SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

Course Name –Software Engineering

II Year / III Semester

Requirements Engineering

- **Requirements Engineering (RE)** refers to the process of defining, documenting, and maintaining requirements in the engineering design process.
- Requirement engineering provides the appropriate mechanism to understand what the customer desires, analyzing the need, and assessing feasibility, negotiating a reasonable solution, specifying the solution clearly, validating the specifications and managing the requirements as they are transformed into a working system.

According to IEEE standard 729, a requirement is defined as follows:

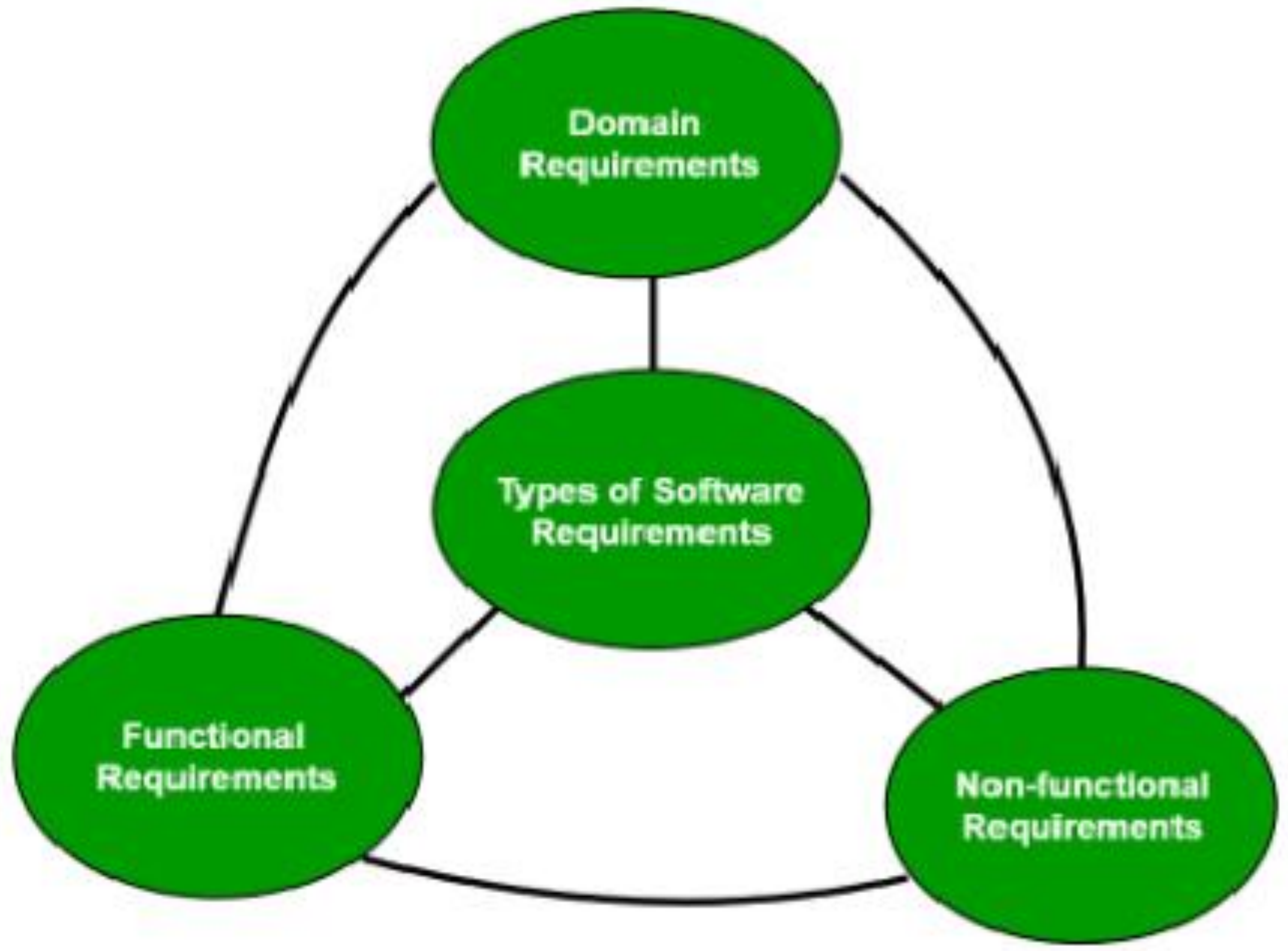
•A condition or capability needed by a user to solve a problem or achieve an objective
•A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification or other formally imposed documents
•A documented representation of a condition or capability as in 1 and 2

A software requirement can be of 3 types:

•Functional requirements

•Non-functional requirements

•Domain requirements



Functional Requirements:

- These are the requirements that the end user specifically demands as basic facilities that the system should offer.
- All these functionalities need to be necessarily incorporated into the system as a part of the contract.
- These are represented or stated in the form of input to be given to the system, the operation performed and the output expected.

Non-functional requirements:

- These are basically the quality constraints that the system must satisfy according to the project contract.
- The priority or extent to which these factors are implemented varies from one project to other.
- They are also called **Non-behavioral requirements**.
- Non-functional requirements are divided into two main categories:

Execution qualities like security and usability, which are observable at run time.

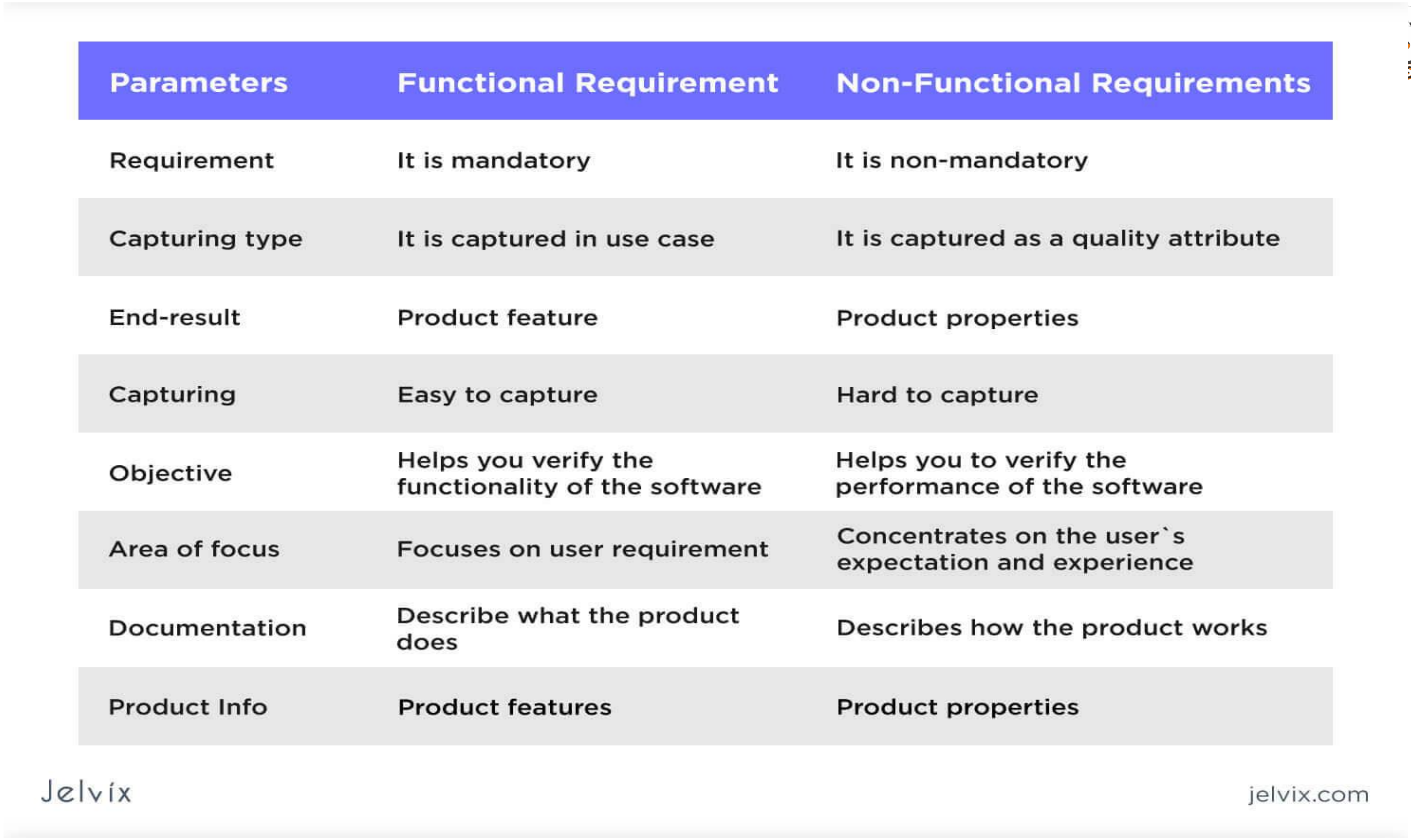
Evolution qualities like testability, maintainability, extensibility, and scalability that embodied in the static structure of the software system.

Non-behavioral requirements basically deal with issues like:

- Portability
- Security
- Maintainability
- Reliability
- Scalability
- Performance
- Reusability
- Flexibility

Domain requirements:

- Domain requirements are the requirements which are characteristic of a particular category or domain of projects.
- The basic functions that a system of a specific domain must necessarily exhibit come under this category.
- For instance, in an academic software that maintains records of a school or college, the functionality of being able to access the list of faculty and list of students of each grade is a domain requirement.



Parameters	Functional Requirement	Non-Functional Requirements
Requirement	It is mandatory	It is non-mandatory
Capturing type	It is captured in use case	It is captured as a quality attribute
End-result	Product feature	Product properties
Capturing	Easy to capture	Hard to capture
Objective	Helps you verify the functionality of the software	Helps you to verify the performance of the software
Area of focus	Focuses on user requirement	Concentrates on the user`s expectation and experience
Documentation	Describe what the product does	Describes how the product works
Product Info	Product features	Product properties

Jelvix

jelvix.com

Enduring requirements

- These are relatively stable requirements that derive from the core activity of the organisation and which relate directly to the domain of the system.
- For example, in a hospital there will always be requirements concerned with patients, doctors, nurses, treatments, etc.
- These requirements may be derived from domain models that show the entities and relations which characterise an application domain (Prieto-Díaz and Arango, 1991, Easterbrook, 1993).

Volatile requirements

- These are requirements that are likely to change during the system development process or after the system has been become operational.
- Examples of volatile requirements are requirements resulting from government health-care policies or healthcare charging mechanisms.

THANK YOU