



**CONTRACT MONTHLY  
CLAIM SYSTEM**

# Project Plan

According to Satzinger, Jackson, and Burd in their book "Systems Analysis and Design In a Changing World" (2016), every system must have a well-organized project plan in order to be developed successfully. The project can efficiently accomplish its goals and deliverables within the specified duration by using a methodical strategy. These principles guided the creation of the project plan that is shown below, which guarantees correct alignment with user requirements, realistic task scheduling, and clarity.

---

**PRESENTED BY:**

**Christinah Chitombi**

# Agenda

---

- Introduction
- Project Overview
- Objectives and Scope
- Project Phases and Timeline
- Overview of Phases
- Detailed Timeline
- Phase 1: Planning and Requirements Gathering
- Tasks and Deliverables
- Key Stakeholders
- Phase 2: Design
- Database Schema Design
- UML Class Diagram Development
- GUI Layout Design
- Phase 3: Prototype Development
- Project Environment Setup
- MVC Structure Implementation
- GUI Prototypes and Database Integration
- Phase 4: Testing and Quality Assurance
- Usability and Functional Testing
- Feedback Integration
- Phase 5: Documentation and Submission
- Final Documentation Preparation
- Submission Process
- Project Management and Monitoring
- Task Dependencies and Resource Allocation
- Risk Management
- Conclusion
- Summary of Deliverables
- Next Steps and Future Recommendations
- References
- Citing Relevant Literature and Resources





# Overview

The Claims Management and Coordination System (CMCS) prototype aims to streamline the process of submitting, reviewing, and managing claims within an academic environment. The system is designed to facilitate the work of lecturers, coordinators, and managers by providing an intuitive interface and robust backend that supports claim submissions, approvals, and document management. Leveraging a web-based platform built using .NET Core MVC, the CMCS will enhance efficiency, transparency, and accountability in the claim management process.

## Objectives

- Develop a functional prototype of the CMCS with essential features.
- Design an intuitive and user-friendly interface for all user roles.
- Implement core backend functionality using .NET Core MVC and a relational database.
- Establish secure user roles with appropriate permissions.
- Conduct comprehensive testing to validate system functionality and usability.
- Prepare detailed project documentation, including design, architecture, and user guides.

## Goals

- Streamline Claim Management:
- Enhance Transparency:
- Improve Efficiency:
- Ensure Data Integrity:

[BACK TO AGENDA PAGE](#)

# Scope

- User Roles:

---

- Claim Management:

---

- Database Design:

---

- User Interface:

---

- Prototype Development:
- Testing and Validation:
- Documentation

[BACK TO AGENDA PAGE](#)



# Timeline

[BACK TO AGENDA PAGE](#)

1 WEEK

## 01. Planning and Requirements Gathering

Briefly outline the project's scope, gather requirements, identify key features, and set a high-level schedule.

## 1 WEEK Documentation and Submission

Finalize all project documentation and prepare the submission package.

2 WEEKS

## 02. Design

Design the database schema, create UML diagrams, and develop GUI layouts to ensure clarity and alignment with project goals.

3 WEEKS

## 03. Prototype Development

Set up the project environment, implement the MVC structure, develop GUI prototypes, and integrate mock data for testing.

1 WEEKS

## 04. Testing and Quality Assurance

Conduct usability and functional testing, integrate feedback, and prepare for the final review and project submission.

8 WEEKS



CMCS

PROJECT PLAN

# DEPENDENCIES

- Requirements Clarity: Clear and detailed requirements from stakeholders are necessary to ensure the system meets user needs.
- Database Design: A well-defined database schema is required to manage data relationships and ensure data integrity.
- Technical Environment: Availability of necessary development tools (e.g., Visual Studio, SQL Server) and a stable environment for coding and testing.
- Stakeholder Involvement: Continuous feedback and approval from stakeholders, including lecturers, coordinators, and managers, to align the system with business processes.

- Resource Allocation: Sufficient time and resources (developers, testers) are needed to adhere to the project timeline and complete all phases.
- Access to Test Data: Access to realistic mock data is essential for testing and validating system functionalities.
- Integration Points: Proper integration with any external systems or APIs if required, such as authentication or payroll systems.
- Approval Processes: Clearly defined workflows for claim approval and rejection processes to ensure smooth operations within the system.
- Security and Compliance: Adherence to security standards and compliance with data protection regulations to safeguard sensitive information.

# MILESTONES

- Requirements Finalization: Complete gathering and documentation of all project requirements.
- Design Approval: Finalize and approve the database schema, UML diagrams, and GUI layouts.
- Prototype Development: Complete initial prototype with core MVC structure and basic GUI functionality.
- Testing Completion: Finish usability and functional testing with all necessary feedback incorporated.
- Documentation and Submission: Finalize all documentation and submit the completed project.

# TASKS

[BACK TO AGENDA PAGE](#)



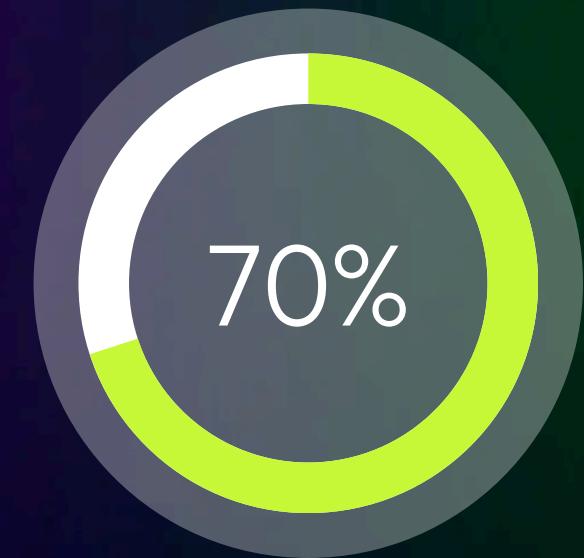
## Phase 1: Planning and Requirements Gathering (1 week)

- Define project scope and objectives
- Gather detailed requirements
- Identify key features and user roles
- Create a high-level project schedule



## Phase 2: Design (2 weeks)

- Database Schema Design (3 days)
- UML Class Diagram Design (2 days)
- GUI Layout Design (5 days)
- Project Plan Documentation (2 days)



## Phase 3: Prototype Development (3 weeks)

- Set Up Project Environment (2 days)
- Implement MVC Structure (1 week)
- Develop GUI Prototypes (1 week)
- Database Integration (3 days)
- Review and Refine Prototype (2 days)



ADD COMPANY NAME

PROJECT PLAN

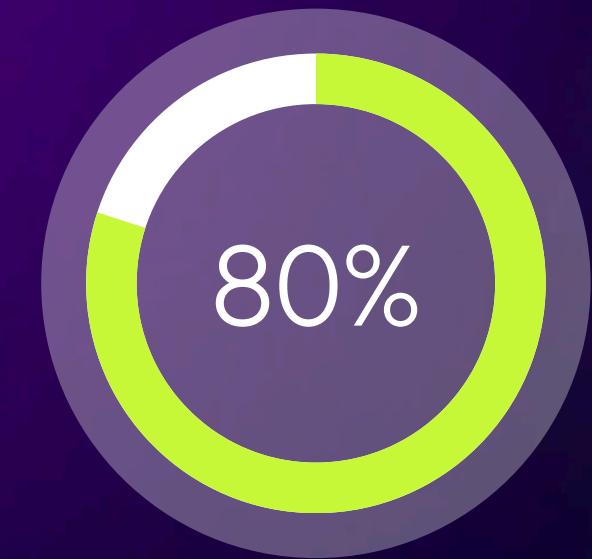
# TASKS

[BACK TO AGENDA PAGE](#)



## Phase 4: Testing and Quality Assurance (1 week)

- Usability Testing (2 days)
- Functional Testing (2 days)
- Feedback Integration (1 day)
- Final Review (1 day)



## Phase 5: Documentation and Submission (1 week)

- Complete Project Documentation (3 days)
- Prepare Submission Materials (2 days)
- Submit Project (1 day)



ADD COMPANY NAME

PROJECT PLAN



---

# For questions, reach out to:

Stella

PROJECT LEAD

---

@Stella@gmail.com

Kim

PROJECT LEAD

---

@Kim@gmail.com

# REFERENCES

---

Satzinger, W,J., Jackson,R,B and Burd,S,D. (2016).  
Systems Analysis and Design In a Changing World. 7th  
ed. Cengage Learning. ISBN: 978-1-305-11720-4I

