

# *Software Engineering*

## *Lecture 5*

### *Software Requirements*

### *Software Requirements*

- *Logically the first stage of software development is to establish precisely what the users of the system want. This stage call the **Software requirements**. The software requirements are description of features.*
- ***Requirements:** The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client's point of view.*
- ***User requirements:** statements of what services the system is expected to provide to the users and the constraints under which it must operate, in a natural language plus diagrams.*

## ***Software Requirements***

*According to the IEEE, 'Requirement is a condition of capability, needed by a user to solve a problem or achieve an objective'. Broadly requirements can be classified as following :*

- ***Functional Requirements.***
- ***Non-Functional Requirements.***
- ***Domain Requirements.***
- ***User Requirements.***
- ***System Requirements.***
- ***External Requirements***

### ***SR: Functional requirements***

1. ***Functional requirements:*** *are the services which the end users expect the final product to provide.*

*These requirements focus on the functionality of the software components that build the system.*

- *Expressing such type of requirements done in some ways like*
  - *Natural language,*
  - *structured or*
  - *formatted languages with no rigorous syntax.*

## ***SR: Non-Functional requirements***

2. ***Non-Functional requirements*** are the constraints imposed on the system **and can be classified as**
  1. ***Product requirements:*** These requirements specify product behavior. ***Example*** include performance requirements on how fast the system must execute and how much memory it requires.
  2. ***Organizational requirements:*** These requirements are derived from policies and procedures in the customers and developers organization. ***Example*** include process standards that must be used.
  3. ***External requirements:*** These may include interoperability requirements that define how the system interacts with systems in other organizations.

## ***SR: Domain Requirements***

3. ***Domain Requirements:*** Some requirements are specific to an application domain e.g. ***hospital, banking, marketing etc.***
  - *These requirements are not user specific and can be identified from that domain model.*
  - ***For example,*** in a hospital there will always be requirements concerning ***Doctors, Staff, Departments and Patients etc.***

## ***SR: User & System Requirements***

4. ***User Requirements:*** *Statements in natural language (NL) plus diagrams of the services the system provides and its operational constraints. Written for customers are written from the user's point-of-view. User requirements define the **information** or **material** that is **input** into the business process, and the **expected information or material as outcome** from interacting with the business process (system), specific to accomplish the user's business goal.*
5. ***System Requirements:*** *A structured document setting out detailed descriptions of the system services. Written as a contract between client and contractor*

## ***Software Requirements***

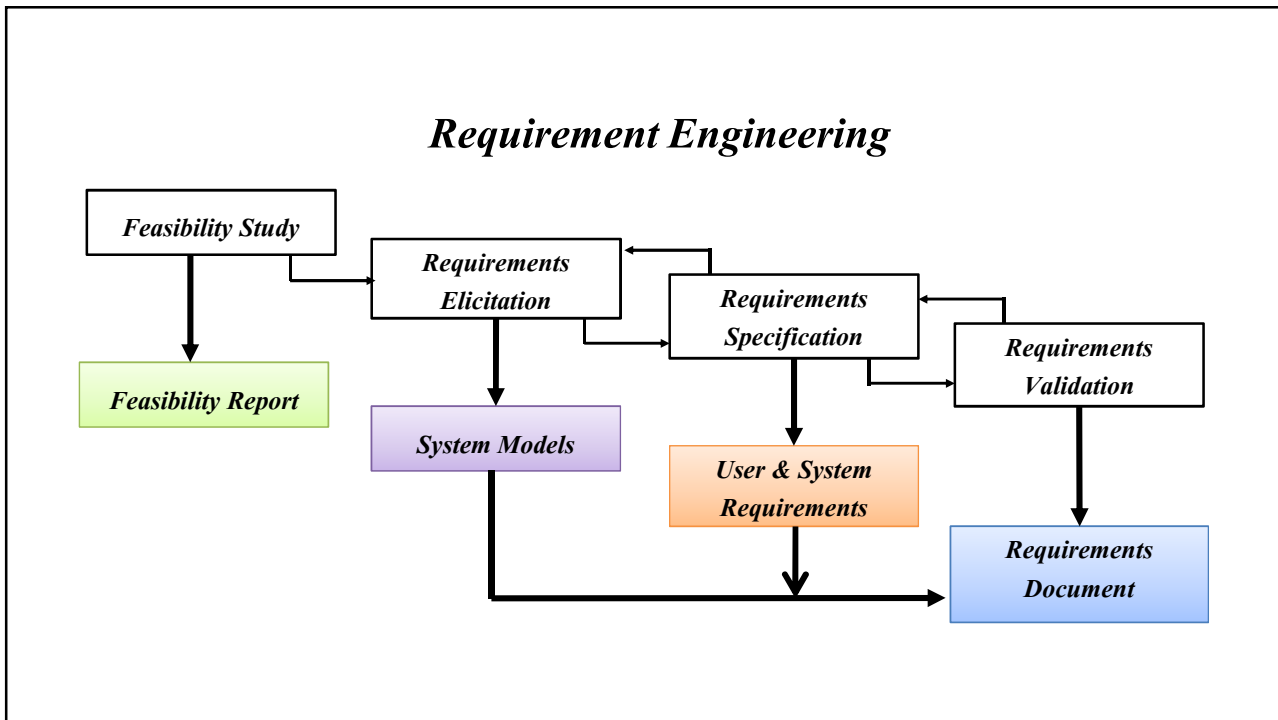
6. ***External requirements:*** *from factor external to the system and its development process*
  1. ***Regulatory requirements:*** *What must be done for the system to be used by a regulator.*
  2. ***Ethical requirements:*** *Ensure the system will be acceptable to its user and the general public.*
  3. ***Legislative requirements:*** *Ensure the system operates within the law.*

## ***Requirement Engineering***

- *The systematic process of documenting requirements through an interactive cooperative process of analyzing a problem is called 'Requirement Engineering'.*
- *The **requirements engineer** must always strive to gather complete, precise, and detailed specifications of system requirements .*
- ***Requirements Engineering:** is a sub-discipline of software engineering that is concerned with determining the goals, functions, and constraints of software systems.*

## ***Requirement Engineering***

- *The Requirement Engineering Process consists of five main activities*
  1. *Feasibility study,*
  2. *Requirements Elicitation,*
  3. *Requirements Specification,*
  4. *Requirements Validation and*
  5. *Requirements Document (Software Requirements Specification, SRS)*



### ***RE: Feasibility study***

- Preliminary investigations examine the feasibility of the project, that the proposed project will be useful to the organization or not. Following are the main feasibility tests:
  1. ***Operational Feasibility.***
  2. ***Technical Feasibility.***
  3. ***Economic Feasibility.***
- The output of this phase should be a feasibility study report that should contain adequate comments and recommendations for management about whether or not the project should be undertaken.

## ***RE: Requirements Elicitation***

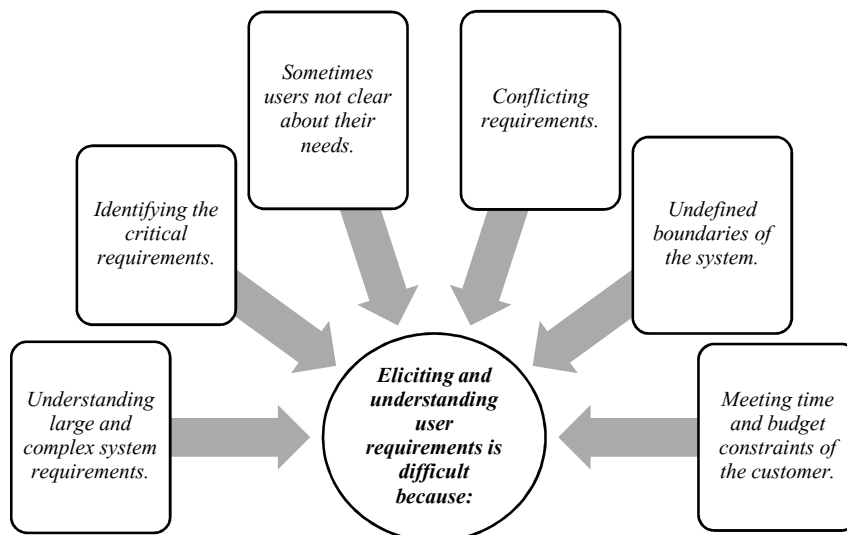
- *If the feasibility report is positive towards undertaking the project, next phase starts with gathering requirements from the user. Analysts and engineers communicate with the client and end-users to know their ideas on what the software should provide and which features they want the software to include. The goal of the requirements phase is to **elicit the requirements from the user**.*
- *This is usually achieved by the **development of diagrams** then user reviews the diagrams to determine if the software developer has understood the requirements.*

## ***RE: Requirements Elicitation***

- *A process activities of the elicitation and analysis process are:*
  1. ***Requirements discovery:** This is the process of interacting with the user in the system to collect their requirements. This requirement discovery involve the **Interviewing, Scenarios, Questionnaires or prototyping** to collect the requirement from the users.*
  2. ***Requirements classification and organization:** This activity takes the unstructured collection of requirements, groups related requirements and organizes them into coherent clusters.*

## ***RE: Requirements Elicitation***

3. ***Requirements prioritization and negotiation Inevitably:*** where multiple users are involved, requirements will conflict. This activity is concerned with prioritizing requirements, and finding and resolving requirements conflicts through negotiation.
4. ***Requirements documentation:*** The requirements are documented. Formal or informal requirements documents may produced.



## ***RE: Requirements Specification***

- *The requirements received from client are written in natural language. It is the responsibility of system analyst to document the requirements in technical language so that they can be comprehended and useful by the software development team. The 'Requirement Specification' activity is required to produce **formal software requirements models**.*
- *The '**Functional**' and '**Non-Functional**' properties of the system are specifying by the requirements models along with the constraints imposed on the system. There are a number of structured and object oriented methodologies that have been proposed to develop these models.*

UNIVERSITY OF DIYALA

## ***RE: Requirements Validation***

- *This activity is usually defined as a process to ensure the constancy of requirements model with respect to user's needs. The validation is done for the final model as well as for the intermediate models.*  
*Following steps should be followed while validating requirements*
  1. **Ensure** that the requirements are not conflicting with other requirements.
  2. **Ensure** that the requirements are complete in all respect.
  3. **Ensure** that requirements are realistic.

## ***RE: Requirements Validation***

- *If the requirements are not validated properly, resultant errors in the requirements definition needs a lot of rework and modifications. Some of the effective ways to validate requirements are :*
  1. **Reviews:** *end users and developers both study and consider the requirements.*
  2. **Prototyping:** *an executable model of the product with minimal set of requirements is built and is demonstrated to the customers. Customers use the prototype to ensure that it meets their needs.*
  3. **Test cases generation:** *From this, it is made sure that it is possible to design test cases for each and every requirement documented otherwise there is a need to reconsider the requirement.*

## ***RE: Requirements Document***

- *Software Requirement Analysis (SRA) ends with a document called 'Software Requirement Specification (SRS) which describes the complete external behavior of the proposed software.*
- *SRS is a document created by system analyst after the requirements are collected from various stakeholders. SRS defines how the intended software will interact with hardware, external interfaces, speed of operation, response time of system, portability of software across various platforms, maintainability, speed of recovery after crashing, Security, Quality, Limitations etc.*

## ***RE: Requirements Document***

- ***Advantages of 'Software Requirement Specification (SRS)***
  1. *SRS establishes the basis for agreement between the client and the supplier on 'what the software product will do'.*
  2. *SRS provides a reference for validation of the final product. The SRS helps to the client to determine whether the software meets the requirements or not.*
  3. *If we want a high quality final software that has few errors, we must begin with a high quality SRS.*
  4. *A high quality SRS reduces the development cost.*