Requirement Engineering

Prof. Dr. Shazia Shoaib

Software requirement engineering

What are requirements, what do we mean by requirements?

We discuss requirements in different perspective.

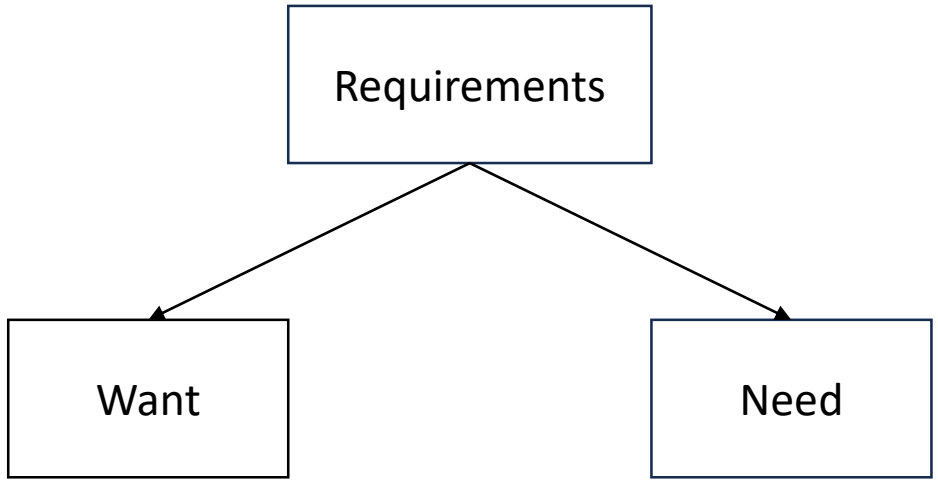
We consider in RE

- Different aspects if SW req.
- How are requirements gather?
- What are the process involved in requirement gathering?

If above 3 tasks done properly & correctly then we can say that the base of SW is strong & we can develop high quality SW. Otherwise, we can face failure & poor-quality SW.

Definition requirement engineering

Requirement mean something wanted or something needed. It is interesting to note that the difference b/w want and need.



Requirement Engineering Process

RE process enables us to systematically determine requirements of the system products.

In requirement process we will talk about

- Different activities
- Problems
- Issues we face during gathering requirements

Above issues we will discuss later.

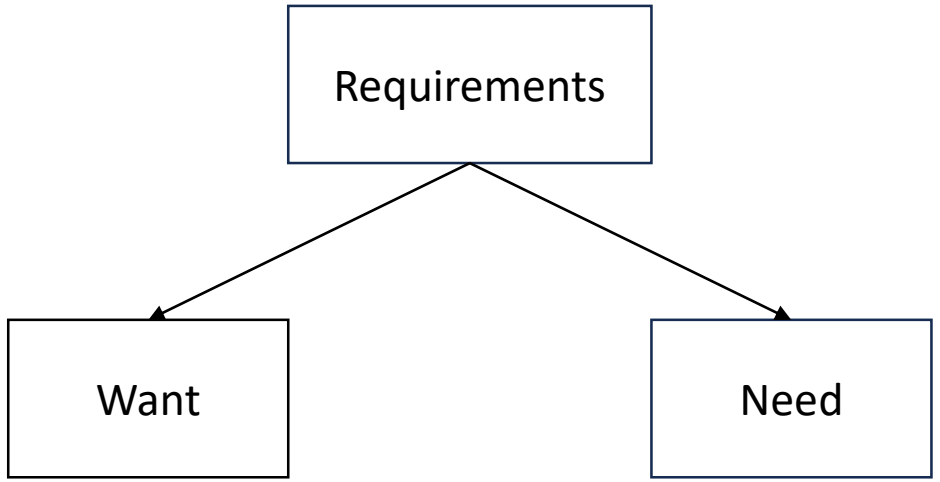
Responsibility of requirement engineering

- Discover
- Understand
- Elaborate
- Exact requirement of the customer for a particular software

Normally customer do know what the requirement are . So, it is necessary to know about requirement definition.

Definition requirement engineering

Requirement mean something wanted or something needed. It is combination of 2 words, want and need. It is necessary to note the difference between want and need.



For Example: many children want to eat burger on pizza, but in reality, they may not need it.

In School app, add animations this is only want of a developer not the need of system.

Which statement specify ‘want’ and ‘need’?

- "The application should support seamless synchronization across multiple devices, ensuring users can access their data from anywhere."
- "The system should have a robust search functionality, enabling users to quickly find relevant information within the platform."
- "Users should be able to easily customize their profiles, providing a personalized experience within the application."

Answer

- Need:

"The application should support seamless synchronization across multiple devices, ensuring **users can access** their data from anywhere."

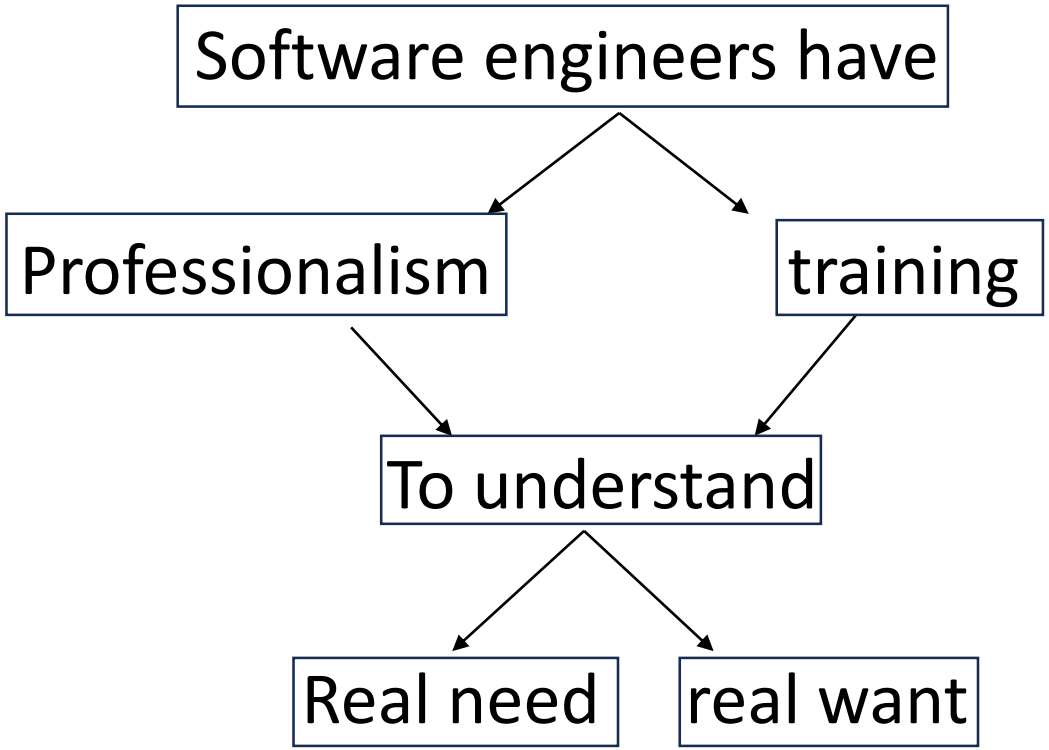
- Need:

"The system should have a robust search functionality, enabling users to **quickly find relevant information** within the platform."

- Want:

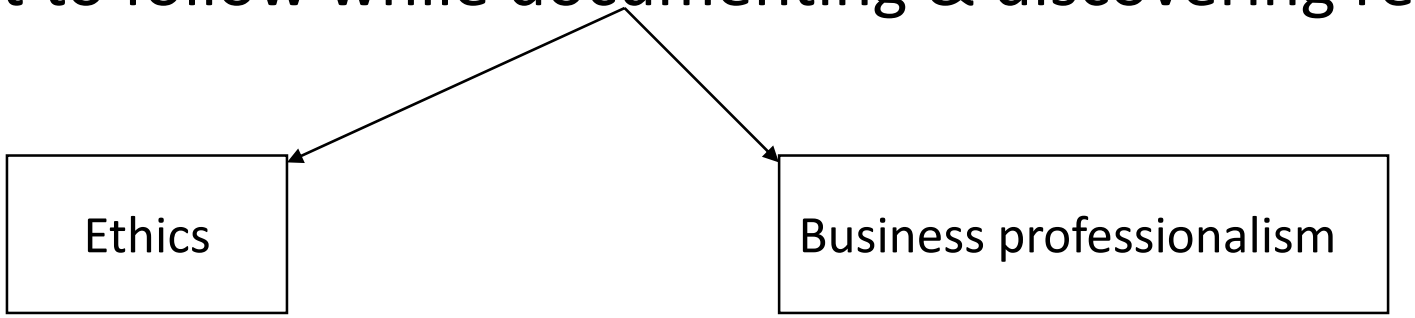
"Users should be able to **easily customize** their profiles, providing a personalized experience within the application."

For developing high quality software, software engineers must have professionalism and training between real need and want of the customers.



SE live and work according to the code of ethics issued by the IEEE computer security.

Important to follow while documenting & discovering requirements



- Avoiding conflict with stakeholders,
- Follow Standards (IEEE, CMM, ISO)

- Team-Work,
- Communication,
- Focus on ethical need

How

system does is the modeling phase/ design phase.

What

Describing **complete specifications** of **external behavior** of the system to be built.

A product viewed by user in different perspective.

If we give input then system respond as you expected. (if want to calculate employee's salary, system calculate the salary.)

Software Requirement tells us:

- What the system does with describing, not how it will do it.

What

- My car runs on an engine

How

- I am not describing how my car run on this engine.

Another way to look SW req. (External behavior)

It describe complete specification of the desired external behavior of software system to be built.

Specification:

- What the system does?
- A product can be viewed by its user in different perspective.
- External behavior has to be:
  - Capture
  - Documented
  - Understand by its users

Definition external behavior

If we give an input, then system show correct or incorrect result.

Important to remember:

User can only understand a product from its external behavior. They don’t know how the product is built. They don’t care how the product is built and they should not know how to product is built. They can only use & judge the product by its external behavior.

Example:

If want to calculate employee’s salary, system calculate the salary, correctly or incorrectly, we knows it from external behavior.

Abstract level detail of what SW req. may be

- Abstract statement & services
- Constraints
- Detail mathematical function or formulas or some other domain

Point is:

We can define requirement at multiple level of abstraction. Simple statement, they can be at different level of detail. E.g, I can say I want a software product which calculate payroll of all employes. So, level of details may be varying.

Requirements at different level of details

- Low level
- High level

Example:

- I want a house build.
- I want detailed map of the house.
- I want 3 beds room.
- I can say its up to you. I want 400 square yard house & bring me map its all up to you & will discuss when you come with document.
- Or other way I can provide you more detail about front elevation, side view etc.

These requirements may be a part of

- Technical document (written when the contract is signed)
- Bidding a contract
- Contract

Requirement may be included in contract i.e business contract although it is not recommended but they may be included in bidding of a product & they may also become part of requirement can operated at different level of abstraction.

Same is applied on software product.

But software product can not be touched, viewed, destroyed.

More attention to be given during software requirement.

- Developing
- Negotiation
- Analyzing
- Understanding

IEEE Definition

A condition or capability that must be met or possessed by a system to satisfied a contract, standard, specification, or other formally imposed document.

- Condition means constraints.
- Capability means must be part of system functionality

Sources of Requirements

- Stack holders ⟶ People affected in some way by the system
- Document ⟶ Developing software for payroll deductions and calculations tax laws and company laws would be very important source of requirements, the tax calculations in the tax regulations vary from time to time and there documented and provided by the government. It is not part of company's policy, basically part of documents which are imposed of by the government.
- Existing System ⟶ manual system and outdated system
- Domain/Business ⟶ area if we calculate provident fund then we do it or include in business area of application domain.

Importance of software requirements

The hardest single part of building a software system is deciding what to built. No other part of work. So cripples the resulting system if does wrong. No other part is difficult to rectify later.

Example of functional requirement (what)

- **What** describe policy matter means functionality aspects.
- **How** define mechanism and to development for the software is the responsibility of designers and developer.

Example:

A system shall maintain record of library material including books, services, newspapers and magazines, videos and audio types, reports, DVD etc.

Does not tell how it will do.

Examples

What: The system should allow users to view their purchase history.

- What functionality is expected as per the "what" statement?
- Why is it important for a system to have a feature to view purchase history? Can you provide real-life scenarios where this functionality would be useful?
- What is the **how** statement for this example?

Answers

- The expected functionality as per the "what" statement is the ability for **users to view** their purchase history.
- It's important for a system to have this feature as **users can track** their past purchases, review order details, and keep records for reference. For example, users may need to verify a previous purchase for warranty or return purposes.
- **How (Non-Functional Requirement - Performance):**

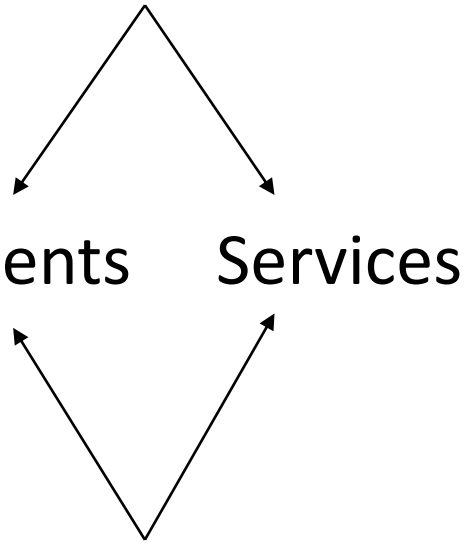
Requirement: The system should ensure that the purchase history page loads within 2 seconds, providing a responsive user experience.

Kind of Requirements

- Functional requirements
- Non-functional requirements
- Domain requirements
- Inverse requirements
- Design and implementation constraints

Functional Requirements

Functional Requirements tells what the system does.



Define external behavior of the system

- **Exceptional behavior** (e.g, invalid username, forget password, timeout) handle in functional requirements.

Enhance maintenance

Better user experience

- **Abnormal behavior** is also documented in proper way, so proper expectational handling can be done.
- Functional requirements should be **complete and consistent**
- **Sequencing and Parallelism** are captured by Functional requirements.

Sequencing
- Step by step
- Order of task

Parallelism (important for system behavior)
Ability of system to perform multiple task parallelly
e.g, multiple people like post on Facebook at the same time

Complete and consistence

- Do not change requirements again and again otherwise price, efforts will increase.