

Project Report Format

1 Introduction

1.1 Project Overview

The Smart SDLC AI Assistant is a web-based application designed to streamline the Software Development Life Cycle (SDLC) by automating tasks such as requirement analysis, code generation, test case creation, bug fixing, and code summarization. It targets software developers, students, and small teams to enhance productivity and reduce errors.

1.2 Purpose

The project aims to simplify SDLC processes, enabling users to focus on logic and creativity by automating repetitive tasks and providing intelligent assistance through AI-driven features.

2 Ideation Phase

2.1 Problem Statement

Software developers, students, and small teams need an efficient tool to streamline the SDLC by automating tasks like requirement analysis, code generation, bug fixing, test case creation, and code summarization to save time, reduce errors, and enhance productivity.

2.2 Empathy Map Canvas

- **Who:** Developers, students, small teams.
- **Says:** “I need a tool to simplify coding and testing.”
- **Thinks:** “Debugging takes too long; I want automation.”
- **Does:** Writes code, tests applications, searches for fixes.
- **Feels:** Frustrated with repetitive tasks, confident with AI support.
- **Pains:** Time-consuming processes, error-prone coding.
- **Gains:** Increased productivity, faster development cycles.

2.3 Brainstorming

Key ideas include AI-driven requirement analysis, multilingual code generation, automated test case generation, bug detection and fixing, code summarization, and a chatbot for SDLC queries, prioritized based on impact and feasibility.

3 Requirement Analysis

3.1 Customer Journey Map

[To be detailed with stages like discovery, input, processing, and output review.]

3.2 Solution Requirement

- **Functional:** Requirement analysis, code generation, test case generation, bug fixing.
- **Non-functional:** Usability (intuitive UI), security (JWT authentication), performance (2-second response time).

3.3 Data Flow Diagram

DFD Level 0: User inputs requirements/code, AI processes data, and outputs structured modules, code, or test cases, stored in MongoDB.

3.4 Technology Stack

- **Frontend:** React, Tailwind CSS.
- **Backend:** Node.js, Express.js, FastAPI.
- **Database:** MongoDB, SQLite.
- **Infrastructure:** Kaggle, VS Code local server.

4 Project Design

4.1 Problem Solution Fit

The solution addresses the need for automation in SDLC, reducing manual effort and errors for developers and small teams.

4.2 Proposed Solution

A web-based platform with AI-driven features for requirement analysis, code generation, testing, and summarization, accessible via a dark-themed UI.

4.3 Solution Architecture

Micro-services architecture with React frontend, Node.js/Express.js backend, and MongoDB for data storage, deployed on Kaggle.

5 Project Planning & Scheduling

5.1 Project Planning

Three 5-day sprints:

- **Sprint 1:** Requirement analysis, initial code generation (10 story points).
- **Sprint 2:** Code editing, test case generation, bug fixing (11 story points).
- **Sprint 3:** Code summarization, chatbot (4 story points).

Velocity: 8.33 story points per sprint.

6 Functional and Performance Testing

6.1 Performance Testing

- **Unit Testing:** Jest (frontend), Mocha (backend).
- **Integration Testing:** Postman for API endpoints.
- **Performance:** Response time under 2 seconds, 99.9% uptime.

7 Results

7.1 Output Screenshots

[Screenshots of UI, generated code, and test cases to be included.]

8 Advantages & Disadvantages

- **Advantages:** Automates SDLC tasks, improves productivity, supports multiple languages.
- **Disadvantages:** Limited language support, potential AI processing delays.

9 Conclusion

The Smart SDLC AI Assistant successfully automates key SDLC tasks, enhancing developer efficiency and reducing errors, with a scalable architecture for future enhancements.

10 Future Scope

- Expand language support for code generation.
- Integrate CI/CD pipelines.
- Enhance AI for faster processing and complex queries.

11 Appendix

- **Source Code:** [GitHub repository link]
- **Dataset Link:** [Kaggle dataset link]
- **GitHub & Project Demo Link:** [Demo link]