நான் முதல்வன்

IBM

9530

ST.MOTHER THERESA ENGINEERING COLLEGE

COMPUTER SCIENCE AND ENGINEERING

NM-ID :1F5902C07E80DFCA24D42E87C655E5DC

REG.NO: 953023104129

DATE:14-09-2025

Completed the project named as

Phase-1

PROBLEM UNDERSTANDING &

REQUIRMENTS

SUBMITTED BY,

A.SUSAIYAMMAL

PH.NO:9360984679

1. Introduction

Every project begins with a clear understanding of the problem it aims to solve. Without defining the problem properly, the solution may fail to address user needs or business objectives. This section provides a structured overview of how the problem is identified, analyzed, and translated into specific requirements.

---

2. Problem Understanding

2.1 Problem Identification

- Clearly define the issue or gap in the current system.
- Example: *In existing student management systems, manual attendance and records often lead to errors, delays, and data duplication.*

2.2 Problem Statement

- A concise, formal statement of the issue.
- Example: *There is a need for a centralized, automated system that reduces manual work, improves data accuracy, and enables real-time access to student records.*

2.3 Root Cause Analysis

Use methods like **5 Whys** or **Fishbone Diagram** to identify the root causes of the problem.

- Lack of automation
- Poor integration of data
- Human errors in manual entry
- Time delays in reporting

2.4 Goals and Objectives

- To design a system that eliminates manual redundancy.
- To provide accurate, reliable, and timely access to information.

- To improve user experience and efficiency.

---

3. Project Scope

3.1 In-Scope

- Features and services that will be included in the project.
- Example: *User authentication, record management, reporting module, and analytics.*

3.2 Out-of-Scope

- Features excluded to prevent scope creep.
- Example: *Integration with third-party systems or mobile applications (may be considered in future).*

3.3 Stakeholders

- Students
- Teachers/Employees

- Administrators
- External users (e.g., parents, customers, regulators)

4. Requirement Gathering

4.1 Techniques Used

- **Interviews** – Direct interaction with stakeholders.
- **Surveys/Questionnaires** – Collecting mass feedback.
- **Observation** – Studying current processes.
- **Document Analysis** – Reviewing existing system reports.
- **Workshops/Brainstorming** – Group discussions for clarity.

4.2 Requirement Prioritization

- **MoSCoW Method** (Must have, Should have, Could have, Won’t have).

- Ensures critical features are developed first.

---

5. Functional Requirements

Functional requirements describe **what the system must do**.

Examples:

1. **User Management** – Users must be able to register, log in, and manage profiles.
2. **Data Management** – System should allow creation, modification, and deletion of records.
3. **Reporting** – The system must generate real-time reports.
4. **Search and Filter** – Users should search records based on criteria.
5. **Notifications** – System must alert users about important updates.

6. Non-Functional Requirements

Non-functional requirements describe **how the system should perform**.

Examples:

1. **Performance** – The system should handle 500 concurrent users.
2. **Security** – User data must be encrypted (AES-256).
3. **Scalability** – The system should support future growth.
4. **Usability** – The interface should be intuitive and user-friendly.
5. **Reliability** – 99.9% uptime must be ensured.
6. **Maintainability** – Easy to update and extend.

---

7. Use Case Modeling

Include **Use Case Diagrams** to represent

system interaction.

- Example actors: *Admin, User, Guest.*
- Example use cases: *Login, Manage Data, Generate Report, View Records.*

8. Requirement Specification (SRS Format)

A detailed **Software Requirement Specification (SRS)** includes:

- Introduction
- System Overview
- Functional and Non-Functional Requirements
- Constraints
- Assumptions
- Acceptance Criteria

9. Challenges in Requirement Gathering

- Ambiguity in stakeholder expectations.
- Conflicting requirements between users.
- Changing scope during development.
- Communication gaps.

---

10. Conclusion

Problem understanding and requirement analysis are the **foundation of project success**. A well-defined requirement document ensures developers, designers, and stakeholders share a **common vision** of the project goals. This minimizes risks, reduces rework, and increases project success rate.
