Apple Mahmood Oni

Task-11

Date: 13th January,2024

1. Requirement is a condition or capability needed by a user to solve a problem or achieve an objective. A **software requirement** is a statement that specifies what a software system must do to be successful.

Types of Requirements:

- **Functional Requirements:** A functional requirement is a specific, measurable capability that a software application or product must possess to be successful. There are 2 types of Functional Requirement:
  - **User Requirements**: are those that pertain to the functionality that users need from the system to complete their tasks.
  - **System Requirements**: pertain the functionality that the system itself needs to work effectively.
- **Non-functional Requirements:** Describe how the software application or products needs to behave. Performance, Security etc are the examples of non-functional requirement.
- **Business/ Domain requirements**: These requirements describe the business goals and objectives that the software system is expected to achieve. Business requirements are usually expressed in terms of revenue, market share, customer satisfaction, or other business metrics.
- **Stakeholder Requirements:** outline the characteristics that must be satisfied in order for the business requirements to be satisfied

2. Requirement analysis is an activity to analyze, refine and scrutinize the gathered requirements to make consistent and unambiguous requirements. There are 4 process involved in requirement analysis.

- **Sorting information:** Analysis of information permits to separate customers requirements, user's requirements and standards information.

- **Modeling information:** Customers' requirements would be formalize in a Functional Analysis System Techniques (FAST) through a FA of the organization. User requirements would be modeled with use cases using UML. Standard information wouldn't be formalized.
- **Structuring requirements:** A reusable form implied considering an implementation of the requirements through a database.
- **Validating requirements**: A comparison between use cases and the last level of the FAST with a pivot table assured the completeness and concordance of the information in the database.

3. A **software requirements specification (SRS)** is a document that describes what the software will do and how it will be expected to perform. It also describes the functionality the product needs to fulfill the needs of all stakeholders (business, users).

Characteristics of a good SRS document:

- **Correctness:** All the features and functionality of the system should be defined accurately.
- **Completeness:** Should contain all features and functionality demanded by the client.
- **Consistency:** Should be free of contradiction and should follow the same terminology and formatting.
- **Unambiguousness:** There should not be any confusion regarding interpretation of the requirements.
- **Traceability:** Should be traceable to other documents such as user stories and use cases.
- **Modifiability:** Update and change should be possible.
- **Verifiability:** Ensuring that requirements are being met through testing and validation.
- **Testability:** Should be written in a way so that requirements can be tested and validated.

- **Design Independence:** Should be an option for multiple design alternatives meaning, should not contain any implementation details.
- **Understandable by the customer:** The language should be kept simple and clear so that a non-technical person understands.

4. A **Context Diagram** shows the interaction between a system and external entities.

The context diagram in the slide represents an student result management system. Administrator and Marks entry operator are external entities/ factors.

The administrator interact with the system by giving entry of student information and subject information. Marks entry operator interacts by entering marks in the system.

The system generates student information report, marksheet and student performance report as output.