглеждане и актуализиране на документите, прикрепени към кораба
FREQ.18.	<p>Преглед на регистъра, с възможност за:</p> <ul style="list-style-type: none">• Търсене и филтриране;• Визуализиране на информация за кораб:<ul style="list-style-type: none">○ Информация от регистъра;○ Информация от предишни инспекции;○ Други специфични данни и документи за кораба;○ Информация за екипажа.
FREQ.19.	<p>За потребители, служители на Възложителя, ще се осигури възможност за ръчно въвеждане/допълване на специфични данни за кораби – функционалността е свързана с предоставяне на възможност на потребителите със съответната роля да могат да допълват данните в Обединения списък на корабите от постъпила информация в следствие на извършени проверки, издадени препоръки и предписания и/или приложени документи от страна на корабособствениците.</p>

Handwritten mark

Handwritten signature

4.2.3. Модул Корабни инспекции и формалности

4.2.3.1. Описание

Основната цел на модула е да осигури възможност за планиране и управление на контролната дейност на инспекторите на ИАМА.

Модулът ще обезпечавя процесите при провеждане на проверки.

Handwritten signature

На проверки подлежат плавателни съдове по вътрешните водни пътища, самоходни и несамоходни, плаващи под чуждо знаме, извършени от българската и румънската Администрации по протежение на общия участък от река Дунав.

Проверки не се прилагат на плавателни съдове за спорт, туризъм и развлечение.

Проверките извършени на борда на плавателни съдове по вътрешните водни пътища от една от Администрациите ще се признават от другата Администрация за целите на определяне на рисковия профил на корабите, подбора на кораби за проверка и ограничаване административната тежест върху операторите от сектора на вътрешноводния транспорт.

Проверки на плавателни съдове се извършват също и на пристанище от българската или румънската Администрация.

Честотата на извършваните проверки зависи от определения рисков профил на кораба.

Всяка Администрация се стреми да осигури отстраняването на установените от извършените проверки неизправности. При условие, че са положени всички възможни усилия за отстраняване на неизправностите, без онези, които представляват явна заплахата за безопасността, здравето или околната среда, на плавателен съд може да бъде позволено да отплава до друго пристанище, където неизправностите могат да бъдат отстранени.

В случай на неизправности, представляващи явна заплахата за безопасността, здравето или околната среда, Администрацията гарантира, че заплахата е отстранена, преди на плавателния съд да бъде позволено да продължи плаването си по вътрешните водни пътища. За целта се предприемат подходящи мерки, които могат да включват задържане или официална забрана на кораб да извършва дадена операция поради установени неизправности, които биха направили тази операция опасна.

При извършване на инспекции се проверява за:

- корабни документи – тяхната пълнота и срокове на валидност,
- документи на екипажа - тяхната пълнота и срокове на валидност,
- товаро-разтоварни операции - спазване на условията за безопасност на труда от членовете на екипажа; товаро-разтоварните работи да се провеждат при съблюдаване на правилата за товарене и разтоварване на корабите, осигуряващи безопасния им престой в пристанището, обръща се особено внимание на корабите, превозващи опасни товари
- изхвърляне на остатъци от товара в пристанищата,
- условия за безопасно слизане и качване на пътници на пътнически кораби,
- спасителни и противопожарни средства,
- съответствие на данните, записани в AIS на всеки кораб със действителните данни на кораба. При проверки на корабите по корабоплавателния път с патрулни катери се проверява за:
- спазване реда за носене на светлини, подаване на сигнали, а също и изпълняване на всички други предписания, произтичащи от Правилата за плаване;
- следи от нерегламентирано изхвърляне на отпадъци и нефтопродукти в реката,

- максималното газене на корабите да отговаря на навигационните условия,
- съответствие на данните, записани в ИС за корабни инспекции за всеки кораб със действителните данни на кораба.

4.2.3.2. *Начин на реализация*

Модулът ще бъде част от:

- външен портал – по отношение на достъпа до ИС за корабособственици и капитани на корабите;
- вътрешен портал – по отношение на достъпа за потребители, служители на Възложителя.

Следва списък с функционалности, които ще бъдат конфигурирани и са относими към процеса по извършване на проверки.

№ по ред	Изискване
FREQ.20.	<p>Модулът ще предоставя възможност за:</p> <ul style="list-style-type: none"> • Конфигуриране на процедури за инспекции; • Поддържане на регистър на идентифицирани рискове – предложено е като допълнителна функционалност (виж т. 4.5.2. Допълнителна функционалност 2: Рисков профил на кораба); • Планиране на инспекции и др. полеви дейности; • Управление на ресурсите, участващи в процеса на инспекция и формалности; • Регистриране на резултати от извършени на инспекции и др. полеви дейности и документите, които ги съпътстват; • Предоставяне на информация от инспекции.
FREQ.21.	<p>Конфигурирането на процедури за инспекция ще включва следните основни моменти:</p> <ul style="list-style-type: none"> • Конфигуриране на видове проверки; • Конфигуриране на стъпките при инспекции; • Конфигуриране на проверовъчни листи.
FREQ.22.	<p>Ще се предостави възможност за поддържане на списък с идентифицирани рискове.</p> <p>Потребител с роля Ключов потребител ще може да създава, преглежда и редактира даден риск, като за целта ще бъде разработен потребителски интерфейс, описващ риска с необходимите атрибути – предложено е като допълнителна функционалност (виж т. 4.5.2. Допълнителна функционалност 2: Рисков профил на кораба).</p>
FREQ.23.	<p>Ще се осигури възможност за регистриране в ИИС на план за предстоящи инспекции и др. полеви дейности от страна на Администрацията от българска страна. Планът ще може да се изготвя за конкретен кораб при:</p> <ul style="list-style-type: none"> • Издадени указания за отстраняване на нередности при предходни проверки; • Идентифицираните рискове; • Във връзка с рисковия профил на кораба и извършени предходни проверки с установени нарушения.
FREQ.24.	<p>Планираните дейности по инспекции ще могат да бъдат синхронизирани с планираните такива от румънска страна.</p>
FREQ.25.	<p>Ще се предостави възможност за изготвяне на интегриран план за инспекции. Той ще включва списък с кораби, подлежащи на проверка.</p> <p>Ще бъде осигурена възможност за синхронизиране на данни с ИИС на Администрацията от румънска страна, по отношение на корабите, подлежащи на проверки. Ще бъде реализирано на базата на уеб услуги.</p>
FREQ.26.	<p>Ще бъде реализирани двупосочен обмен на информация за съгласуване на проверки и други полеви дейности.</p>

Handwritten mark at the top of the page.

FREQ.27.	Модулът ще осигурява възможност за управление на ресурсите, участващи в процеса на инспекция и формалности. Ще бъде реализиран: <ul style="list-style-type: none">• списък на ресурсите;• календар на ресурсите;• валидиране на заетост на ресурсите. Заетостта на ресурсите се определя от функционалността за планиране на инспекционните дейности.
FREQ.28.	Модулът ще осигури възможност за регистриране на резултатите от извършените инспекции и др. полеви дейности. При попълване на информацията в ИИС, за по-голяма леснота на потребителите при работа с нея, ще се предвиди възможност за автоматично попълване на контролните листи с информация от обединения списък на корабите където е приложимо и проведени предишни инспекции.
FREQ.29.	Ще бъде разработен потребителски интерфейс за въвеждане на данни от проверовъчни листи от извършени инспекции.

Handwritten scribble at the bottom left.

Handwritten scribble at the bottom right.

Handwritten scribble at the bottom center.

FREQ.30.

Проверовъчния лист ще съдържа:

- Място и час на проверката
- Основна информация за плавателния съд:
 - Име на кораба;
 - УЕИН/Официален номер;
 - Регистрационен номер;
 - Пристанище на регистрация;
 - Място на домуване;
 - Знаме;
 - Тип на кораба;
 - Година на построяване;
 - Зона на плаване по вътрешните водни пътища;
 - Дължина (m);
 - Ширина (m);
 - Максимално газене (m);
 - Обща номинална мощност (kW);
 - Водоизместване (cm)/ Дедуейт (t);
 - Име и адрес на собственика/ оператора;
 - Име и адрес на превозвача (само за опасни товари);
 - Превозван опасен товар (UN-/номер на веществото).
- Отбелязване на резултати от проверката дали са спазени директивите на ЕС относно:
 - Корабни документи;
 - Екипаж и документи на екипажа;
 - Корабни дневници/инструкции/планове;
 - Корпус/палуби/надстройка;
 - Рулева рубка;
 - Машинно отделение;
 - Товарно помещение;
 - Спални помещения/кухни;
 - Рулевата система (рулевите инсталации)
 - Оборудване;
 - Спасителни средства;
 - Противопожарно оборудване;
 - Електрическо оборудване;
 - Котвено оборудване.

При извършване на проверки на сухотоварни плавателни съдове и танкери, ще се проверява дали са спазени ADN изискванията и ще се отбелязват данни за:

- Документите на кораба;
- Спазени ли са разпоредбите в областта на транспорта.

FREQ.31.	Модулът ще дава възможност за предоставяне на информация от инспекции: <ul style="list-style-type: none">• Съобщаване за констатирани нарушения на администрацията на флага;• Предоставяне на резултати от инспекции на румънските партньори.
FREQ.32.	Ще се предостави възможност за управление на Административно наказателна дейност, включваща регистриране в системата на: <ul style="list-style-type: none">• издадени АУАН• издаване на наказателни постановления със срок на валидност и размер на нанесените глоби. ИИС ще предоставя възможност за генериране на уведомителни писма, които следва да бъдат входирани в деловодната система на Възложителя. Не е предвидена интеграция с деловодната система на Възложителя.

4.3. Модул Управление и статистически данни

Модулът ще осигури реализацията на следните функционалности:

Handwritten signature

№ по ред	Изискване
FREQ.33.	Модулът ще осигурява възможност за управление на: <ul style="list-style-type: none">• потребителски акаунти• потребителски роли• достъпите на потребителите от различните потребителски роли до типовете документи• Настройки на системата;• номенклатури и основни данни;• шаблони на документите, които ще се генерират автоматично от Системата и формуляри;• Преглед на Логовете на Системата.
FREQ.34.	Ще се предостави възможност за генериране на справки и статистики. Ще бъде реализирано с имплементирането на решение за генериране на справки и отчети.
FREQ.35.	Потребителите ще могат да задават: <ul style="list-style-type: none">• филтри по различни критерии преди извличане на резултатите• начин на сортиране на резултатите. Резултатите ще се представят в табличен вид.
FREQ.36.	Ще се предостави възможност да се указват колоните, които ще бъдат включени в справка и техните параметри.
FREQ.37.	Възможност за отпечатване по предварително дефинирани образци.
FREQ.38.	Справките трябва да могат да се експортират във файлове в стандартен формат (минимум .xls).
FREQ.39.	Справките трябва да могат да се визуализират по прегледен начин на екран, а също така да се съхраняват локално на компютър, минимум във формат на структурирана Excel таблица.

Handwritten mark

Handwritten mark

4.4. Обмен на данни с други системи

Ще се реализират следните видове интеграция за обмен на данни с други системи:

- Български корабен регистър – получава данни за кораби;
- Корабен регистър на ИАРА – получава специфични данни за риболовни кораби;
- Европейски регистър на корабите (EHDB) – получава данни за кораби;
- Румънския компонент на ИИС – двупосочен обмен;
- Предоставяне на публични данни от регистъра на компетентни организации чрез уеб-услуга.

Handwritten signature

4.5. Допълнителни функционалности

Към изпълнението на дейността, предлагаме да включим следните функционалности:

- Доклад от извършена проверка;
- Поддържане на рисков профил на кораб;
- Регистриране на Уведомления за лимбиране.

4.5.1. Допълнителна функционалност 1: Доклад от извършена проверка

Функционалността ще отговаря на следните изисквания:

000083

№ по ред	Изискване
FREQ.40.	Доклад от извършена проверка ще може да се генерира по конкретна проверка след приключването ѝ.
FREQ.41.	Потребителите, които ще имат необходимите права и ще генерират такъв доклад са с роля Инспектор.
FREQ.42.	<p>Докладът от извършената проверка ще включва следните данни:</p> <ul style="list-style-type: none"> • Данни за кораба: <ul style="list-style-type: none"> ○ Име на кораба ○ Флаг на кораба ○ Пристанище на регистрация ○ УЕИН/Регистров номер ○ Тип на кораба / Година на построяване ○ Водоизместване (m³)/ Дедуейт ○ Газене (m) • Данни за капитана: <ul style="list-style-type: none"> ○ Име на капитана ○ Свидетелство за правоспособност №: ○ Валидно за ○ Дата на валидност • Детайли за проверката <ul style="list-style-type: none"> ○ Проверяващ орган ○ Дата на проверката/ Място на проверката • Вид на проверката • Проверени зони и оборудване <ul style="list-style-type: none"> ○ Мостик ○ Товарни помещения/танкове ○ Екипировка за безопасност ○ Кабини / Кухни ○ Навигационно оборудване ○ Електрическо оборудване ○ Палуба/Надсторйка • Установени несъответствия – Да/Не • Несъответствия – Да/Не • Списък на несъответствията • Взети мерки <ul style="list-style-type: none"> ○ Наказание ○ Корабът е задържан - <i>Дата на издаване на заповедта за задържане</i>

4.5.2. Допълнителна функционалност 2: Рисков профил на кораба


На всеки плавателен съд ще се поддържа рисков профил, който ще обуславя неговия приоритет за проверка, интервала от време между проверките и техния обхват.

Срокът за извършване на допълнителна проверка се определя от рисковия профил на кораба, както следва:

- За високорискови кораби – между 3-4 месеца след последната допълнителна инспекция, считано от 3-я месец;
- За кораби със средни рискове – между 6-8 месеца след последната допълнителна проверка, считано от 6-я месец;
- За кораби с нисък риск – между 10-12 месеца след последната допълнителна инспекция, считано от 10-я месец.

При избора на плавателен съд за проверка, Администрациите определят приоритета им за проверката.

Приоритет за допълнителна проверка ще бъде определян за всеки кораб от Интегрираната система за корабните проверки:

- 
- Кораби, участващи в инциденти като: сблъсък, засядане, замърсяване или пожар;
 - Кораби, които нарушават правилата, известията до корабоводителите или правилата за плаване по река Дунав;
 - Кораби, за които е получена информация от друга държава;
 - Кораби, които не могат да бъдат идентифицирани в базата данни на Интегрираната система за корабните проверки
 - Плавателни съдове, работещи по начин, застрашаващ безопасността на корабоплаването;
 - Кораби, които са били предмет на сигнал или жалба от страна на капитан, член на екипажа или всяко лице или организация, които имат законен интерес от безопасното функциониране на кораба, условията на живот и работа на кораба или предотвратяване от замърсяване, освен ако докладът или жалбата очевидно не са основателни.

Анализите за оценка на рисковете ще се извършват независимо от и извън ИС за корабни проверки, алгоритмите за изчисляване на рисковете не са предмет на настоящата поръчка.

4.5.2.1. Начин на реализация

Изисквания към функционалността:

№ по ред	Изискване
FREQ.43.	<p>Ще бъдат реализирани потребителски интерфейси за въвеждане на основни данни, свързани с рисковия профил на кораба. Те могат да бъдат следните:</p> <ul style="list-style-type: none">• Тип на риска• Тежест на риска• Индикатори за настъпване на даден риск• Видове дейности, при които може да се случи даден риск• Нарушения <p>Основните данни ще бъдат детайлизирани на етап Анализ и проектиране при изпълнението на поръчката.</p>
FREQ.44.	<p>При данните в Обединения списък на корабите ще се съдържа секция „Рисков профил“. Тези данни ще бъдат достъпни за потребители с роля Ключов потребител. Те ще могат да въвеждат данни описващи рисковия профил на даден кораб.</p>

4.5.3. Допълнителна функционалност 3: Уведомяване за лимбирания

№ по ред	Изискване
FREQ.45.	<p>В модул Корабни инспекции и формалности ще се предостави възможност за поддържане на списък с декларирани лимбирания.</p> <p>Елементите на този списък ще бъдат постъпилите и потвърдени уведомления за лимбиране от страна на корабоводителите.</p> <p>Списъкът ще съдържа данни най-малко за:</p> <ul style="list-style-type: none">• Кораба;• Очаквана дата и час за извършването на лимбирането;• Вида на транспортирания товар;• Количество за прехвърляне.
FREQ.46.	<p>Ще бъде реализиран потребителски интерфейс за въвеждане на данни за лимбиране. Корабоводителите ще имат възможност за подаване на уведомление за лимбиране чрез профилите си на външния портал след успешен вход в системата.</p>
FREQ.47.	<p>Ще бъде реализиран потребителски интерфейс за потвърждаване на въведените от корабоводителите в уведомлението данни за лимбиране. Служители на Възложителя ще имат възможност за преглед на подадените данни и да потвърдят или откажат лимбиране.</p>
FREQ.48.	<p>Ще бъде реализиран процес по уведомяване за лимбиране, който ще включва следните стъпки:</p> <ol style="list-style-type: none">1. Попълване на данни за лимбиране от корабоводител;2. Изпращане на уведомлението до администрацията, която ще извърши дейността;3. Получаване и преглед на уведомлението от контролиращия орган;4. Потвърждаване на извършване на лимбиране, с посочени дата и час на провеждане на операцията или въвеждане на мотивиран отказ.

4.5.3.1. Обосновка

Предлаганите от Кандидата допълнителни функционалности ще покрият дейността на Възложителя по извършване на инспекции на кораби, ще се постигне цялостност и завършеност на процесите по отношение на изпълнението на инспекционната дейност на администрациите от българска и румънска страна. Това от своя страна ще спомогне за постигане на по-добро планиране и управление на контролната дейност на инспекторите на Възложителя.

С реализирането на доклада от извършени проверки ще се предостави възможност за събиране на информация, която би могла да послужи за последващи анализи и оценка на работата на служителите на Възложителя.

000087

Handwritten signature

Поддържане на рисковия профил на кораба, който обуславя приоритетите за проверка, интервала от време между проверките и техния обхват ще предостави възможност за по-прецизното им планиране, по-добро проследяване на изпълнението на предписаните препоръки по отстраняване на открити нередности, проследяване на наложените мерки и санкции на нарушителите и не на последно място адекватно и обективно прецизиране на дейностите при последващи проверки.

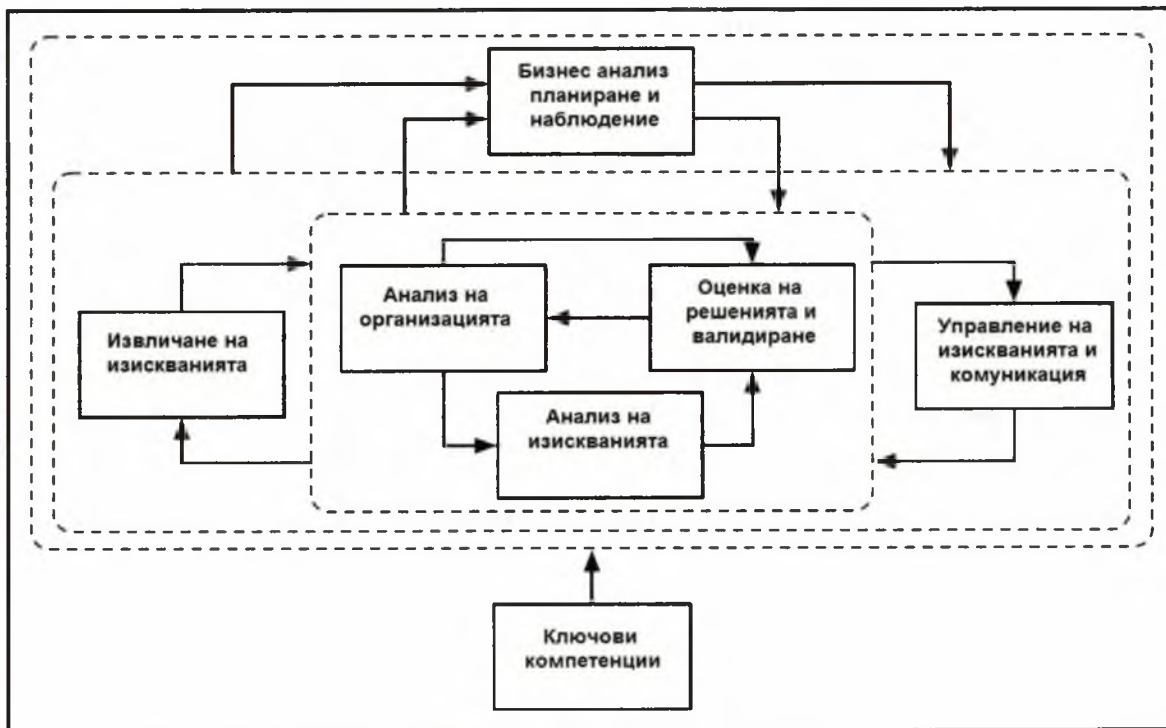
Уведомленията за лимбиране ще допринесат за подобряване на взаимодействието между институциите, ангажирани с корабни инспекции по река Дунав и/или с корабособствениците и корабоводителите.

5. Подход за бизнес анализ

5.1. Методика за бизнес анализ

Подходът за анализ на изискванията, който използваме се придържа към BABOK (Business Analysis Body of Knowledge на Международния институт по бизнес анализ) и е описан по-долу. Връзките между процесите в бизнес анализа са показани на следващата фигура.

На диаграмата се вижда по-ясно, че процесите от по-горе са итеративни и взаимосвързани. Извличането е водещо, като анализа преминава през серия под-процеси, които се адаптират към конкретния продукт в зависимост от всяко конкретно изискване, конкретната ситуация и др. За това адаптиране са отговорни бизнес анализаторите. То се извършва по време на работата, като анализаторите сами определят коя техника да приложат (например интервюиране, въпросник и т.н.) или кой процес да използват. В BABOK документирането е съпровождащ анализа процес. В контекста на обществените поръчки той е съществен. Процесът по валидиране се изпълнява последен, по начина описан тук.



Handwritten signature

Handwritten signature

Handwritten signature

Връзки между процесите в бизнес анализа

Планиране и наблюдение на бизнес анализа

Бизнес анализ 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, и как да се направи оценка за напредъка на работата. Извършените дейности в тази фаза на анализа управляват изпълнението на всички други задачи за бизнес анализ. (BABOK) изброява над четиридесет техники за изискванията за планиране, извличане, анализ, валидиране и комуникация. Бизнес анализ, планиране и мониторинг описва процеса на това как един бизнес анализатор определя кои дейности ще бъдат необходими за завършване на бизнес анализа.

Извличане на изискванията

В допълнение към казаното за този процес по-горе (Извличане), в неговите рамки се определя 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. Най-често необходимата информация се събира чрез провеждане на интервюта.

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. 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.

След като информацията от интервюта се обобщи тя трябва да се съгласува 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 (Enterprise Analysis)

0000389

Процесът по анализът на организацията (Enterprise Analysis) е свързан с това, как бизнес анализаторите идентифицират бизнес нуждите на организацията, усъвършенстват и изясняват определението на тази бизнес необходимост и се определя обхвата, който реално може да бъде описан. 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, тя 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".

Една от първите задачи на екипа, 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 които 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

Анализа на изискванията включва дейности по приоритизиране и доразвитие на изискванията.

Целите на анализа са да се опишат изискванията в една структура, класифицират, организира техните атрибути и приоритети, да се разгледа всяко едно от тях, обхватът на влиянието му върху други изисквания, определят предпоставките за реализацията им и възможните "тесни места", да се направи повторна проверка, да се дискутират със заинтересованите страни и да се утвърдят от тях.

Техниките, които предвиждаме да използваме за анализ на изискванията са:

- Анализ на решенията – при тази техника се осъществява развитие на решението на базата на структурния анализ, който позволява да се оценят алтернативните решения
- Анализ на риска – изискванията които са оценени като рискови, се подлагат на допълнително изясняване или се изпълняват още в началото така, че своевременно да бъдат предприети действия по изменението им.
- MoSCoW Анализ – при анализа изискванията се разделят в четири групи:
 - Изисквания, които трябва да се включат в решението
 - Изисквания, с висок приоритет, които е необходимо да се включат в решението или при създаване на висок риск да бъдат заменени с други
 - Изисквания, които са желателни да бъдат включени в решението, но не е задължителна тяхната реализация

- Изисквания, за които е постигнато съгласие, че няма да бъдат реализирани в текущата реализация, а ще бъде осъществени в бъдещо доразвитие.
- Анализ на бизнес правилата – при него ще бъдат отделени бизнес изискванията от технологичните, организационни и други подобни изисквания.
- Анализ на алтернативните решения
- Дефиниране и приемане на критерии за оценка
- Функционална декомпозиция
- Моделиране на събитията и състоянията
- Подробен анализ на нефункционалните изисквания
- Сценарии и варианти на използване

При анализа на изискванията ще изпълняваме основните задачи съобразно методологията:

- Приоритизиране на изискванията
- Структуриране на изискванията
- Специфициране и моделиране на изискванията
- Определяне на допусканията и ограниченията
- Проверка на изискванията
- Утвърждаване на изискванията от заинтересованите страни

В резултат от анализ на изискванията ще получим набор от дефинирани, симулирани изисквания, които са структурирани, проверени и утвърдени от заинтересованите страни, придружени от съответните предпоставки и ограничения.

Оценка на решенията и валидиране

Целта на дейностите по оценка и валидация е да гарантира, че решението отговаря на изискванията към него и да улесни прилагането му.

Нашето първоначално виждане е при оценката и валидацията да ползваме следните техники:

- Дефиниране и приемане на критерии за оценка
- Анализ на бизнес правилата
- Функционална декомпозиция
- Анализ на алтернативните решения
- Сценарии и случай на използване

В процеса на изпълнение, при необходимост ще бъдат включени и допълнителни техники препоръчвани от методологията BABOK. Оценката и валидацията ще бъдат осъществени посредством изпълнението на 3 основни задачи:

- Оценка на предлаганото решение - в оценката ще бъде определена степента, в която решението отговаря на изискванията на заинтересованите страни
- Разпределение на изискванията

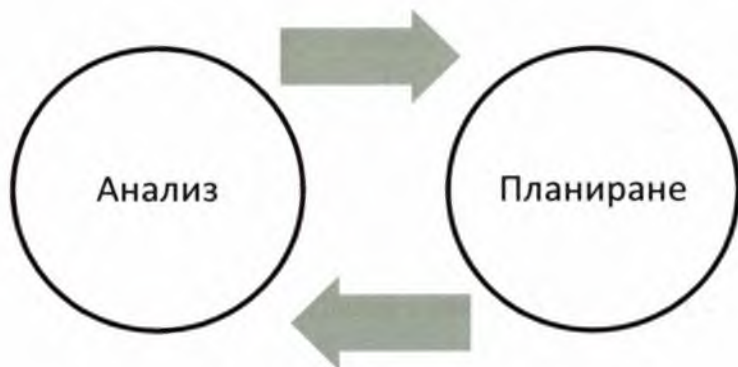
000091

хотя

- Валидиране на решението - гарантира, че решението отговаря на изискванията поставени към него. Идентифицираните проблеми, чрез утвърждаване на решение Проблемите, които са идентифицирани чрез утвърждаване на решение са документирани и приоритизирани, което дава възможност за предприемане на най-подходящите действия от екипа.

5.2. Планиране на анализа

Анализът и планирането е част от всяка итерация, но в началото на проекта трябва да му бъде отделено най-голямо внимание. Тези две дейности са взаимосвързани, тъй като планирането изисква детайлен анализ и същевременно може да породви допълнителни изисквания (обикновено това са ограничения) и така 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 възниква необходимост от допълнителен анализ, тъй като по време на разработката се откриват пропуски или ограничения налагащи промени в изискванията. Също така е възможно Възложителя да промени своите изисквания. Тези промени се управляват с формалния процес по управление на промените и налагат извършване на допълнителен анализ и допълнително планиране, които 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 разработка в момента, като промяната да повиши възможността за възникване на програмни грешки. Дори и да няма никакви промени при реализирането на софтуерните проекти се извършва определено ниво допълнителен анализ и планиране, тъй като потребителските изисквания трябва да бъдат сведени до алгоритми и там възможните вариации са съществени, както от гледна точка на функциониране на системата, така и от гледна точка на управление и вложени ресурси.

В рамките етапа „Анализ на системните изисквания“ трябва да бъдат извършени определени конкретни дейности. Те ще бъдат групирани по итерации/периоди на изпълнение по време на изготвянето на „Детайлен план за изпълнение“. Тези конкретни дейности са както следва:

- 1) Сформиране екипите от експерти, които ще вземат участие в разработката;
- 2) Определяне на необходимите ресурси от страна на Възложителя (технически, човешки и др.), необходими за реализацията;
- 3) Разработване на детайлен план за изпълнение на дейността в допълнение към общия план за реализация;

- Blk*
- 4) Уточняване и ясно дефиниране на обхвата на разработката;
 - 5) Прецизиране и стабилизиране в достатъчна степен на функционалните и технически изисквания, които ще послужат като основа при Проектирането и Разработката.

5.3. Извличане и документиране на изискванията

Подходът за изготвяне на функционалната и техническа спецификация е планът за управление на изискванията и определяне как те ще бъдат анализирани, документирани и управлявани по време на проекта, т.е. целия процес по тяхното изготвяне. Изготвянето на функционалната и техническа спецификация е процесът по дефиниране на изискванията към проекта и документирането им в тази спецификация.

Предпоставките за успешното изпълнение на процеса са:

- 1) Проектната документация (Договор, Техническо задание и предложение);
- 2) Планът за управление на проекта;
- 3) Сформиран работен експертен екип;
- 4) Списък на заинтересованите лица.

Резултати от процеса:

- 1) Документация на изискванията на заинтересованите лица;
- 2) План за управление на изискванията;
- 3) Матрица за проследимост на изискванията.

Техники, които могат да бъдат използвани:

- 1) Интервюта;
 - 2) Фокусни групи;
 - 3) Семинари;
 - 4) Техники за групова креативност;
 - 5) Техники за групово взимане на решения;
 - 6) Въпросници и анкети;
 - 7) Наблюдение;
 - 8) Прототипи.
- Handwritten mark*

Прототипите са предпочитани поради тяхната разбираемост и възможност за по-добра комуникация при тяхното наличие, а и поради възможността за по-лесна верификация.

Извличане на изискванията

Описание на процеса

Извличането на изискванията представлява процес на събиране на информация за предлаганата и съществуващите системи и извличане на потребителските и системните изисквания от нея.

Handwritten mark
Цел

Разглеждане, идентифициране и документиране на потребностите на заинтересованите лица.

Кратко представяне на процес „Извличане“

Задача	Цел	Входни данни	Изходни данни
Подготовка за извличане на изискванията	Подготовката за извличането включва осигуряване на всички необходими ресурси и организирането и планирането на всички дейности по извличане на изискванията	<ul style="list-style-type: none">✓ Списък на заинтересовани страни✓ Роли и отговорности на заинтересованите страни✓ Определение на бизнес проблема✓ План на извличането	<ul style="list-style-type: none">✓ Планиране на ресурсите✓ Материали за извличането
Провеждане на извличането на изискванията	Срещи със заинтересованите страни за събиране на изискванията свързани с техните нужди	<ul style="list-style-type: none">✓ Материали за извличането✓ Определение на бизнес проблема✓ Стандарти, вътрешни правила и процедури в организацията	<ul style="list-style-type: none">✓ Резултати от извличането✓ Предположения, ограничения, рискове, проблеми✓ Документация базирана на (интервюта, бележки, резултати от срещи, анкети и т.н.)
Документиране на извличането на изискванията	Записване на информацията предоставена от заинтересованите страни за целите на анализа	<ul style="list-style-type: none">✓ Резултати от извличането	<ul style="list-style-type: none">✓ Описани изисквания
Верификация на извличането на изискванията	Потвърждаване на информацията със заинтересованите страни с цел правилно разбиране и определяне на обхвата	<ul style="list-style-type: none">✓ Описани изисквания	<ul style="list-style-type: none">✓ Одобрени описани изисквания

Най-често използваните техники за извличане на бизнес изисквания са:

- Събеседвания (brainstorming)
- Преглед на съществуваща документация
- Целеви групи
- Анализ на интерфейси
- Интервюта
- Наблюдение на процеса от близо
- Прототипи
- Анкети

Анализ и документиране

Описание на процеса

Анализът на изискванията и тяхното документиране е процес, при който се описва как екипа на проекта постепенно да изгради решение, което ще отговори на нуждите на заинтересованите страни. За да се постигне това, трябва да се анализират обявените (извлечените) изисквания на страните, да се гарантира, че те са правилни спрямо текущото състояние на бизнеса, за да се препоръчат подобрения и в крайна сметка да се потвърдят като резултат.

Цел

- Да се разработят ясни изисквания с достатъчно ниво на детайлност, за определения бизнес проблем, в определения обхват
- Да се верифицира, че изискванията отговарят на бизнес нуждите
- Да се провери, че изискванията отговарят на определено качество

Кратко представяне на процес „Анализ на изискванията и документиране“

Задача	Цел	Входни данни	Изходни данни
Организиране на изискванията	Структуриране и организиране на изискванията в логически групи. Организирането може да се основава на определянето на множество „нива“ на изискванията, групиране на функции и др.	<ul style="list-style-type: none"> ✓ Бизнес сценарии ✓ Обхват на решението ✓ Изисквания 	✓ Структурирани изисквания
Приоритизиране на изискванията	Определяне на бизнес приоритет на изисквания (включително гласуване, класиране, анализ на ползите и т.н.). Идентифициране на логически зависимости между изисквания и изисквания на логическите групи.	<ul style="list-style-type: none"> ✓ Изисквания ✓ Бизнес сценарии 	✓ Приоритизирани изисквания
Специфициране и моделиране на изискванията	Използване на стандартни практики за писане на текстови изисквания и създаване на модели и диаграми.	✓ Изисквания	✓ Специфицирани и моделирани изисквания

Задача	Цел	Входни данни	Изходни данни
Определяне на допускания и зависимости	Да се отделят онези изисквания, които са допускания и тези, които са зависимости	✓ Изказвания на заинтересованите страни	✓ Допускания и зависимости
Верификация на изискванията	Определя дали изискванията са правилно и напълно определени	✓ Специфицирани и моделирани изисквания	✓ Верифицирани изисквания
Валидиране на изискванията	Потвърждаване, че изискванията отговарят на бизнес нуждите	✓ Верифицирани изисквания	✓ Валидирани изисквания

Документиране на изискванията

В таблицата по-долу е дадено общо съдържание на най-често използваните документи, като резултат от бизнес анализа

Тип документ	Бизнес изискване	Модел на изискванията	Текущ бизнес модел	Потребителски изисквания	Функционални изисквания	Изисквания за осигуряване и о	Допускания и зависимости	Други
Визия	✓							
Спецификация на функционалните изисквания		✓		✓	✓	✓	✓	
Документ с бизнес изисквания		✓		✓	✓	✓	✓	
Техническо предложение/техническо задание	✓							✓
Описание на бизнес процесите			✓	✓	✓			
Инструмент за описание на изискванията		✓			✓	✓	✓	

000097

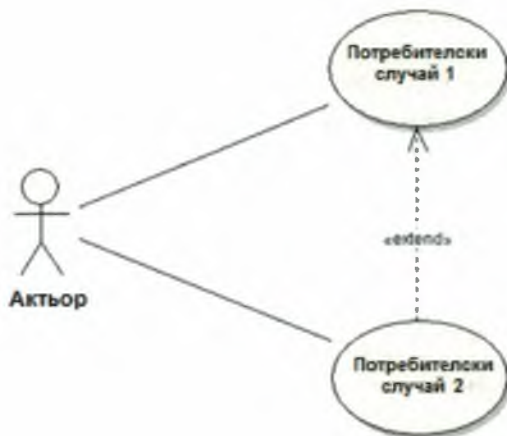
Handwritten signature

5.4. Инструменти, които ще се използват за извършване на функционално и техническо специфициране

За описание на изискванията ние използваме няколко техники.

- 1) За описание на всички изисквания, като цяло се ползва обикновен текст. Самите изисквания се разделят на функционални и нефункционални. За всяко изискване има определен номер и префикс, който е свързан с определена дейност от самия проект напр. FR-ACT2-2 (за функционално изискване), LR (нефункционално изискване излизащо от правна рамка), SR (изискване към сигурността), NFR (всички нефункционални изисквания без класификация). От всяко изискване произлиза една или повече функционалности или от няколко изисквания произлиза една функционалност, така че във функционалната спецификация е необходимо да се опише т.нар. „Матрица на съответствията“, в която се посочва, кое изискване от коя функционалност се покрива.
- 2) За описание на потребителските изисквания използваме т.нар. Потребителски случаи (Use cases). Първоначално се създава общ модел на всички потребителски случаи и актьори в тях, за да се представи общата картина на всички потребителски случаи. За описанието на потребителските случаи се използва специализиран инструмент.


Всички потребителски случаи се описва със следните елементи:



Съдържание:

- Общи бележки и записки, описващи сценария
- Изисквания – Функционалните изисквания, които потребителския сценарии трябва да предложи на крайния потребител. Всяко изискване трябва да е описано във функционалната спецификация
- Ограничения – Формални правила и ограничения, с които потребителския случаи трябва да се съобразява. Това включва:

Handwritten signature

- 
- Предусловия – условия, които трябва да бъдат изпълнени преди да стартира текущия потребителски случай
 - Пост-условия – условия, които трябва да са изпълнени в края на изпълнението на потребителския случай
 - Сценарии – формално описание на стъпките за изпълнението на потребителския случай. Те биват основни и алтернативни.
 - Диаграми на потребителския случаи – графично описание на стъпките за изпълнението на потребителския случай


5.4.1. Моделиране с UML


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

По-подробно методиката за моделиране е разгледана в т. 5.2.4 Моделиране с UML в Приложение 2

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).

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.

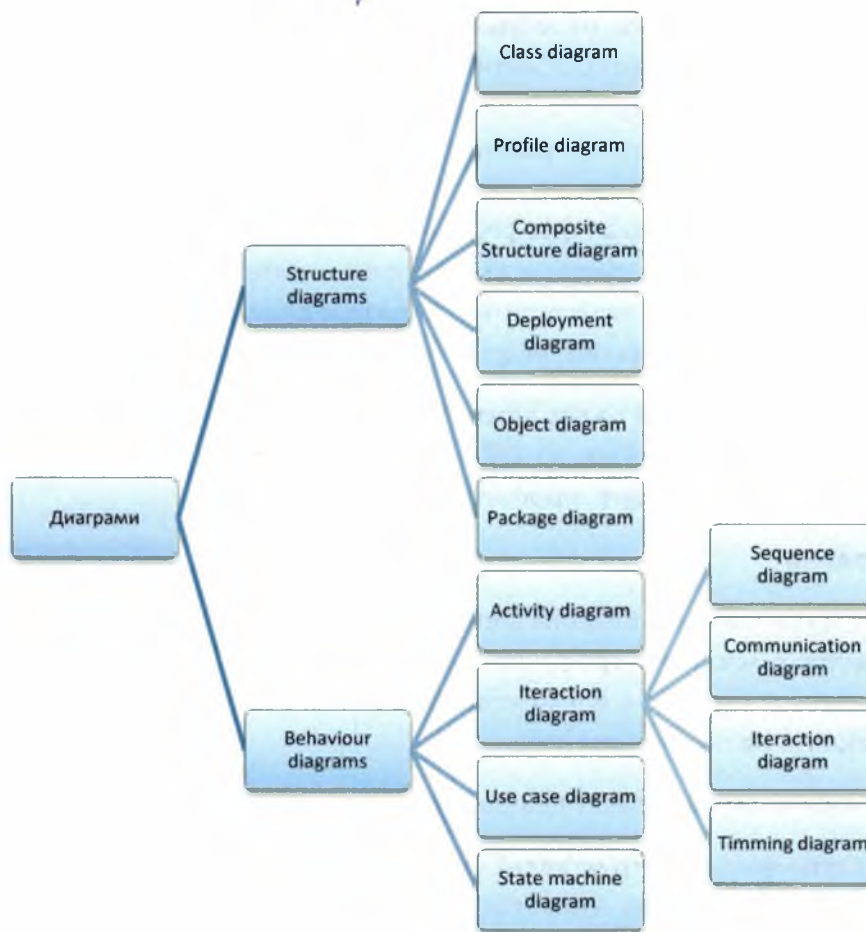
UML модели

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:



Диаграми в UML 2.2

/

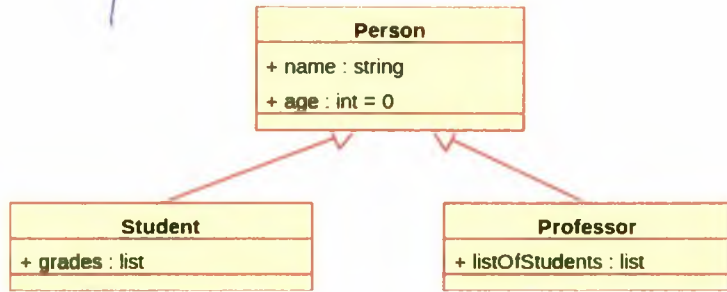
UML не ограничава UML типовете елементи да принадлежат към конкретна диаграма. Като правило всеки UML тип на елемент може да се появи на почти всички типове диаграми. Тази гъвкавост е частично ограничена в UML 2.0. UML профилите могат да дефинират допълнителни типове диаграми или да разширяват съществуващи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

Структурни диаграми

Структурните диаграми наблягат на нещата, които трябва да са налични в моделираната система. Тъй като те представят структурата, се използват изключително за документиране на софтуерната архитектура на софтуерни системи. Това са:

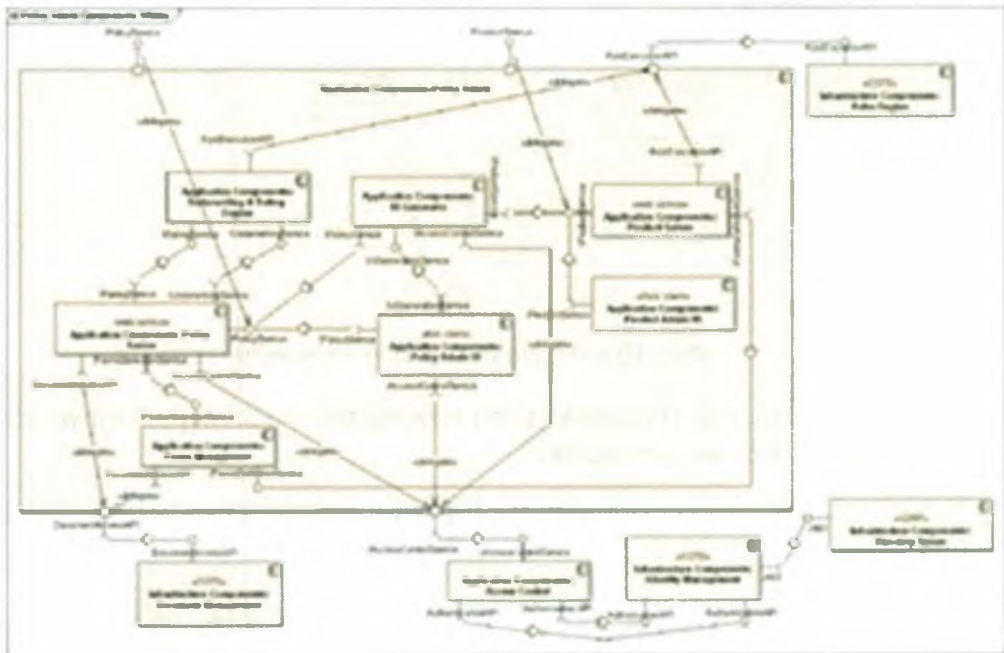
- *Клас диаграма* - описва структурата на системата, показвайки наличните в нея класове, техни атрибути и взаимовръзките между класовете

Handwritten signature



Фиг. Пример за клас диаграма

➤ **Компонентни диаграма** – описва как софтуерната система ще бъде разделена на компоненти и показва връзките между тези компоненти

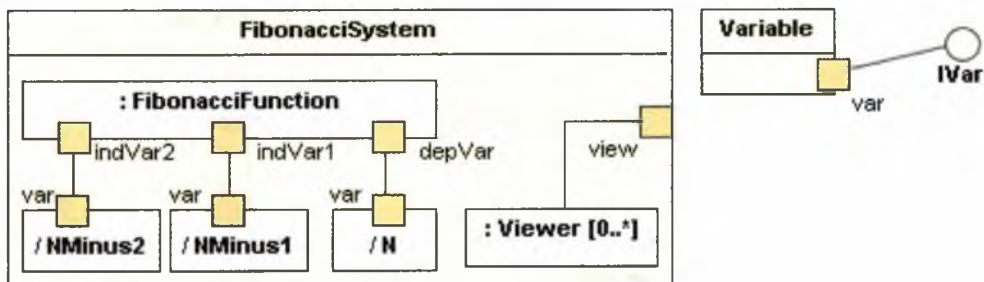


Handwritten signature

Фиг. Пример за компонентна диаграма

Handwritten signature

Диаграма на композитната структура – описва вътрешната структура на класа и взаимодействията, които тази структура прави възможни

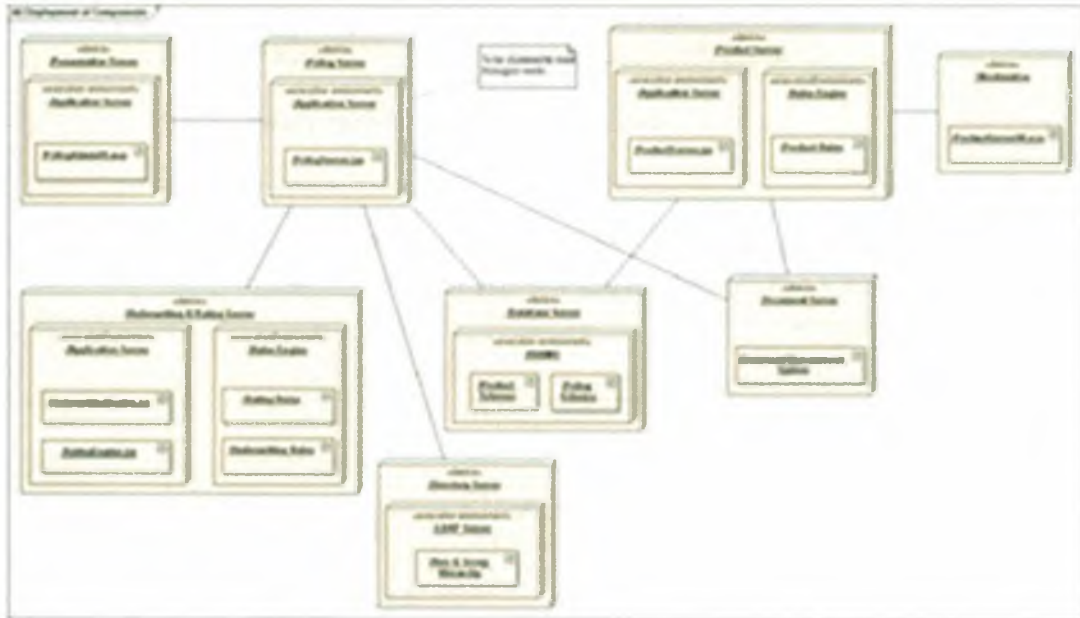


Handwritten signature

Handwritten signature

Фиг. Пример за диаграма на композитна структура

- **Диаграма на внедряването** – описва хардуера, използван от имплементираната система и средата на изпълнение, както и артефактите инсталирани на хардуера



Фиг. Примерна диаграма на внедряването

- **Обектна диаграма** – показва пълен или частичен пример на структурата на системата в определен момент от времето

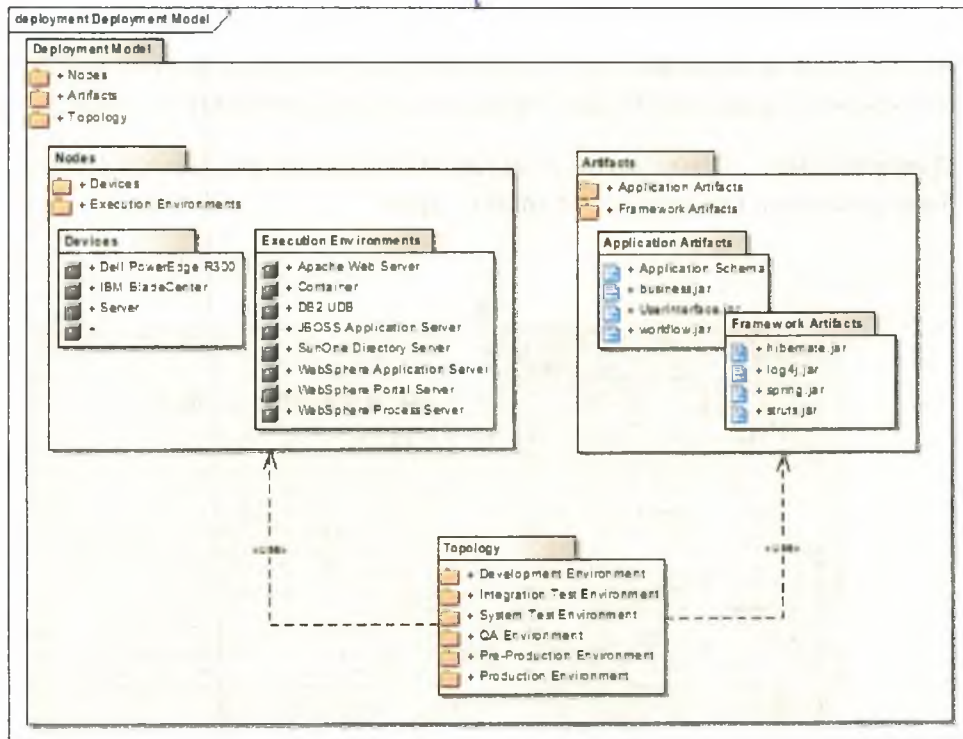
Object Diagram



Фиг. Примерна обектна диаграма

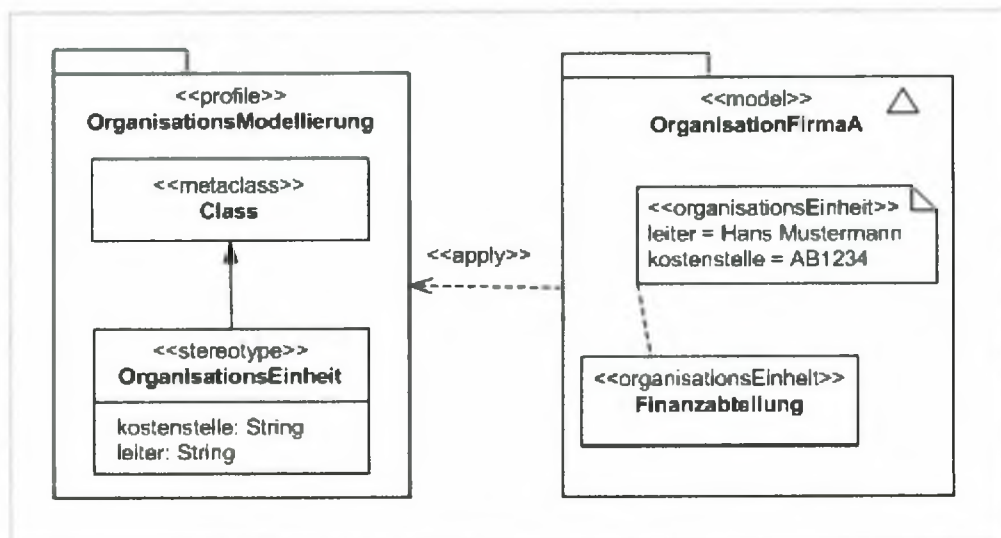
- **Диаграма на пакетите** – описва как системата е разделена на логически групи, показвайки връзките между тях

Handwritten signature



Фиг. Пример на диаграма на пакетите

- *Диаграма на профилите* – оперира на нивото на мета-модела и показва стереотипите, като класове със стереотип <<stereotype>>, и профилите като класове със стереотип <<profile>>. Връзката за разширяване (плътна линия със затворена плътна стрелка) указва кой елемент от мета-моделът разширява стереотипа

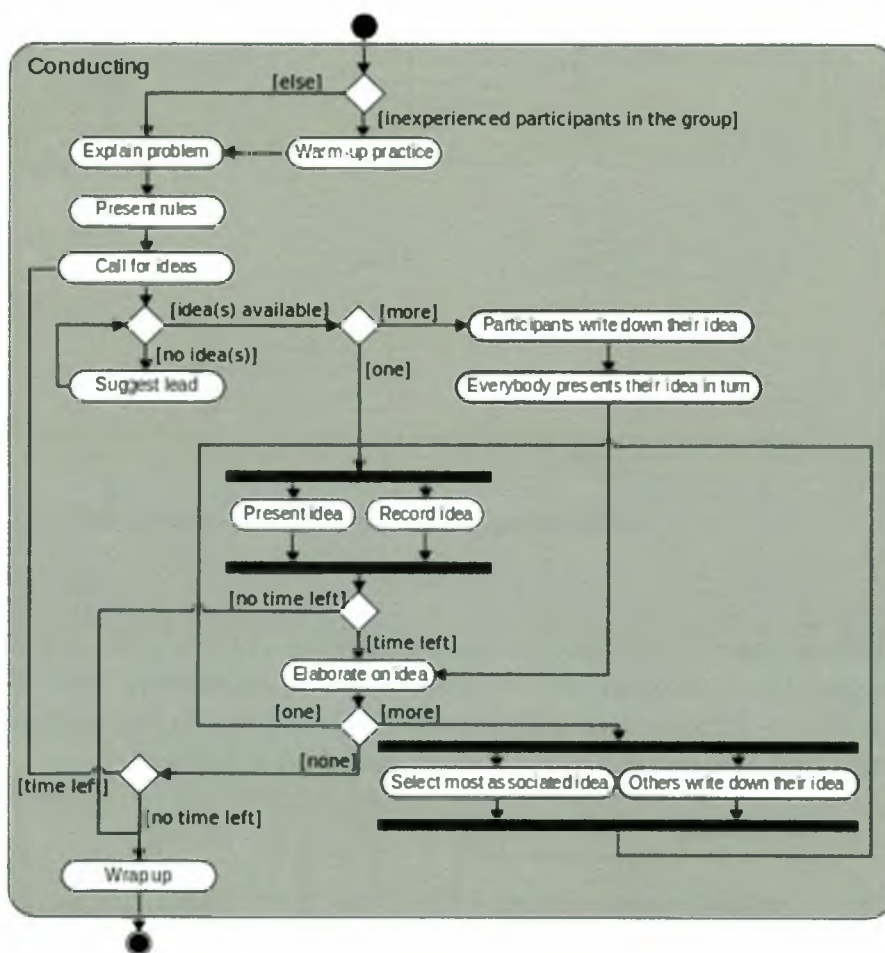


Фиг. Пример на диаграма на профилите

Поведенчески диаграми

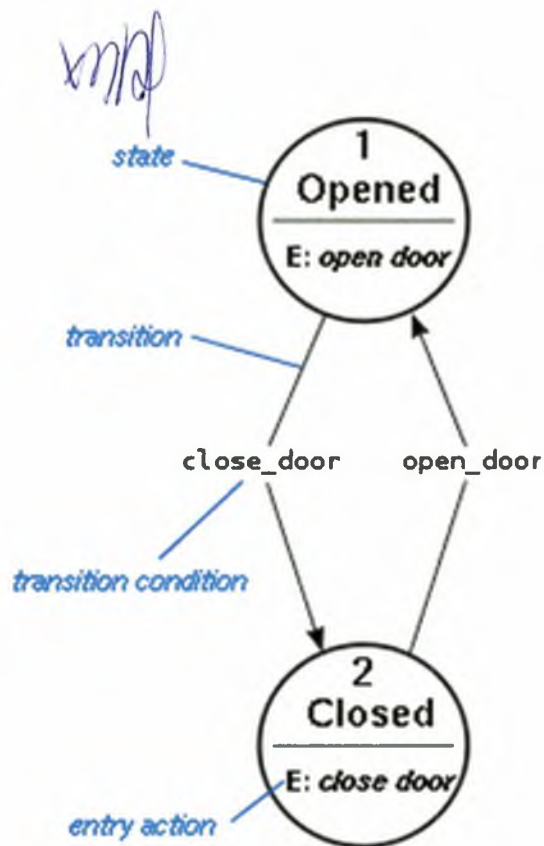
Диаграмите на поведението наблягат на това какво трябва да се случва в моделираната система. Тъй като диаграмата на поведението илюстрира поведението на системата, те се използват изключително много за описване на функционалността на системата. Това са:

- *Диаграма на действията* – описва работата и операциите стъпка по стъпка в система. Тази диаграма показва контролния поток



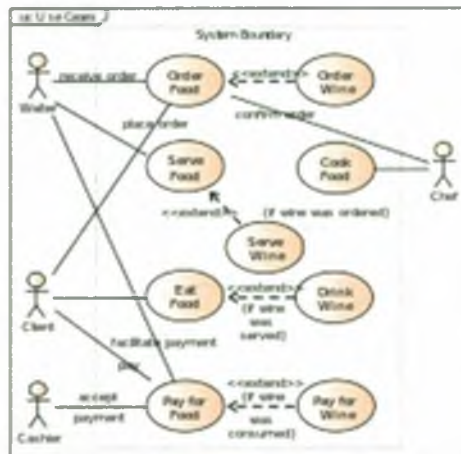
Фиг. Пример на диаграма на действията

- *Диаграма на състоянията* – описва състоянията и преходите между тях



Фиг. Пример на диаграма на състоянията

- **Диаграма на случаите на употреба** – описват функционалността, предоставена от системата в термините на актьори, техните цели представени като „начини на използване“ и всякакви зависимости между тези начини на използване

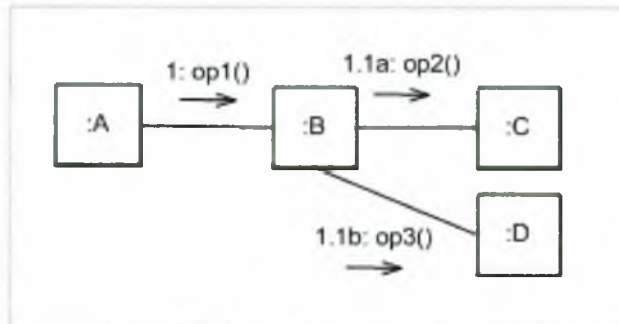


Фиг. Пример на диаграма на случаите на употреба

Диаграми на взаимодействието

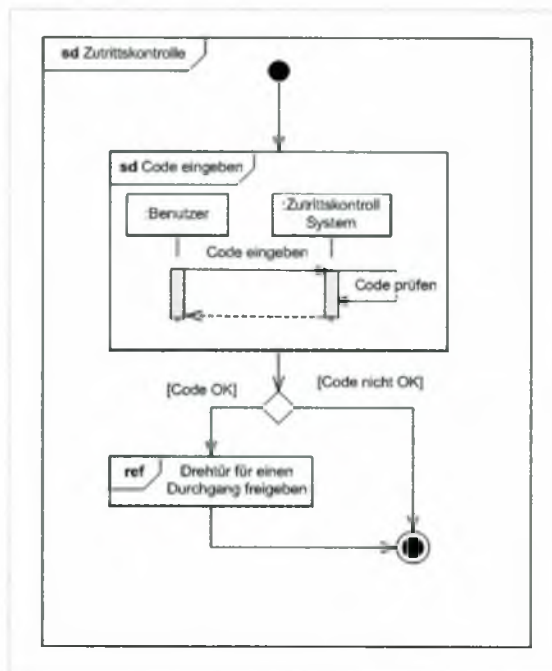
Диаграмите на взаимодействието са подмножество на диаграмите на поведението, наблюдаващи на контролния поток и данните. Това са:

- Handwritten signature*
- **Диаграма на комуникацията** – показва взаимодействията, между обектите или частите в термините на последователни съобщения. Те представляват комбинация от информация, взета от Клас диаграма, Диаграмата на последователностите и на Случаите на използване и описват както статичната структура, така и начина на поведение на системата



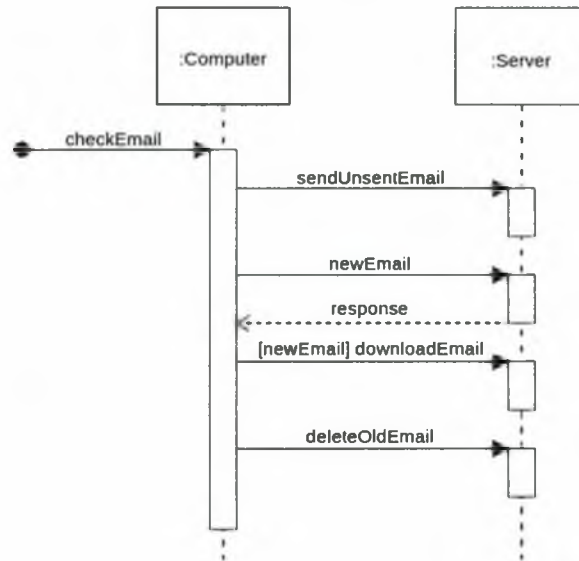
Фиг. Пример на диаграма на комуникацията

- **Диаграма преглед на начините на взаимодействие** – предоставя общ поглед, в който възлите представляват комуникационни диаграми



Фиг. Пример на диаграма на преглед на начините на взаимодействие

- *Диаграма на последователностите* – показва как обектите комуникират един с друг в термините на последователност от съобщения. Също така показват продължителността на живот на обектите, свързани с тези съобщения



Фиг. Пример на диаграма на последователностите

- *Диаграма на времетраенето* – специфична диаграма на взаимодействията, в която фокусът е върху ограниченията във времетраенето

Диаграмата на състояния на протокола е подвариант на диаграмата на състоянията. Тя може да бъде използвана за моделиране на мрежови комуникационни протоколи.

Като инструмент за моделиране ще бъде използван програмният продукт на Sparx Systems Enterprise Architect. Той ще бъде използван за моделиране на бизнес процеси (Business Process Modeler), моделиране на UML диаграми (UML Modeler), генериране на документи (Document generator) и управление на изискванията (Requirements manager).

5.4.2. 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 управление 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.

5.5. Съдържание на Спецификация на изискванията

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

Спецификацията на изискванията ще включва всички изисквания – бизнес правила, функционални и нефункционални, изисквания към сигурността, системна и софтуерна архитектура, както и ограниченията и предположенията, формиращи успешното конфигуриране и адаптиране на Системата.

По-долу е представено примерно съдържание на спецификацията:

Съдържание

1. Въведение
 - 1.1. Цел на документа
 - 1.2. Цел на системата
 - 1.3. Използвани термини и съкращения
2. Основни предположения и ограничения
3. Концепция и бизнес архитектура на системата
 - 3.1. Обхват на системата
 - 3.2. Основни функции на системата и действащи лица
 - 3.3. Логически модел
 - 3.4. Карта на сайта/ порталното решение
4. Информационни обекти
 - 4.1. Видове информационни обекти
 - 4.2. Жизнен цикъл на основните информационни обекти
5. Бизнес правила
6. Описание на бизнес процесите
 - 6.1. Процес 1
 - 6.2. Процес 2
7. Функционални изисквания
 - 7.1. Модул 1
 - 7.1.1. Обща диаграма на функционалностите
 - 7.1.2. Функционално изискване 1
 - 7.1.3. Функционално изискване 2



7.2. Справки

8. Интеграции с външни системи

8.1. Интеграция с [име на система]

9. Нефункционални изисквания

9.1. Технически изисквания

9.2. Изисквания за натовареност на системата

9.3. Изисквания за производителност

9.4. Изисквания за оперативност и преход към новата система

9.5. Изисквания за сигурност

9.6. Изисквания за използваемост

9.7. Изисквания за надеждност и наличност

9.8. Изисквания за конфигуриране

10. Приложения

10.1. Шаблони на документи

10.2. Шаблони на мейл нотификации

10.3. Номенклатурни списъци




5.6. Проследяване на изисквания


Проследяването на изискванията е важна част от жизнения цикъл на проекта и има за цел да осигури доставката на качествен продукт в края на проекта. Извършва се посредством различни техники.

За мониторирането статуса на изискванията по време на изпълнението на проекта ще използваме Матрица за проследяване на изискванията (Requirements traceability matrix).

Това е документ, в който се дефинират взаимовръзките между всички компоненти на системата даващ възможност за добро управление на обхвата на проекта. В него са включени всички потребителски изисквания към системата, описани във вид на случаи на употреба, с възможност за дефиниране на връзките помежду им. Матрицата за проследяване на изискванията гарантира, че всяко изискване добавя бизнес стойност, като я свързва с целите на бизнеса и проекта. Тя осигурява средство за проследяване на изискванията през целия жизнен цикъл на проекта, което спомага за това изискванията, одобрени в техническата спецификация, да се предоставят в края на проекта. И не на последно място, тя предоставя структура за управление на промените в обхвата на проекта.

Този процес включва, но не се ограничава до проследяване на:

- Изисквания към бизнес нуждите, възможностите и целите;
 - Изисквания към целите на проекта;
- 

- 
- Изисквания към обхвата на проекта / продуктите на WBS;
 - Изисквания към дизайна на продукта;
 - Изисквания за разработване на продукти;
 - Изисквания към стратегията за тестване и сценариите за тестване;
 - Изисквания на високо ниво към по-подробни потребителски случаи на употреба.


Създаването на Матрица за проследяване на изискванията се извършва още в началото на проекта.

5.7. Обосновка за предложените подход и инструменти за бизнес анализ

Прилагайки правила за извършване на бизнес анализ, компаниите успяват да направят жизнеспособни промени, за да подобрят организацията си и да постигнат своите цели. При използването на този подход се спомага за намаляването на разходите по проектите чрез увеличаване на възвръщаемостта на инвестициите.

Предлаганите от нас подход и инструменти за извършване на бизнес анализ са базирани на световно утвърдени стандарти и добри практики. При изпълнение на задачите, свързани с дейностите ще се придържаме към методиката BABOK (Business Analysis Body of Knowledge) на Международния Институт по бизнес анализ (МИБА). Тя е разработена от бизнес анализатори и е изцяло базирана на натрупан опит при разработката на продукти в софтуерното инженерство.

По-долу са изброени най-основните предимства при прилагането на подхода:


- Повишаване на възвращаемостта на инвестициите за проекта;
 - Успешно завършен проект;
 - По-ниски разходи;
 - Добро коопериране между заинтересованите страни.
- 

Задачите свързани с бизнес анализа се изпълняват по време на целия жизнен цикъл на проекта, съобразени с гъвкав подход на разработване на системите, което допринася за своевременно отстраняване на допуснати неточности, така че в края на проекта да се постигне изцяло изпълнени цели на проекта, в срок, бюджет и обхват.

5.8. Подход за прилагане на continuous integration практики

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.



[Handwritten signature]


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 непрекъсната интеграция, въпреки че това може да се направи и ръчно. За много хора, непрекъснатата интеграция е синоним на използването на сървър за автоматизирана непрекъсната интеграция, който автоматично стартира процеса по компилиране.
 - 6) Компиляцията 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 конфигурация 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.

5.9. Dependency management инструмент

С напредването на разработката се увеличава експоненциално и броя на взаимодействия и зависимостите. Независимите задачи стават по-малко на брой и повечето от задачите стават все по-зависими от изпълнението на други задачи.

В един момент управлението на всички взаимозависимости става тежък и трудоемък процес и при разработчиците възниква необходимостта от елегантно представяне на зависимостите по проекта.

Тук възниква необходимостта от наличието на Dependency management инструмент, с помощта на който разработчиците да дефинират по-точно и по-прецизно зависимостите между отделните елементи с цел по-лесното им проследяване и управление. Управлението на зависимостите Изпълнителят ще управлява посредством Dependency Structure Matrix (DSM) (позната още и като Design Precedence Matrix или Problem Solving Matrix).

Dependency Structure Matrix (DSM) е квадратна матрица, използвана за представяне на зависимостите по проекта. DSM представя информацията по начин по който лесно и бързо да може да се придобие представа за това коя задача от коя зависи, т.е. кои задачи зависят от изхода на други задачи. DSM се използва с цел представяне на обратна връзка и циклични зависимости.

В редовете и колоните се попълват задачи, като те трябва да бъдат едни и същи и подредени в еднаква последователност. За всяка клетка, където задачите, зависещи една от друга се засичат отбелязваме като X, ако задачата в колоната зависи от задачата в реда.

Blue

Задача	A	B	C	D	E
A		X			X
B	X			X	
C	X	X			
D			X		
E	X	X	X		

[Handwritten signature]

Изграждането на добре обмислен DSM би ни позволило да използваме компютърни средства, за да дадем приоритет на изпълнението на задачите, чиито редове имат максималните "X".

Когато се използва правилно, DSM предоставя много компактен начин за представяне на задачите и ни позволява да направим управлението на софтуерната разработка и проекта като цяло по-ефективно.

6. Гаранционна поддръжка

Кандидатът ще осигури за своя сметка гаранционна поддръжка за период от минимум 24 месеца след приемане в експлоатация на системата.

Гаранцията ще включва всички необходими дейности за поддръжка на работоспособността на системата. При необходимост, по време на гаранционния 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, заложиени в спецификациите.

Минималният обхват на гаранционната поддръжка трябва да включва:

- Извършване на диагностика на докладван проблем с цел осигуряване на правилното функциониране на системите и модулите;
- Отстраняване на дефекти в информационната система;
- Подмяна или ремонт на дефектирали хардуерни компоненти в рамките на гаранционните условия на производителя за всеки доставен компонент;
- 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;
- По време на гаранционния период Изпълнителят трябва да осигури използването на уеб базирана система за регистриране и проследяване на инцидентите, която да се ползва съвместно от представители на Изпълнителя и Възложителя.

Приоритетите на проблемите се определят от Възложителя в зависимост от влиянието им върху работата му. В зависимост от приоритета на проблема се изисква следната достъпност на услугата и време за реакция:

MLC

Приоритет	Описание	Достъпност на услугата	Време за реакция
Висок	<p>Въздействие върху работните процеси вследствие нарушаване на основните функционалности на структуроопределящите компоненти, водещо до пълната системна неработоспособност;</p> <p>Системата е напълно неработоспособна или работата ѝ предизвиква невъзстановимо записване на грешни данни.</p>	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	<p>Въздействие върху работните процеси вследствие частично нарушаване на функционалностите на структуроопределящите компоненти, водещо до частична системна неработоспособност;</p> <p>Част от съпътстващата функционалност на системата е неработоспособна, например справки и др.</p>	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	<p>Въздействие върху работните процеси вследствие нарушаване на функционалности на неструктуроопределящите компоненти, водещо до частична системна неработоспособност;</p> <p>Има отпаднал елемент на системите за мониторинг на абонатна станция;</p> <p>Системата е работоспособна, но забелязаният недостатък предизвиква неудобство и затруднение при работа.</p>	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.

Гаранционната поддръжка НЕ включва:

- разработка на нова функционалност;
- помощ на крайните потребители на системата;
- оперативно обслужване информационната система и доставеното оборудване.

000118