REQUIREMENTS

➢ Requirements Engineering

Requirements engineering is a crucial part of software engineering that involves the identification, analysis, specification, validation, and management of the requirements of a software system. Requirements are the foundation of any software project, and they provide the basis for designing and developing a system that meets the needs of its stakeholders.

Requirements engineering is an iterative process that involves ongoing communication and collaboration between the stakeholders, the software developers, and the software engineers. The goal of requirements engineering is to ensure that the requirements of the system are complete, accurate, and unambiguous, and that they can be traced throughout the software development life cycle.

➢ Requirements

A requirement is a statement that describes what a software system or product should do or how it should behave. Requirements are used to specify the functionality, performance, usability, reliability, and other characteristics of the system or product. They are the foundation of any software project, and they provide the basis for designing and developing a system that meets the needs of its stakeholders.

Requirements can be elicited from various sources, such as interviews, surveys, focus groups, and observation. They can also be documented using different techniques, such as use cases, user stories, and requirements specifications. Requirements engineering is a critical process in software engineering that involves the identification, analysis, specification, validation, and management of requirements throughout the software development life cycle.

➢ Types of Requirements

Requirements are divided into several types, where each of them is responsible for something specific. Below there are listed some important types of requirements and a short definition for some of them.

1. User Requirements

User requirements are a type of requirements in software engineering that describe the needs and expectations of the end-users or customers of a software system. User requirements are important because they define what the users want and expect from the system, and they help ensure that the system meets the needs of its intended users.

2. System Requirements

System requirements describe the hardware, software, and other resources that are needed to run a software system. System requirements specify the minimum and recommended configurations that are required to run the system efficiently and effectively. These requirements are important because they ensure that the software system is compatible with the hardware and software infrastructure of the organization.

3. Functional Requirements

Functional requirements are a type of requirements in that describe the specific behaviors and functions that a software system must perform to meet the needs of its users or stakeholders. Functional requirements specify the features and capabilities of the system and describe how it should behave in response to various inputs and conditions

4. Non-Functional Requirements

Non-functional requirements are a type of requirements that describe the qualities and characteristics that a software system must possess, rather than specific behaviors or functions. Non-functional requirements specify the performance, reliability, usability, security, and other aspects of the system that are important to its stakeholders but are not related to its primary functions.

There are several types of non-functional requirements:

- Product requirements
- Organizational requirements
- External requirements

5. Product Requirements

Product requirements refer to the set of non-functional requirements that describe what a software product should do and how it should behave. Product requirements are typically derived from the needs of the customers, users, and stakeholders and are used as the basis for designing, developing, and testing the software product.

There are some categories of product requirements which will be listed and analyzed below.

- Usability Requirements

Usability requirements refer to the set of non-functional requirements that describe how easy it is for users to learn and use a software system. Usability requirements are important because they ensure that the software system is user-friendly and meets the needs of its users.

- Dependability Requirements

Dependability requirements refer to the set of non-functional requirements that describe how reliable, available, and maintainable a software system should be. These requirements ensure that the software system can be trusted to perform its intended function, even under adverse conditions.

- Security Requirements

Security requirements refer to the set of non-functional requirements that describe how a software system should protect against unauthorized access, data breaches, and other security threats. These requirements ensure that the software system can be trusted to protect sensitive data and maintain confidentiality, integrity, and availability.

- Efficiency Requirements

Efficiency requirements refer to the set of non-functional requirements that describe how a software system should perform efficiently in terms of response time, throughput, and resource utilization. These requirements ensure that the software system can meet performance expectations and handle user and system load efficiently.

- Performance Requirements

  Performance requirements refer to the set of non-functional requirements that describe how a software system should perform in terms of speed, throughput, and capacity.

- Space Requirements

  Space requirements refer to the set of non-functional requirements that describe how much disk space, memory, or other storage resources a software system must consume or provide. These requirements ensure that the software system can function correctly and store the necessary data and files.