



Изисквания

Изисквания към
разработката на
софтуерни продукти



Теми

- Видове изисквания
 - Системни и потребителски
 - Функционални и нефункционални
 - Организация на изискванията в документи
- Техники за събиране на изискванията

Обхват на проекта

- Управление на обхвата на проекта
 - ▣ дефиниране и контролиране на това, което се включва в проекта
- Събиране на изискванията
- Дефиниране на обхвата
- Създаване на WBS

Определяне на изискванията

- ◆ Определяне на изискванията (requirements engineering) – процес на установяване на
 - услугите, които потребителите изискват от с-та;
 - ограниченията при които системата работи и/или се разработва.

Requirements



**The descriptions of the
system services and
constraints**

that are generated during the
requirements engineering
process

Дефиниране на изисквания

- Изискване към разработката на софтуерен продукт - варира от абстрактно описание (описание на високо ниво) на услуга или ограничение на системата до детайлна спецификация на определена функция.
 - Може да бъде оферта за бъдещ договор – тогава трябва да бъде отворено за интерпретация
 - Може да бъде основа за самия договор – тогава трябва да бъде дефинирано в детайли

Потребителски и системни изисквания

■ Потребителски изисквания

- Изрази на естествен език в комбинация с диаграми. Те описват какви услуги се очаква да предоставя системата и ограниченията, с които трябва да се съобразява. Изготвят се за потребителите

■ Системни изисквания

- Структуриран документ, който установява детайлно описание на функциите и услугите, които предоставя системата, както и на оперативните ограничения. Дефинира какво трябва да се разработи. Може да е част от договора между клиента и договарящия.

Функционални и нефункционални изисквания

- **Функционални изисквания**

- Набор от услуги, които системата трябва да предоставя; как ще реагира на особени входни данни и как ще се държи при по-особени ситуации.

- **Нефункционални изисквания**

- Ограничения върху услугите и функционалността, предлагани от системата - за времето, върху процеса на разработка, стандарти и др.

- **Изисквания на областта**

- Произтичат от приложната област на системата и отразяват характеристики на тази област.

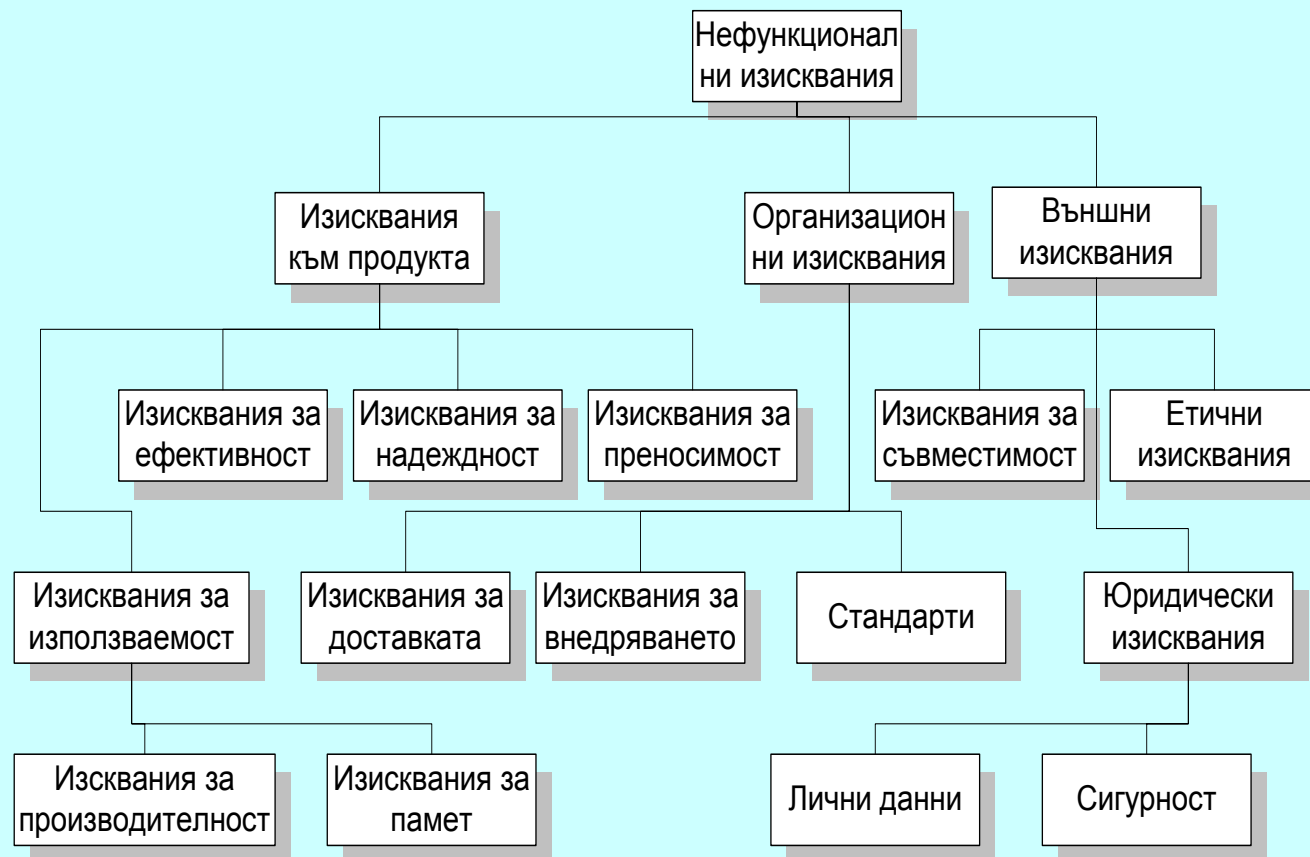
Нефункционални изисквания

- Дефинират системни характеристики и ограничения:
 - характеристики - надеждност, време за отговор или необходима памет.
 - ограничения върху системата, като възможности на устройствата за вход/изход или начина за представяне на данните.
- Ненфункционалните изисквания могат да бъдат по-критични от функционалните.
- При описанието им може да се използват CASE средства, програмен език, метод за разработка.

Нефункционални изисквания - видове

- **Изисквания към продукта**
 - Определят специфично поведение на продукта - изисквания за производителност; бързина, памет; сигурност; приемлива степен на неуспех; преносимост и използваемост.
- **Организационни изисквания**
 - Произтичат от политиките на организациите на клиентите и разработчиците - стандарти за процеси, които трябва да се спазват; изисквания, свързани с реализацията – език за програмиране, определен метод за проектиране; изисквания за предаване.
- **Външни изисквания**
 - Произлизат от фактори, които са външни за с-та и нейната разработка. Определят как с-та трябва да си взаимодейства със с-те на другите организации; законодателни изисквания и т.н.

Типове нефункционални изисквания



Цели и изисквания

- Нефункционалните изисквания могат да бъдат трудни за дефиниране и проверяване
- Цел
 - “Системата трябва да бъде лесна за използване”
- Проверимост на нефункционалните изисквания

Въвеждане на метрики

Свойство	Мярка
Скорост	Брой транзакции / секунда Потребител / събитие време за отговор Период за обновяване на екрана
Преносимост	Брой целеви системи
Размер	Kbytes, Mbytes, GBytes Брой RAM чипове
Използване	Времето за обучение Наличие на „Помощ“ и колко е подробна
Надеждност	Вероятност от провал Наличност Процентно отношение на провали
Устойчивост	Време за рестартиране след провал Процент на събитията причиняващи провал Вероятност от повреда на данните при провала

Взаимодействие м/у изискванията

- Конфликт между нефункционални изисквания и функционални изисквания.
- Пример: Изискване за максималната памет за системата - не по-голяма от 4MB, а друго изискване определя, че при разработката трябва да се използва Ada.
 - Не е възможно програма, която е написана на Ada да се вмести в 4MB. Трябва да се направи някакъв компромис между двете изисквания - да се използва друг език за програмиране или да се увеличи паметта.

Изисквания за областта

- Произлизат от приложната област, а не от нуждите на потребителите.
- Могат да бъдат:
 - Нови функционални изисквания
 - Ограничения върху съществуващи вече функционални изисквания
 - Указания как трябва да се извършат определени операции

Проблеми при изискванията на областта

- Разбираемост

- Изискванията се описват на езика на приложната област
- Това често не се разбира от софтуерните инженери, разработващи системата.

- Неявност

- Специалистите в областта я разбират много добре и не се сещат да изразят явно изискванията на областта.

Потребителски изисквания

- Потребителските изисквания за системата описват функционалните и нефункционалните изисквания по начин разбираем за потребителите, които нямат техническо познание. Те определят само външното поведение и избягват детайлно описание на системните характеристики.
- Дефинират се с помощта на естествен език в комбинация с прости форми и диаграми, а не с модели за реализация .

Проблеми при използване на естествен език

- Липса на яснота
 - Трудно се използва естествен език за точно и недвусмислено описание на изисквания, без документите да станат многословни и тежки за четене.
- Объркани изисквания
 - Функционалните и нефункционалните изисквания, целите на системата и информацията за дизайна може да не са добре разграничени.
- Обединяване на изисквания
 - Няколко различни изисквания могат да бъдат описани заедно в един документ.

Системни изисквания

- Дават по-подробни спецификации на системните функции, услуги и ограничения отколкото потребителските изисквания
- Те са предназначени да бъдат основа за проектирането на системата.
- Могат да бъдат включени в системния контракт
- Могат да бъдат дефинирани или илюстрирани като се използват системни модели.

Проблеми със спецификациите на естествен език

- Двусмислие
- Свърх гъвкавост
 - Едно и също нещо може да бъде казано по много различни начини.
- Липса на структура
 - Естествените езици са неадекватни за структуриране на ситемните изисквания

Алтернативи на естествения език

Означения	Описание
Структуриран естествен език	Този подход зависи от дефиниране на стандартни формуляри или модели за изразяване на спецификацията на изискванията.
Езици за описание на проект	Този подход използва език като програмен език, но с повече абстрактни свойства чрез дефиниране на операционен модел на системата. Този подход не се използва широко, макар че би бил полезен за спецификация на интерфейс.
Графични означения	Използва се графичен език, допълнен с текстуални анотации, за да дефинира функционалните изисквания за системата. Ранен пример за такъв език е SADT. Сега се използват диаграми на използване и на последователностите.
Математически означения	Това са означения основани на математически понятия такива като крайни автомати или мрежи.. Тези недвусмислени спецификации намаляват споровете м/у клиент и разработчик по въпроса за системната функционалност. Обаче много клиенти не разбират формалните спецификации и не са склонни да ги приемат като контракт за системата.

IEEE стандарт за изискванията

- IEEE/ANSI 830-1998
- Дефинира основната структура на документа за изискванията.
 - Въведение.
 - Основно описание.
 - Специфични изисквания.
 - Приложения.
 - Индекс.



Структура на техническото задание

- Предговор
- Въведение
- Глосарий
- Дефиниция на потребителските изисквания
- Архитектура на системата
- Спецификация на системните изисквания
- Модели на системата
- Развитие на системата
- Приложения
- Индекс

Requirements Document

Table of Contents

1. Project Preliminaries

- 1.1 Purpose and Scope of the Product
- 1.2 Business Context
- 1.3 Stakeholders
- 1.4 Ideas for Solutions
- 1.5 Document Overview

2. System Services

- 2.1 The Scope of the System
- 2.2 Function Requirements
- 2.3 Data Requirements

3. System Constraints

- 3.1 Interface Requirements
- 3.2 Performance Requirements
- 3.3 Security Requirements
- 3.4 Operational Requirements
- 3.5 Political and Legal Requirements
- 3.6 Other Constraints

4. Project Matters

- 4.1 Open Issues
- 4.2 Preliminary Schedule
- 4.3 Preliminary Budget

Appendices

- Glossary
- Business Documents and Forms
- References

Обобщение

- Изискванията показват какво системата трябва да прави и дефинират ограниченията върху нейната работа и разработка
- Функционалните ограничения показват услугите, които системата трябва да осигури.
- Нефункционалните ограничения ограничават системата или процеса на разработка.
- IEEE стандартът е полезна отправна точка за дефинирането на по-подробни специфични стандарти за изискванията.

Традиционни методи за определяне на изискванията

- Видове традиционни методи
 - Интервюта с клиенти и експерти в областта
 - Въпросници
 - Наблюдение
 - Разучаване (преглеждане) на документация и налични софтуерни системи
- Използваемост
 - Прости и ефективни
 - При добре дефинирани проекти и ниски нива на риск

Интервю

- Чрез **формални и неформални интервюта** екипът за определяне на изискванията поставя въпроси към заинтересованите лица (stakeholders) относно с-та, която се използва и тази, която ще се разработва.
- Видове интервюта:
 - **Затворено интервю** – отговори на предварително дефинирани въпроси
 - **Отворено интервю** – няма предварително дефинирани въпроси; обсъждат се въпроси, повдигнати от заинтересуваните лица.

Практически насоки

- Прилага се смесица от отворени и затворени интервюта.
- Интервютата са подходящи за цялостно разбиране на дейността на с-та
- Интервютата не са добри при определяне на изискванията на областта (domain requirements)
- Интервюиращите трябва:
 - Да бъдат толерантни, да изслушват, да не са предубедени
 - Да подпомагат интервюираните с подходящи въпроси, а не просто да очакват отговор на въпроси като 'Какво искате от нас?'.

Интервюиране на клиенти и домейн експерти

- С клиенти – за определяне на use case изисквания
- С домейн експерти – директна обмяна на знания
- Структурирано (формално) интервю
 - отворени въпроси
 - затворени въпроси
- Неструктурирано (формално) интервю
- Неприемливи въпроси
 - Догматични (do we have to do things the way we do them?)
 - Предубедени (you are not going to do this, are you?)
 - Внушаващи (you do things this way, don't you?)
- Summary report (с коментарии) от интервюто се изпраща 1-2 дни след провеждането му

Въпросници

- Допълнения към интервюта (Questionnaires)
- Пасивна техника
 - (+) – време за разглеждане на въпросите
 - (-) – невъзможност за пълно изчистване на въпросите и отговорите
 - Кои хора не отговарят? Защо?
- Затворени въпроси
 - Multiple-choice (могат да се добавят допълнителни коментари)
 - Рейтинг на отговорите на въпросите (напълно съгласен, съгласен,....)
 - Степенуване на въпросите (номериране, %)

Наблюдение

- Допълнения към интервютата (и въпросниците)
- Подходящи, когато потребителите не могат да предоставят достатъчно информация и/или имат само частични познания
- **Форми**
 - Пасивно
 - без прекъсвания и директна намеса
 - видео камера
 - Активно
 - Обясняващо
- Провежда се продължителен период по различно време и на различни работни места

Разучаване на документация и налични софтуерни системи

- Прилага се винаги (понякога само върху част от системата)
 - Use case изисквания
 - Организационни документи
 - бизнес форми, процедури, политики, планове, диаграми, кореспонденция
 - Системни форми и отчети (System forms and reports) ако има предишна система
 - Записи от заявките за промяна (change requests) дефекти и усъвършенствания
 - Знания за предметната област
 - Документация, книги, Интернет

Модерни методи за определяне на изискванията

- Предлагат по-добро вникване при по-висока цена и повече усилия
- Подходящи за проекти с висока степен на риск (неясни цели, недокументирани процедури, нестабилни изисквания, неопитни разработчици, недостатъчна ангажираност на клиента)
- Видове:
 - Прототипиране
 - Brainstorming
 - Joint Application Design (JAD)

Прототипиране

- Прототип
 - Прототипът представлява модел на част от системата, който се представя пред клиентите. Има за цел да стимулира тяхното мислене, да подобри диалога, да разбере повече за проблема и неговото разрешаване.
- Видове прототипи:
 - Изследователски (*Throw-away Prototype*) & Еволюционен (*Evolutionary Prototype*)
 - Хоризонтален и вертикален
- Подходящ при комплексни и иновативни проекти

Brainstorming

- Групова техника, основана на метода на свободното асоцииране, чиято цел е откриването на нови и оригинални идеи върху определен проблем.
- Формиране на нови идеи и намиране на решение на специфичен проблем чрез избягване на критицизма. Всяка идея трябва да бъде приета и разгледана внимателно.
- Достигане на консенсус между участниците
- 12 - 20 участника, разположени около кръгла маса
 - равнопоставеност
 - водещ (facilitator) – един от всички

Brainstorming

- Провеждане:
 - Преди срещата водещият (facilitator) представя **probortunity statement** (the problem/opportunity area)
 - Дават се писмени отговори на trigger въпроси, които преминават последователно през участниците
 - Отговорите/идеите са анонимни
 - Отговорите/идеите се дискутират
 - Гласуване на отговорите/идеите
 - Консенсус между stakeholders

JAD – Joint Application Design

- Техника, подобна на Brainstorming, която акцентира на груповата динамика
- Групата увеличава продуктивността, елиминира повече грешки, взема решения с по-малък риск, интегрира хората
- Участници
 - Лидер (комуникативни умения, познания за областта, не е stakeholder)
 - Секретар (бележки, CASE tool, софтуерни умения)
 - Клиенти (Потребители & Мениджъри)
 - Разработчици