



Software Engineering -1

Dr. Sameh Zarif

Lecture 6



Chapter 4

Requirements Engineering

Topics covered



- ✧ Introduction
- ✧ Functional and non-functional requirements
- ✧ Requirements engineering processes
- ✧ Requirements elicitation
- ✧ Requirements specification
- ✧ Requirements validation
- ✧ Requirements change

Requirements engineering



- ✧ The process of establishing the **services** that customer requires from a system and the **constraints** under which it operates and is developed.
- ✧ The system requirements are the descriptions of the **system services** and **constraints** that are generated during the requirements engineering process.

Requirements engineeringwhy?



“ Without requirements or design, programming is the art of adding bugs to an empty text file. ” - Louis Srygley

SOFTWARE TESTING
without requirements

What is a requirement?



- ✧ It may **range** from a **high-level abstract** statement of a service or of a system constraint to a **detailed mathematical** functional specification.
- ✧ Requirements may serve a dual function
 - May be the basis for a bid for a contract - therefore must be open to interpretation;
 - May be the basis for the contract itself - therefore must be defined in detail;
 - Both these statements may be called requirements.

Types of requirement



✧ User requirements

- Statements in natural language plus diagrams of the services the system provides and its operational constraints. Written for customers.

✧ System requirements

- A structured document setting out detailed descriptions of the system's functions, services and operational constraints. Defines what should be implemented so may be part of a contract between client and contractor.

User and system requirements



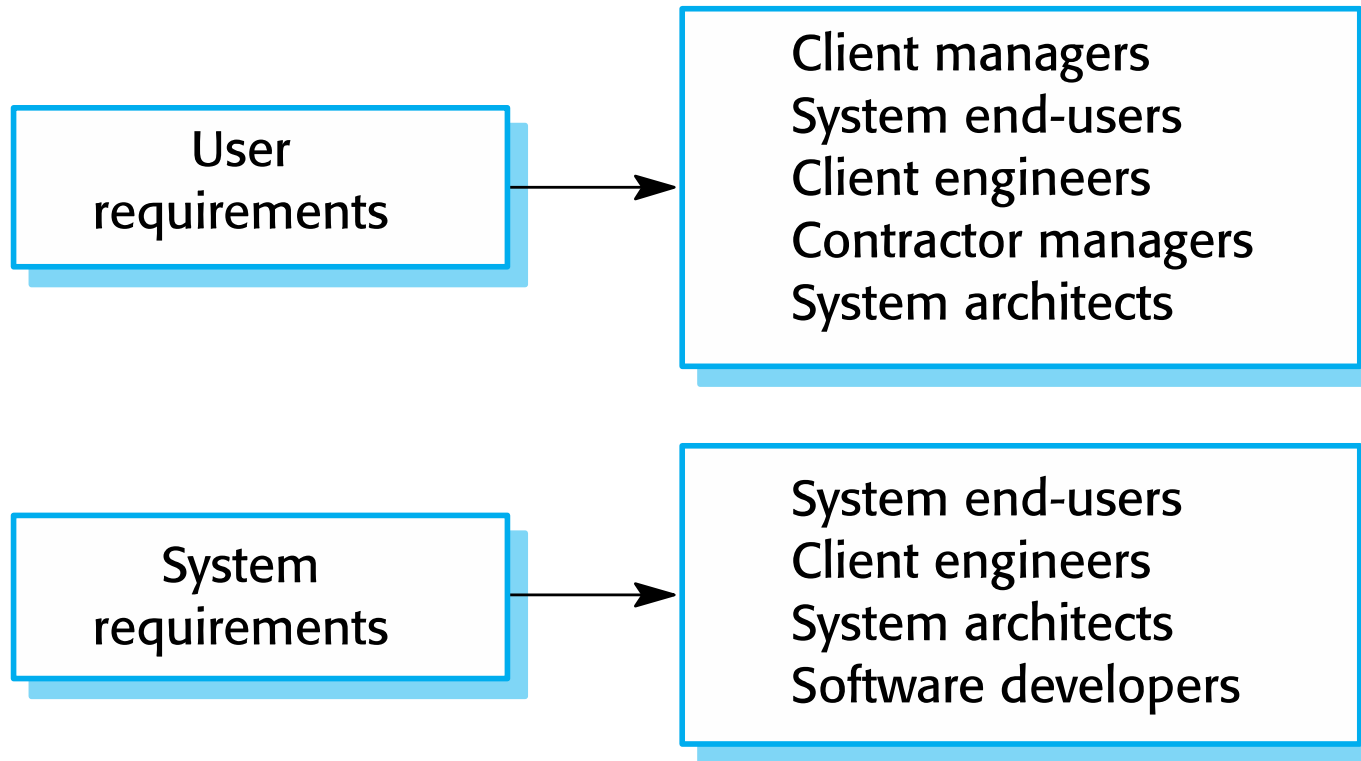
User requirements definition

- 1.** The Mentcare system shall generate monthly management reports showing the cost of drugs prescribed by each clinic during that month.

System requirements specification

- 1.1** On the last working day of each month, a summary of the drugs prescribed, their cost and the prescribing clinics shall be generated.
- 1.2** The system shall generate the report for printing after 17.30 on the last working day of the month.
- 1.3** A report shall be created for each clinic and shall list the individual drug names, the total number of prescriptions, the number of doses prescribed and the total cost of the prescribed drugs.
- 1.4** If drugs are available in different dose units (e.g. 10mg, 20mg, etc) separate reports shall be created for each dose unit.
- 1.5** Access to drug cost reports shall be restricted to authorized users as listed on a management access control list.

Readers of different types of requirements specification



System stakeholders



- ✧ Any person or organization who is affected by the system in some way and so who has a legitimate interest
- ✧ Stakeholder types
 - End users
 - System managers
 - System owners
 - External stakeholders

Stakeholders in the Mentcare system



- ✧ **Patients** whose information is recorded in the system.
- ✧ **Doctors** who are responsible for assessing and treating patients.
- ✧ **Nurses** who coordinate the consultations with doctors and administer some treatments.
- ✧ **Medical receptionists** who manage patients' appointments.
- ✧ **IT staff** who are responsible for installing and maintaining the system.



Functional and non-functional requirements

Functional and non-functional requirements



✧ Functional requirements

- Statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.
- May state what the system **should not do**.

✧ Non-functional requirements

- Constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, etc.
- Often apply to the system **as a whole** rather than individual features or services.

Functional requirements



- ✧ Describe functionality or system services.
- ✧ Depend on the type of software, expected users and the type of system where the software is used.
- ✧ Functional user requirements may be high-level statements of what the system should do.
- ✧ Functional system requirements should describe the system services in detail.

Mentcare system: functional requirements



- ✧ A user shall be able to search the appointments lists for all clinics.
- ✧ The system shall generate each day, for each clinic, a list of patients who are expected to attend appointments that day.
- ✧ Each staff member using the system shall be uniquely identified by his or her 8-digit employee number.

Requirements imprecision



- ✧ Problems arise when functional requirements are not precisely stated.
- ✧ Ambiguous requirements may be interpreted in different ways by developers and users.
- ✧ Consider the term 'search' in requirement 1
 - User intention – search for a patient name across all appointments in all clinics;
 - Developer interpretation – search for a patient name in an individual clinic. User chooses clinic then search.

Requirements completeness and consistency



- ✧ Requirements should be both complete and consistent.
- ✧ Complete
 - They should include descriptions of all facilities required.
- ✧ Consistent
 - There should be no conflicts in the descriptions of the system facilities.
- ✧ In practice, because of system and environmental complexity, it is impossible to produce a complete and consistent requirements document.

Non-functional requirements



- ✧ These define system properties and constraints e.g. reliability, response time and storage requirements. Constraints are I/O device capability, system representations, etc.
- ✧ May also be specified a particular programming language or development method.
- ✧ Non-functional requirements may be **more critical** than functional requirements. If these are not met, the system may be useless.

Non-functional classifications



✧ Product requirements

- Requirements which specify that the delivered product must behave in a particular way e.g. execution speed, reliability, etc.

✧ Organisational requirements

- Requirements which are a consequence of organisational **policies** and procedures e.g. process standards used, implementation requirements, etc.

✧ External requirements

- Requirements which arise from **factors** which are **external** to the system and its development process e.g. the system is approved for use, the system operates within the law,

Examples of nonfunctional requirements in the Mentcare system



Product requirement

The Mentcare system shall be available to all clinics during normal working hours (Mon–Fri, 0830–17.30). Downtime within normal working hours shall not exceed five seconds in any one day.

Organizational requirement

Users of the Mentcare system shall authenticate themselves using their health authority identity card.

External requirement

The system shall implement patient privacy provisions as set out in HStan-03-2006-priv.

Goals and requirements



- ✧ Non-functional requirements may be very difficult to state precisely and imprecise requirements may be difficult to verify.
- ✧ Goal
 - A general aim of the user such as ease of use.
- ✧ Verifiable non-functional requirement
 - A statement using some measure that can be objectively tested.

Usability requirements example



- ✧ The system should be easy to use by medical staff and should be organized in such a way that user errors are minimized. (Goal)
- ✧ Medical staff shall be able to use all the system functions after four hours of training. After this training, the average number of errors made by experienced users shall not exceed two per hour of system use. (Testable non-functional requirement)

Metrics for specifying nonfunctional requirements



Property	Measure
Speed	Processed transactions/second response time Screen refresh time
Size	Mbytes Number of ROM chips
Ease of use	Training time Number of help frames
Reliability	Mean time to failure Probability of unavailability Rate of failure occurrence Availability
Robustness	Time to restart after failure Percentage of events causing failure Probability of data corruption on failure



Requirements engineering processes

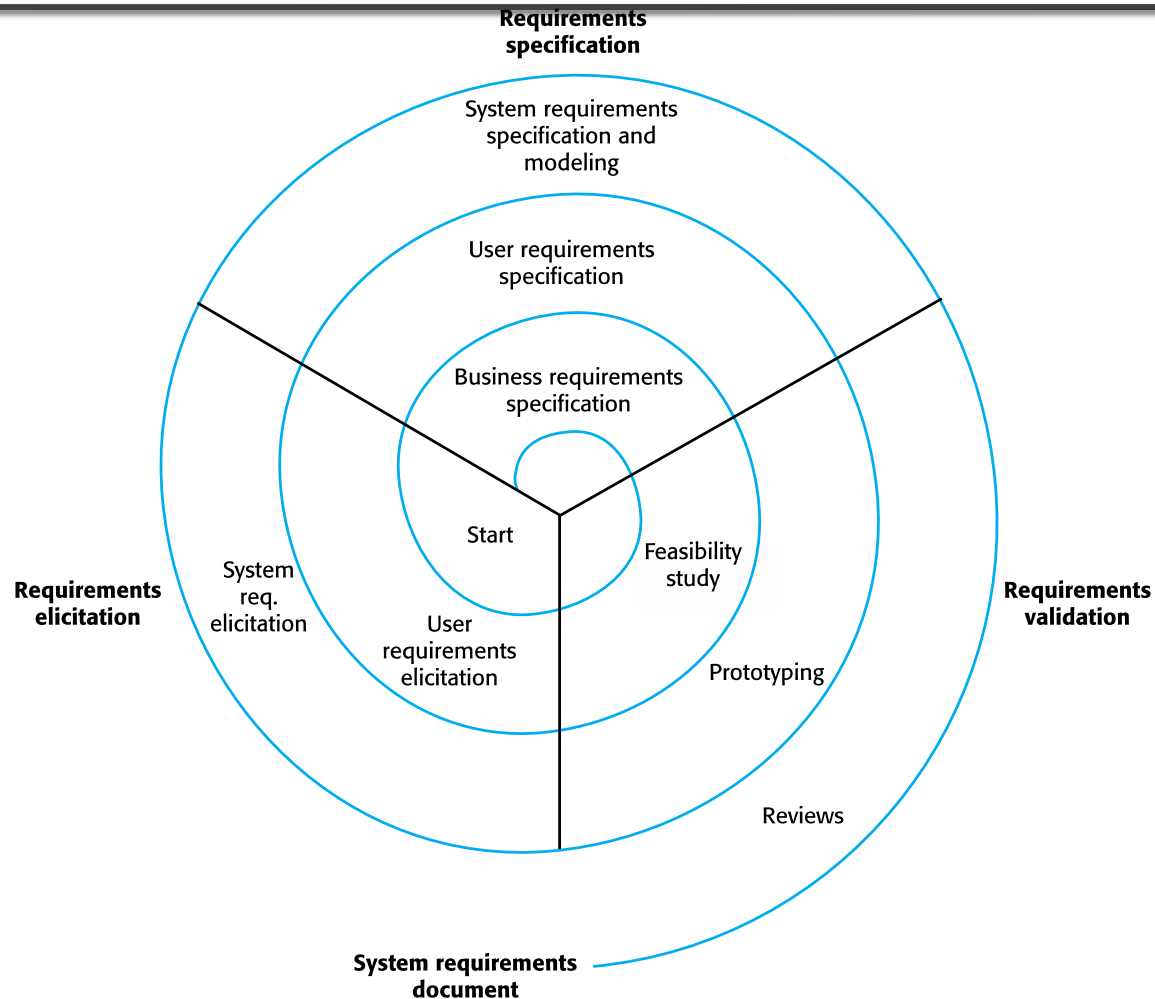
Requirements engineering processes



- ✧ The processes used for RE vary widely depending on the **application domain**, the **people involved** and the **organisation developing** the requirements.

- ✧ However, there are a number of **generic** activities common to all processes
 - Requirements elicitation;
 - Requirements analysis;
 - Requirements validation;
 - Requirements management.

A spiral view of the requirements engineering process





Requirements elicitation

Requirements elicitation and analysis



- ✧ Sometimes called requirements **elicitation** or requirements **discovery**.
- ✧ Involves technical staff working with customers to find out about the application domain, the services that the system should provide and the system's operational constraints.
- ✧ May involve end-users, managers, engineers involved in maintenance, domain experts, trade unions, etc. These are called *stakeholders*.

Requirements elicitation



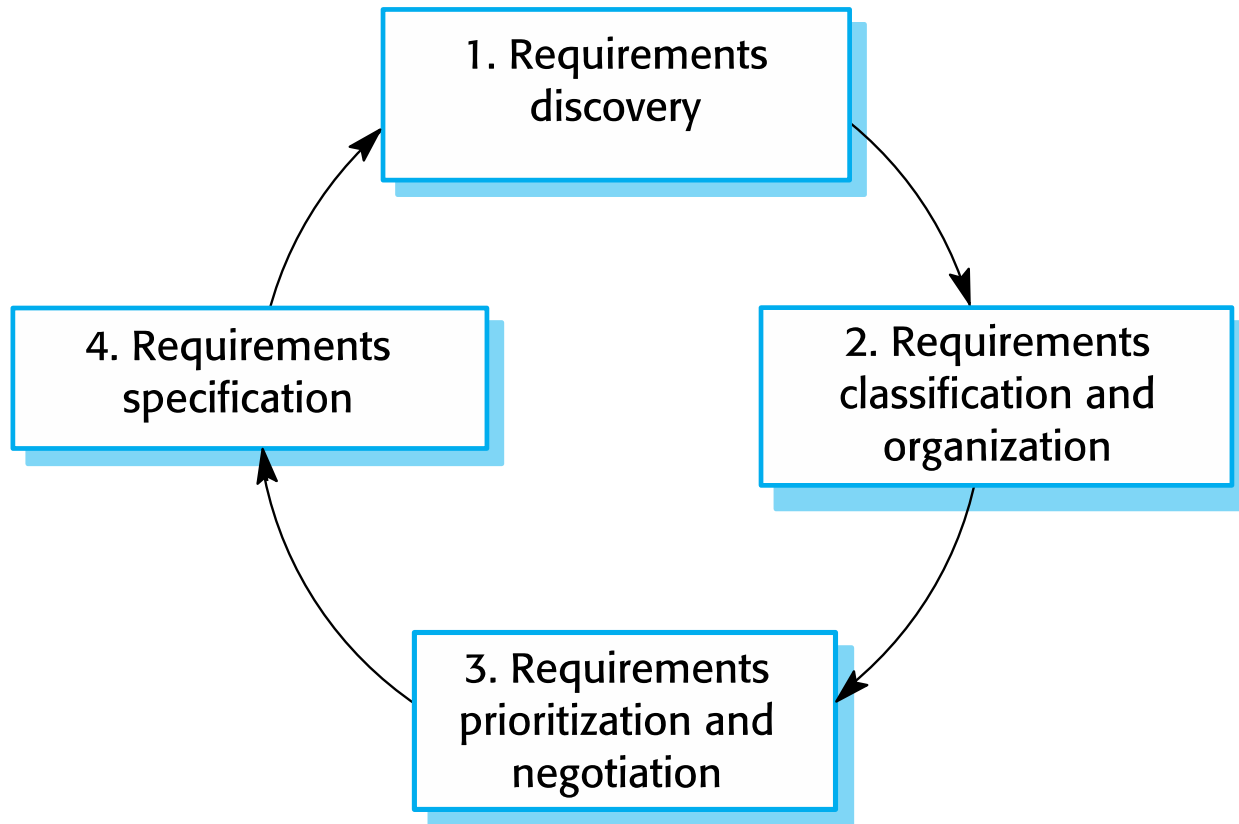
- ✧ Software engineers work with a range of system stakeholders to find out about the application domain, the services that the system should provide, the required system performance, hardware constraints, other systems, etc.
- ✧ Stages include:
 - Requirements discovery,
 - Requirements classification and organization,
 - Requirements prioritization and negotiation,
 - Requirements specification.

Problems of requirements elicitation



- ✧ Stakeholders **don't know** what they really want.
- ✧ Stakeholders express requirements in their **own terms**.
- ✧ Different stakeholders may have **conflicting** requirements.
- ✧ Organisational and political factors may **effect** the system requirements.
- ✧ The requirements change during the analysis process. **New stakeholders** may **emerge** and the business environment may change.

The requirements elicitation and analysis process



Process activities



✧ Requirements discovery

- Interacting with stakeholders to **discover** their requirements. Domain requirements are also discovered at this stage.

✧ Requirements classification and organisation

- **Groups** related requirements and **organises** them into coherent clusters.

✧ Prioritisation and negotiation

- **Prioritising** requirements and **resolving** requirements **conflicts**.

✧ Requirements specification

- Requirements are **documented** and input into the next round of the spiral.

Requirements discovery



- ✧ The process of **gathering** information about the required and existing systems and **distilling** the user and system requirements from this information.
- ✧ Interaction is with system stakeholders from managers to external regulators.
- ✧ Systems normally have a range of stakeholders.

Interviewing



- ✧ Formal or informal interviews with stakeholders are part of most RE processes.
- ✧ Types of interview
 - **Closed interviews** based on pre-determined list of questions
 - **Open interviews** where various issues are explored with stakeholders.
- ✧ Effective interviewing
 - Be **open-minded**, avoid pre-conceived ideas about the requirements and are **willing to listen** to stakeholders.
 - **Prompt the interviewee** to get discussions going using a springboard question, a requirements proposal, or by working together on a prototype system.

Problems with interviews



- ✧ Application specialists may use language to describe their work that isn't easy for the requirements engineer to understand.
- ✧ Interviews are not good for understanding domain requirements
 - Requirements engineers cannot understand specific domain terminology;