



Automated Code Review System (Project Plan)

**SOEN 6841 – Software Project Management
By
Prof. Joumana Dargham**

Submitted By – Group 18

NAME	STUDENT ID NUMBER
Dhruv Panchal	40226430
Udisha Kaura	40266183
Khushal Nirmal Jain	40233877
Aman Kumar	40278443

TABLE OF CONTENTS

Milestones.....	3
Deliverables	4
Resource Allocation	6
Critical Dependencies.....	7
Work Breakdown Structure	8

Project Plan (WBS):

Top-Down Planning for Automated Code Review System

1. Milestones

The milestones outlined below serve as pivotal markers throughout the construction phase of the project, representing significant achievements in the development and implementation of the Automated Code Review System (ACRS). These milestones encompass key deliverables and objectives that signify progress towards realizing the project's goals and objectives. Each milestone represents a crucial step in the construction process, indicating the successful completion of specific tasks and the attainment of important project milestones. By achieving these milestones, the project team demonstrates its commitment to delivering a high-quality, functional, and efficient code review system that meets the needs and expectations of stakeholders.

- **Initiation Phase**

- **Milestone: Technical Feasibility Assessment Completed**

- The completion of the technical feasibility assessment marks a significant milestone in the initiation phase of the project. This milestone signifies that the team has thoroughly evaluated the technical aspects of implementing the Automated Code Review System (ACRS). Key achievements include identifying potential challenges and constraints related to technology requirements, assessing the compatibility of the proposed solution with existing systems, and estimating the resources required for conducting the feasibility study.

- **Requirement Phase**

- **Milestone: Completion of Requirement Documentation**

- The milestone of completing requirement documentation symbolizes the culmination of an extensive process of gathering, analyzing, and documenting user needs and system requirements. It represents a thorough understanding of the project objectives, scope, constraints, and stakeholders' expectations. The comprehensive documentation serves as a foundation for the subsequent phases of design, development, and implementation, ensuring alignment with stakeholders' requirements and project goals.

- **Design Phase**

- **Milestone: Completion of System Design**

- The milestone of completing system design marks the finalization of the system architecture and user interface design. It represents the translation of conceptual requirements into tangible design specifications, encompassing system components, modules, interfaces, and user interactions. The system design documentation outlines the structural framework, functionality, data flow, and user experience principles, guiding the development team in the implementation phase.

- **Construction Phase**

- **Milestone: Completion of Core Functionality Implementation**

- The milestone of completing core functionality implementation signifies the successful development of essential system features and functionalities. It represents the realization of the system design specifications into functional software components, including frontend and backend modules. The core functionality implementation ensures the fulfillment of primary user requirements and system objectives, laying the groundwork for subsequent feature enhancements and refinements.

- **Testing Phase**

- **Milestone: Completion of Testing Phase**

- The milestone of completing the testing phase indicates the conclusion of rigorous testing activities, including unit testing and integration testing. It represents the comprehensive validation and verification of the system's functionality, performance, reliability, and security aspects. The testing phase ensures the detection and resolution of defects, bugs, and inconsistencies, thereby enhancing the overall quality and robustness of the automated code review system.

- **Implementation Phase**

- **Milestone: Successful Deployment of Automated Code Review System**

- The milestone of successful deployment marks the culmination of the implementation phase, with the automated code review system ready for operational use. It represents the seamless integration of the developed software components into the production environment, ensuring accessibility, availability, and functionality for end-users. The successful deployment signifies the realization of project objectives and the delivery of value to stakeholders, paving the way for increased efficiency and productivity in software development processes.

- **Project Closure**

- **Milestone: Project Sign-Off and Handover**

- The milestone of project sign-off and handover signifies the formal acceptance and closure of the project, marking the transition from development to maintenance and support phases. It represents the acknowledgment and approval of project deliverables by stakeholders, including the completion of post-implementation review, documentation, and knowledge transfer activities. The project closure ensures the seamless transition of ownership and responsibility, facilitating ongoing maintenance, updates, and enhancements to the automated code review system.

2. Deliverables

The deliverables listed below represent concrete results and essential objectives for the Construction Phase of the project, aimed at advancing the development and implementation of the Automated Code Review System (ACRS). Each deliverable is a crucial element in the construction process, contributing to the overall functionality, usability, and efficiency of ACRS.

- **Initiation Phase**

Technical Feasibility Report : The Technical Feasibility Report is a comprehensive document that presents the findings of the detailed study conducted during the Initiation Phase of the project. This report outlines the team's assessment of the technical feasibility of implementing the Automated Code Review System (ACRS). It includes an analysis of potential challenges and constraints related to technology requirements, compatibility with existing systems, and resource estimation for conducting the feasibility study.

- **Design Phase**

1. **System Architecture Design:** Detailed **architectural** diagrams, **models**, and specifications defining the system's structure, components, interfaces, and interactions.
2. **User Interface Design:** Prototypes, wireframes, and mock-ups illustrating the visual layout, navigation flow, and user experience design of the automated code review system.

- **Construction Phase**

1. **Frontend Development:** Developed frontend components, user interfaces, and interactive elements adhering to design specifications and usability standards.
2. **Backend Development:** Implemented backend functionality, data processing logic, database management systems supporting the core features of the automated code review system.

- **Testing Phase**

1. **Unit Testing Reports:** Documentation of unit test cases, test scenarios, and test results verifying the functionality and behavior of individual software modules.
2. **Integration Testing Reports:** Documentation of integration test cases, test suites, and test outcomes validating the interoperability and integration of system components.

- **Implementation Phase**

1. **Deployment Plan:** Detailed plan outlining deployment procedures, configurations,

environments, and rollback strategies for the smooth deployment of the automated code review system.

2. **System Deployment Documentation:** Documentation of deployment processes, installation instructions, and system configuration settings facilitating the successful 'deployment and setup of the software solution.

- **Project Closure**

1. **Post-Implementation Review Report:** Evaluation and analysis of project performance, outcomes, lessons learned, and recommendations for future improvements.

2. **Documentation and Knowledge Transfer Materials:** Comprehensive documentation, user manuals, and training materials facilitating the transfer of project knowledge, insights, and best practices to stakeholders and support teams.

3. Resource Allocation

This plan encompasses the allocation of human, financial, and technological resources required to effectively carry out frontend and backend development tasks, database integration, and quality assurance activities. By delineating resource allocation, the project ensures optimal utilization of available resources to meet project objectives within specified timelines and budgetary constraints.

- **Initiation Phase**

1. **Business Analyst:** Responsible for gathering, analyzing, and documenting user requirements and project objectives.
2. **Technical Writer:** Responsible for documenting requirement specifications, use cases, and functional requirements in a clear and concise manner.

- **Design Phase**

1. **System Architect:** Responsible for designing the system architecture, defining components, interfaces, and system behaviors.
2. **UI/UX Designer:** Responsible for designing intuitive, user-centric interfaces and visual elements to enhance user experience.

- **Construction Phase**

1. **Frontend Developer:** Responsible for implementing frontend components, user interfaces, and client-side functionality.
2. **Backend Developer:** Responsible for developing backend services, data models, and server side logic supporting the system's core functionality.

- **Testing Phase**

1. **Quality Assurance Engineer:** Responsible for designing and executing test cases, analyzing test results, and ensuring software quality and reliability.

2. **Test Engineer:** Responsible for planning, executing, and managing integration tests, system tests, and acceptance tests to validate system functionality.

- **Implementation Phase**

1. **DevOps Engineer:** Responsible for configuring deployment environments, automating deployment processes, and ensuring system scalability, reliability, and availability.
2. **Deployment Team:** Responsible for coordinating and executing deployment activities, monitoring deployment progress, and resolving deployment-related issues.

- **Project Closure**

1. **Project Manager:** Responsible for overseeing project closure activities, conducting post-implementation reviews, and ensuring project deliverables meet stakeholder expectations.
2. **Technical Writer:** Responsible for documenting project outcomes, preparing project closure reports, and facilitating knowledge transfer sessions with stakeholders and support teams.

4. Critical Dependencies

The critical dependencies highlighted below are key factors that significantly impact the progress and success of the project. These dependencies represent essential milestones and tasks that must be completed to facilitate smooth progression through the project lifecycle.

- The completion of requirement documentation is critical for initiating the design phase and providing clear guidelines for system design and development.
- System design finalization is essential before commencing the construction phase to ensure alignment with user requirements and project objectives.
- Successful implementation depends on the completion of testing activities and the identification and resolution of defects and issues discovered during testing.
- The project closure process relies on the successful completion of post-implementation review, documentation, and knowledge transfer activities to ensure project outcomes meet stakeholder expectations and facilitate ongoing maintenance and support.

5. Work Breakdown Structure

For the Automated Code Review System (ACRS), a Work Breakdown Structure (WBS) is essential to systematically organize and manage the project's scope, tasks, and deliverables. The WBS for ACRS will categorize the project into distinct phases, such as requirements gathering, design, development, testing, and deployment. Each phase will be further broken down into specific tasks and activities, such as conducting user interviews, creating wireframes, coding frontend components, implementing backend functionality, performing system testing, and configuring deployment environments.

