



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Проект „Повишаване квалификацията на служителите от администрацията на централно ниво чрез усъвършенстване на знанията и практическите им умения за управление на софтуерни ИТ проекти в съответствие със съвременните методологии“, осъществяван с финансовата подкрепа на Оперативна програма „Административен капацитет“ (ОПАК), съфинансирана от Европейския съюз, чрез Европейския социален фонд”,
Договор № К13-22-1/05.03.2014 г.

съгласно

НАРЪЧНИК

ДЕЙНОСТ 1.

ПРОВЕЖДАНЕ НА ОБУЧЕНИЕ ЗА БИЗНЕС- АНАЛИТИК ЗА 62 СЛУЖИТЕЛИ НА ЦЕНТРАЛНАТА АДМИНИСТРАЦИЯ И ИЗДАВАНЕ НА СЕРТИФИКАТИ ЗА ПРОВЕДЕНОТО ОБУЧЕНИЕ

Изготвен в изпълнение на Договор № Д-37/11.12.2014 г.

между

МИНИСТЕРСТВО НА ТРАНСПОРТА,
ИНФОРМАЦИОННИТЕ ТЕХНОЛОГИИ И
СЪОБЩЕНИЯТА

и

„КОНСОРЦИУМ ИТ ОБУЧЕНИЯ 2015“ ДЗЗД





Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

„КОНСОРЦИУМ ИТ ОБУЧЕНИЯ 2015“ ДЗЗД

**София 1040, ж.к. Изток, бул. Драган Цанков 36, СТЦ Интерпред, блок А, ет.6; тел:
024210040; имейл: ittraining2015@newhorizons.bg;**

Авторски колектив:

Костадин Купенов, Елена Бугова

Одобрил: Николай Пенев – ръководител на проекта

София, Май 2015 г.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Съдържание

1. Съкращения	4
2. Речник на термините	4
3. Въведение	5
4. Модул 1: 'Бизнес анализатор' като роля. Въведение в ролята на бизнес анализатора	6
5. Модул 2: Методологии и роли в разработката на софтуер. Rational Unified Process (RUP, Унифициран процес за разработка на софтуер).....	8
6. Модул 3: Какво е Бизнес анализ? Въведение в бизнес анализа.	15
7. Модул 4: Въведение в изискванията. Какво е изискване?	19
8. Модул 5: Видове изисквания	20
9. Модул 6: Бизнес анализ документи.....	24
10. Модул 7: Процесът по писане на изисквания	28
11. Модул 8: Enterprise Analysis (Анализ на предприятието/организацията)	36
12. Модул 9: Дефиниране на бизнес проблем.....	37
13. Модул 10: Дефиниране на обхвата.....	40
14. Модул 11: Визуализиране на обхвата	43
15. Модул 12: Бизнес казус	47
16. Модул 13: Потребителски изисквания и изисквания на заинтересованите страни.....	51
17. Модул 14: Техники за извличане на потребителски изисквания	54
18. Модул 15: Представяне на потребителски изисквания	58
19. Модул 16: Функционални изисквания.....	61
20. Модул 17: Структуриране на изискванията и техните атрибути	66
21. Модул 18: Практика: Извличане на функционални изисквания от потребителски изисквания и use cases	69
22. Модул 19: Практика: Извличане на функционални изисквания от артефакти.....	72
23. Модул 20: Нефункционални изисквания.....	74
24. Модул 21: Въведение в UML.....	81
25. Модул 22: Въведение в структурните UML модели.....	84
26. Модул 23: Практика: Клас диаграма.....	88
27. Модул 24: Въведение в поведенческите UML модели.....	90
28. Модул 25: Практика: Диаграма на състоянията.....	106
29. Модул 26: Практика: Диаграма на дейностите	107
30. Модул 27: Верифициране на изисквания	108
31. Модул 28: Валидиране на изисквания	109
32. Модул 29: Комуникация на изисквания.....	111



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

33. Модул 30 Управление на изисквания	113
34. Приложение	117
35. Източници	120

1. Съкращения

BA - Business Analyst

BABOK - Business Analysis Body of Knowledge

BPMN- Business Process Modeling Notation

IBM - International Business Machines Corporation

International Institute of Business Analysis (IIBA)

RACI - Responsible, Accountable, Consulted, Informed

RFI (Request for Information)

RFP (Request for proposal)

RUP - Rational Unified Process (Унифициран процес за разработка на софтуер)

SMART - Specific, Measurable, Achievable, Realistic, Testable

SME - Subject Matter Expert

SWOT(Strengths, Weaknesses, Opportunities, Threads)

UML - Unified Modeling Language

2. Речник на термините

Agile Business Analyst - бизнес аналитик/анализатор в Agile проекти

Agile Методология - начин за организиране на процеса по разработка на софтуер, според който продукта се доставя поетапно и на части с цел намаляване на риска и по-бързо доставяне на резултат

Assumptions and Constraints - твърдения, свързани с изискванията, които приемаме за верни и ограничения, най-често свързани с избраната технология

Business Analyst - роля в проектите, която трансформира бизнес нуждите в изисквания и следи дали крайното решение ги покрива

Business Owner - роля в проектите, която представлява клиента

Business requirement - изисквания на високо ниво, които изразяват бизнес цел

Business rule - изисквания, които определят как работи бизнеса - правила, условия, формули



Европейски съюз



ОПАК Експерти в действие



Европейски социален фонд
Инвестиции в хората

Customer - клиентът , който плаща за крайния продукт и/или ще го ползва

Feature Creep - неконтролирана промяна на обхвата в проекта

Functional requirement - изисквания, които определят какви функции и данни ще поддържа крайното решение

Gold-Plating - предлагане и разработка на функционалности, които не отговарят на бизнес нуждата и водят до промени в бюджета и сроковете

Non-functional requirement - изискване, което дефинира качествена работа на избраното решение

Product Owner - роля в Agile(Scrum) проекти, представлява интересите на клиента

Project Manager - роля в Waterfall проекти, която управлява проекти и отговаря за доставянето им навреме и спрямо планирания бюджет

Regulatory requirements - изисквания от външни институции, които определят правилата, по които да работи бизнеса

Reporting requirements - изисквания за справки

Scrum Master - роля в Agile(Scrum) проекти, следи за планирането и ефективността на екипите

Stakeholder - засегната страна в проекта (ИТ екип, Клиент, Потребители, други отдели в организацията)

Stakeholder requirements - изисквания, чийто източник са различни засегнати страни

Transition requirements - временни изисквания, които обикновено се дефинират при замяна на една система с друга

User requirement - под група на stakeholder requirements

Waterfall Методология - начин за организиране на процеса по разработка на софтуер, според който продукта се доставя еднократно. Етапите в проекта се изпълняват последователно. Един етап започва само ако предходния е завършен напълно.

3. Въведение

Наръчник за обучение по Бизнес анализ съдържа учебни материали, упражнения и препратки, речник с термини, полезни връзки и множество примери илюстриращи разглежданите теми. Целта е да се улесни придобиването на умения за ролята на бизнес анализатор (BA) или за ролята на системен бизнес анализатор (BSA) в модерната организация. Примерите демонстрират как бизнес анализаторът работи в ИТ



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

екип и как формулира бизнес изисквания, достатъчно ясни за ИТ, как да дефинира бизнес нуждите и как да ги представи в текстови и визуален формат.

4. Модул 1: 'Бизнес анализатор' като роля. Въведение в ролята на бизнес анализатора

- Нуждите на клиентите и нуждите на ИТ
- Основните познания и умения, необходими на Бизнес анализатора

4.1. Цел

Запознаване с ролята 'бизнес анализатор', очакванията към ролята, уменията, които са необходими

Често ролята на бизнес анализатор се поема от някой в екипа от проекта, без да има обособена позиция 'Бизнес анализатор'. Очакванията към тази роля могат да бъдат разнообразни - да съвместява Бизнес анализ с управление на проекти или с изграждане на софтуерна архитектура например.

В рамките на този Наръчник се разглеждат добрите практики в Бизнес анализа, без той да се съвместява с друг вид роли от проектите.

4.2. Ролята на бизнес анализатора - Възможни наименования:

- Бизнес аналитик (Business Analyst)
- Бизнес анализатор (Business Analyst)
- Системен аналитик (System Analyst)
- Функционален аналитик (Functional Analyst)
- Бизнес системен аналитик (Business System Analyst)
- Аналитик

Организациите наричат по различен начин ролята, но най-разпространеното наименование е Бизнес аналитик или Бизнес анализатор. В този документ ще употребяваме основно Бизнес анализатор.

4.3. Очаквания към ролята:

- Бизнес анализатора като преводач между Бизнеса и ИТ специалистите
- Бизнес анализатора като защитник на интересите на клиента/потребителите
- Бизнес анализатора като гарант за разработката/имплементацията на правилните изисквания

4.4. Нуждите на клиентите и нуждите на ИТ

Какви са нуждите на Бизнеса/организацията?

Кой е Бизнесът?

Бизнесът/организацията обикновено е клиента в проекта.

Клиентът е тази страна в проекта, която плаща за реализацията и/или ползва крайния продукт.

Какво е характерно за Бизнеса/организацията?

Слаби или никакви технически познания

Очаква разработката/имплементацията да е готова веднага

Затруднява се да приоритизира изискванията си

Не различава изисквания от желания

Често не е наясно какво точно иска - предлага решение за части от цялостния проблем

Бизнес анализаторът подпомага Бизнеса/организацията като:

Разбира от какво наистина Бизнесът/организацията има нужда

Подпомага Бизнеса/организацията в избора на най-подходящото решение

Подпомага Бизнеса/организацията в приоритизирането на нуждите

Какви са нуждите на ИТ специалистите?

Кои са ИТ специалистите?

ИТ специалистите в проектите са: програмисти, дизайнери, архитекти, тестери, проектни мениджъри, т.е. хората с експертни познания, които могат да доставят крайното решение, от което Бизнеса има нужда.

Какво е характерно за ИТ специалистите?

Не познават индустрията/администрацията

Трудно разбират нуждите на Бизнеса/администрацията

Винаги изискват повече време и ресурси

Не обичат да четат документация

Често изпадат в крайности - имплементират по-малко или повече

Не обичат да комуникират (въпроси, проблеми, предложения)

Бизнес анализаторът подпомага ИТ специалистите като:

Подпомага ИТ специалистите в разбирането на бизнес нуждата

Представя изискванията и бизнес нуждата по начин, който ще улесни ИТ специалистите

Съгласува възможността за имплементация на всяко от изискванията

Подпомага ИТ специалистите в етапа на разработката

Гарантира, че правилните изисквания са имплементирани

4.5. Основните познания и умения, необходими на Бизнес анализатора

- Да разбира бизнес нуждата в дълбочина (проблемът зад проблема)
- Комуникационни умения
 - Да умее да решава конфликти
 - Да умее да мотивира
 - Да познава културни особености
- Да умее да води ефективни срещи и интервюта
 - Да умее да задава правилните въпроси
 - Да разпознава емоции и настроения
- Презентационни умения
- Аналитични умения (да разглежда средата в нейната цялост)
- Знания в бизнес домейна/областта на съответния бизнес/спецификата на организацията
- ИТ Познания

5. Модул 2: Методологии и роли в разработката на софтуер. Rational Unified Process (RUP, Унифициран процес за разработка на софтуер).

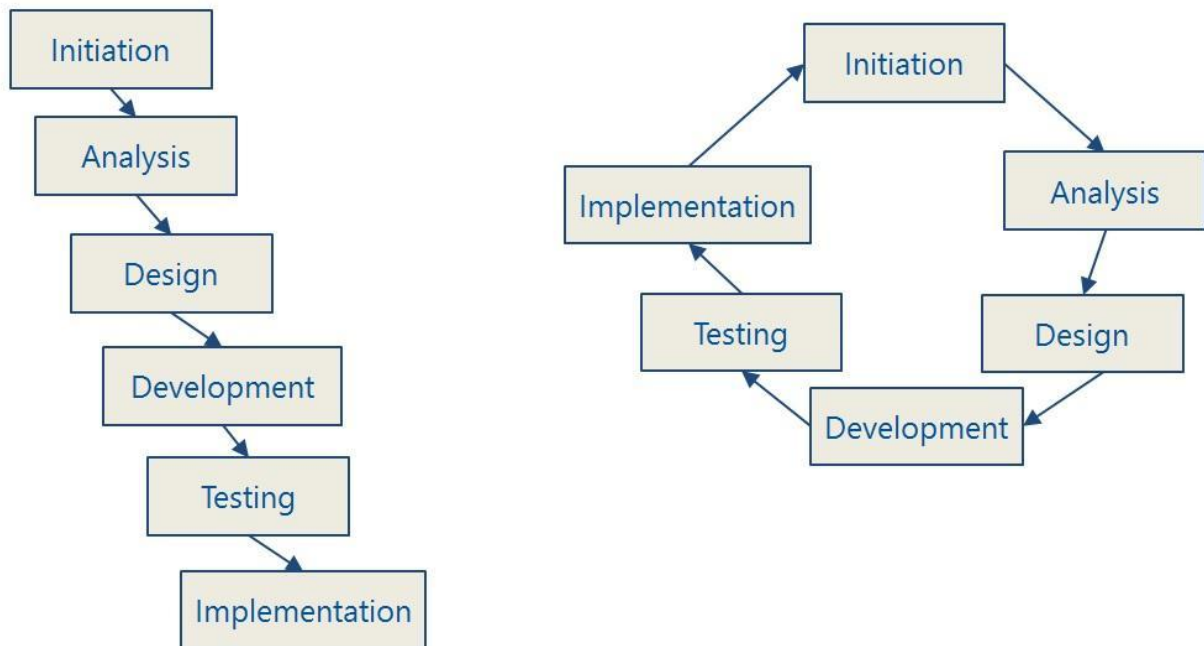
- Agile и Waterfall методологиите и Бизнес анализатора
- Унифициран процес за разработка на софтуер. Фази.
- Rational Unified Process (RUP) - елементи, артефакти и документи.

- Добрите практики на RUP.

Цел

Запознаване с методологии за разработка на софтуер - Waterfall и Agile, Rational Unified Process (RUP).

5.1. Agile (Гъвкав, Адаптивен) и Waterfall („Водопад“, Каскаден) методологиите и Бизнес анализатора



Ако стандартно всеки проект има следните фази:

- Initiation - Начало (решава се концепцията на проекта)
- Analysis - Анализ (изясняване на изисквания)
- Design - Дизайн (проектиране на системата)
- Development - Разработка
- Testing - Тестване
- Implementation - Внедряване

Начинът, по който се изпълняват те, определя модела на процеса по разработка на софтуер.

Waterfall или наричан още инкрементален модел представлява последователно изпълнение на фазите в проектите по разработка на софтуер. Текущата фаза трябва да се изпълни напълно, за да може да започне следващата фаза. Waterfall е по-старият модел от двата (счита се, че се прилага от 1970 насам), взаймства се от манифактурните индустрии.

Agile модела възниква през 2001 (т. нар. ‘Agile Manifesto’) като реакция на експерти в областта срещу недостатъците на Waterfall модела.

При Agile модела (итеративен подход) крайният продукт се доставя на части, с цел по-голяма гъвкавост.

Предстои да сравним двата модела:

- по различни критерии - продължителност, гъвкавост..
- ролите, които участват
- Бизнес анализът и при двата модела

Waterfall и Agile - сравнение

Критерий	Waterfall	Agile
Срок	няколко месеца	няколко седмици
Резултат	крайният продукт е готов в самия край на проекта	крайният продукт се доставя на фази-модули
Гъвкавост за промени	ниска	висока
Промени	търсят се начини да не бъдат допускани никакви промени	очакват се промени и се приветстват
Време за откриване на проблеми	дълго (заради последователността на фазите)	кратко (проблемите се идентифицират в ранните фази)

Роли в Waterfall проекти

Business Owner (Собственик на проекта)	Финансово отговорен за проекта. Слабо участие в проекта. Получава регулярни статуси за прогреса по проекта от Ръководителя на проекта . Взема решение при ескалации.
Project Manager (Ръководител проект)	Планира и управлява ресурсите в проекта, за да бъде доставено очакваното решение в срок.
Team (Проектен екип)	Имплементира продукта от проекта. Обикновено в статус срещите и при взимането на решения учатват само Лидери на екипи/Team Leads. Екипитите обикновено са големи - над 30 човека.

Business/User (Потребители) Subject Matter Experts (Експерти)	На разположение за консултации през целия проект. Често участват в тестовете на продукта преди релийза му.
--	---

Роли в Agile (Scrum) проекти

Product Owner (Собственик на продукта)	<ul style="list-style-type: none"> ● Дефинира нови функционалности и датите за публикуване на нови версии ● Приоритизира функционалности ● Замества Клиента ● Приема или отхвърля работа ● Отговорен е за финасовия резултат
Scrum Master (Скръм Мастер)	<ul style="list-style-type: none"> ● Фасилитира процеса по планиране (не традиционната роля на Проджект мениджър) ● Следи за ефективността на процеса и екипите ● Премахва пречки за работния процес
Team (Екип)	<ul style="list-style-type: none"> ● 7+- 2 участника в екипите ● организира и изпълнява сам задачите си ● комуникира тясно с Product Owner-a ● участва в планирането на спринтовете и demo срещите
Customer (Клиент)	<ul style="list-style-type: none"> ● Дефинира списък от нови функционалности ● Работи заедно с екипа (демо срещи, планиране)

Роли в Waterfall и Agile проекти - сравнение

Waterfall	Agile
Business Owner/BA Собственик на проекта/ Бизнес анализатор	Product Owner (Собственик на продукта)
Project Manager (Ръководител проект)	Scrum Master (Скръм Мастер)
BA (Бизнес Анализатор)	Agile BA („Гъвкав“ бизнес анализатор)
Team (Екип)	Team (Екип)

Business/User (Потребители) Subject Matter Experts (Експерти)	Customer (Клиент)
---	-------------------

Бизнес анализ в Waterfall и Agile проекти

Критерий	Waterfall	Agile
Взаимоотношения	Връзка между Бизнеса и ИТ	Тясно работи с Собственика на продукта и екипа
Участие в проекта	Най-вече в началото на проекта Кратко участие по време на разработката (при необходимост от разясняване на изисквания) В тестовете на крайния продукт, за да валидира, че правилния продукт е доставен.	През цялото време на проекта
Изисквания	Всички изисквания да бъдат събрани предварително. Изискванията остават замразени до края на проекта.	Стратегически важните изисквания ще бъдат открити преди да започне разработката. Голяма част от изискванията ще бъдат уточнени по време на разработката.
Документация	Подробна документация	Почти липсва (описват се само кратки сценарии - user stories). Целта е да се поддържа валиден регистър/backlog с добре дефинирани сценарии
Модели, Прототипи	Предварително изготвени модели, use cases (use cases -Документ за случаите на употреба)и прототипи на екрани. Очакват се минимални промени.	Моделите и прототипите се имплементират директно. Променят се в движение.

5.2. Унифициран процес за разработка на софтуер. Фази

Методология за организиране на процеси по софтуерна разработка (Rational Unified Process, RUP). Изготвена е от Rational Software Corporation, през 90-те години на 20 век.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Създадена е с цел повишаване на продуктивността и намаляване на риска по време на софтуерните разработки. От 2003 година насам, RSC е подразделение на IBM.

Проектите по RUP методология се състоят от 4 фази:

1. Предварителна фаза (Phase Inception)

- Изготвя се бизнес казус
- Визия за решението
- Основните функционалности

2. Фаза Разработка (Elaboration phase)

- Use Case (Потребителски сценарии) модел
- Не-функционални изисквания

3. Фаза Конструирание (Construction phase)

- Разработка/имплементация (Създаване на код)
- Изготвяне на ръководство за експлоатация

4. Преходна фаза (Transition phase)

- Дейности по прехвърляне на решението в продукционна среда
- Обучения

5.3. Rational Unified Process (RUP) - елементи, артефакти и документи

Елементите са различните задачи и техните изпълнители - например изготвяне на изисквания, архитектура, дизайн

Артефактите - модели, програмен код, use case модел

Документи – vision (визия) документ, бизнес казус, документ с архитектура, ориентирана към потребителите документация,

5.4. Добрите практики на RUP

1. Разработване на софтуер поетапно (итеративно)

Намалява риска, осигурява фокус за екипа и свобода за промени в изискванията и графика

2. Управление на изисквания

Описва как се извличат, организират и документират изискванията



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

3. Архитектура изградена от компоненти

Описва как да се изгради еластична архитектура, която може да се използва повторно

4. Визуално моделиране на софтуера

Предоставя структура и поведение на системата.

5. Верифициране качеството на софтуера

Подпомага дейностите свързани с планиране, изпълнение и оценка на тестове за качеството на работа на системата

6. Контрол върху промените по софтуера

Предписания за контрол върхи промените

Запознайте се с методологията за разработка на софтуер в организацията от примерния казус (тя е пример за приложение на RUP).

ВА Банк е българска банка, със седалище в София. Банката разполага с 210 структурни единици в цялата страна.

Банката разполага със собствен ИТ отдел. Софтуерната разработка в ВА Банк следва RUP методология и включва следните етапи, валидни за всеки проект:

1. Начална фаза

По време на този етап Изпълнителят следва да извърши анализ на потребностите и да дефинира целта, обхвата и основната функционалност на софтуерното приложение – бизнес изисквания, потребители на системата, бизнес процесите, които ще бъдат електронизирани, бизнес казус за проекта.

2. Проектиране

По време на етап проектиране, Изпълнителят трябва да изготви Системен проект (детайлна техническа спецификация) до тридесет дни от датата на подписване на договора. Същия трябва да включва минимум следните задачи:

- Дефиниране на детайлни изисквания и бизнес процеси, които трябва да се разработят
- Дизайн на информационната система, хардуерна и комуникационна инфраструктура
- Изготвяне на план на техническа реализация
- Дефиниране на изисквания към потребителския интерфейс
- Дефиниране на политика за сигурност и защита на данните

При документиране на изискванията, с цел постигане на яснота и стандартизация на документите, е необходимо да се използва стандартът за описание на бизнес модели UML или еквивалентна стандартна и широко призната нотация.

Изготвеният системен проект се представя за одобрение на Възложителя. В случай на забележки, корекции или допълнения от страна на Възложителя, Изпълнителят е длъжен да ги отрази в срок не по-късно от 10 (десет) дни.

3. Разработка

Етапът на разработка трябва да съответства на методологиите и архитектурата, утвърдени в системния проект. Етапът на разработката включва изпълнение на следните задачи:

- Софтуерна разработка
- Изготвяне на техническа документация
- Тестване на системата (функционални, интеграционни и тестове за производителност). В рамките на тази фаза се отстраняват разминаванията между изискванията и функционалността на системата, както и откритите програмни грешки.

4. Внедряване

Етапът на внедряване включва изпълнението на следните дейности:

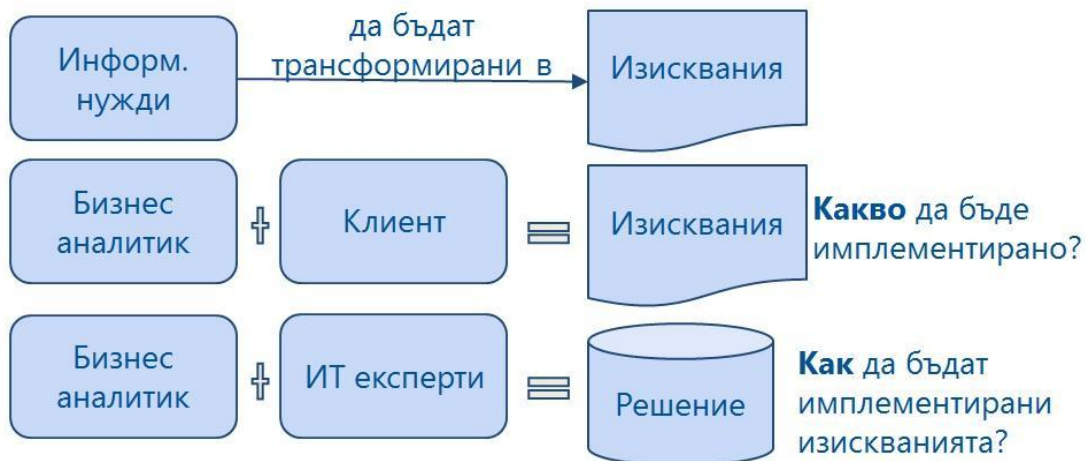
- Миграция на данни (ако се налага)
- Инсталиране и настройка на софтуерните приложения в реална среда
- Извършване на приемателни тестове на системата
- Провеждане на обучение

6. Модул 3: Какво е Бизнес анализ? Въведение в бизнес анализа.

- Цели на бизнес анализа.
- Последичите от недобре изготвен бизнес анализ.
- Какво не трябва да прави бизнес анализаторът.

6.1. Цели на Бизнес анализа

Целта на Бизнес анализа е да трансформира нуждите на клиента/потребителите в изисквания, които да могат да бъдат разработени/имплементирани.



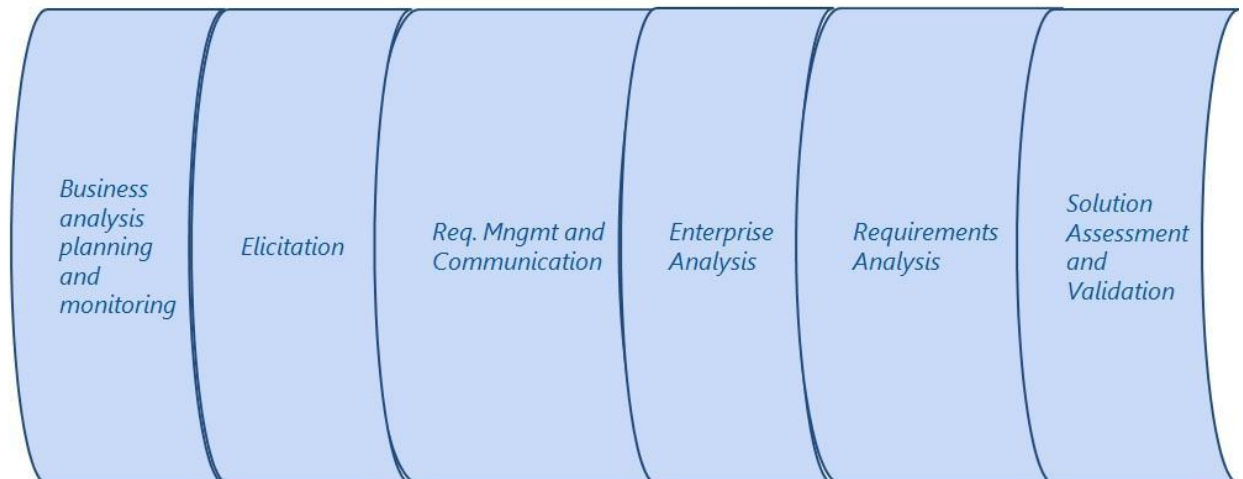
- Бизнес анализът трансформира нуждите в изисквания
- Заедно с Клиента, Бизнес анализаторът изяснява **какво** трябва да бъде разработено
- Заедно с ИТ експертите, Бизнес анализаторът изяснява **как** ще бъдат имплементирани изискванията (решението за реализация)

BAВOK Области на знания

Business Analysis Body of Knowledge (BAВOK) стандартизира добрите практики в Бизнес анализа. Поддържа се от International Institute of Business Analysis (ИВА). [BAВOK v.2.0](#) (издадена 2009 г.) представя следната дефиниция за това какво е Бизнес анализ:

Съвкупност от задачи и техники, използвани за осъществяване на връзка между засегнатите страни, с цел разбиране на структура, политики и операции в организацията и предлагане на решения, които да подпомогнат организацията в постигането на поставените цели.

BAВOK дефинира следните зони на знание:



- **Business analysis planning and monitoring** - задачи и техники по планиране и наблюдение

пример: изготвяне на график за писане на изисквания, план за комуникация

- **Elicitation** - задачи и техники по извличане на изисквания

пример: провеждане на интервюта с потребителите на бъдещата система

- **Requirements management and communication** - задачи и техники по управление на изискванията и тяхната комуникация

пример: Преглед от колеги / Peer review

- **Enterprise analysis** - задачи и техники за анализ на ниво организация

пример: изготвяне на бизнес казус, SWOT (Strengths, Weaknesses, Opportunities, Threads) анализ

- **Requirements Analysis** - задачи и техники за анализ на изискванията

пример: детайлизиране и приоритизиране на изисквания

- **Solution Assessment and Validation** - задачи и техники по оценка на решението за реализация



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

пример: изготвяне на тест планове, тестове по разработката/имплементацията на най-важните бизнес изисквания

6.2. Последиците от никакъв или не добре изготвен бизнес анализ

- Клиентът не участва достатъчно в проекта и съответно доставения продукт е различен от очакванията
- Потребителите на системата не участват в проекта достатъчно и в крайна сметка крайния продукт не се ползва
- Програмистите предлагат не поискани функционалности (Gold plating), което води до закъснения в графика
- Промените в изискванията не са контролирани (feature/scope creep). Пропуска се да се оцени въздействието и крайния продукт е с ниско качество.
- Пропуснати важни изисквания, предизвикат допълнителни доработки и отклонения - сроковете и бюджета не се спазват

6.3. Какво бизнес анализаторът НЕ прави!

- Не се ангажира и не обещава срокове на клиента
- Не променя, изтрива, добавя изисквания по собствена преценка (съгласува и комуникира всяка промяна!)
- Не дефинира изисквания, които не могат да се измерят и тестват
- Не представя изискванията като равни по приоритет. Но, подпомага клиента да приоритизира изискванията си
- Не игнорира информационните нужди на участниците в проекта (изготвя допълнителни материали - диаграми, таблици, презентации; отговаря на възникнали въпроси)

Най-общо като полза в проектите, в които участват бизнес анализатори още е ,че видимостта се подобрява - по-ясно е какъв е смисъла от проекта, какъв е статуса на всяко изискване, кои са отворените въпроси, кой участва и какъв е приносът му.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

7. Модул 4: Въведение в изискванията. Какво е изискване?

- Работа по извличане на изисквания
- Цената на зле написаните изисквания

7.1. Цел

Изграждане на разбиране за това какво е ‘изискване’. Изискването като базов елемент в Бизнес анализа.

Изискването изразява възможност за изпълнението на дадена нужда.
Пример: ‘Системата ще позволява на потребителите със съответни права да търсят хотелска резервация по следните критерии: дата, клиент’

Изискванията са независими от решението за разработка/имплементация (дизайна на системата). Анализът дефинира какво (самото изискване), а дизайнът определя как (решението, което изпълнява изискването).

Като премахнем дизайн елементите от примера по-горе, изискването получава следния вид:

Пример: ‘Системата ще позволява на потребителите със съответни права да търсят хотелска резервация, чрез ~~попълване на изтечащи екрани~~ за дата, клиент и бутони ‘Потвърди’, ‘Откажи’

Внимание! Избягвайте подробности от дизайна при дефинирането на изискванията. (Като задрасканите) Изискванията следва да са независими технологично. Дизайнът (начинът за имплементация (кодиране)) следва да се уточни отделно.

Feature (Възможности, функционалности)- съвкупност от изисквания, които изграждат бизнес значима употреба. Пример:: поръчка без регистрация, навигационно меню със секция Новини

(Според BABOK v2.0) **Изискване е:**

1. условие или качество, необходимо на дадено заинтересовано лице, за да реши проблем или да постигне цел
2. условие или качество, което трябва да бъде изпълнено или поддържано от някакво решение или компонент на решение, което удовлетворява договор, стандарт, спецификация, или друг формален документ
3. документирано условие или качество, съгласно 1 или 2

8. Модул 5: Видове изисквания

- Изисквания, бизнес правила и ограничения
- Бизнес изисквания
- Изисквания на потребителите
- Функционални изисквания
- Нефункционални изисквания
- Изисквания за справки
- Преходни изисквания
- Източници на изисквания

8.1. Цел: Запознаване с видовете изисквания

Бизнес изисквания (Business requirements)	Бизнес правила (Business rules)
Изисквания на заинтересованите страни (Stakeholder requirements)	Регулаторни изисквания (Regulatory requirements)
Потребителски изисквания (User requirements)	Предположения и ограничения (Assumptions and Constraints)
Функционални изисквания (Functional requirements)	Изисквания за справки (Reporting requirements)
Нефункционални изисквания (Non-functional requirements)	Преходни изисквания (Transition requirements)

Изискванията, които пишем трябва да бъдат SMART (Specific/Конкретни, Measurable/Измерими, Achievable/Постижими за проекта, Realistic/Реалистични спрямо обхвата, Testable/Възможно е да се изтестват) !

8.2. Бизнес изисквания

- Бизнес изискванията изразяват бизнес нужди
- Бизнес изискванията са по-обща



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Бизнес изискванията са технологично независими

‘Искам да подобрим процеса по управление на клиентски контакти’

‘Искам да намалим времето за пресмятане на клиентските задължения’

‘Системата трябва да управлява хотелски резервации’

8.3. Изисквания на заинтересованите страни

- Представяват нуждите и очакванията на определен човек/група от заинтересовани страни

‘Всички финансови експерти в нашата организация ще трябва да могат да създават месечни ДДС отчети’

‘В интерес на проекта е да се използва отделна, специално оборудвана зала за взимане на решения относно промени в обхвата, времето и бюджета.’

‘Системата трябва да предоставя възможност за обединение на контрагенти, така че, да е възможно генерирането на групи получатели.’

8.4. Потребителски изисквания

- Потребителските изисквания са по-конкретни от бизнес изискванията
- Потребителските изисквания се обуславят от бизнес изискванията
- Потребителските изисквания често описват последователност от стъпки и действия, които системата трябва да поддържа
- Потребителските изисквания са технологично независими

‘Създаването на хотелска резервация се състои от следните стъпки ..’

‘Преди да приеме резервацията, служителя трябва да провери дали на клиента не се полага някаква отстъпка’



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

‘Трябва да мога да извлека справка за клиентите, като избира различни критерии, например град, възраст, продукт’

8.5. Функционални изисквания

- Функционалните изисквания са функциите, които системата трябва да поддържа, за да реализира бизнес и потребителските изисквания
- Функционалните изисквания дефинират и данните, които системата трябва да поддържа
- Функционалните изисквания са технологично независими

‘Системата ще позволява на потребителите със съответни права да търсят хотелска резервация по следните критерии: дата, клиент’

‘Преди да потвърди закупуване на услуга, системата ще проверява за стари клиентските задължения. Системата ще представя статус на проверката’

8.6. Нефункционални изисквания

- Нефункционалните изисквания изразяват качеството, което системата трябва да поддържа, за да покрива бизнес нуждите
- Нефункционалните изисквания могат да бъдат технологично зависими (верифицират се с ИТ експерти)

‘Системата трябва да е налична 24/7, с изключение на планираните профилактики’

‘Системата трябва да поддържа българска и английска езикови версии.’

‘Системата трябва да поддържа различни нива на достъп според потребителските групи.’

8.7. Бизнес правила

Бизнес правилата изразяват процеси, процедури, формули, дефинирани от външни източници (регулации) или установени от съответната организация

‘Само клиенти с двугодишен договор може да ползват услуга X’

‘Само бизнес клиенти получават безплатна инсталация на закупената услуга.’

‘Бонус ‘Лоялен клиент’ е в размер на 5% от депозирани средства до момента.’

8.8. Регулаторни изисквания

Изисквания, наложени по силата на закон или друга регулация.



‘Ставката на данъка е 20 на сто за:

- 1. облагаемите доставки, освен изрично посочените като облагаеми с нулева ставка;*
- 2. вноса на стоки на територията на страната;‘*

‘Водачът на пътно превозно средство не трябва да се движи без основателна причина с твърде ниска скорост, когато по този начин пречи на движението на другите пътни превозни средства.’

8.9. Предположения и ограничения

Предположения - неща, за които не сме напълно сигурни, но ги приемаме за верни

‘Приемаме, че регулация X няма да се измени в следващите 6 месеца и съответно ще калкулиране клиентските задължения както следва’

Ограничения - обикновено това са технически ограничения, които трябва да имаме предвид

‘Ще преизползваме сегашната инфраструктура, която изпълнява до 100 заявки в минута’

8.10. Изисквания за справки

Изискванията за справки са необходими, за да се оцени успеха на нов продукт или резултатите на организацията като цяло

При уточняване на справки следва да съобразим:

- Какви данни да включва справка, за какъв период
- Да има ли филтри, ако да - какви
- Колко често да се генерира
- Къде да се поддържа/Как да се получава
- Колко време да се пази
- В какъв формат да се генерира
- За кого да е достъпна
- Колко актуални да са данните

8.11. Преходни изисквания

Преходни изисквания се пишат обикновено при миграция на системи. Тези изисквания отпадат веднага след като приключи миграцията.

Примери:

- Миграционен план
- Изисквания за архив/back-up на данни
- Карта - кои стари данни, в кои нови полета да се запишат при миграцията

Пример за конкретно изискване и неговия вид.

Изискване	Вид
1. Системата ще поддържа помощно меню за всяка отделна функционалност	Нефункционално изискване (за по-лесна работа със системата)
2. Информацията трябва да се представя в три измерения - време, клиенти, продукти	Изисквания за справки
3. Омръзнало ми е да попълване по няколко пъти едни и същи клиентски данни	Потребителско изискване
4. За всяка продажба трябва да се калкулира ДДС 20%	Бизнес правило
5. Новите графика трябва да се имплементират на HTML 5 технология.	Ограничение
6. Клиентския номер ще отговаря на Account ID поле в новата система	Преходно изискване
7. С разработката/имплементацията на новата система за документооборот, очаквам понижаване на разходите с 20%	Бизнес изискване (изразява бизнес цел/полза)
8. Приемаме, че очакваното изменение в закона няма да измени начина, по който се изчислява лихвата върху вземането	Предположение
9. Системата ще поддържа информация за: <ul style="list-style-type: none"> - клиенти - доставчици - поръчки 	Функционално изискване (изброява данните, които ще се поддържат)
10. Системата ще позволява достъп на потребителите само в работните часове на деня	Нефункционално изискване за сигурност

9. Модул 6: Бизнес анализ документи

❖ Видове документи и техните употреби:



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- документ за визия и обхват
- документ за бизнес изисквания

9.1. Цел

Преглед на по-известните документи, които бизнес анализаторите пишат. Запознаване със стандартното им съдържание.

Обикновено всяка организация сама определя документите и тяхната структура, според целите, които се стреми да постигне. Следва преглед на стандартните документи, които бизнес аналитиците изготвят и тяхното примерно съдържание.

9.2. Видовете документи (според BABOK, v.2.0)

1. Vision - Документ за визията за промяната.

Включва Бизнес изискванията и възможни алтернативи за постигането им. Обикновено такъв документ се изготвя когато се изяснява концепцията за проекта - кой точно проблем искаме да решим, какви са възможностите за това.

2. Business Requirements Document - Документ с детайлни изисквания.

Включва бизнес изисквания и изисквания на заинтересованите страни. Обикновено се изготвя след като е ясна концепцията.

3. Use cases document - Документ за случаите на употреба.

Включва изискванията на заинтересованите страни, предствени като взаимодействие потребител-система (use cases)

4. Software requirements specification - Документ със системните изисквания

Функционални изисквания към системата, технически ограничения.

5. Supplementary specification - Документ с нефункционалните изисквания към системата.

Тези изисквания до голяма степен зависят от избраното решение така, че често са в допълнителен документ

6. Requirements for vendor selection - Изисквания за избор на доставчици/изпълнители

RFI (Request for Information) - Документ-запитване за това как даден изпълнител решава определени информационни нужди

RFP (Request for proposal) - Документ - запитване дали и как точно даден изпълнител изпълнява конкретни изисквания



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

И двата документа се ползват за случаите, в които искаме да се допитаме и да изберем възможен доставчик на бъдеща система

9.3. Съдържание на Vision документ (според Rational Unified Process методология)

1. Introduction/Въведение
 - 1.1 Purpose/Цел
 - 1.2 Scope/Обхват
 - 1.3 Definitions, Acronyms, and Abbreviations/ Дефиниции, акроними и абривиатури
 - 1.4 References/Референции
 - 1.5 Overview/Преглед

2. Positioning/Позициониране
 - 2.1 Business Opportunity/Бизне възможност
 - 2.2 Problem Statement/Излагане на проблема
 - 2.3 Product Position Statement/Позициониране на продукта

3. Stakeholder and Customer Descriptions/ Клиент и засегнати страни
 - 3.1 Market Demographics/ Демография на пазара
 - 3.2 Stakeholder Summary/ Обобщение на засегнатите страни
 - 3.3 Customer Summary/ Обобщение за клиента
 - 3.4 Customer Environment/ Клиентска среда
 - 3.5 Stakeholder Profiles/ Профили на засегнатите страни
 - 3.6 Customer Profiles /Профили на клиента
 - 3.7 Key Stakeholder or Customer Needs/ Ключови нужди
 - 3.8 Alternatives and Competition / Алтернативи и конкуренция
4. Business Modeling Objectives / Цели на бизнес моделирането

5. Constraints / Ограничения
6. Quality Ranges/ Качество
7. Precedence and Priority/ Предимства и приоритет
8. Other Requirements/ Други изисквания



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- 8.1 Applicable Standards / Приложими стандарти
- 8.2 System Requirements / Изисквания към системата
- 8.3 Performance Requirements / Изисквания за производителност на системата
- 8.4 Environmental Requirements / Изисквания към средата

Виж още RUP документи и тяхната структура

тук: <http://sce.uhcl.edu/helm/rationalunifiedprocess/process/templates.htm>

Съдържание на Business requirements документ (според Wiegers, ‘Software Requirements’)

- 1. Business Requirements/Бизнес изисквания
 - 1.1 Background/Въведение
 - 1.2 Business Opportunity/Бизнес възможност
 - 1.3 Customer or Market Requirements/Клиентски или маркетингови изисквания
 - 1.4 Value Provided to Customers/Стойност за клиентите
 - 1.5 Business Risks/Рискове
- 2. Vision of the Solution/ Визия за решението
 - 2.1 Vision Statement/ Визия
 - 2.2 Major Features/ Основни функционалности
 - 2.3 Assumptions and Dependencies/ Предположения и Зависимости
- 3. Scope and Limitations/ Обхват и ограничения
 - 3.1 Scope of Initial Release/ Обхват на първоначалната версия
 - 3.2 Scope of Subsequent Releases/ Обхват на следващите версии на продукта
 - 3.3 Limitations and Exclusions/Ограничения и Изключения
- 4. Business Context/ Бизнес контекст
 - 4.1 Customer Profiles/ Клиентски профили
 - 4.2 Project Priorities/ Приоритети на проекта

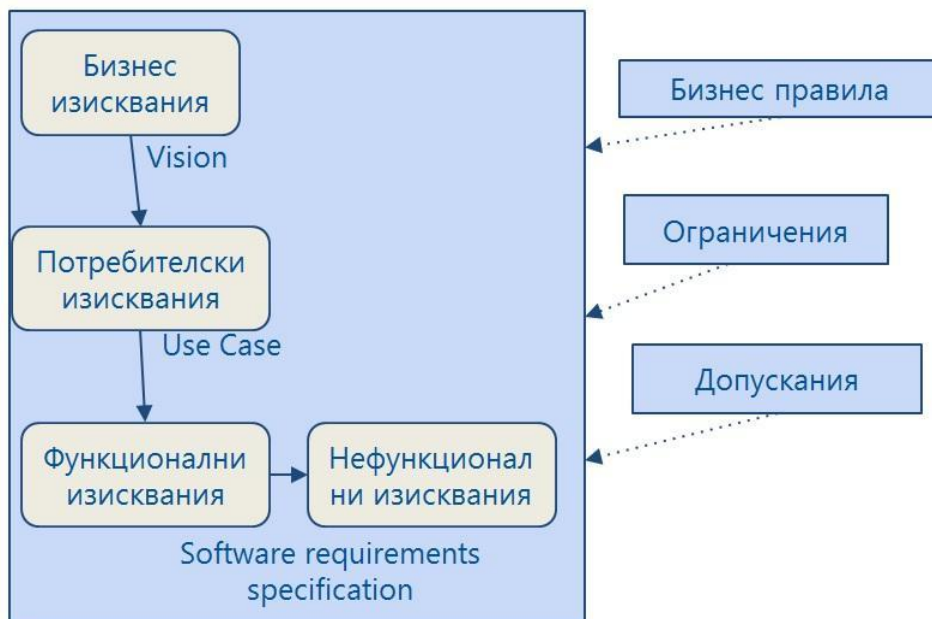
5. Product Success Factors / Фактори за успеха на продукта

10. Модул 7: Процесът по писане на изисквания

- Планиране на работата, свързана с изисквания
- Текущо състояние срещу бъдещо състояние
- Идентифициране на изисквания
- Изключения при изисквания - извън обхвата
- Поетапен или повтарящ (итеративен) подход
- Речник на термините
- BPMN - език за моделиране на бизнес процеси.

10.1. Цел

Въведение в процеса по писане на изисквания



Процесът по писане на изисквания започва с изясняването на бизнес изискванията и нуждите, които те трябва да решат!

Преди за започнем да пишем изисквания се запознаваме с текущото положение в организацията:

- Процесът
- Организационната структура
- Системите

- Текущите проблеми

Анализираме текущата ситуация в няколко насоки:

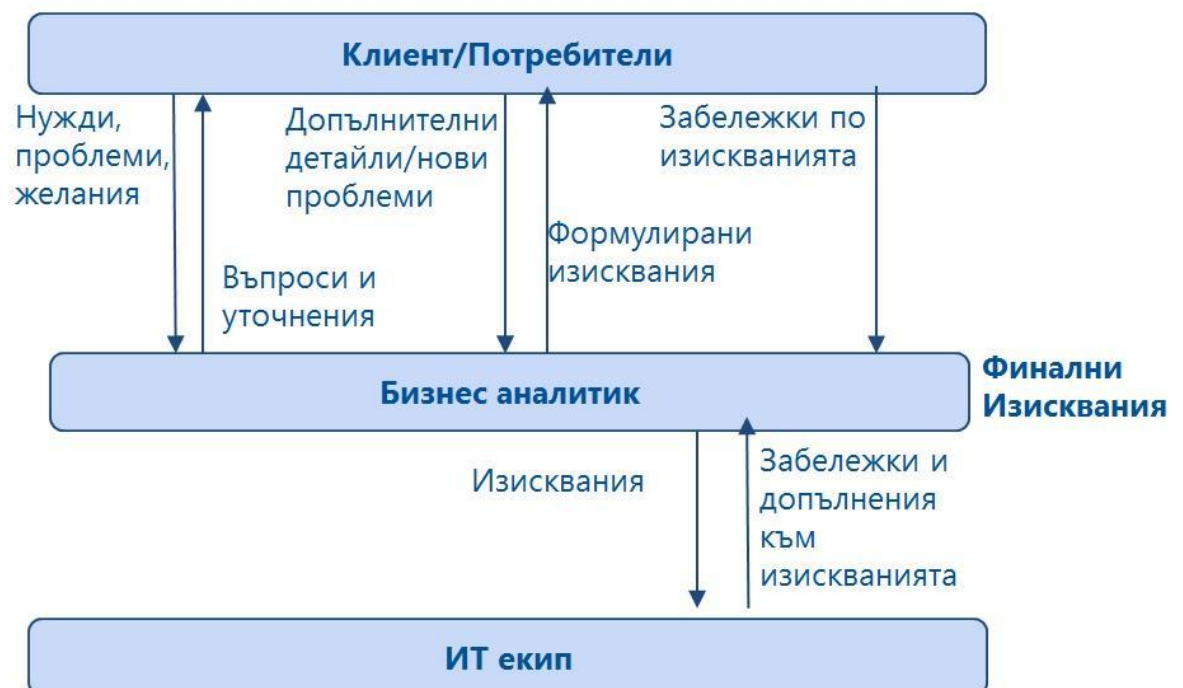
- Какво трябва да **Продължим да правим**
- Какво трябва да **Добавим**
- Какво трябва да **Променим**
- Какво трябва да **Премахнем**

На база откритото дефинираме изисквания

Съвети:

- Не бързайте да измисляте решения - често проблемът стои по-надълбоко.
- Не допускайте каква е бизнес нуждата - проверете разбиранията си

10.2. Процесът по писане на изисквания е итеративен!



Писането и изясняването на изисквания става итеративно, защото много от въпросите възникват докато Бизнес анализатора дефинира изискванията и/или при прегледа на изискванията се правят допълнения.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Както Клиентът, така и ИТ експертите участват в процеса по писане на изисквания. Това гарантира, че изискванията са реализирани и изпълняват бизнес нуждата, заради която са възникнали.

10.3. Планиране на работата, свързана с изисквания

- Списък с документи, които планираме да изготвим по време на проекта
- График на писане на изисквания
- Планираме време за:
 - извличане на изисквания
 - писане на документа
 - период за преглед и одобрение на изискванията

Често с RACI¹ (Responsible (служител, който ще свърши работата), Accountable (служител, който е отговорен работата да се свърши), Consulted (служител, който ще консултира тези, които ще извършват работата), Informed (служител, който трябва да бъде информиран по въпроси от развитието на проекта)) матрица се представя разпределението на отговорностите в екипа.

Например за задачата по изготвяне на документа с изисквания:

- Бизнес анализаторът е Responsible (служител, който ще свърши работата) да напише изискванията,
- Project Manager-ът е Accountable (служител, който е отговорен работата да се свърши) и Informed (служител, който трябва да бъде информиран по въпроси от развитието на проекта)
- ИТ експертите и Клиента/Потребителите на системата са Consulted (служител, който ще консултира тези, които ще извършват работата)
- Период, в който източниците на изисквания са ни необходими

Бизнес анализаторът трябва да изгради реалистични очаквания сред хората, с които ще работи по време на писането на изискванията. Обикновено най-активната

¹ Чете се РАЦИ



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

комуникация е в периода на извличане на изискванията и по време на прегледа на изискванията, т.е. докато изискванията не бъдат завършени и одобрени

- По какъв начин ще общуваме с участниците в проекта – workshop (семинар, работна среща), интервюта, само по e-mail, колко често

10.4. Речник на термините

Изготвя се при писането на изисквания. Уеднаквява разбирането

- Развива се още в начална фаза на проекта
- Осигурява:
 - Еднакво разбиране
 - Изграждане на общ речник на термините
- Включва:
 - Бизнес термини
 - Абревиатури
 - Синоними
 - Дефиниции

10.5. BPMN - език за моделиране на бизнес процеси

Представянето на изискванията чрез модели улеснява разбирането. Business Process Modeling Notation (BPMN) е стандартна нотация, която разработчиците на приложения за моделиране са се съгласили да използват, с цел улеснение за крайния потребител.

BPMN е създаден през 2004 от Business Process Management Initiative, в момента се поддържа и развива от Object Management Group.

BPMN обхваща моделиране на процеси. Цели да улесни разбирането между Бизнеса/организацията и ИТ експертите. BPMN процеси биха могли да се автоматизират чрез BPMN двигатели (BPMN engines)

Елементи

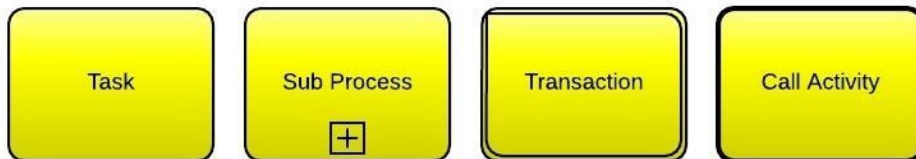
- **Видове събития (Events)**



- **Start/Начало**
- **Intermediate/Междинно - междинно събитие може да бъде**
- **End/Край**

Събитие за начало, междинно събитие или събитие за край може да бъде - получаване на съобщение, грешка, включване на таймер.

- **Дейности (Activity)**



- **Task/Задача**
- **Sub Process/ Под процес**
- **Transaction/ Трансакция**
- **Call Activity/Дейност, която стартира друга дейност**

Според това какъв процес моделираме, BPMN предлага възможност да използваме различни видове дейности:

стъпките да са отделни неделими задачи,

стъпките да са подпроцеси (за да капсулираме сложността),

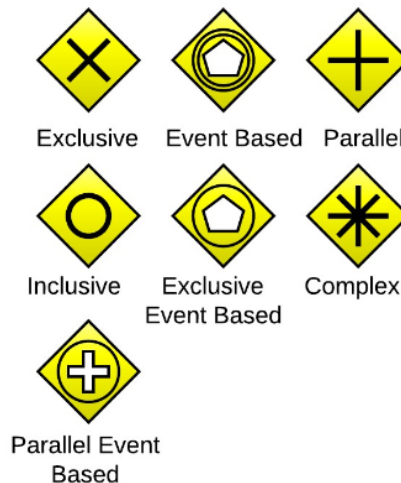
трансакции

стъпките да са трансакции

стъпките да са референции към други дейности (за модулиране и повторна употреба)

- **Решение (Gateway)**

- **Exclusive (Изключващо)** - използва се за действия, които се изключват взаимно, т.е. избира се едното или другото. Пример: ще се движим по море или по въздух
- **Inclusive (Включващо)** - използва се за действия, които не се самоизключват, т.е. изпълняват се и двете. Пример: попълвам въпросник и за продукт 1 и за продукт 2
- **Parallel (Паралелно)** - използва се за представяне на едновременни действия. Например - отдел 1 и отдел 2 преглеждат документите паралелно
- **Complex (Комплексно)** - използва се, за да опрости ситуациите, в които е необходимо да се приложат няколко решения. Complex се описва с повече текст
- **(Parallel/Exclusive) Event Based (Зависещо от събитие)** - решенията зависят от дадено събитие



- **Връзки (Connections)**

- **Sequence Flow/ Последователност**

Моделира посоката на отделните стъпки. Използва се за представяне на комуникацията между обекти от един вид, например отделите в една организация

- **Message Flow /Предаване на съобщение**

Използва се за представяне на съобщения между различни обекти - например организация и клиент

- **Association/ Връзка**

Използва се например за представяне на връзка към Data обекти, т.е. в случаи в които не е необходимо представяне на посока



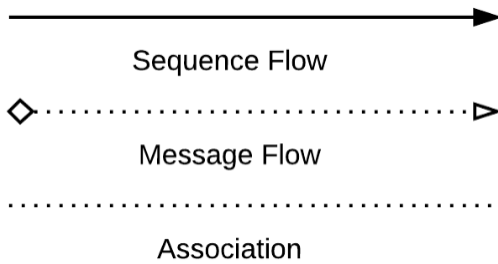
Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората



За повече информация:

<http://www.bpmn.org/>

10.6. Упражнение:

Текста по-долу представя текущата ситуация в банка VA Bank и работата, която се очаква бизнес аналитиците да свършат.

Прегледайте текста и подгответе план за писане на изисквания (т.е. в какъв срок, с кои засегнати страни, по какъв начин бихте извлекли изискванията, какви документи трябва да бъдат изготвени).

Запознаване с текущата ситуация

През изминалото 6 месеци, VA Bank отвори 5 нови клона и се наложи спешно да назначи 40 нови служители.

Банката среща вътрешно-организационни проблеми, ефекта от които е оплаквания от клиенти, главно заради бавното обслужване в офисите.

Политиките за сигурност и контрол в банката изискват различни нива на достъп в основната банкова система. Често обслужването на клиента изисква участието на две или три роли, които да одобряват отделни аспекти от заявената от клиента услуга.

Потока на документи вътре в структурите на банката и извън тях, не е автоматизиран. Комуникацията с външни лица не се проследява, което води до пропускане на срокове и допълнителни разходи.



Липсва средство за разпределяне на задачи към служителите.

Липсва отчетност и прозрачност за мениджмънта на банката за това как протичат процесите и как се изпълняват задачите.

Текущата системата не позволява на клиентите сами да инициират добавянето на услуги, те трябва да посетят офис. Банката отчита пропуснати възможности от този факт.

Мениджмънта е твърдо решен да намери решение на текущите проблеми и затова нае екип от бизнес аналитици, които да се заемат с въпроса.

Георги Петров (отдел 'Иновации'), пое ролята на Клиент в тази инициатива.

Изготвяне на план за писане на изисквания

Имате поща!

От: Георги Петров

До: Бизнес анализ екип

Относно: Бизнес анализ - банка BA Bank

Здравейте,

гордея се, че толкова силен екип ще съдейства в решаването на текущите проблеми на банката.

Жизнено важно е като начало да се съсредоточим в проблемите и възможните им решения.

Вярвам, че срок от две седмици е напълно достатъчен за тази цел.

На ваше разположение за въпроси са колегите Ана Велкова (отдел 'Клонова мрежа') и Димитър Станчев (отдел 'ИТ'). Може да се свържете с тях по мейл или на място в централния офис.

Като резултат от анализа очаквам да получа vision документ (може би вече сте запознати, че използваме RUP методология).



Въпросния документ трябва да бъде представен пред мениджърския съвет след две седмици. Тъй като нося отговорността за успеха на проекта, държа да преглеждам резултатите от анализа периодично!

Желая ви успех,

Г. Петров
отдел 'Иновации'

Отговор:

Очаква се екипа от бизнес аналитици да изготви vision документ за две седмици.

Лица за контакт са: Ана Велкова (отдел 'Клонова мрежа') и Димитър Станчев (отдел 'ИТ')

Начин за комуникация: предвид краткото време, най-подходящо е да се планират срещи за бързо изясняване на визията

Комуникация: през цялото време информираме за развитието Георги Петров, той е засегната страна с голямо влияние върху задачата

План: първата седмица от срока - запознаваме се с визията и изготвяме първа версия на vision документа. Цялата втора седмица планираме за преглед и отразяване на промени по документа.

11. Модул 8: Enterprise Analysis (Анализ на предприятието/организацията)

- Извършване на анализ на ниво Enterprise - какво се е променило, кой е засегнат

11.1. Цел

Запознаване с целите и дейностите в Enterprise анализа

Enterprise analysis означава да анализираме организацията като едно цяло. Какви проблеми, нужди, възможности има пред организацията. Къде сме сега и къде искаме да стигнем като отговор на конкурентно предимство например.

Enterprise analysis включва активностите свързани с:

- Дефиниране на бизнес проблем



- Root-Cause analysis - техника за анализ на проблем в дълбочина и цялост
- Document analysis - например да се запознаем с нова регулация и да анализираме дали изисква промени в организацията
- Идентифициране на бизнес възможност
 - Benchmarking (Оценка чрез сравнение) - сравняваме се с конкурентите - има ли бизнес възможност пред нас, с която може да се отличим от конкурентите.

Например ако конкурентите ни нямат мобилно приложение дали това е възможност за нас да се отличим сред тях?

- Идентифициране на възможни решения за решаване на проблема/възползване от възможността
 - SWOT (Strengths, Weaknesses, Opportunities, Threads) analysis - за всяка алтернатива оценяваме силни и слаби страни, възможности, които ни разкрива, заплахи. Дава възможност за сравнение на алтернативни решения с цел избор на най-подходящото.
- Изготвяне на Бизнес казус - измерване и оценка на прогнозните ползи и разходите от имплементирането на избраното решение

12. Модул 9: Дефиниране на бизнес проблем

- Проблемното изявление (Business Requirements Problem Statement)
- Засегнати страни

12.1. Цел

Запознаване и прилагане на стандартен подход за дефиниране на бизнес проблем

Често неуспехите в проектите са заради грешно или непълно формулиран проблем, резултатът е частично решение на проблема. Следният начин за формулиране на проблема съдържа: дефиниция на самия проблем, засегнатите страни, ефекта от проблема и предложение за успешното му решение.

Дефинирането на бизнес проблема, осигурява еднакво разбиране за това, какво ще постигнем в края на проекта, какво ще решим.

Проблемът	(описваме проблема)
------------------	---------------------

засяга	(изброяваме хората, които са засегнати)
като последствията са	(изброяваме последствията от проблема)
и успешно решение би било	(изброяваме основните бизнес ползи от избраното решение)

Пример:

Проблемът	в закъснените доставки
засяга	всички клиенти, които поръчват доставка до вкъщи
като последствията са	недоволни клиенти и отменени поръчки
и успешно решение би било	оптимизиране на процеса по изпълнение на клиентски поръчки и доставка

● Пример - VA Bank:

Проблемът	<ul style="list-style-type: none"> ● Бавното обслужване на клиенти ● Невъзможност за проследяване на задачи, процеси
засяга	служителите на банката и нейните клиенти
като последствията са	<ul style="list-style-type: none"> ● недоволни клиенти, ● липса на отчетност, ● пропуснати възможности
и успешно решение би било	<ul style="list-style-type: none"> ● единна система за документооборот и разпределяне на задачи ● —

● Примерен вариант на Vision документ се съдържа в Приложението

12.2. Защо е възникнал проблема - Root Cause Analysis (Анализ на причинно-следствените връзки)

Root-Cause Analysis е техника за анализ на ефекта от даден проблем. Според съответния случай се избират подходящите категории, като например:

- **4Ps** - Process/Процеси, People/Хора, Policies/Политики, Plant/Technology/Технологии
- **6Ms** - Machines/Машини, Methods/ Методи, Measurement/Измерване, Materials/Материали, Manpower/ Работна ръка, Mother nature/Природа
- **Process steps (Процесни стъпки)** - Determine customers/Кои са клиентите, Advertise product/Реклама, Purchase/Закупуване, Sell/Продажба, Ship/Доставка, Upgrade/Развитие на продукта

Категориите служат, за да ни насочат за различните аспекти за анализ на проблема. Бизнес анализаторът избира коя категория е най-подходяща.

Резултатът от Root-Cause анализа са нещата, които трябва да променим, за да решим проблема.

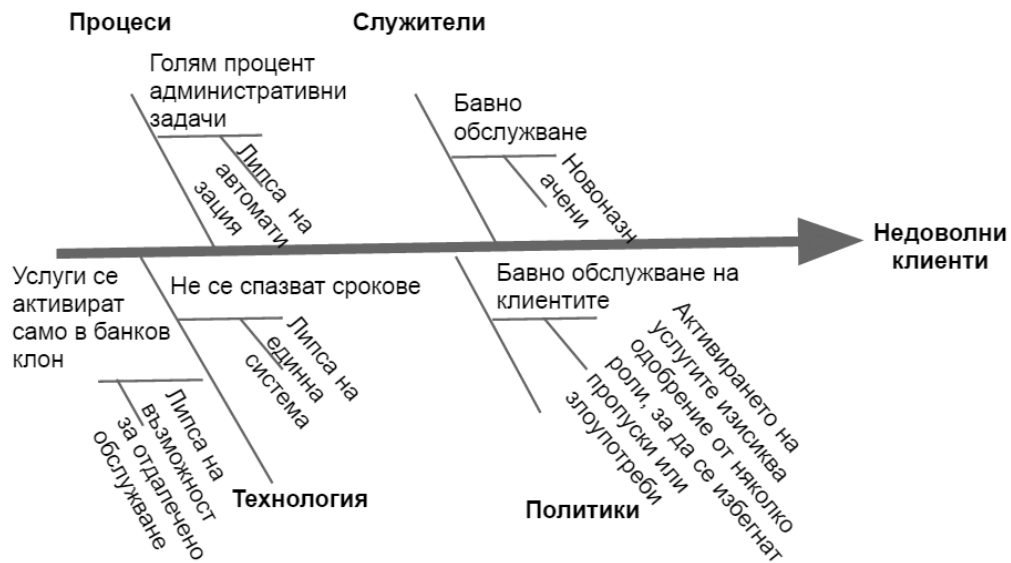


4Ps – Общ вид



4Ps пример

Root Cause Analysis



4Ps пример

13. Модул 10: Дефиниране на обхвата

- Дефиниране какво е в проекта и какво е извън проекта



13.1. Цел

Защо е важно да се дефинира обхвата на проекта? Техники за приоритизиране на изисквания.

Ясното дефиниране на обхвата е най-доброто оръжие срещу неконтролирани промени (feature-creep) по време на проекта.

Какво е в обхвата на проекта:

- Списък с основните групи функционалности
- Приоритети на функционалностите

Често Клиентът прави компромис между нуждите и реалистичните очаквания.

Добра практика е да се изброи и какво остава извън обхвата на проекта - списък с функционалностите, които остават извън обхвата.

Често клиентът се затруднява да приоритизира, ето няколко техники, които улесняват приоритизирането. Бизнес анализаторът преценява коя е подходяща за съответния случай:

1. Гласуване -

заинтересованите страни /клиента и потребителите/ гласуват за това коя функционалност е по-важна

2. MoScOw

За всяко изискване се избира подходящия приоритет - Must, Should, Could, Won't, където:

- [M]ust - изискванията, които трябва да бъдат реализирани, за да считаме проекта за успешен
- [S]hould - изисквания, които трябва да бъдат включени в проекта, когато е възможно (високо приоритетни изисквания, които могат да бъдат покрити по други начини)
- [C]ould - изисквания, които могат да бъдат считани за желани когато времето и ресурсите позволяват.

- [W]on't - изисквания, които отпадат от този релийз, но може да бъдат взети предвид в бъдещето

3. Приоритизиране по клъстери (Prioritization clustering approach)

Техниката дефинира кои изисквания са необходими, за да се изпълни даден случай на употреба. Изискванията, които са необходими за изпълнение на по-важните случаи на употреба са най-приоритетни.

Техниката ни дава поглед и за това как минимално усилие за посрещане на дадено изискване може да донесе значителна допълнителна полза - пример: ако разработим изискване 3, това ще достави Бизнес сценарий 3 (Use Case 3) за клиента ни.

	Бизнес сценарий 1 (Use Case 1)	Бизнес сценарий 2 (Use Case 2)	Бизнес сценарий 3 (Use Case 3)
Изискване 1	Y	N	N
Изискване 2	Y	Y	Y
Изискване 3	N	N	Y
Изискване 4	Y	N	N
Изискване 5	N	N	N

4. Сравнителна класация (Comparative ranking)

По-лесен начин за определяне на приоритети за няколко изисквания е като ги сравняваме едно с друго. Изискването, което сме избирали повече пъти като по-важно е най-приоритетното.

Общо Класация

Req. 1							
Req. 2	A						
Req. 3	B	C					
Req. 4	D	E	F				
	Req. 1	Req. 2	Req. 3	Req. 4			

1. Нека Изискване 1 (Req.1) е по-важно от Изискване 2 (Req.2). Попълваме в клетката A – Req.1
2. Нека Изискване 1 (Req.1) е по-важно от Изискване 3 (Req.3). Попълваме в клетката B – Req.1
3. Нека Изискване 4 (Req. 4) е по-важно от Изискване 1 (Req.1). Попълваме в клетката D - Req4 и т.н.
4. Колона Общо, показва, че най-много сме избрали като по-важно Req.4, за това и то No1 в Класацията

Общо Класация

Req. 1					2	2	Req. 1
Req. 2	Req. 1				1	3	Req. 2
Req. 3	Req. 1	Req. 2			0	0	Req. 3
Req. 4	Req. 4	Req. 4	Req. 4		3	1	Req. 4
	Req. 1	Req. 2	Req. 3	Req. 4			

14. Модул 11: Визуализиране на обхвата

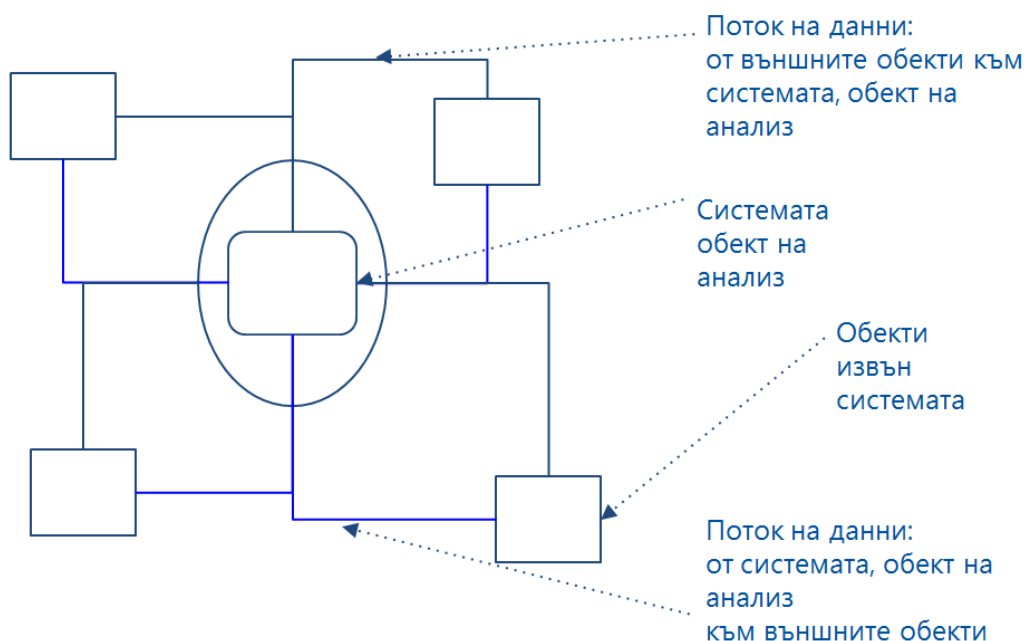
- Визуализиране на обхват Диаграмата за визуализиране на обхвата на проекта (Context diagram)

14.1. Цел

Запознаване с и прилагане на Context diagram (Контекстна диаграма) с цел визуализиране на обхват.

Context diagram (Контекстна диаграма) представя визуално какво е в и извън обхвата на системата обект на анализ. Често в проектите се забравя, че системата взаимодейства с външни обекти, което води до доработки и допълнителни разходи.

Елементи:

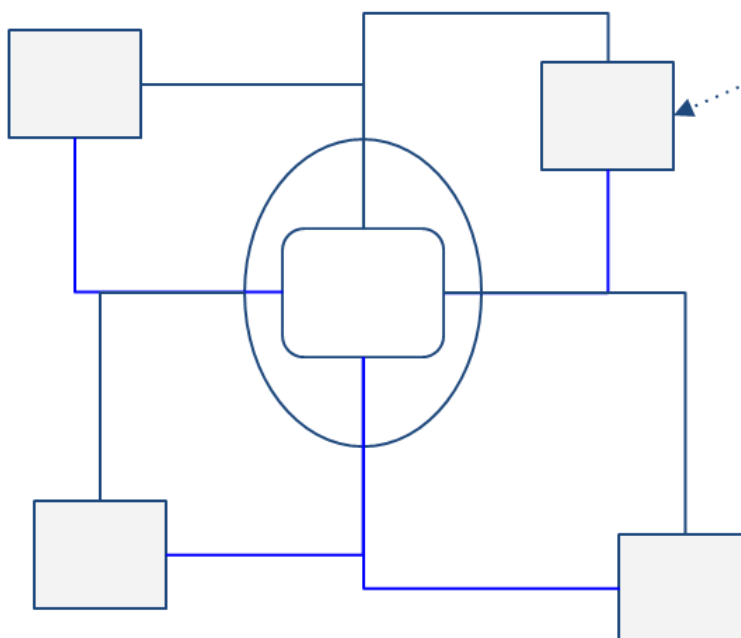


Пример:

Ако системата ни обект на анализ е Банкомат, то външните обекти, с които тя си взаимодейства са Клиент и Банкова сметка. В обхвата на проекта се включва: системата Банкомат и данните, които се прехвърлят с външните обекти (Карта, Пин, Сума, Изтеглени средства, Бележка, Наличност).



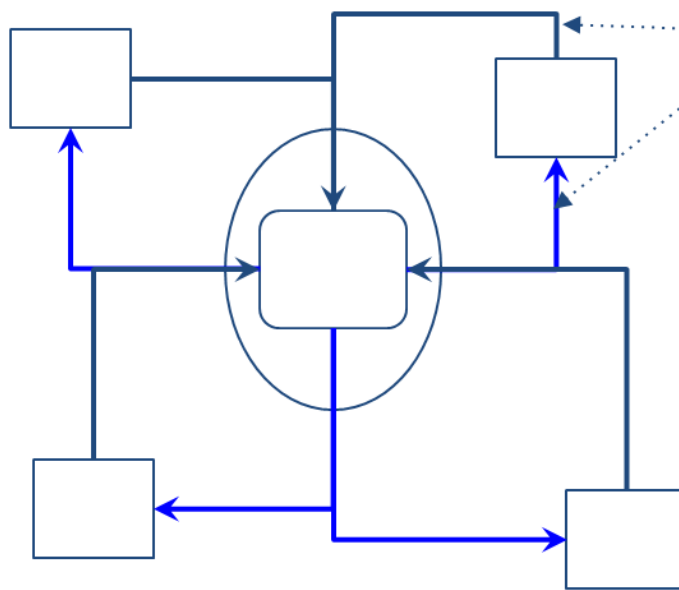
Външни обекти в Context Diagram:



Външен обект

- Външен източник или дестинация, която е извън системата, която се анализира
- роля, отдел, система, база, сензор
- източник и/или производител на данни
- системата, обект на анализ има отговорност към този външен обект

Поток на данни в Context Diagram



Поток на данни

- Представя информацията, която се движи между системата и външните обекти
- Не представя информация кога и какво тригерира потока на данни
- посоката се представя със стрелка с един връх
→

Примерна контекст диаграма на проекта за разработка и внедряване на система за електронно управление на документи:

Context diagram - Система за електронно управление на документи





Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

15. Модул 12: Бизнес казус

- Въведение в писането на бизнес казус
- Техники за оценяване на работата на екипа (Estimation techniques)

15.1. Цел

Запознаване с елементите на бизнес казуса. Изготвяне на бизнес казус.

Бизнес казусът:

- Дефинира резултата, който един проект носи
- Представя се пред мениджмънта
- Служи като основа за информирано решение, какви действия да предприеме компанията

Възможни елементи на Бизнес казуса:

- Преглед/Обобщение
- Текуща ситуация
- Дефиниция на проблема
- Цели на проекта
- Обхват на проекта
- Засегнати страни (Stakeholders)
- **Възможни решения за разработка/имплементация**
- **Бизнес ползи**
- **Разходи**
- План за разработка/имплементация
- Анализ на въздействието
- Риск план

Възможните решения, Оценка на бизнес ползите и Разходите са трите най-важни елемента, на които по-подробно ще се спрем.

15.2. Възможните решения за разработка/имплементация

Обикновено алтернативните решения попадат в някоя от следните групи:

- Оптималното решение според най-новите технологии
- Нестандартното решение, което е евтино, но покрива голяма част от текущите проблеми
- Незабавното решение, което адресира най-критичните проблеми
- Решението 'остава както е сега' (Stay as it is) - т.е. избираме да не правим нищо

15.3. Как оценяваме разходите за всяко от възможните решения?

Най-важното е да има предвид всички разходи за проекта. Те могат да бъдат:

- Разходите за изпълнението на проекта
 - разходи за човешки труд (планираното време работа)
 - разходи за ресурси (за закупуване на техника)
 - разходи за служителите (обучения)
 - разходи за доставчици (устройства, услуги)
 - непланирани разходи - резерв

- Разходите след като проекта се изпълни
 - поддръжка
 - годишни такси за лицензи
 - цена за закупуване на нови лицензи

15.4. Как оценяваме разходите за човешки труд (планираното време работа)? Техники за оценяване (Estimation techniques)

Обикновено времето за разработка в проектите е най-съществен разход. Следните техники предлагат възможни начини за оценка на разходите:

- Исторически анализ
Обръщаме се към исторически данни - колко е струвал подобен вече приключил проект?
- Експертна оценка
Допитваме се до експерти в областта, за да вземем оценка.
- Delphi оценка
Допитваме се до експерти областта независимо, без да знаем кой е дал оценката
- Оптимистична, реалистична, песимистична оценка
Взимаме средна стойност измежду трите оценки
Често изброените техники се комбинират!

Как оценяваме бизнес ползите от дадено решение? Примери за бизнес ползи, които трябва да измерим



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

За да направим бизнес казус (оценка на ползата от дадено решение), ние следва да дадем измерители както на разходите, така и на възможните ползи. Ползите от имплементирането на дадено решение могат да бъдат различни и не винаги е толкова лесно да бъдат измерени. Ето и някои примери:

- Оптимизация/спестяване на разходи - ползата е сумата на спестените разходи
- Избягване на загуби/глоби - ползата е сумата на глобата, която ние спестяваме
- Печалби - нови клиенти, обем продажби - ползата е обема продажби, които прогнозираме, че ще имаме
- Задържане на клиенти - ползата са сегашните печалби, които ще запазим и занапред
- Повишаване на клиентската удовлетвореност - задържане на клиентите
- Решаване на клиентски проблем - задържане на клиентите
- Позициониране на марка/продукт - задържане на клиентите и нови печалби

За да направим бизнес казус (оценка на ползата от дадено решение), ние следва да дадем измерители както на разходите, така и на възможните ползи. Ползите от имплементирането на дадено решение могат да бъдат различни и не винаги е толкова лесно да бъдат измерени.

Варианти разбира се има - ето и някои примери:

Как сравняваме ползите, спрямо разходите от разработка/имплементация на дадено решение? Оценка

Net Present Value и Return Of Investments са много известни методи за оценка на ползи-разходи в проектите.

Net Present Value парите се обезценяват във времето. Този метод дава възможност да измерим стойността на парите, които очакваме да спечелим в бъдеще, днес.

$$pv = fv / (1+d)^n$$

n = number of years in the future /брой години

d = discount rate/ дисконтов процент

Return of Investments (ROI) (Възвръщаемост на инвестициите) - Този метод ни дава възможност да измерим възвръщаемостта

$$ROI = b / c$$

$$\text{Annual ROI} = ROI / n$$

b = net present value of benefits, i.e. total savings/сума на ползите

c = net present value of costs, i.e. total costs/сума на разходите

n = deployed life of solution in years/ години, през които решението се имплементира

Примерен бизнес казус

Ако времето за разработка/имплементация на система за управление на документи е 5 месеца.

Банката ще спести 400 000 лв. А разходите са:

- Разработка в ИТ – 200 000 лв
- Резерв – 20 000 лв
- Разходи за хардуер – 40 000 лв
- Не се очакват разходи за лицензи

Да се пресметне ROI и Net Present Value (d=4%) за 1 година

Отговори:

1. Бизнес Ползи

С постигане на основната цел на проекта, а именно внедряването на тази система, ще се постигнат следните бизнес ползи:

1. Облекчаване на административната работа на служителите.
2. Повишаване капацитета на организацията като цяло и отчетността на работните процеси в структурата на BA Bank
3. Повишаване на ефективността и прозрачността на работата на всички звена на банката чрез въвеждане на електронно управление и електронен обмен на данни. Банката ще спести 400 000 лв.

- **Net Present Value**

$$pv = fv / (1+d)^n$$

n = number of years in the future

d = discount rate = 4 %

$$pv = 400\,000 / (1+0.04)$$

$$pv = 384\,616,4$$

- **Return of Investments (ROI)**

$$ROI = b / c$$

b = net present value of benefits, i.e. total savings

n = deployed life of solution in years

c = net present value of costs, i.e. total costs

$$pv \text{ разходи} = 260\,000 / (1+0.04)$$



$$pv = 250\,000$$

$$ROI = 384\,616,4 / 250\,000,$$

$$ROI = 1,54 * 100 = 154\%$$

16. Модул 13: Потребителски изисквания и изисквания на заинтересованите страни

- Кой са потребителите на системата?
- Характеризиране на потребителите

16.1. Цел

Техники за анализ на засегнатите страни и потребителите

16.2. Кой са засегнатите страни?

Това са всички в проекта:

- Клиент
- Потребители
- Екипа в проекта
- Организационни структури
- Доставчици на услуги

Кой са потребителите на системата?

Users vs. Stakeholders: Не всички засегнати страни са потребители на системата!

За да улесним анализа, идентифицираме кои са потребителските групи, това включва:

1. Дефиниране на потребителски атрибути
 - Честота на ползване на системата
 - Географска локация
 - Права
 - Експертиза
 - Умения
2. Какви услуги на системата трябва да могат потребителите да достъпват, за да изпълняват задълженията си.

Кой са заинтересованите страни? Няясно ли сме - кой са те, каква е тяхната роля, какви очаквания имат, кои от тях ще бъдат потребители на системата. Тази информация може да бъде записана в таблица от вида:

Име	Роля	Очаквания	Атрибути
-----	------	-----------	----------

--	--	--	--

16.3. Анализ на заинтересованите страни

Тази техника разпределя засегнатите страни, според тяхното влияние и засегнатост от проекта. Техниката ни дава насоки как да се държим с отделните групи, според квадранта, в който те попадат.

- Да се държат доволни - това обикновено е Клиента
- Да се управляват интензивно - например екипа в проекта
- Да се наблюдават - слабо засегнатите отдели в организацията
- Да се държат информирани - например потребителите на системата



16.4. Пример с попълнени потребителските атрибути за всяка от ролите в BA Bank

Потребителска роля	Услуги	Потребителски атрибути
Касиер	Обработка на плащания и вземания, работа с пари в	Няма опит със система

Потребителска роля	Услуги	Потребителски атрибути
	брой и чекове	<p>Ползване на системата за няколко часа в края на работния ден</p> <p>Ползване на системата само от банковия клон</p> <p>Експертиза - слаба- използва основната банкова система</p> <p>2-ма касиери</p>
Фронт офис	<p>Пренасочване на клиентски запитвания,</p> <p>Използване на компютъризирана система за обновяване на счетоводни данни,</p> <p>Документоборот,</p> <p>Продажба на банковите финансови продукти и услуги на клиент</p>	<p>Работи със системата през целия работен ден</p> <p>Ползване на системата само от банковия клон</p> <p>Експертиза - висока, използва основната банкова система, участва в почти всички активности, засягащи клиенти или външни институции</p> <p>4-ма служители на тази позиция</p>
Кредитен консултант	<p>Обслужване на клиентски запитвания за кредити,</p> <p>Предлагане на индивидуални за клиента кредитни условия,</p> <p>Използване на компютъризирана система за счетоводни данни</p>	<p>Ползва системата няколко пъти на ден</p> <p>Ползване на системата само от банковия клон</p> <p>Експертиза - висока, използва основната банкова система</p> <p>2-ма кредитни консултанти</p>
Отговорен счетоводител	<p>Надзор и контрол върху документоборота,</p> <p>Използване на компютъризирана система за</p>	<p>Ползва системата няколко пъти на ден</p> <p>Ползва основни контролни и надзорни функции - с по-високи</p>

Потребителска роля	Услуги	Потребителски атрибути
	счетоводни данни	права Ползване на системата само от банковия клон Експертиза - висока, използва основната банкова система 1 служител на тази роля
Охрана	Следи за реда по време на работния процес	Не използва система

17. Модул 14: Техники за извличане на потребителски изисквания

- Въпросници
- Интервюта
- Подходящи въпроси за интервю
- Brainstorming
- Ролеви игри
- Workshop
- Прототипи

17.1. Цел

Запознаване със и прилагане на техники за извличане на потребителски изисквания

Техники за извличане на изисквания, които ще разгледаме по-подробно:

- Документален анализ
- Въпросници
- Наблюдение
- Интервюта



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Мозъчна атака /Brainstorming/
- Ролеви игри
- работна среща
- Прототипи

17.2. Документален анализ

Използва се за събиране на изисквания чрез преглед на наличната документация за съществуващи подобни решения.

17.3. Въпросници

Въпросниците са средство за едностранна комуникация

- Предимства:
 - Може да достигнат до голям кръг от хора
 - Резултатите подлежат на статистически анализ
 - Не изисква пряк контакт
- Недостатъци:
 - Изисква умение за правилното конструиране на въпросите
 - Липсва свободата за допълнителни или проучвателни въпроси
 - Не можем да сме сигурни, че можем да се доверим на отговорите

Въпросниците могат да са подходящо средство за извличане на изисквания, ако се комбинират и с интервю/работна среща.

17.4. Наблюдение

Възможно е да усетим, че не можем да се доверим напълно на информацията, която получаваме от заинтересованите страни, или пък да установим, че има много противоречия. В тези случаи е подходящо да приложим техниката Наблюдение.

- Предимства:

Виждаме как точно се случват нещата, не разчитаме на обяснение от страна на потребителите

- Недостатъци:
 - Изисква сериозен ресурс от време



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Наблюдаваният може да върши работата си по-различно от обикновено, заради това, че е наблюдаван
- Може да се появят изключения

17.5. Интервю

Интервюто е една от най-разпространените техники. Обикновено Бизнес анализаторът прави Интервюта и след това продължава с Работни срещи / Workshop/

- Предимства:
 - Дава възможност за промяна посоката на разговора
 - Можем да следим реакциите на интервюирания
 - Можем да използваме мета въпроси, за да повлияем за успеха на интервюто
 - *Задавам ли твърде много въпроси?*
 - *Вие ли сте подходящия човек, който може да отговори на този въпрос?*
 - Можем да ги провеждаме във формална или неформална обстановка (според социалния тип на интервюирания)
- Недостатъци:
 - Може да изисква доста време
 - Трудно може да се реши конфликт по време на интервю
 - Изисква добри умения за слушане
- Подходящи въпроси за интервю - въпроси свободни от контекст, абстрактни, със свободен край
 - За начало на интервю
Знаете ли защо сте тук и какво се опитваме да постигнем?
Хората ще се по-спокойни, когато разберат целта на интервюто
 - Общо за организацията
Какво работи добре днес?
Кои според вас са трите неща, които пречат на работата?



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Мета-въпроси

Задавам ли твърде много въпроси?

Вие ли сте, човека, който може да отговори на този въпрос?

17.6. Brainstorming /Мозъчна атака/

Тази техника се използва обикновено за решаване на отворени въпроси

- Стъпки:
 - Задава се тема
 - Генериране на идеи: всички идеи се записват без да се критикуват
 - Категоризиране на идеи: събраните идеи се категоризират и редуцират от екипа
- Съвети:
 - Няма лоша идея, но може тя да е неподходяща за случая
 - Идеите се редуцират след като е минал етапа на генериране на идеи
 - Brainstorming-а обикновено протича в рамките на 90 минути, по-дълъг период не е продуктивен

17.7. Ролеви игри, Прототипиране

17.8. Ролеви игри

Когато нямаме възможност да говорите с потребителите на системата (клиенти, географски ограничения)

- Участниците получават роля
- Разиграват се различни случки, които биха могли да се случат в реална среда

Съвет: Внимавайте с разпределянето на ролите! Някои роли могат да са обидни за участниците

17.9. Прототипиране

Демонстрация на работно ниво на системата или базови екрани, могат да помогнат да подобрим разбирането на бизнес изискванията и да уеднаквим очакванията.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Прототипите често с цел са черно-бели и несъвършени. Целта е да се изгради разбиране за структура на данните и да се открият изискванията, които са пропуснати до момента.

17.10. Работна група /Workshop/

Когато имаме множество потребители, с конфликтни изисквания

- Структура
 - Малка група представители на потребителските групи (7-10)
 - Обикновено протича като brainstorming
 - Води се от неутрален фасилитатор
 - Има ясна и реалистична цел

- Подготовка
 - Правилните хора
 - Дъска, материали
 - Участниците са ангажирани и отдадени на темата

- По време на работната група/workshop
 - Водят се записки видими от всички
 - Прекъсват се дискусии различни от темата
 - Не се водят паралелни дискусии
 - Разпределят се задачи
 - Спазва се определеното време

- След работната група
 - изпраща се паметна записка/мемо с решенията и задачите

18. Модул 15: Представяне на потребителски изисквания

- Примерни употреби
- Основни и алтернативни сценарии
- Добри практики в писането на use cases

18.1. Цел

Представяне на потребителски изисквания като случаи на употреба /use cases/

18.2. Use Cases /Случаи на употреба/

- Случаите на употреба представят взаимодействието между потребителя и системата разбито (представено) на стъпки. Допълват изискванията, донасяйки контекст
- Последователност от стъпки се нарича сценарий
- Един случай на употреба може да има няколко възможни сценария
- Основен сценарий
- Алтернативни сценарии - един и същ резултат може да се постигнем по различен начин
- Сценарии, в които възникват изключения (грешки)
- Един случай на употреба /use case/ може да има няколко актьора.
- Системите също могат да са актьори
- Един use case може да има системен или бизнес контекст

18.3. Елементи на Use Cases/Случаи на употреба

Use Case Name/ Име на случаи на употреба	Пример: Активиране на услуга
Actor/Актьор	Ролите, които си взаимодействат - хора, системи (Пример: Клиент, Система, Асистент в офис)
Preconditions/Предусловия	Условие, което трябва да се изпълни, за да започне use case-a (Пример: клиента да има право да ползваа услугата)
Main Scenario/Основен сценарий	1 2 3
Alternate Scenario/Алтернативен сценарий	1a (алтернатива или exception (грешка) на стъпка 1) 2a (алтернатива или exception (грешка) на стъпка 2) 3a (алтернатива или exception (грешка) на стъпка 3)
Postconditions/Послусловия	Състоянието, след като се е изпълнил use case (пример: Услугата е активирна успешно)

Добри практики в писането на use cases



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Използвайте сегашно време
- Пишете кратки стъпки
- Един use-case е добър, когато има до 9-10 стъпка
- Случаите на употреба обикновено са до 15, над този брой са трудно четими
- Не включвайте в стъпките описание на системния дизайн
- Внимавайте с употребите на include (включване) и extend (разширение), може да затруднят аудиторията

18.4. Упражнение

Подгответе Use case на база следните записки от интервю със служител за Фронт Офис

,Системата, с която работим е много стара, нищо не сканираме. Отделните документи ги гледаме по бланки.

Ако клиентът е за каса - направо с документа на каса и остава в касовия дневник и клиента получава копие. Слагаме касов печат и подписи. Документите се добавят в Касов дневник.

Ако клиента не е за каса - оставяме документа в счетоводен дневник, който пускаме на края на работния ден.

Независимо какъв е документът/дневникът на другия ден счетоводният дневник се преглежда от колега и после и от Отговорен счетоводител.

На две години Вътрешен отдел проверяват произволни документи и констатираят нарушения и грешки. Накрая изготвят доклад.

Но в повечето случаи грешките се хващат на момента или в рамките на работния ден и си сторнираме сами операциите.

Отговор

Use Case Name	Обслужване на клиент в банков клон
Actors	Фронт-офис, Касиер, Гл. Счетоводител
Preconditions	Клиент посещава банков клон и изисква да ползва банкова услуга
Main Scenario	<ol style="list-style-type: none"> 1. Клиент изисква услуга на каса 2. Фронт офис служителят изготвя необходимите документи и ги подписва и подпечатва 3. Клиентът подписва документите и отива на каса 4. Касиерът изпълнява операциите, спрямо документите на клиента. 5. Фронт офис добавя документите в Касова книга 6. Гл. Счетоводител проверява изготвените документи. 7. Документите са коректно попълнени

	<p>Край</p> <p><i>На всеки 2 години вътрешен одит проверява случайни документи и изготвя доклад с предписания.</i></p>
Alternate Scenario	<p>1а. Изискваната от клиента услуга не е за каса</p> <p>1а.1. Фронт офис подготвя необходимите документи, подписва ги и ги печата</p> <p>1а.2. Клиентът подписва документи</p> <p>1а.3. Фронт офис добавя подписаните копия в Счетоводната книга</p> <p>Продължи от стъпка б.</p> <p>1б. Клиентът се интересува от кредитна услуга.</p> <p>1б.1. Фронт офис прехвърля обслужването на клиента при Кредитен консултант</p> <p>Край</p> <p>7а. Гл. Счетоводител намира грешка в документите</p> <p>7а.1. Фронт офис сторнира операцията и създава нова коректна транзакция</p> <p>Край</p>
Postconditions	Клиентът е обслужен. Банковата услуга е обработена коректно

19. Модул 16: Функционални изисквания

- Правила за писане на качествени функционални изисквания
- Думите, които трябва да се избягват
- SMART Изисквания
- Добри практики за писане на изисквания (неясни думи, синоними, наречия и т.н.)

Цел

Представяне на правила за писане на качествени функционални изисквания

19.1. Граматика – време

Съществуват различни варианти за изписване на изискванията:

Системата ще...



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Системата трябва...

Потребителят ще...

Потребителят трябва...

Системата...

Системата може да...

Потребителят може да...

Препоръка: изберете си един от вариантите и използвайте само него.

SMART изисквания

S (Specific) – описанието на изискванията е точно, ясно и недвусмислено. Може да се интерпретира само по един начин от всички

M (Measurable) – съществуват критерии (тестове), които могат да бъдат приложени към изискването, за да се провери дали е изпълнено

A (Achievable) – изпълнимо е в рамките на проекта

R (Realistic) – изискването съответства на целите на проекта

T (Traceable) – изискването може да се проследи за собственик, да се свърже със система, модул, тест и т.н.

Препоръки за документиране на изисквания

Обичаен и опростен (изчистен) език

Прецизно използване на едни и същи термини, акроними, съкращения и бизнес език

Колкото може повече графики, особено за представяне на данни

Пишем „консистентно“

Грижа за качеството на изискванията

Грижа за качеството на документа

Изискванията не са място за демонстрация на креативност и оригиналност.

Написани ли са качествено тези изисквания?

Системата ще предостави в реално време информация за притежателя на кредитната карта.

Системата ще предоставя в реално време информация за височината на самолета на навигационната система.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Системата ще показва фактура, за да позволи на потребителите да въвеждат информация в нея. След това ще калкулира всички суми за тях.

Системата ще предоставя достъп до служителския/персонален файл, на одобрени мениджъри.

Системата няма да позволява на неодобрени потребители, да променят кредитните записи на клиентите.

Системата ще е високо надеждна за всички спешни телефонни оператори.

Системата ще предоставя удобен интерфейс за поръчки онлайн.

Системата ще предоставя падащо меню на потребителите, за да определят правилния избор с мишката на екрана.

Граматика – двусмислени думи и изрази

- Системата ще предостави в реално време информация за притежателя на кредитната карта.

- Означава, че:

- Информацията ще се показва в реално време

ИЛИ

- Информацията за притежателя на кредитната карта ще е актуална

Какво означава „реално време“?

Препоръка: внимавайте за думи, като удобен за потребителя (user-friendly), автоматично (automatic), актуализиран (up-to-date).

Граматика – местоимения

- Системата ще визуализира фактура, за да позволи на потребителите да въвеждат информация в нея. След това ще калкулира всички суми за тях.

- Нея – системата или фактурата

- Суми – кои са тези суми

- Тях – за фактурата, потребителите...

- Препоръка: внимавайте с местоименията.

19.2. Граматика – синоними

- Плащания, чек, пари, компенсации, трансакции, възнаграждение, кеш, валута, клиенти (customer, client, subscriber)

- Плащанията трябва да пристигат в компанията, в счетоводните сметки

- Препоръка: опишете всички синоними в речник и използвайте един от тях.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

19.3. Граматика – отрицания

- Системата няма да позволява неодобриени потребители да променят кредитните записи на клиентите.
- Системата никога няма да позволява на потребители, които вече не са служители, да получават нови IDs.
- Системата няма да показва никакви непубликувани номера на телефони или номера, които не включват правилния код.
- Препоръка: избягвайте двойно или тройно отрицание.

Граматика – наречия и прилагателни

- Ефективен и ефикасен – сравнено с какво?
- Висока степен на надеждност – никога не спира работа? Или твърде рядко?
- Бързо – колко бързо е бързо?
- Понякога – колко често?
- Препоръка: използвайте повече количествени измерители.

Пример. Дадени са следните функционални изисквания. Те ще бъдат преработени според правилата за писание на качествени функционални изисквания.

1. Системата трябва да може да управлява потока от документи (входящи, изходящи и вътрешни) и оперативна информация (отчети, заявки, становища, справки, писма, резолюции, заповеди и др.)
2. Системата ще съблюдава движение на документи вътре в организацията във връзка с изпълняваните дейности
3. Приложението ще съдържа задачите и контрола по тяхното изпълнение
4. Трябва да има и данни за потребителите на системата и информация за тях
5. Също за всички външни лица и организации, с които потребителите кореспондират
6. Във връзка с документооборота, Фронт офис служителите трябва да създават инициращ документ/преписка
7. Други документи са: документ към преписка (подчинен документ), кредитно досие и други.
8. За преписката, досието и другите документи, Фронт офисите ще:
 - a. ползват образци от базата данни на системата
 - b. редактират, регистрират, адресират документи
9. Добре е също да може да задаваме кореспондент, заявка за подпис, съгласуване или одобрение
10. Да има подписване и съгласуване
11. Преместване на документи
12. Поставяне на резолюции – срокове, права, нотификации, адресиране, статус
13. Вероятно най-често Главният Счетоводител и Директорът ще възлагат задачи



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

14. Те трябва да могат да отбелязват изпълнението
15. Също и да контролират изпълнението на задачи и разпореждания по документи

Преработени изисквания:

Системата трябва да може да поддържа:

Видове данни

1. Потока от документи (входящи, изходящи и вътрешни) и оперативна информация (отчети, заявки, становища, справки, писма, резолюции, заповеди и др.)
2. Движение на документи вътре в организацията във връзка с изпълняваните дейности
3. Задачите и контрола по тяхното изпълнение
4. Данни за потребителите на системата съобразно тяхната роля в структурата на организацията
5. Данни за всички външни лица и организации, с които потребителите кореспондират
6. За всеки документ следва да се поддържат следните статуси: нов, регистриран, изисква подпис, завършен, проверен, книга в която документа попада (Касова/Счетоводна)
7. Атрибути на документите - входящ номер и дата, изходящ номер, свързани документи

Видове функции:

Системата трябва да поддържа следния набор от функции:

- a) Създаване на инициращ документ
- b) Създаване на документи по зададени образци от базата данни на системата
- c) Създаване на документ към преписка (подчинен документ)
- d) Редактиране на документ
- e) Регистриране на документ
- f) Адресиране на документ
- g) Добавяне на документ към съответната книга - Касова или Счетоводна
- h) Заявка за подпис, съгласуване или одобрение
- i) Подписване и съгласуване
- j) Преместване на документи в преписка
- k) Поставяне на резолюции – срокове, права, нотификации, адресиране, статус
- l) Попълване на атрибути на документа - номер, дата и час, тип на документ, относно, допълнителни забележки, данни за кореспондент
- m) Възлагане на задача
- n) Отбелязване на изпълнението
- o) Контрол на изпълнението на задачи и разпореждания по документи
- p) Водене и приключване на преписки по кореспонденти, поддържане на преписки и дела с документи
- q) Дефиниране на права на достъп, преглед, редакция, справка на ниво – потребителска роля, отдел или най-ниско ниво потребител

За потребители със съответните права, системата ще позволява:



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- a. проверка на комплекти документи
- b. ред на въвеждане на документите
- c. търсене на документи по различни критерии

20. Модул 17: Структуриране на изискванията и техните атрибути

- Примерна структура на изискванията
- Атрибути на изискванията
- Атрибути за проследяване на изискванията

20.1. Цел

Примерна структура на изискванията

Съществуват разнообразни методи за структуриране на изискванията в документацията, като най-популярни от тях са следните:

- По заинтересовано лице, отдел
- По системни модули
- По функции
- По случаи на употреба
- По фаза на проекта
- Със секция за „извън обхвата“

Атрибути на изисквания

Атрибутите на изискванията представляват допълнителна информация за самите изисквания с цел тяхната по-добра проследимост. Примерни атрибути са:

- Идентификационен атрибут
 - Число - идентификационния атрибут може да бъде число, например цяло число - 1, 2, 3, ...
 - Комбинация от число и букви - например
 - F.1 –F.# - Функционални изисквания
 - NF.1 –NF.# -Нефункционални изисквания
 - Йерархия - подразбира се според идентификационни атрибут, например, за следващите изисквания се приема, че са подизисквания на F.1
 - F.1.0



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

•F.1.1

•F.1.2

- Автор на изискването
- Сложност
- Собственост
- Приоритет
- Риск
- Източник
- Стабилност
- Спешност

Атрибути за проследяване на изискванията

Използването на атрибути към изискванията подпомага проследяването на изискванията. Например: кои са всички изисквания с източник X, кои са всички изисквания свързани с даден риск Y.

Проследяването на изискванията се осигурява чрез автоматизирани системи (например IBM Doors, Rational Req Pro).

20.2. Упражнение

Приложете подходяща структура на функционалните изисквания от предходното упражнение.

20.3. Отговор

(Подходяща структура за изискванията - *Атрибути на документ, Управление (CRUD) на задачи, Управление (CRUD) на преписки и дела, Права и достъпи, Търсене и Справки*):

Системата трябва да може да поддържа:

Видове данни

1. Потока от документи (входящи, изходящи и вътрешни) и оперативна информация (отчети, заявки, становища, справки, писма, резолюции, заповеди и др.)
2. Движение на документи вътре в организацията във връзка с изпълняваните дейности
3. Задачите и контрола по тяхното изпълнение
4. Данни за потребителите на системата съобразно тяхната роля в структурата на организацията
5. Данни за всички външни лица и организации, с които потребителите кореспондират



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

6. За всеки документ следва да се поддържат следните статуси: нов, регистриран, изисква подпис, завършен, проверен, книга в която документа попада (Касова/Счетоводна)
7. Атрибути на документите - входящ номер и дата, изходящ номер, свързани документи

Видове функции:

Системата трябва да поддържа следния набор от функции:

Управление (CRUD) на Документи

Създаване на инициращ документ

1. Създаване на документи по зададени образци от базата данни на системата
2. Създаване на документ към преписка (подчинен документ)
3. Редактиране на документ
4. Регистриране на документ
5. Адресиране на документ
6. Добавяне на документ към съответната книга - Касова или Счетоводна
7. Заявка за подпис, съгласуване или одобрение
8. Подписване и съгласуване
9. Преместване на документи в преписка
10. Поставяне на резолюции – срокове, права, нотификации, адресиране, статус
11. Атрибути на Документи
12. Попълване на атрибути на документа - номер, дата и час, тип на документ, относно, допълнителни забележки, данни за кореспондент

Управление (CRUD) на задачи

1. Възлагане на задача
2. Отбелязване на изпълнението
3. Контрол на изпълнението на задачи и разпореждания по документи

Управление (CRUD) на преписки и дела

Водене и приключване на преписки по кореспонденти, поддържане на преписки и дела с документи

Права и достъпи

Дефиниране на права на достъп, преглед, редакция, справка на ниво – потребителска роля, отдел или най-ниско ниво потребител

Търсене и Справки

За потребители със съответните права, системата ще позволява:

1. проверка на комплекти документи



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

2. ред на въвеждане на документите
3. търсене на документи по различни критерии

21. Модул 18: Практика: Извличане на функционални изисквания от потребителски изисквания и use cases

- Писане на функционални изисквания, с различни източници (заинтересованите страни, use cases)
- Ниво на детайлност за функционалните изисквания
- Къде можем да открием функционални изисквания?
- Писане на функционални изисквания, извлечени от изисквания на заинтересованите страни

21.1. Цел

Запознаване с различни методи за извличане на функционални изисквания от потребителски изисквания, от изискванията на заинтересованите страни и случаи на употреба.

Ниво на детайлност за функционалните изисквания

Нивото на детайлност до което трябва да бъдат разписани функционалните изисквания е предмет на обсъждане на почти всички форуми, разискващи проблемите на бизнес анализа. Едно мнение доминира над всички и това е, че нивото на детайлност е в пряка зависимост от желанията на различните заинтересовани страни, т.е. трябва да се детайлизира до тогава, докато заинтересованите страни разбират и одобряват представените изисквания.

Източници за изисквания

Потенциални източници на изисквания са:

- Взаимоотношения със заинтересованите страни – интервюта, наблюдения, фокус групи
- Процеси и случаи на употреба – документираните детайли за потока на работата
- Артефакти - форми, отчети
- Модели
- Съществуваща функционалност
- Прототипи

Stakeholders -> Функционални изисквания



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Задачата е от даденото изискване на заинтересовано лице да се извлекат колкото може повече функционални изисквания:

•Изискване на заинтересовано лице: Да се трансферират детайлите на избрания доставчик (номер, име, адрес на регистрация, адрес на доставка, телефон) от базата данни, за да се попълни RFP документа.

Извлечени функционални изисквания:

- Ф1. Системата ще трябва да запита потребителя за идентификатор на доставчик – номер или име
- Ф2. Системата ще трябва да покаже всички намерени записи, които отговарят на заявката
- Ф3. Системата ще трябва да даде възможност на потребителя да избере един от доставчиците, ако има повече от един запис
- Ф4. Системата ще трябва да трансферира данните от базата данни на доставчика – адрес на регистрация,...
- Ф5. Системата ще трябва да запълни полетата в RFP формата, с трансферираните данни.

Случаи на употреба -> Функционални изисквания

•Случаите на употреба описват последователност от стъпки, при взаимодействието на потребителя със системата, с цел да се изпълни някакъв резултат

•Една конкретна последователност се нарича сценарий

•Основен път

•Алтернативен път

•Път за изключения

Примерен фрагмент от случай на употреба. Целта е да се открият и генерират множество от функционални изисквания директно породени от случая на употреба. Няма ограничения в броя на извлечените изисквания.

Use Case ID:	UC 2.1
Use Case:	АКТИВИРАНЕ НА НС АБОНАТ
Обобщение:	Този случай на употреба се изпълнява от служител Обслужване на клиенти, когато клиент влиза в магазина да сключи договор на НС услуга.
Актьори:	Физическо лице (Клиент), Служител Обслужване на клиенти.
Тригер:	Клиентът влиза в магазина за да сключи договор за НС услуга.
Пред-условия:	Всички системи са работещи (Remedy 6, ADX Billing&CRM).
Основен сценарий:	<ol style="list-style-type: none">1. Служител Обслужване на клиенти проверява дали в системата ADX CRM съществува случай за проверка на изпълнимост на заявката.2. Служител Обслужване на клиенти се уверява, че резултата от проверката е положителен за адреса на клиента.3. Служител Обслужване на клиенти обяснява на Клиента условията по договора.4. Клиентът избира да сключи договор.5. Служител Обслужване на клиенти идентифицира Клиента в системата ADX и предоставя услугата на Клиента. <i>/**Всички услуги трябва да бъдат активирани под един и същи BAN*/</i>6. Клиентът подписва договора.7. Служител Обслужване на клиенти затваря диалоговия прозорец за сключване на договор и изписва оборудването (аналогов/DECT телефон).

Примерни функционални изисквания, които могат да бъдат извлечени от горепосочения фрагмент от случай на употреба.

- Ф1. Системата трябва да позволява активиране на случай за проверка за проверка на изпълнимост на заявка.
- Ф2. Системата трябва да позволява промяна на статуса на случай за проверка за проверка на изпълнимост на заявка.
- Ф3. Системата трябва да позволява проверка на статуса на случай за проверка за проверка на изпълнимост на заявка.
- Ф4. Системата трябва да позволява селекция на приложимите за конкретния клиент услуги.



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Ф5. Системата трябва да позволи отпечатване на договор за подписване от клиента.
- Ф5.1 Системата трябва автоматично да добави данните на клиента, във формуляра за договора.
- Ф5.2 Системата трябва да позволява редактиране на данните, в определени полета на формуляра за договора.

21.2. Упражнение

Какви функционални изисквания може да извлечете от потребителските изисквания от упражнението в модул 15?

21.3. Отговор

1. За всеки документ следва да се поддържат следните статуси: нов, регистриран, изисква подпис, завършен, проверен, книга в която документа попада (Касова/Счетоводна)
2. За потребители със съответните права, системата ще позволява:
 - a. проверка на комплекти документи
 - b. ред на въвеждане на документите
 - c. търсене на документи по различни критерии

22. Модул 19: Практика: Извличане на функционални изисквания от артефакти

- Писане на функционални изисквания, с различни източници - артефакти

22.1. Цел

Запознаване с различни методи за извличане на функционални изисквания от артефакти. Целта на модула е да се демонстрира методи и средства за откриване и детайлизиране на функционални изисквания от разнообразни артефакти - репорти, таблици и др.



Европейски съюз



ОПАК Експерти в действие



Европейски социален фонд
Инвестиции в хората

ИТ общ, 2013__Сменен режим.xls [Compatibility Mode] - Microsoft Excel

1	Полож в работни дни през 2013								Платени часове извънреден труд в почивни дни и на сменен режим през 2013																			
	месец								ОБЩО 2013 г.	месец													ОБЩО 2013 г.					
2	08	09	10	11	12	I-во три	II-ро три	III-то три		VI-то три	01	02	03	04	05	06	07	08	09	10	11	12		I-во три	II-ро три	III-то три	VI-то три	
3	Име, Презиме и Фамилия																											
5									0.0					12.0													12.0	
6								0.5	35.0	0.5	36.0											28.0	44.0		35.0	107.0		
7					2.0						2.0				6.0				6.0								12.0	
8											0.0			5.0													5.0	
9								4.0			4.0																0.0	
10											0.0			12.0													12.0	
11								29.5			29.5																0.0	
12								30.5			30.5											8.0					8.0	
13									0.5	0.5								38.0	44.0	1.0	14.0					97.0		
14						2.5			0.5	3.0														25.0	40.0		65.0	
15										0.0						6.0											6.0	
16										0.0				12.0													12.0	
17								7.5		7.5																	0.0	
18					2.0					2.0																	0.0	
19					5.0					5.0																	0.0	
20					5.0					5.0				5.0					6.0								11.0	
21								0.5	6.5	0.5	7.5													44.0	50.0		94.0	
22								6.0	4.0	0.5	10.5											28.0	50.0		40.0	118.0		
23										0.0				12.0													12.0	
24										0.0																	0.0	
25										0.0																	0.0	
26										0.0																	0.0	
27		0.0	10.0	0.0	0.0	4.0	2.5	11.0	113.0	2.5	143.0	0.0	0.0	0.0	0.0	58.0	0.0	12.0	0.0	0.0	0.0	12.0	0.0	102.0	182.0	26.0	179.0	571.0

Артефактът, който ще разгледаме, представлява таблица за пресмятане на отработено време от служителите.

- Ф1. Системата трябва да поддържа няколко типа служители:
 - На нормирано работно време
 - На ненормирано работно време
 - На сменен режим
- Ф2. Системата трябва да събира информация за отработените часове за всеки месец и да отчита извънредния труд
- Ф3. Системата трябва да обобщава резултати за часовете извънреден труд, по тримесечия
- Ф4. Системата трябва да обобщава информацията за часовете извънреден труд, по области

22.2. Пример за извличане на функционални изисквания от следния артефакт

I. Регистрация на първоначалния /инициативен/ документ			II. Регистрация на междинните документи и отговори. Движение на документите	
1. Наш регистрационен индекс - дата	2. Кореспондент – адрес	3. Вид документ	9. Изх. №, Дата, Изпратен до – Получен от – Предаден на, Начин на изпращане	
161/03.02.2012 г.	Висш съдебен съвет Гр. София	Писмо – ел. поща	03.02.2012 г. – П. Стоянова – подпис 03.02.2012 г. – Д. Димова – подпис	
4. Резолиран от		5. Кореспондентски рег. индекс - дата		
И. Стоева		Изх. № 91-00 018/ 03.02.2012 г.	Изх. № 93/24.02.2012 г. – изпратено по ел. поща – Д. Димова – подпис Изх. № 93/24. 02.2012 г. – RID – поща – 005U16 D	
6. Относно				
Въпросник за СФУК за 2011г.				
7. Резолюция /задача/				
За сведение и изпълнение на съдебен администратор и гл. счетоводител				
8. Срок за изпълнение				
29.02.2012 г. – на е-тап и по				

Примерни функционални изисквания, които могат да се извлекат от посочения артефакт:

1. Системата трябва да позволява автоматично генериране на регистрационен индекс във формат XXX/дата, където XXX е цяло положително число, което се увеличава с 1 за всяка нова регистрация.
2. Системата трябва да позволява въвеждане на адрес на Кореспондент в свободен текст до 200 символа.
3. Системата трябва да позволява на потребителя избор на вид документ за регистрация.

23. Модул 20: Нефункционални изисквания

- Кои изисквания са нефункционални?
- Видове нефункционални изисквания
- Добри практики при писането на нефункционални изисквания
- Източници на нефункционални изисквания

23.1. Цел

Запознаване с най-популярните нефункционални изисквания



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Нефункционалните изисквания са условия, които нямат директна връзка с поведението или функциите на разработваната система.
- Нефункционалните изисквания са условия на средата, при които функционалността на разработваната система, трябва да остане с необходимото качество.
- Наричат се още supplementary или quality of service.

Източници на нефункционални изисквания

- Взаимодействие със заинтересованите лица и техните изисквания
- Процеси и случаи на употреба
- Артефакти
- Регулации, законова рамка
- Външна среда
- Операции
- Сигурност

Изисквания за надеждност

- Fault tolerance (Устойчивост при откази) - дефинира изисквания за системата да продължи работа при възникване на отпадане на компонент
- Resiliency and redundancy (Устойчивост и дублиране) - дефинира изисквания за възстановяване след отпадане
- Disaster recovery (Възстановяване) - възстановяване след мащабно отпадане на системата
- Information availability (Наличие на информация) - дефинира изисквания за наличие на определена информация
- Business continuity (Непрекъсваемост на бизнеса) - осигуряване на наличието на бизнес критични функции
- Uptime vs. downtime (експлоатационно спрямо неексплоатационно време)
- Backup and restore functions (Архивиране и възстановяване на функциите)

Примери:

- Системата трябва да бъде 99.999% от времето активна, което е 8 часа общо неактивност за 365 дни
- Системата трябва да архивира всички данни в 23.30 всеки ден.

Изисквания за производителност



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Processing volume (Обем обработка) - дефинира се необходимия обем данни, които трябва да могат да се обработват за единица време
- Processing time (Време за обработка) - дефинира се времевия отрязък, в който трябва да се изпълни дадена операция
- Transmission speeds (Скорост на пренос) - дефинира скоростта, с която трябва да се пренасят данни
- Throughput - дефинира обема данни които се трансферират за единица време
- Utilization thresholds (Гранични стойности за използване) - дефинира граничните стойности за използване на даден компонент
- Buffer capacities (Буферен капацитет) - дефинира напр. капацитета на буфера от данни, който може да се използва
- Storage capacities (Капацитет за съхранение) - дефинира капацитета, който се отделя за съхранение на данни

Примери:

- Системата трябва да обработва 1000 записа за 1 час.
- Системата трябва да може да отпечата 4500 страници за 8 часов период.

Изисквания за оперативност и използваемост

- Ease of use (Лесна за употреба) - дефинира изисквания за лесна и удобна употреба на системата
- Number of concurrent user access (Брой едновременен достъп до системата) - дефинира изисквания за броя на едновременно достъпващите системата
- Licensing policies (Политики за лицензиране) - дефинира политиките за лицензиране
- User access methods (Методи за достъп до системата) дефинира видовете достъп на потребителите до системата
- Future growth and capacity (Бъдещ растеж и капацитет) - дефинира изисквания за бъдещ растеж на системата и необходимия капацитет
- Operating conditions (Условия за работа) - дефинира изисквания за оптимални условия за работа на системата
- Help features (Помощни функции) - дефинира изисквания за наличие на помощни функции
- Levels of knowledge (Ниво на знания) - дефинира изисквания за нивото на знания необходими за използването на системата
- Look and feel (Изглед и усещане) - дефинира изисквания за графичен изглед и усещане на визуалния интерфейс



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

•Experience and proficiency (Опит и умения) - дефинира изисквания за опит и умения на потребители за използване на системата

Примери:

- Системата трябва да предоставя помощно меню за всяка функционалност.
- Системата трябва да позволява 75% от населението да се ориентира в наличните функционалности в рамките на 5 минути.

Изисквания, свързани със сигурността

- Access levels and capabilities (Нива на достъп и права) - дефинира изисквания за правата и ролята на достъп до функционалностите на системата
- Restrictions (Ограничения) - дефинира ограниченията, които се налагат върху използването на функционалността
- Policies (Политики) - дефинират се политиките за сигурност
- Authorizations (Оторизация) - дефинират се механизмите за оторизация на права за достъп до функционалността на системата
- Validation (Валидация) дефинират се механизмите за валидация на потребителите в системата
- User verification (Проверка на потребителя) - дефинират се механизми за верификация на потребителите
- Security breach policies (Политики при нарушаване на сигурността) - дефинират се политики при опит за нарушаване на сигурността
- Monitoring (Мониторинг) - дефинират се изисквания за мониториране на софтуерни или хардуерни компоненти
- Access timeframes (Времеви достъп) - дефинира се периода в който могат да се използват дадени технологични ресурси
- Access locations (Места за достъп) - дефинират се локации от които е позволено достъпа до дадени технологични ресурси
- Hacking/cracking prevention (Предотвратяване на хакерски атаки) - дефинират се мерки за предотвратяване на хакерски атаки
- Data storage protection (Защита на данни) - дефинират се изисквания за защита от достъп/промяна/изтриване и др. на съхранени данни
- Intruder detection (Засичане на неототоризирано присъствие) - дефинират се механизми за засичане на неототоризирано присъствие в помещения със специален достъп (сървърна зала например)

Примери:



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Системата ще верифицира потребителския идентификационен номер преди да се даде достъп до приложението.
- Системата ще блокира достъпа до приложението при 3 последователни въвеждания на грешна парола.

Изисквания, свързани със съвместимостта

- Hardware connections (Хардуерни връзки) - дефинират се връзките между хардуерни компоненти
- Software connections (Софтуерни връзки) - дефинират се връзките между софтуерни компоненти
- Data transmissions (Предаване на данни) - дефинират се изисквания и формати за предаване на данни
- Operating systems (Операционни системи) - дефинират се операционните системи приложими за разработваната информационна система
- Applications (Приложения) - дефинират се необходимите приложения за работата на информационната система
- Query and access methods (Заявки) - дефинират се методите и средствата за осъществяване на достъп и заявки към базата данни
- Toolsets (Инструментални средства) - дефинират се множеството инструментални средства, необходими за работата на системата
- File sharing (Споделяне на файлове) - дефинират се методи и средства за споделяне на файлове
- Remote access (Отдалечен достъп) - дефинират се методи и средства за отдалечен достъп до системата
- External system interactions (Взаимодействие с външни системи) - дефинират се изисквания за взаимодействие с външни информационни системи

Примери:

- Системата трябва да може да бъде инсталирана върху операционна система Windows 2000.
- Системата трябва да може да работи при налични 512 Мб оперативна памет.

Изисквания, свързани с поддръжката

- Maintenance needs (Нужда от поддръжка) - дефинират се изискванията свързани с поддръжката на системата
- Ease of changing components or program (Леснота на смяна на компоненти или програми) - дефинират се изисквания за лесна смяна/подмяна на компоненти или програми



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Code or component re-use (Многократна употреба) - дефинират се изисквания с които се постига възможност за многократна употреба на сорс-код и/или компонент
- Maintenance timing (Време за поддръжка) - дефинира се времева рамка в която се извършва поддръжката
- Administration needs (Администрация) - дефинират се изисквания за определени административни операции
- Enhancement options (Подобрения) - дефинират се изисквания за възможности за въвеждане на подобрения в системата
- Troubleshooting and diagnosis (Отстраняване на проблеми и диагностика) - дефинират се изисквания за процеса по отстраняване на проблеми и диагностика на информационната система
- Testing process (Процес по тестване) дефинира се процеса по тестване на информационната система
- Upgrades (Надграждане) - дефинират се изисквания към възможностите за надграждане на информационната система
- Scalability (Скалируемост) - дефинират се изисквания за скалируемост (добавяне на още потребители, данни и др. в бъдещата експлоатация на системата)

Примери:

- Системата ще трябва да може да се поддържа от не повече от 3 системни администратора.
- Системата ще трябва да се обновява само в петък в 23.59 часа.

Изисквания, свързани с трансфер (Transferability and Portability)

- Adaptability (Адаптивност) - дефинира изисквания за лесна адаптивност при трансфер на информационната система
- Data migration (Миграция на данни) - дефинира необходимостта от миграция на данни при трансфер на информационната система
- Application migration (Миграция на приложения) - дефинира изискванията породени от необходимостта от миграция на приложения
- Hardware support (Поддръжка на хардуер) - дефинира изискванията за хардуерните компоненти необходими за поддръжката на системата
- Operating system support (Поддръжка на операционната система) - дефинира изискванията за операционната система необходими за поддръжката на системата
- Installation criteria (Критерии за инсталация) - дефинира важните критерии за инсталация на информационната система
- Uninstallation processes (Процеси по деинсталация) - дефинира процесите които се изпълняват по време на деинсталация на системата



- Environment needs (Нужди на средата) - дефинира всички необходими изисквания на средата за оптималното използване на системата
- Object-oriented design (Обектно-ориентиран дизайн) - дефинира изисквания към дизайна на системата, напр. използване на обектно-ориентиран дизайн
- Standard vs proprietary components (Стандартни срещу лицензирани компоненти) - дефинира изисквания за използване на стандартни, свободни, лицензирани и др. тип компоненти

Примери:

- Системата ще трябва да позволява пренос на данни използвайки ASCII символи.
- Системата ще трябва да премахва всички приложени файлове при деинсталация.

Транзитивни(Преходни) изисквания

- Транзитивните изисквания са преходни изисквания, необходими за трансформацията от старото към новото решение.
- Обикновено новото решение се въвежда с цел подмяна на старото решение.
- В периода на преход между двете, организацията навярно ще има нужда и двете решения да бъдат налични с цел например миграция на данни, обучения за потребителите и т.н. Тези изисквания са важни за успешното включване на новото решение и понеже са различни по същност от всички други изисквания се разглеждат отделно.
- Data migration and transmission plans (Планове за миграция и пренос на данни) - разработени детайлни планове за миграция и пренос на данни
- Server storage from one system to another (Сървър за съхранение) - дефинира се необходимостта от използването на сървър за съхранение на данни при миграция от една система към друга
- Training requirements for stakeholders (Изисквания за обучения) - Изисквания за обучение на заинтересованите страни
- Parallel work environments (Паралелни работни среди) - дефинира необходимостта от въвеждането на паралелни работни среди в периода на миграция от предишно към ново решение
- Contingency and backout plans - разработени детайлни планове за извънредни ситуации в процеса на миграция от предишно към ново решение
- Manual process vs automated process preparedness (Ръчен или автоматизиран процес на готовността) - дефинира необходимия ръчен или автоматизиран процес за проверка на готовността за миграция към новото решение
- Communication needs (Комуникационни нужди) - дефинира комуникационните нужди на всички заинтересовани страни в процеса на миграция към новото решение



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- System configuration changes immediately prior to implementation (Конфигурационни промени по система непосредствено преди разработката/имплементацията) - дефинира всички необходими конфигурационни промени, които трябва да се извършат непосредствено преди имплементацията
- System backups (Архивиране на системата) - дефинира необходимите действия по архивиране на важни за системата данни
- System cutover plans - детайлни планове за цялостната миграция на системата

23.2. Пример за пренаписване на нефункционални изисквания

Тези нефункционални изисквания (получени от потребител на системата) могат да се подобрят/пренапишат, за да отговарят на правилата за писане на изисквания

- Да е лесна за работа
- Да е автоматизирана
- Да е на български език
- Да е сигурна

23.3. Подобрени/пренаписани:

- Системата трябва да поддържа различни права на достъп, преглед, редакция, справка на ниво потребител.
- Системата трябва да се поддържа на български език
- Системата трябва да е лесна за използване - 80% потребители без опит със системата да обработват документ за по-малко от 5 минути.

24. Модул 21: Въведение в UML

- Възникване
- Какво в UML
- Начини на употреба на UML
- Видове модели

24.1. Цел

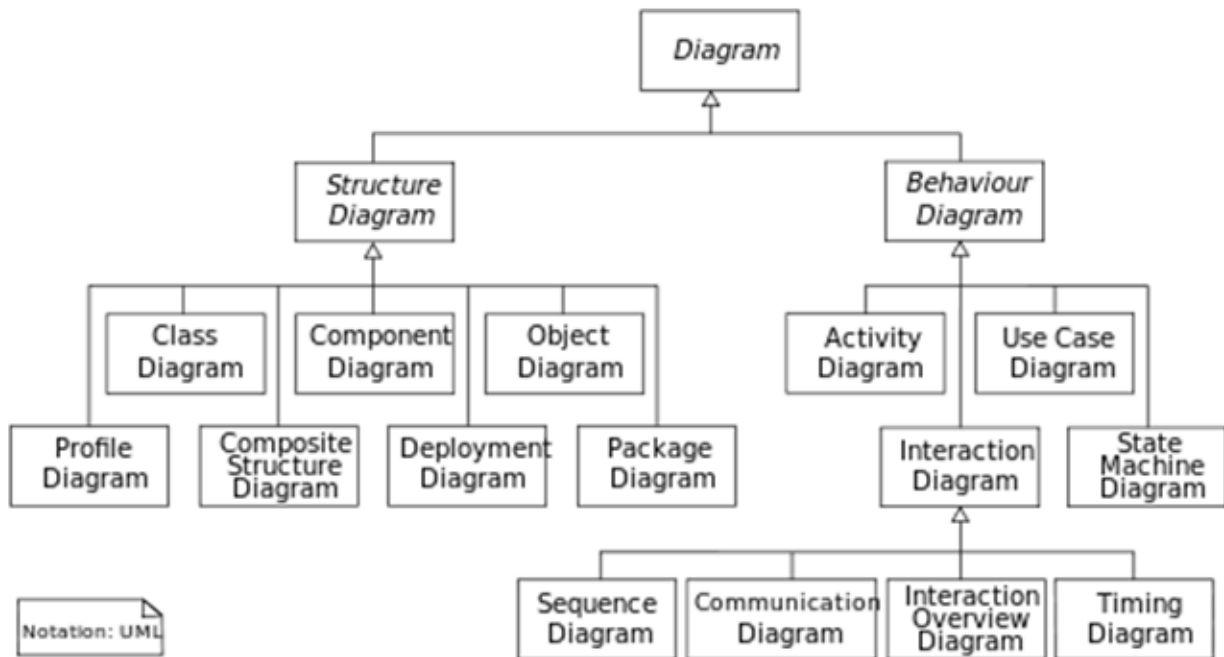
Целта на този модул е да се направи въведение в унифицирания език за моделиране UML. Официалната страница на UML е uml.org, откъдето може да бъде изтеглена спецификацията. Препоръчителна книга за Основи на UML е UML Distilled от Martin Fowler.

UML е стандартен (ISO/IEC 19501:2005), език за моделиране с общо приложение в областта на софтуерните инженерство, създаден от Grady Booch, Ivar Jacobson и

James Rumbaugh в Rational Software през 1990те, последващо придобит от Object Management Group (OMG) в 1997, където се управлява и досега

От 2000 година UML се приема от International Organization for Standardization (ISO) като индустриален стандарт за моделиране на софтуерни системи

Текущата версия на UML е 2.4.1 публикувана от OMG през август 2011



24.2. Модел – дефиниция и цели

Какво е модел?

Моделът е опростено представяне на реалността.

Моделът представлява проект на системата.

Моделът включва елементите и аспектите на системата от съществено значение, и не се фокусира върху маловажни страни.

Защо моделираме системите?

Визуализираме система с желаните качества.

Определяме структурата и поведението на системата.

Създаваме шаблон за конструиране на системата.

Документираме взетите решения до този момент.

24.3. Какво е моделиране?

Моделиране означава да създадем абстракция на действителността

Абстракциите представляват опростен образ на реалността

Те игнорират несъществените детайли

Запазват само съществените

Кое е съществено и кое - несъществено зависи от предназначението на модела

Моделирането е средство за справяне със сложността



Картата изобразява само елементи, които са важни - имената на улиците, посоките, кварталите, като пренебрегва маловажните неща като дървета, пешеходци, коли.

Примери за моделиране са моделите на софтуерната архитектура на изгражданата информационна система, потребителските графически интерфейси, дефиниране на процеси.

25. Модул 22: Въведение в структурните UML модели

- Преглед на видовете структурни UML модели
 - Клас диаграма (Class diagram)
 - Диаграма на обектите (Object diagram)
- Клас диаграма - свойства, множественост, асоциации

25.1. Цел

Този модул въвежда в правилата и дава указания за създаване на структурни UML модели.

Структурни UML модели:

UML съдържа 8 структурни диаграми:

- Диаграма на класовете (Class diagram)
- Диаграма на обектите (Object diagram)
- Диаграма на компонентите (Component diagram)
- Диаграма на профила (Profile diagram)
- Диаграма на разгръщането (Deployment diagram)
- Диаграма на пакетите (Package diagram)
- Композитно-структурна диаграма (Composite structure diagram)

Добри практики за диаграмите:

- Диаграматите трябва да бъдат лесно разбираеми и четими.
- Но не твърде елементарни.
- Комбинация от статичните и динамичните диаграми.
- Диаграмата да се именува съобразно предназначението □.
- Диаграмите се организират йерархично в пакети.
- Да се въздържа от използването на пресичащи се линии.
- Логическите свързани елементи се изобразяват близо един до друг.
- Използвайте различни цветове и допълнителни бележки, за да фокусирате вниманието върху важни елементи в диаграмата.

Класовете са множества обекти, които имат общи характеристики и свойства. Диаграмата на клас изобразява типовете обекти в разглежданата област и различните взаимоотношения, които съществуват между тях.

Интерфейсите представляват класове, които не се имплементират, всичките им операции са абстрактни (без разработка/имплементация).

Диаграмата на класовете показва как различните елементи на системата (класове, интерфейси и връзки) си взаимодействат или казано с други думи, тя представя статичната структура на системата.

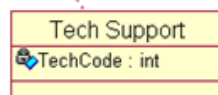
Диаграмата на класовете може да се използва както при визуализирането на абстрактните класове, т.е. обектите от бизнес средата, така и при визуализиране на класовете, които ще се използват при програмирането на системата. Тези две групи от класове често се припокриват като имена на класове и основни връзки между тях. Основната разлика е при методите на класовете, които са по-технически определени във втората група (ползваните при програмирането).

Диаграмата на класовете е една от най-използваните диаграми при моделирането на обектно-ориентирани системи. Тя обрисова лексиката на софтуерната система и като такава се използва като база за диаграмата на компонентите и диаграмата за внедряване.

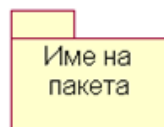
Условни символи в диаграмата на класовете:

Задължение на клас

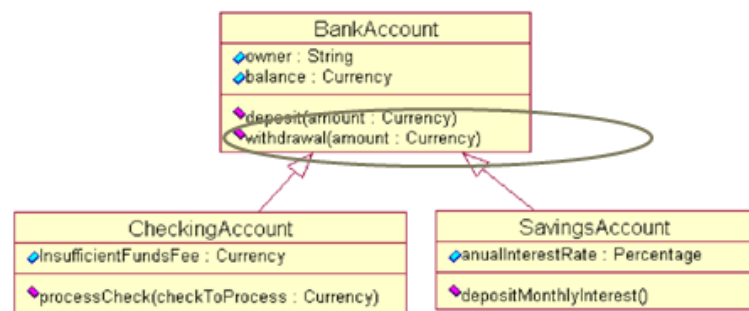
Responsibilities
-- maintain the ATM on the regular base
-- maintain the ATM on emergency



Пакет



Наследствена връзка



Задължение на клас (Responsibilities)

Задължения на класа са отговорностите, които той, респективно неговите обекти трябва да изпълняват. Добра практика при дефинирането на класовете е в началото да се опишат класовете като имена и отговорности, а по-късно да се уточнят техните атрибути и методи.

Диаграма на класовете – асоциации

Алтернативен начин за означаване на свойство е асоциацията. Информацията, която се показва в атрибутите, може да бъде изобразена и с асоциация. Асоциацията се визуализира с непрекъсната линия между два класа. Името на свойството се поставя върху линията или ако се представя с роли - в двата края на линията.

Асоциативна връзка - имена



Асоциативна връзка - роли



Асоциативна връзка – разнообразие



Асоциативна връзка – просто обобщение цяло-част

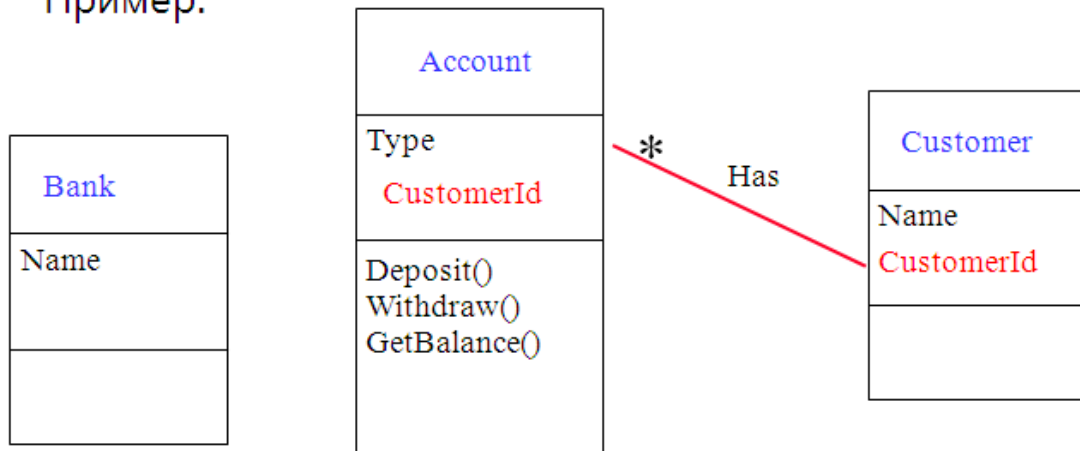


Асоциативно обобщение – не могат да съществуват без цялото



Основни стъпки при създаването на диаграма на класове:

Пример:



- 1) идентифицираме обекти в областта на анализ
- 2) Намираме техните име, атрибути, методи
- 3) Намираме асоциациите между обектите
- 4) Задаваме имена на асоциациите
- 5) Задаваме множественост на асоциациите

Диаграма на обектите

Диаграмата на обектите се състои от обекти и връзки. Както всяка друга диаграма, в нея могат да се използват възли и ограничаващи условия. Може да се наложи за подобряване на четимостта на диаграмата да се добавят пакети и класове.

Обектите в UML се обозначават като правоъгълник, в който се описва името на обекта и класа към който принадлежи този обект, като между тях се поставят две двоеточия

Име на обекта: Име на класа

В допълнителна секция на правоъгълника могат да се опишат конкретните стойности, които имат атрибутите на обекта.

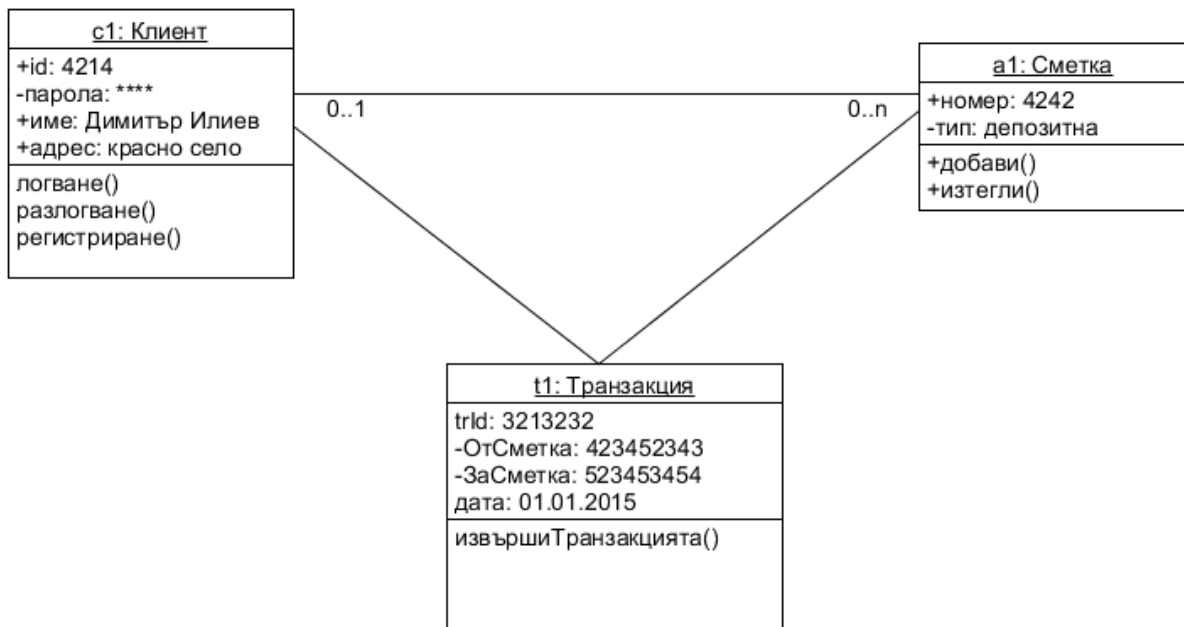
Име на атрибут = Стойност

Диаграмата на обектите моделира отделните обекти, които се съдържат в класовете. Тази диаграма показва отделните обекти и техните връзки в определена точка от времето.

Диаграмата на обектите се използва за моделиране на статичен дизайн или статично определен процес на системата, като се представя нейното моментно състояние чрез визуализиране на обектите, които съществуват в този момент, тяхното състояние и връзките между тях. Тази диаграма е изключително полезна за обрисване

на етапи в сложни процеси с цел разграничаване на съществуващите обекти и техните характеристики, които се използват в дадения момент. Основно грешките при обектно-ориентираните системи се базират не на погрешна логика, а на неправилното взаимодействие на обектите. Тези грешки лесно могат да се избегнат чрез моделиране на по-сложните части от системата с диаграми на обектите. Както и диаграмата на класовете, диаграмата на обектите е елемент от статичното представяне на системата. За разлика от нея диаграмата на обектите е насочена към функционалните изисквания на системата, т.е. описва функционалност на системата в определен момент. Целта е с тази диаграма да се моделира структурата на обектите в определена част на функционалността.

Примерна диаграма на обектите:



На диграмата са изобразени 3 обекта със съответните им атрибути и стойности на атрибутите. Обект c1 от клас Клиент и атрибути id със стойност 4214, име: Димитър Илиев и адрес „красно село“, обект a1 от клас Сметка и атрибут номер 4242 и тип депозитна и обект t1 от клас Транзакция със съответни попълнени атрибути.

26. Модул 23: Практика: Клас диаграма

- Изготвяне на Клас диаграма (Class diagram)

26.1. Цел

Този модул следва въведението в структурните UML диаграми като го допълва със практическа работа по създаване на клас диаграми и диаграми на обекти по



разглежданите до този момент казуси (напр. създаването на документооборотна информационна система).

На следващата клас диаграма са изобразени следните класове:

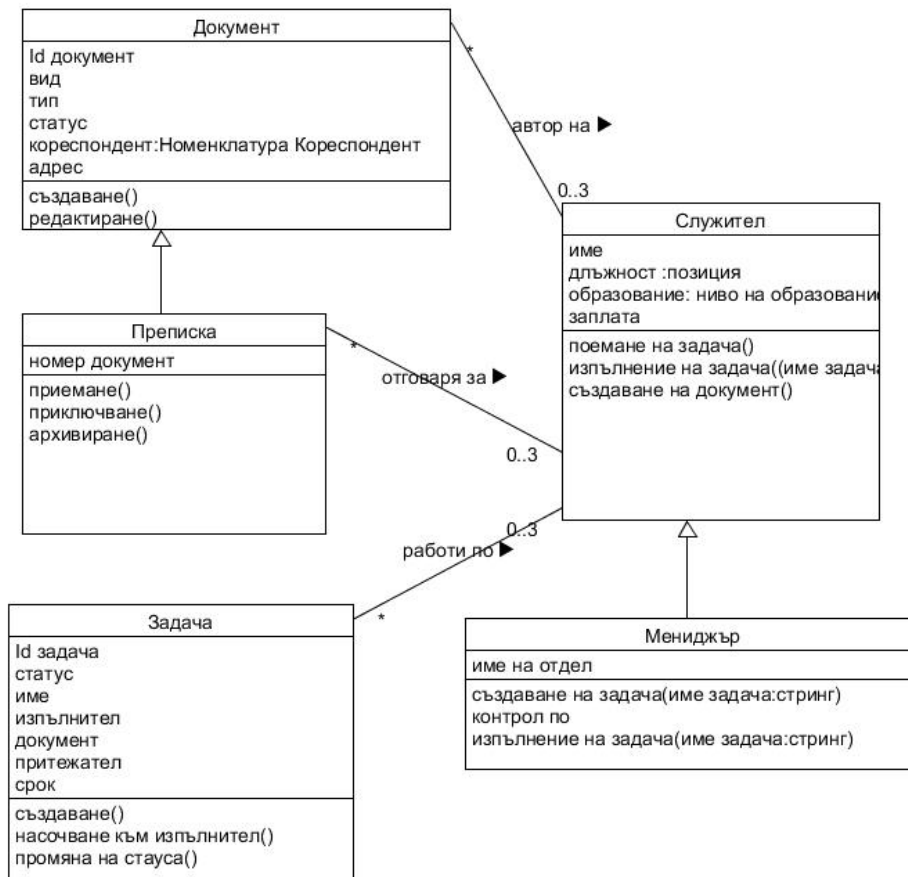
- Документ – с атрибути Id документ, вид, тип и т.н
- Преписка – с атрибути номер документи
- Служител – с атрибути име, длъжност, образование и т.н
- Мениджър - с атрибут име на отдел
- Задача – с атрибути Id задача, статус, име, изпълнител и т.н.
- Номенклатура Кореспондент
- Позиция
- Ниво на образование

Между класовете могат да се забележат следните наследявания:

- Класът Преписка наследява класа Документ, т.е наследява всички негови атрибути и операции
- Класът Мениджър наследява класа Служител, т.е наследява всички негови атрибути и операции

Между класовете могат да се забележат следните асоциации:

- Между класа Документ и класа Служител има асоциация „автор на“
 - Между класа Преписка и класа Служител има асоциация „отговаря за“
 - Между класа Задача и класа Служител има асоциация „работи по“
- Може да се отбележи също и че класовете Номенклатура Кореспондент, Позиция и Ниво на образование са в пакет Номенклатури.



Номенклатури::Номенклатура Кореспондент
име адрес
изпратен документ() получен отговор()

Номенклатури::Позиция
служител деловодител мениджър директор

Номенклатури::Ниво на образование
висше средно

27. Модул 24: Въведение в поведенческите UML модели

- Преглед на видовете поведенчески UML модели
- Диаграма на различните случаи на употреби (Use case diagram)
- Диаграма на последователността (Sequence diagram)
- Диаграма на времето (Timing diagram)
- Изготвяне на диаграма на състоянията (State Machine Diagram)
- Кога се използват диаграмите на състояния
- Изготвяне на диаграма на дейностите (Activity Diagram)
- Кога се използват диаграмите на дейностите

27.1. Цел

Този модул въвежда в правилата и дава указания за създаване на поведенчески UML модели.

Основните възможности, които трябва да притежава една система, която се разработва, се наричат поведение на системата.

Поведението на системата се документира чрез създаване на use case модел, представящ:

- възможностите й, които предвиждаме да разработим (use cases (случаи на употреба)),
- обкръжението й (actors (актьори)), и
- връзките между тях, визуализирани чрез use case (случаи на употреба) диаграми.

Създаването на use case модела започва още в първата фаза на жизнения цикъл с описание на основните действащи лица от обкръжението на системата и предвижданите, в този първоначален етап, възможности на системата. В следващите фази на жизнения цикъл този модел се детайлизира с по-подробни описания, за да се достигне до модел, който напълно описва софтуерната система от функционална гледна точка.

Основна роля на този модел е не само описанието на функционалността на системата. Той представлява отлично средство за по-лесно визуализиране и представяне на тази функционалност както за работния екип, така и пред клиентите и заинтересованите лица. Т.е. use case модела е средство за обсъждане между потребителите и разработващия екип на функционалността и поведението на системата.

Use case диаграма

Условни символи в диаграмата:

Actor – Действащо лице



Лица или обекти, които взаимодействат със системата

Use case – Потребителски сценарии





Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Представя част от функционалността на системата, която е логически завършена – има начало и край.

връзки между елементите

Основни предназначения на диаграмата:

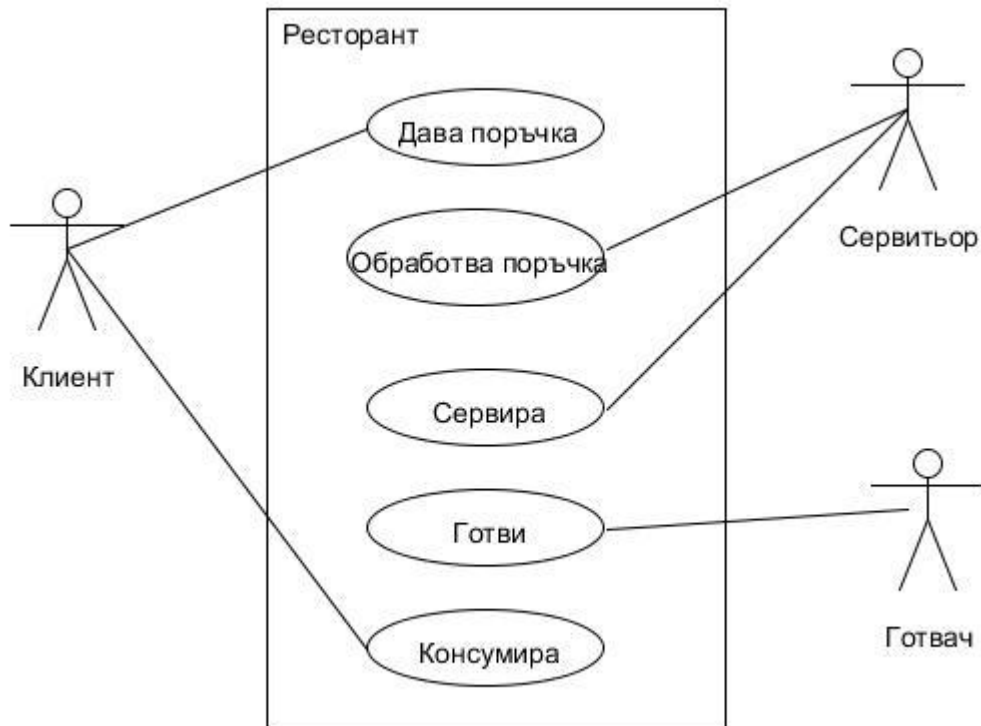
- За моделиране на съдържанието на системата
- За моделиране на изискванията към системата

Един use case описва една от функционалностите на системата

Състои се от:

- Уникално име
- Свързан е с актьори
- Има входни условия
- Съдържа поток от действия (процес)
- Има изходни условия
- Може да има и други изисквания

Пример:



Use-case текстово описание

Име: Клиентът дава поръчка:

Актьори: Клиент, Сервитьор, Готвач

Входни условия:

- Клиентът е гладен
- Клиентът има пари

Изходни условия

- Клиентът е доволно нахранен

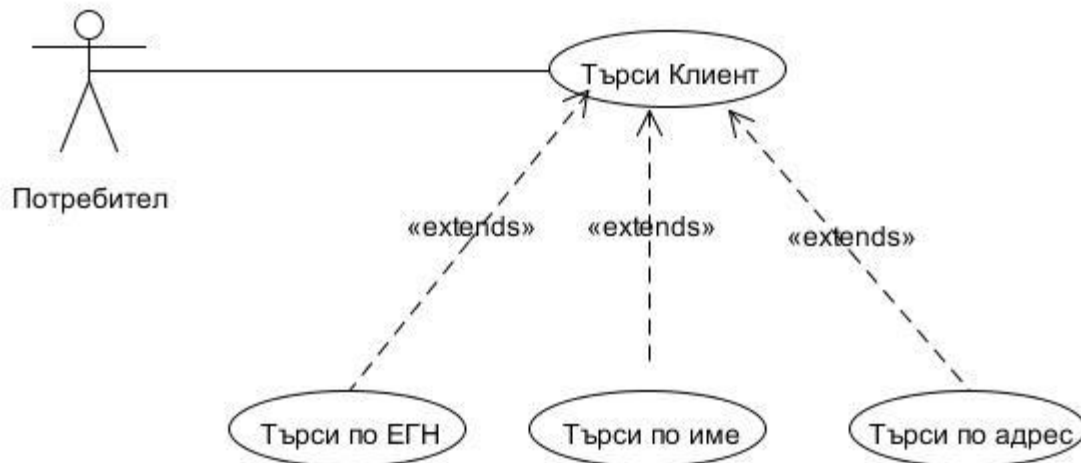
Поток от действия (описание на процеса):

1. Клиентът избира ресторант
2. Клиентът влиза в ресторанта
3. Сервитьорът дава меню на клиента
4. Клиентът поръчва
5. Сервитьорът записва поръчката

Релация <<extends>>

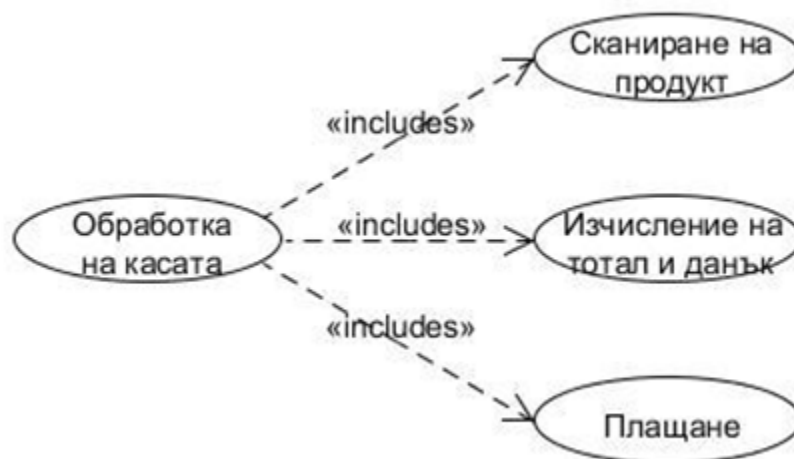
<<extends>> представя изключение или рядко възникващ случай.

Процесът за обработка на изключителна ситуация е извън основния случай на употреба.



Релация <<includes>>

- <<includes>> е поведение, извадено извън даден use case
- Позволява преизползване на споделена функционалност

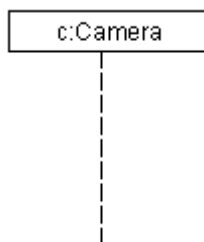


Диаграма на последователността

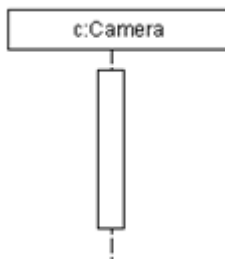
Диаграмата на последователността (sequence diagram) описва динамичните аспекти на системата, т.е. описва различните взаимодействия между множествата обекти, връзките и данните, предавани между тях. Тези диаграми визуализират поведението на системата или на отделни нейни функционалности. Диаграмата на

последователността се фокусира върху това каква е последователността на информацията, която се предава в течение на времето.

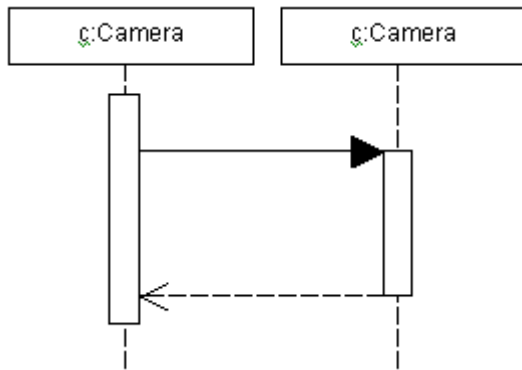
Графическото представяне на диаграмата на последователността много прилича на таблица, в която по хоризонтала се описват обектите, а по вертикала информацията, която те си обменят с течение на времето. Изграждащи елементи на диаграмата на последователността са обектите, връзките и обменената информация или съобщения. Обектите в тази диаграма имат продължителност на живот, която се обозначава с прекъснатата линия, представлява момента на генериране, съществуване и край на живота на съответния обект. Последователността са обектите, връзките и обменената информация или съобщения.



В диаграмата се посочват и точките на контрол, изобразени с тънък вертикално разположен правоъгълник, който посочва в какъв период от време обектът извършва някакво действие.



Информацията (съобщенията), които се предават между обектите се визуализира с плътна линия завършваща със стрелка, като тя започва от обекта който предава съобщението и стига до обекта, към който се предава съобщението. Над тази линия се поставя име на съобщението, което го описва. Добре е да се включи и уникална номерация на съобщенията, представляваща тяхната последователност във времето, което подпомага лесното разбиране на диаграмата.



За да се визуализира взаимовръзката между действащите лица и системата по време на избраната функционалност, действащите лица също се включват в диаграмата като обекти. С тях се работи по същия начин както и с вътрешните обекти на системата.

Диаграмата на последователността най-често се използва при детайлно описание на use case-и, системи или подсистеми, за да визуализира последователността от обмена на информация между обектите във времето. Едно от основните правила при изграждането на този тип диаграма е да се предпазим от усложняване, което води до трудно разбиране на диаграмата. Тя трябва да е лесно разбираема. Ако логиката, която диаграмата описва, е много сложна, препоръчително е да се разбие на под-диаграми. Разделението на под-диаграми може да се направи на база последователни периоди от време, т.е. всяка под-диаграма да изобразява определен период от време, а комбинацията на всички диаграми да ни даде описанието на пълната функционалност.

Диаграма на последователността - елементи

- Класовете се представят с колони
- Съобщенията се представят чрез стрелки
- Участниците се представят с широки правоъгълници
- Състоянията (във времето) се представят с пунктирана линия

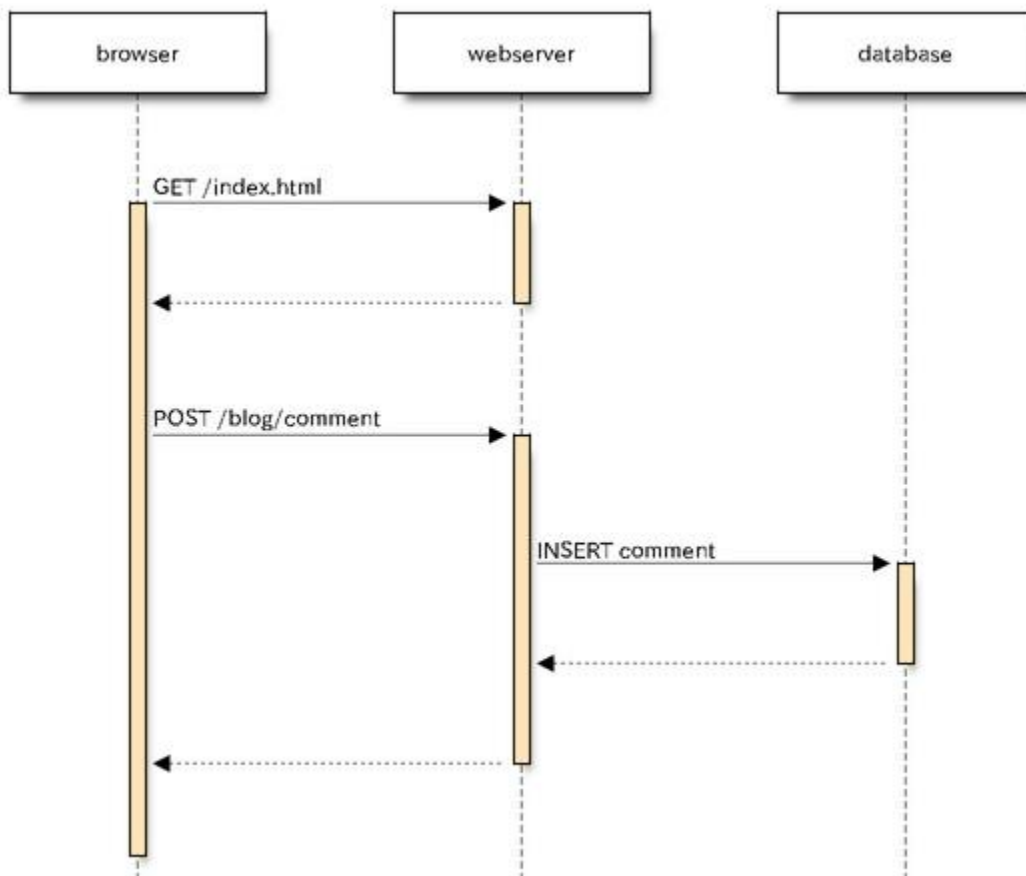
Стъпки за дефиниране на диаграма на последователността

- Избира се функционалност, която ще се моделира.
- Описание на обектите, които присъстват в този механизъм и изпълняват някаква дейност в него.
- Определя се продължителността на живот на всеки един обект.
- За тези обекти, които се създават и/или унищожават по време на визуализираната функционалност, това трябва да се обозначи със съответните условни знаци.

- Като последователност във времето се изобразяват всички съобщения, които се предават между обектите. Определя се наименованието на всяко съобщение и се поставят индикатори за тяхната последователност.
- За обектите се изобразяват точките им на контрол.
- Ако е необходимо, се записват предварителни и последващи условия за съобщенията.
- Преценява се дали е необходимо диаграмата да се раздели на под-диаграми.

Пример:

На следващата диаграма на последователността са изобразени три класа – browser (браузер), web server (уеб сървър), database (база от данни). Първото съобщение, което се разменя между обектите е GET/index.html което се изпраща от browser към web server. По използваната нотация на стрелката се разбира, че съобщението е синхронно. С пунктирна линия се отбелязват съобщенията които се разменят като отговор на друго съобщение. Хронологически следва съобщението POST/blog/comment от browser към web server. След него следва съобщението INSERT comment от webserver към database, като комуникацията завършва с 2 отговора на последните 2 съобщения.

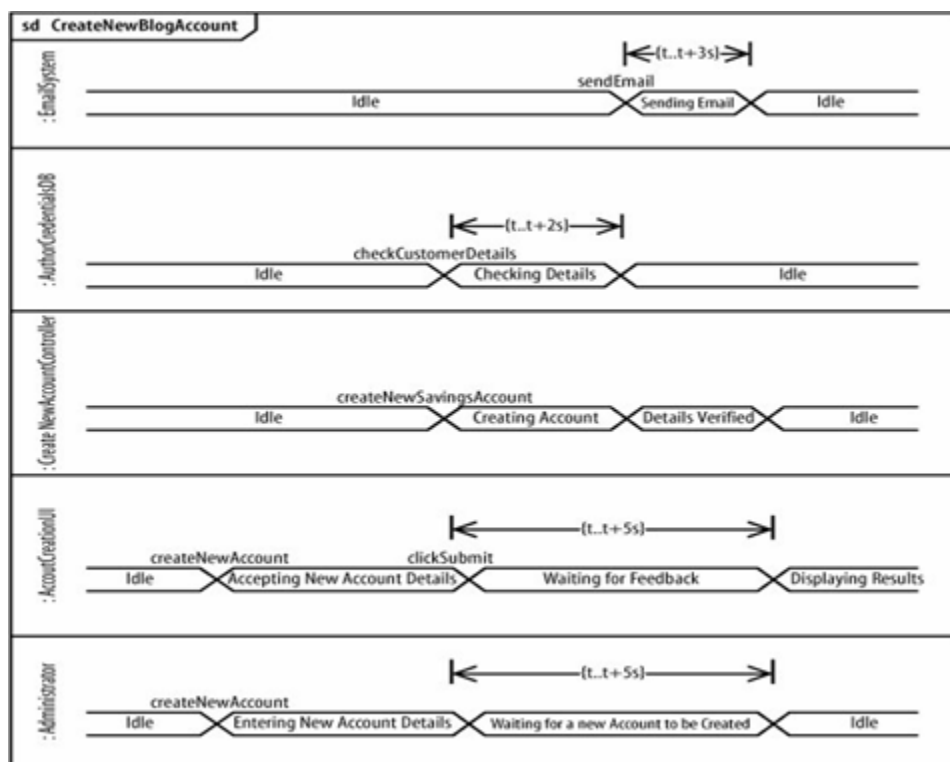


Диаграма за време

Диаграмата за време в UML е специфичен тип диаграма на взаимодействие (Interaction diagram), където фокусът е върху времевите ограничения.

Диаграмите за време се използват за проучване на поведението на обекти в продължение на определен период от време. Диаграмите за време е специална форма на диаграма последователност (Sequence diagram). Разликата между диаграмите за време и диаграмите на последователност е, че осите са разменени, така че времето се увеличава от ляво на дясно, а обектите в отделни отделения разположени вертикално.

Календарното време в диаграмите за време тече от ляво надясно. На следващата примерна диаграма за време може да забележи, че дейностите Checking Details и Creating Account се изпълняват едновременно за едно и също време – 2 секунди, дейностите Sending Email и Details Verified също едновременно за 3 секунди, а Waiting for Feedback и Waiting for a new Account to be Created – за 5 секунди.



Диаграма на състоянието

- Описва поведението чрез състояния и преходи между тях.
- Визуализира поведението на конкретен обект в системата.

• Не се създава за всеки клас в системата, а само за тези, които имат изключително динамично поведение, по време на своето съществуване (т.е. променят често своето състояние).

• Състоянието на един обект в неговия жизнен цикъл, представя какви условия изпълнява обектът в този етап от живота си, какви дейности изпълнява и дали очаква някакви събития.

Диаграмата на състоянията моделира динамичното представяне на системата. Докато диаграмите на взаимодействието представят определен сценарий и взаимодействието на обектите по време на този сценарий, диаграмата на състоянията е на по-ниско ниво. Тя визуализира поведението на конкретен обект в системата. Поведението на обекта се моделира чрез описване на:

- състоянията, през които обектът преминава,
- събитията и информацията, които предизвикват преминаването от едно състояние в друго, и
- действията, които са резултат от промяната на състоянието на обекта.

Този тип диаграма не се създава за всеки клас в системата, а само за тези, които имат изключително динамично поведение по време на своето съществуване (т.е. променят често своето състояние). Разглеждайки създадените диаграми на последователността и кооперирането, лесно могат да се определят обектите в системата, които изпращат и получават най-много информация. Това са и обектите с най-динамично поведение. Препоръчително е всеки един от тях да се представи чрез диаграма на състоянията.

Състоянието на един обект в неговия жизнен цикъл представя какви условия изпълнява обекта в този етап от живота си, какви дейности изпълнява и дали очаква някакви събития. Състоянието може да се характеризира със стойностите на атрибутите на този обект и връзките, които има с други обекти.

В UML състоянието се описва като правоъгълник със заоблени краища, в който се поставя името на състоянието.



Промяна на състоянието се представя в UML като плътна линия, започваща от старото състояние и стигаща до новото състояние, която завършва със стрелка.



Стъпки за дефиниране на диаграма на състоянието

- Избира се елемент, който ще се моделира (система, подсистема, клас, обект или use case).

- Определят се началното и крайното състояние на елемента.

- Определят се „стабилните състояния“ на елемента, т.е. определяне при какви условия елементът, ще остане в определено състояние за даден период от време.

- Поддредат се стабилните състояния по време на жизнения цикъл на елемента.

- Определят се събитията, които водят до промяна на състоянието на елемента. Визуализиране на събитията.

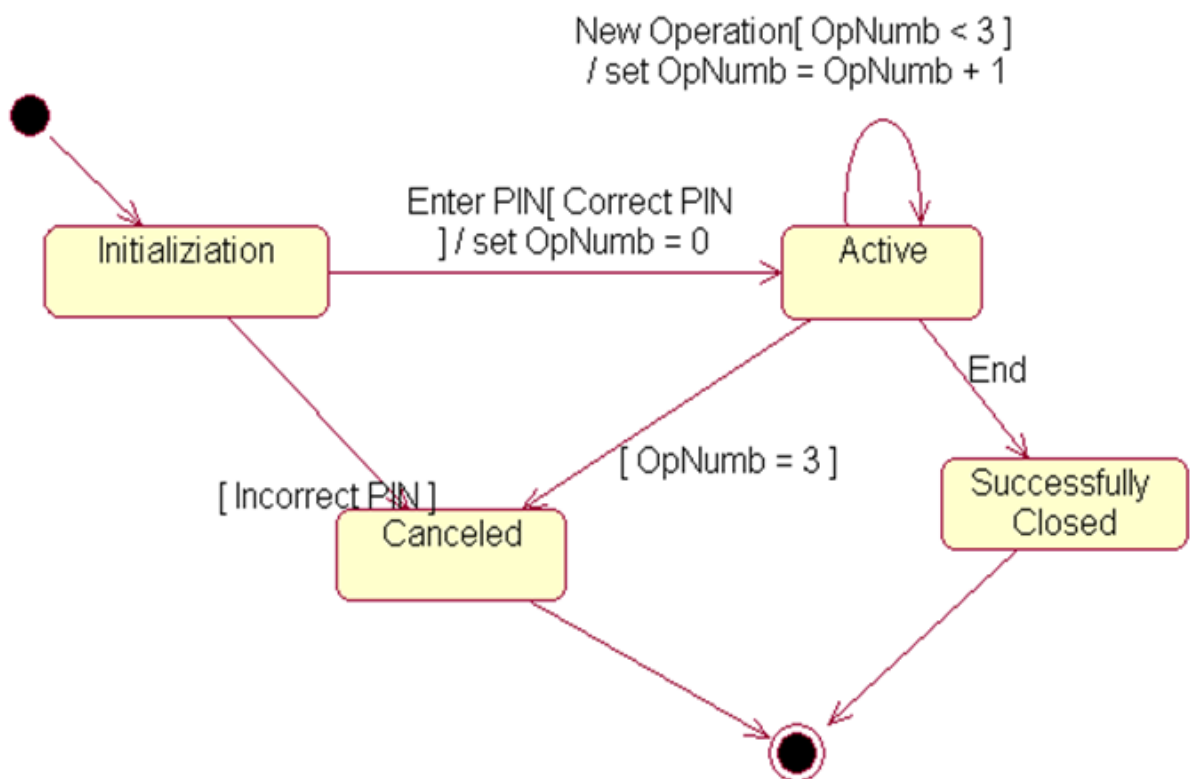
- Именува се събитията, които предизвикват промяната на състоянието.

- Проверява се дали всички състояния, изобразени на диаграмата са достижими при определени условия, т.е. няма състояние, което не може да се достигне. Ако има такова, то или не е стабилно състояние на елемента (и се изтрива от диаграмата) или не са определени всички промени на състоянията коректно (добавят се промените и действията, които ги предизвикват).

- Проверява се, че няма състояние, което да е крайна точка (с изключение на крайното състояние), т.е. всяко състояние трябва при определени събития да води до друго състояние.

- Проверява се логиката на модела.

Пример:



Първото състояние на обекта в примерната диаграма е Initialization. При въвеждане на некоректен PIN обекта преминава в състояние Canceled. При въвеждане на коректен PIN, обекта преминава в състояние на Active и остава в това състояние до 3 пъти при изпълнението на нова операция (New Operation), като при всяко изпълнение променливата OpNumb се увеличава с 1. От тук обекта може да премине в състояние Successfully Closed при избор на операция End или да попадне в състояние Canceled когато променливата OpNumb стане равна на 3.

Диаграма на дейностите

Целта на диаграмата на дейностите е да моделира последователността от действия, които описват определена функционалност на системата.

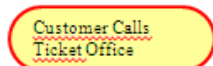
Основно диаграми на дейностите се използват за по-детайлно описание на use case-те, включени в use case модела на системата. Освен това диаграми на дейностите могат да се използват независимо от use case-ите за моделиране на функционалности на бизнес ниво. Този тип диаграми описват динамичния аспект на системата чрез последователност от действия.

Въпреки, че в UML диаграмите на последователността представят подобна информация като диаграмата на действията, основната разлика е, че диаграмата на дейностите е по-удобна за описание на процеси на бизнес ниво, поради две причини:

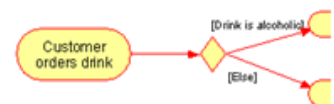
- Показва всички възможни развития на процеса (успешен и не успешен завършек) – докато при диаграмата на последователността основно се описва само успешния клон на развитие.
- По-лесно се приема от хората на бизнес ниво – този тип диаграма не е „техническа”, т.е. описва логиката на процесите, без да описва средствата или градивните елементи.

Символи в диаграмата:

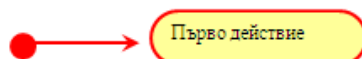
Действие



Вземане на решение



Начално състояние



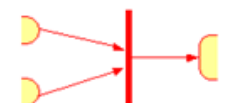
Синхронизирано състояние



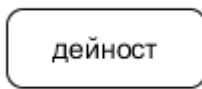
Крайно състояние



Асинхронно състояние

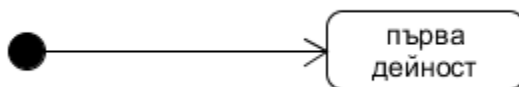


Условни знаци и тяхното значение в диаграмата на дейностите:



Дейност се представя като правоъгълник със заоблени краища, текстът в него описва самото действие.

Тъй като диаграмата на дейностите показва последователност от действия, тази последователност има начална точка и в UML тя се нарича начално състояние (initial state). Това е точката, от която се започва разчитането на последователността от дейности. Условният знак за начално състояние е запълнена точка със стрелка към първото действие в диаграмата.



Важно е да се отбележи, че в диаграмата на дейностите може да съществува само едно начално състояние и само една транзакционна стрелка може да го свързва с действие.

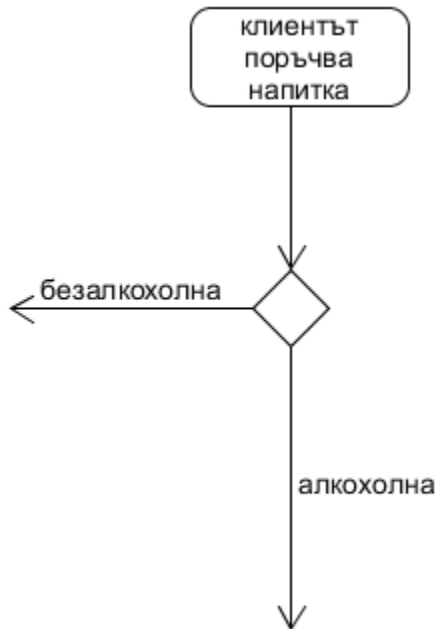
Транзакционната стрелка посочва посока в диаграмата на последователността на действията.

Последователността от действия приключва, когато транзакционната стрелка на последното действие се свърже със символа за крайно състояние (final state), който представлява запълнена точка оградена от кръгче.

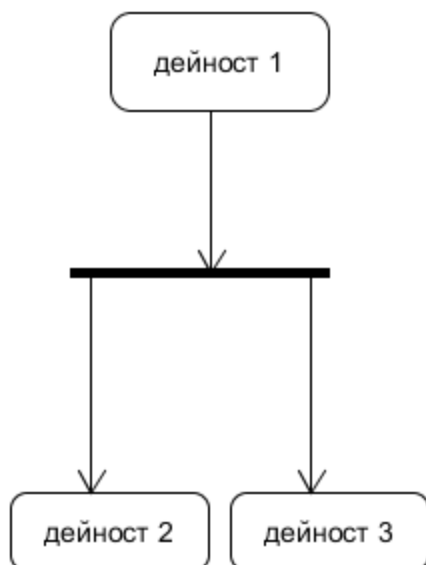


Всяка диаграма на дейностите трябва да има поне едно крайно състояние. В противен случай при разглеждане на тази диаграма може да се предположи, че тя не е завършена без изобразена на нея крайна точка. От друга страна, е възможно в една диаграма да има няколко крайни състояния. Това означава, че последователността от действия може да се прекъсне в няколко различни клона на логиката, т.е. описаната функционалност може да завърши по различни начини.

При визуализиране на последователност от дейности често се налага описването на вземане на решения, базиращи се на резултата от предходни действия. За такива случаи UML предлага символа Решение (decision point) – ромб, като от него излизат две транзакционни стрелки върху всяка, от които се описва кой клон на логиката представя.

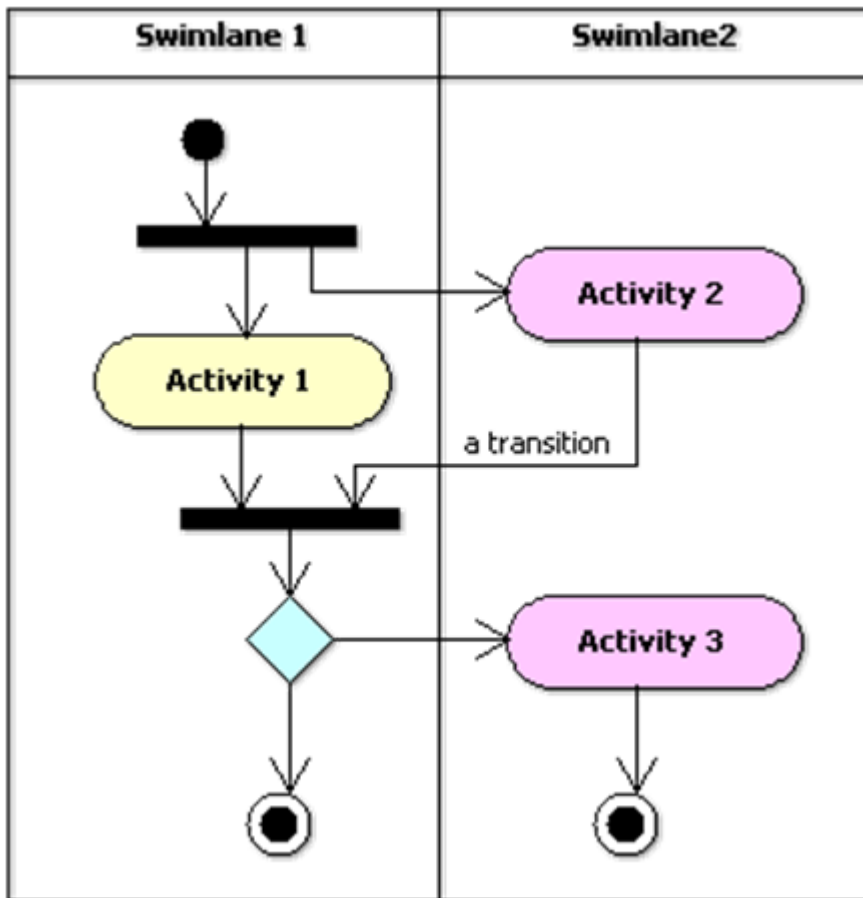


Понякога при описване на последователност от действия се налага да се посочи, че определени действия трябва да се извършат паралелно. За обозначаване на паралелни действия се използва Синхронизирано състояние – плътна линия, която показва че след нея две или повече дейности ще се извършат паралелно.



В диаграмата на дейностите е полезно да се определи не само последователността от действия, но и кой обект контролира и изпълнява съответното действие (банкова система, организацията, клиента и т.н.). За да се визуализира това, диаграмата се разделя на коридори (swimlanes) чрез вертикални линии. Всеки коридор има име, което е името на обекта, който контролира действие. За всеки обект, който изпълнява определено действие от диаграмата, трябва да съществува отделен коридор.

На следващата примерна диаграма на дейностите се илюстрира с нотацията fork (вилаца), че Activity 1 и Activity 2 са две дейности, които се изпълняват паралелно, като наличието на join (свържи) показва, че те взаимно изчакват приключването си за да се продължи процеса. Activity 3 произлиза от decision node (възел-решение), което показва, че е опционално в процеса.

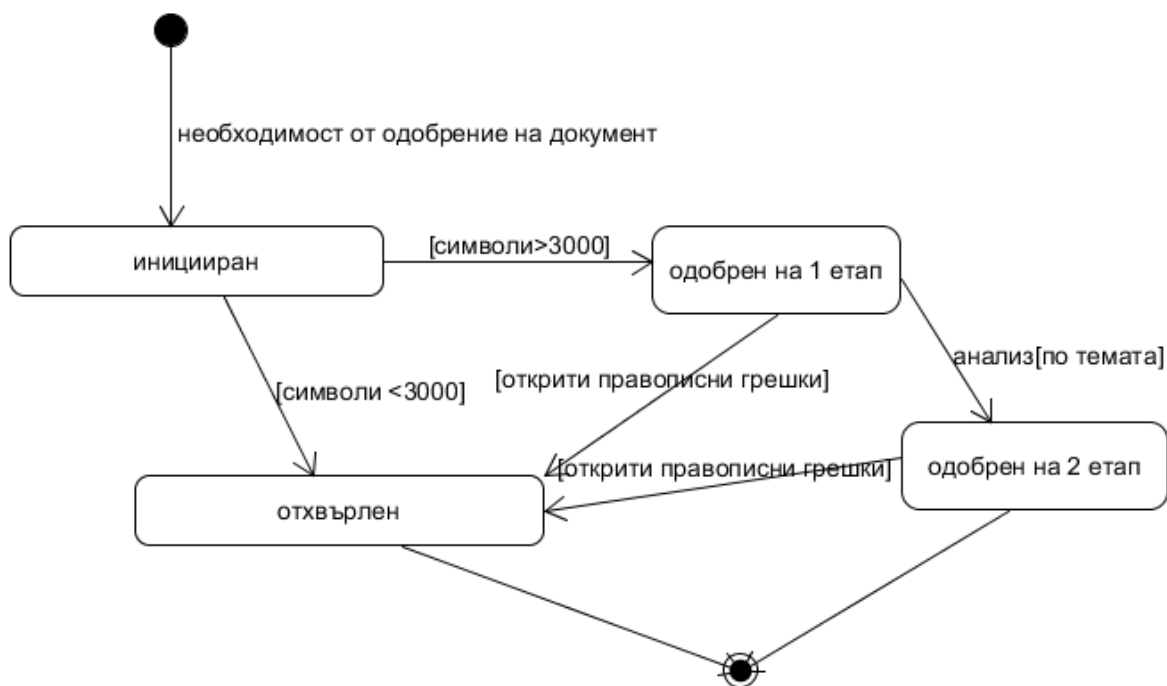


28. Модул 25: Практика: Диаграма на състоянията

- Изготвяне на диаграма на състоянията (State Machine Diagram)

28.1. Цел

Този модул следва въведението в поведенческите UML диаграми като го допълва със практическа работа по създаване на диаграми на състоянията по разглежданите до този момент казуси (напр. създаването на документооборотна информационна система).



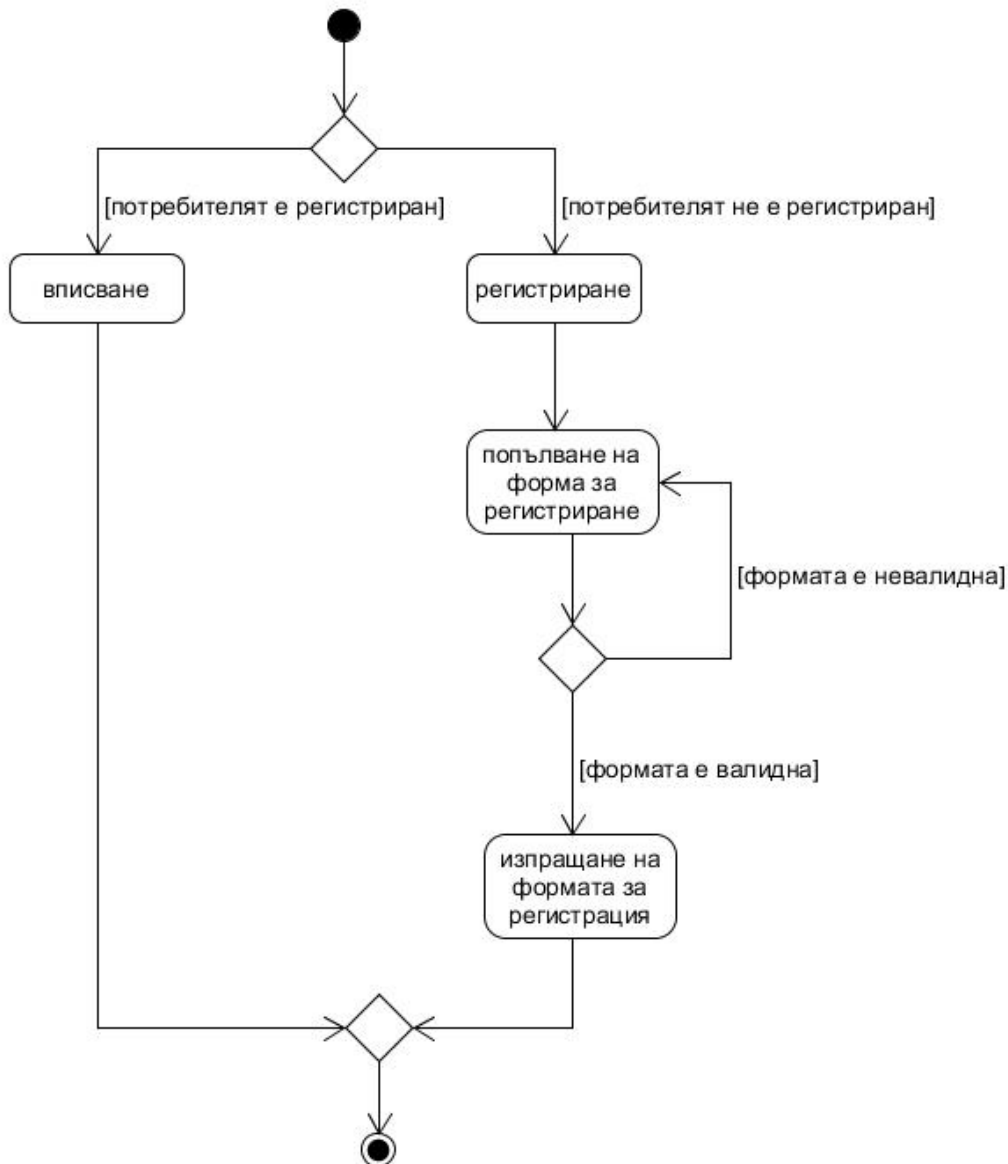
Обектът (документът) в примерната диаграма на състоянията може да съществува в 4 състояния: иницииран, одобрен на 1 етап, одобрен на 2 етап, отхвърлен. В квадратни скоби [] се отбелязва условието при което обекта преминава от едно състояние в друго.

29. Модул 26: Практика: Диаграма на дейностите

- Изготвяне на диаграма на дейностите (Activity diagram)

29.1. Цел

Практикуване на създаване на диаграми на дейностите по разглежданите до този момент казуси (напр. Процес по регистрацията и вписване).



Процесът започва с проверка дали потребителя е регистриран или не. Ако е регистриран следва вписване и край на процеса. Ако не е регистриран следва регистрация и попълване на форма за регистрация. Осъществява се проверка за валидация на формата. Ако е валидна се изпраща формата и с това приключва процеса, а ако не е валидна се връща процеса в попълване на формата.

30. Модул 27: Верифициране на изисквания

- Кога считаме изискванията са верифицирани?
- Peer Review
- Обобщение за заинтересованите страни
- Инспекция на изисквания

30.1. Цел

Този модул запознава с принципите на верификация на изисквания и техниките за извършването и. Разглеждат се характеристиките за качествени изисквания.

Верификацията на изискванията удостоверява, че са спазени всички одобрени стандарти за описание и моделиране на изискванията. Верификацията доказва, че изискванията са описани коректно и са с необходимото качество. Изискванията, които не отговарят на заложените качествени стандарти трябва да се преработят, с цел отстраняване на дефектите. След верифициране на изискванията, може да се счита, че те са готови за формален преглед и валидация от заинтересованите страни, и че са във вид, който е достатъчно добър за имплементиране на решението.

30.2. Характеристики на качествените изисквания

- cohesive (атомична) - да се отнасят за едно и също нещо, за един проблем, процес, организационна единица
- пълни - да включва всички изисквания, да не е пропуснато изискване
- консистентни - без противоречия в отделните изисквания
- коректни - верни изисквания
- изпълними - които могат да бъдат имплементирани
- модифицируеми - могат лесно да бъдат променени
- непротиворечиви - без възможност да бъдат интерпретирани по различен начин
- с възможност да бъдат тествани

30.3. Техники

- Peer Review - преглед и корекция на разработени изисквания от специалисти, които са колеги или равнопоставени с автора на изискванията.
- Обобщение за заинтересованите страни - преглед на обобщени или част от изискванията, които касаят конкретна група заинтересовани лица
- Инспекция на изискванията - подробен и задълбочен преглед на изискванията



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

- Checklist - техника с която се удостоверяват за изпълнени основни елементи в процеса на събиране на изисквания. Гарантира поне минимално покритие на качеството.
- Обследване на дефекти - идентифициране на слабости в изискванията, като всяко изискване се измерва с критериите за качествени изисквания (виж. Характеристика на качествените изисквания)
- Оценка на дефектите и слабостите - идентифицираните дефекти се оценяват относно значението и въздействието им върху целия проект. Включва се оценка за големината на дефекта, вероятността да се случи отново и възможността на организацията да не бъде пряко засегната от дефекта.
- Критерии за приемане (Acceptance and Evaluation Criteria Definition) - дефинира множеството изисквания, които трябва задължително да бъдат изпълнени за да се счита проекта за приемлив за заинтересованите лица.
- Metrics and Key Performance Indicators - дефинира метриците, които определят и измерват кои изисквания считаме за качествени.
- Прототипиране - използва се за удостоверяване, че са разбрани правилно изискванията.
- Риск анализ - относно риска, който носят изпълнението/неизпълнението или самата разработка/имплементация на група изисквания
- Structured Walkthrough (Цялостен преглед) - групова техника при която се прочитат (излседват) всички налични изисквания едно след друго.
- Одобрение на изискванията - техника при която се постига консенсус относно приоритетите на изискванията.

Например Peer Review е подходяща техника, с която да проверим с наш колега, доколко изискванията/диаграмите, които сме изготвили по време на анализа са ясни. Това е начин да осигурим качеството на нашия документ.

31. Модул 28: Валидиране на изисквания

- Обследване на дефекти
- Оценка на дефектите и слабостите
- Критерии за приемане
- Одобрение на изискванията

31.1. Цел

Този модул запознава с принципите на валидация на изисквания и техниките за извършването .

Валидирането на изискванията удостоверява, че изискванията носят полза на организацията, изпълняват целите си и са в синхрон с нуждите на засегнатите лица. Валидацията е постоянен процес, който уверява, че изискванията на заинтересованите лица са в синхрон с бизнес изискванията. В случай, че някои заинтересовани лица имат конфликтни изисквания, бизнес анализатора трябва да се опита да разреши конфликтните точки.

31.2. Техники

- Обследване на дефекти - идентифициране на слабости в изискванията, като всяко изискване се измерва с критериите за качествени изисквания (виж. Характеристика на качествените изисквания)
- Оценка на дефектите и слабостите - идентифицираните дефекти се оценяват относно значението и въздействието им върху целия проект. Включва се оценка за големината на дефекта, вероятността да се случи отново и възможността на организацията да не бъде пряко засегната от дефекта.
- Критерии за приемане (Acceptance and Evaluation Criteria Definition) - дефинира множеството изисквания, които трябва задължително да бъдат изпълнени за да се счита проекта за приемлив за заинтересованите лица.
- Metrics and Key Performance Indicators (Метрики и индикатори за представяне) - дефинира метриците, които определят и измерват кои изисквания считаме за качествени.
- Прототипиране - използва се за удостоверяване, че са разбрани правилно изискванията.
- Риск анализ - относно риска, който носят изпълнението/неизпълнението или самата разработка/имплементация на група изисквания
- Structured Walkthrough (Цялостен преглед) - групова техника при която се прочитат (изследват) всички налични изисквания едно след друго.
- Одобрение на изискванията - техника при която се постига консенсус относно приоритетите на изискванията.

Пример

Бизнес анализатора трябва винаги да може да свърже всяко едно изискване с бизнес целта, която те покриват. Пример със системата за управление на документи:

Бизнес цел	Изискване
Да се облекчи административната работа на служителите	Изискване номер 24 - Управление на преписки и дела

Да се повиши капацитета на организацията	Всички изисквания (1 до 26)
Да се повиши отчетността на работните процеси в структурата на BA Bank	Изискване номер 26 - Търсене и справки

32. Модул 29 Комуникация на изисквания

- Защо да комуникираме изискванията
- Нива на комуникация
- Модели и методи за комуникация (Structured Walkthrough)
- Семинари за изисквания

32.1. Цел

Този модул запознава с моделите и методите на комуникация на изисквания, нивата на комуникация и най-популярните техники за постигане одобрение на изискванията.

Цел на комуникацията

- Да се осигури, че всички изисквания са събрани
- Да се осигури, че всички ненужни изисквания са извън обхвата
- Да верифицира, че изискванията са пълни, консистентни, специфични, недвусмислени
- Да валидира, че изискванията покриват целта на проекта
- Да верифицира, че изискванията са постижими, реалистични и приоритизирани
- Да намали риска на проекта относно цената, времето на разработка, както и евентуалния провал
- Да осигури ниво на отговорност, собственост и приемане на изискванията

32.2. Базов модел за комуникация:

- encode - кодиране на съобщението
- transmit - транспортиране на съобщението
- decode - разкодиране на съобщението
- acknowledge - маркиране на прието съобщение
- feedback - обратна връзка относно полученото съобщение



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

32.3. Методи на комуникация:

- push - разпространение на съобщение без обратна връзка
- pull - достъп до информация по желание на получателя
- interactive - интерактивна комуникация

32.4. Нива на комуникация:

- Peer review (Преглед от колега)
 - Неформална сесия за преглед на изискванията
 - Отбелязват се изискванията, които имат нужда от изясняване, уточняване и подобрене
 - Конструктивна критика - персонална обида
- Structured walkthrough (цялостен преглед)
 - Формална сесия с потребители и заинтересовани страни
 - Обратна връзка за всяко едно от изискванията
 - Може да са необходими повече от една сесия, толкова колкото са необходими
 - Готовност за промени, допълнения и/или елиминиране на изисквания
- Formal Requirements Inspection (Семинари за изискванията)
 - Внимателно се подбира съставът - ключови потребители и заинтересовани лица
 - Предварително изпращане на документацията - уверете се, че са я получили и прочели
 - Фасилитиране - назначете си фасилитатор
 - Протоколиране - скрайбер, “parking lot”, “follow-up”
- Sign off (Подписване) – одобрение
- Sign Off (Подписване)
 - Формалното одобрение на изискванията от оторизирани лица, които позволяват последващото развитие на проекта
 - отбелязва, че изискванията са прегледани и одобрени
 - изискванията са с нужното качество
 - всички проблеми, неясноти и пропуски са отстранени
- Baseline (Базова линия)
 - Одобрена версия на изискванията
 - Служи като основа за имплементиране и тестване

Изискванията се променят единствено, след като са минали формален процес по одобрение

Постигане на одобрение на изискванията в стъпки:

1. Използвайте резултата от анализа ви - изисквания, use cases, диаграми
2. Изберете начин за комуникиране на изискванията (Peer Review, Structured Walkthrough, Facilitated session) - приложете някоя от изброените техники
3. Целта е да получите формално одобрение - подпис от съответните stakeholders, с което те потвърждават, че изискванията са одобрени от тях

33. Модул 30 Управление на изисквания

- Как изискванията стават неуправляеми
- Проследяване
- Контрол на промяната и политики за контрол на промяната
- ССМ инструменти

33.1. Цел

Този модул запознава с техники за управление и проследяване на променящи се изисквания. Предоставя инструменти за контрол и управление на промяната.

Изискванията не са статични - те се променят и тази промяна трябва да се управлява!

Защо се променят изискванията:

- Променя се бизнес средата
- Промяна или нови закони и регулация
- Борба с конкуренцията
- Желания на заинтересованите страни

Без управление на изискванията, хаосът е сигурен!

Потребители и клиенти



директен диалог между клиенти и
разработчици



Програмисти



добавяне/промяна на
изисквания от различни
източници (понякога
неоторизирани)



добавяне/промяна на
изисквания от
разработчиците

Включването на бизнес анализатор в процеса на комуникация между възложителите (потребители, клиенти) и изпълнителите (програмисти, тест специалисти и други) въвежда строга организация и ред в диалога между засегнатите страни.

Бизнес анализаторът може да инициира или да ръководи процеса на взимане на решения относно промяната на основни компоненти на проектите (обхват, време и цена), като той/тя може да бъдат част от ключовите заинтересовани страни, които взимат решения, а може и да бъде само фасилитатор на групата, отговорна за поемане на решения.



33.2. Проследяване

Проследяване на изискванията е техника в бизнес анализа, която се използва за да предпази обхвата на проекта от scope creep, и да не се събират изисквания, които не са свързани с целта на проекта/инициативата.

Проследяването установява и поддържа връзки (relationships) между бизнес целите, изискванията, компоненти на решението и други резултати от проекта. Изискванията могат да се свържат с други изисквания, до компоненти от разработката/имплементацията и до тестовите сценарии, които верифицират изискванията. Проследимостта е възможността да се идентифицира лесно тази връзка и зависимост. С помощта на проследимостта, лесно може да се установи дали всяко изискване е свързано с конкретна бизнес цел, къде се имплементира и дали има имплементирана функционалност, за която не могат да се открият изисквания.

- трябва да може да се направи проследяване от горе-надолу - от целите, нуждите и изискванията до сорс-кода и тестовете
- и обратно, от тестовете до това, коя цел изпълняват те



33.3. Контрол на промените

Промяната само по себе си не е проблем, а проблем може да бъде неконтролируемата промяна!

За това проектите трябва да имат:

- отговорник по промените (Change control board)
- процес по промените (роли, процедури)

Контролата по промените зависи силно от типа и жизнения цикъл на проекта.

Инструментите за контрол ССМ (Configuration and Change Management Tools) трябва:

- да осигуряват достъп до информация по проекта, само на упълномощени лица
- да дават възможност за промени по документите, само на упълномощени лица
- да пазят версии на документите
- да предоставят на мениджърите статус на задачите по проекта

Пример:



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

Защо е полезно да управляваме изискванията и да поддържаме проследимост?

В хода на тестване на проекта се установява невъзможността да се репродуцират няколко тест сценария. Техниката Проследяване на изисквания ще ни даде отговор за това, кои кореспондиращи изисквания са засегнати от невъзможността да се репродуцират тест сценариите, както и целите, които евентуално няма да могат да бъдат постигнати вследствие на неизпълнението на изискванията. В допълнение може да се проследи и съответния source code или модул който е засегнат.

34. Приложение

I. Документ - Визия

В Vision документа са попълнени само релевантните точки от стандартния темплейт, който се разглежда в модул 6 - Бизнес анализ документи)

1. Въведение

Този документ включва текущите бизнес проблемите на VA Bank и възможни алтернативи за решаването им

1.1. Обхват

Системата трябва да поддържа следния набор от функции:

- о Потока(създаване, регистриране, редактиране, изпращане, подписване) от документи (входящи, изходящи и вътрешни) и оперативна информация (отчети, заявки, становища, справки, писма, резолюции, заповеди и др.)

- о Движение на документи вътре в организацията във връзка с изпълняваните дейности

- о Задачите и контрола по тяхното изпълнение

- о Данни за потребителите на системата съобразно тяхната роля в структурата на организацията

- о Данни за всички външни лица и организации, с които потребителите кореспондират

засяга	служителите на банката и нейните клиенти
като последствията са	<ul style="list-style-type: none">● недоволни клиенти,● липса на отчетност,● пропуснати възможности
и успешно решение би било	<ul style="list-style-type: none">● единна система за документооборот и разпределяне на задачи

3. Засенати страни

3.1. Профили

Роля	Представител	Отговорности
<i>Клиент</i>	<i>Георги Петров</i>	<i>Одобрява бюджета и обхвата на проект</i>
<i>Потребители</i>	<i>Ана Велкова</i>	<i>Представя интересите на потребителите на системата.</i>
<i>ИТ експерти</i>	<i>Димитър Станчев</i>	<i>Гарантира разработката на системата да отговаря на уточнените изисквания и да спазва добрите практики за разработка</i>

Изисквания към средата

Учебни материали и обучения

- o Изготвени и предоставени ръководства за работа със системата
- o Проведено обучение на 200 потребители на системата

Хардуерни ресурси

Хардуерни ресурси за системата със следната минимална конфигурация:

- 4 бр. Сървър по 4 ТВ и минимум 5 мрежови карти
- 2 бр. Сторидж устройства
- 2 бр. Устройства за обмяна (exchanger)
- 2 бр. Суича
- 2 бр. Рутер
- 2 бр. Рак-шкафа – 19 инча съобразени с броя на изделията софтуерен продукт за документооборот, потребители, ИТ специалисти, администратори



Европейски съюз



ОПАК. Експерти в действие



Европейски социален фонд
Инвестиции в хората

35. Източници

1. Agile Manifesto, Agile Alliance
2. [BABOK v.2.0, IIBA](#)
3. BPMN, OMG
4. RUP, Rational Software White Paper
5. Software Requirements, Wiegers
6. UML Distilled, Third Edition, Addison Wesley