

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

STUDENT NO ST10448454

STUDENT FULL NAME AFEZIWE THANDANI

PROJECT TITLE CONTRACT MONTHLY CLAIM SYSTEM

PORTFOLIO OF EVIDENCE PART 1

MODULE NAME PROGRAMMING 2B

MODULE CODE PROG6212

LECTURE MRS A. PHEWA

DATE 28 AUGUST 2025

Table of Contents

[INTRODUCTION 3](#_Toc207259269)

[What is CMCS 3](#_Toc207259270)

[Project objective 3](#_Toc207259271)

[Design explanations 3](#_Toc207259272)

[Database Structure 3](#_Toc207259273)

[GUI Layout 3](#_Toc207259274)

[Assumptions and Constraints 4](#_Toc207259275)

[Assumptions 4](#_Toc207259276)

[Constraints 4](#_Toc207259277)

[UML Diagram 5](#_Toc207259278)

[Project Plan 6](#_Toc207259279)

[GUI Screenshot 7](#_Toc207259280)

# INTRODUCTION

The purpose of this project is to design and develop a .NET WPF application called the Contract Monthly Claim System (CMCS).

## What is CMCS

The contract Monthly system is designed to streamline the process of claim submission, verification, and approval. It is tailored for lectures, program coordinators, academic managers

### Project objective

My goal is to create an efficient, reliable, and user-friendly system that simplifies the claim process while ensuring transparency and accuracy for all stakeholders.

# Design explanations

The design of the Contract Monthly Claim System (CMCS) was guided by the need to make the process of submitting and approving claims clear, user-friendly, and role based. The system is built in WPF because it allows for modern and attractive user interfaces with support for custom styles and branding.

## Database Structure

The database was designed around three main groups of users: lecturers, coordinators, and managers.

* Lecturers can submit claims and upload supporting documents.
* Coordinators can review and verify claims.
* Managers can approve or reject claims after review.

The Claims table links all these roles by recording details of hours worked, rates, total amounts, and approval status. Supporting documents are stored in a separate table so that each claim can have multiple attachments. This structure ensures data integrity, separation of roles, and easy tracking of claim progress.

## GUI Layout

The GUI was designed with three dashboards to separate the views for different roles. The landing page gives users the choice to enter as a Lecturer, Coordinator, or Manager. Each dashboard is color-coded and uses icons to make navigation easier. This role-based separation ensures that users only see the features relevant to their role. For example:

* Lecturers can only submit and view their own claims.
* Coordinators can view all lecturer submissions and verify them.
* Managers can approve or reject verified claims.

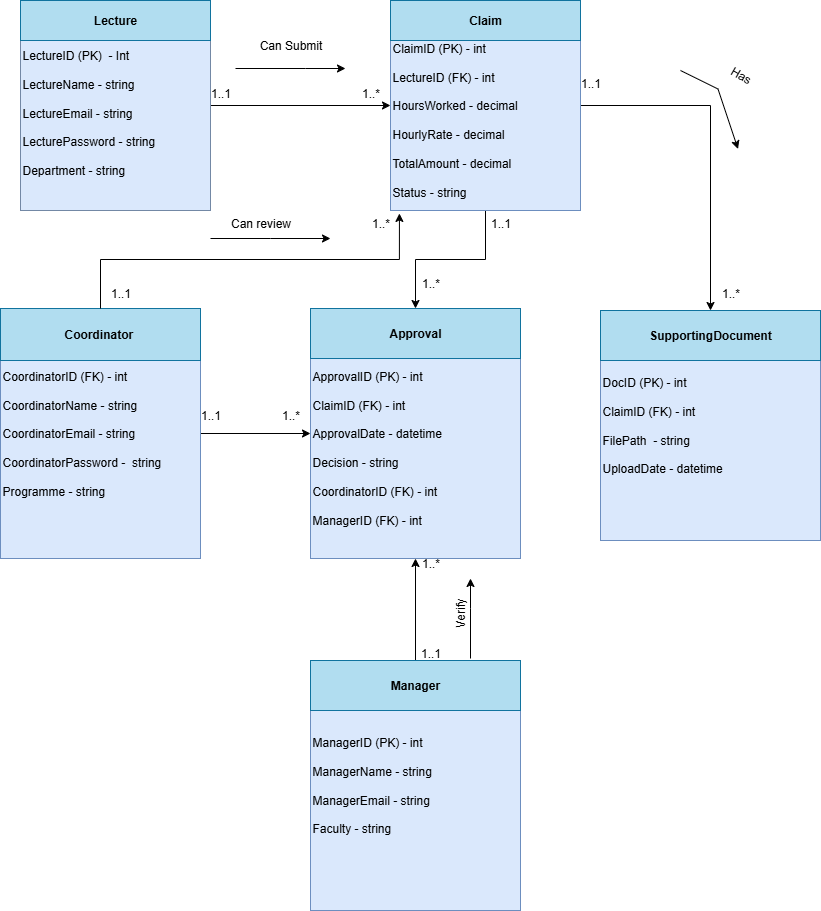
# Assumptions and Constraints

## Assumptions

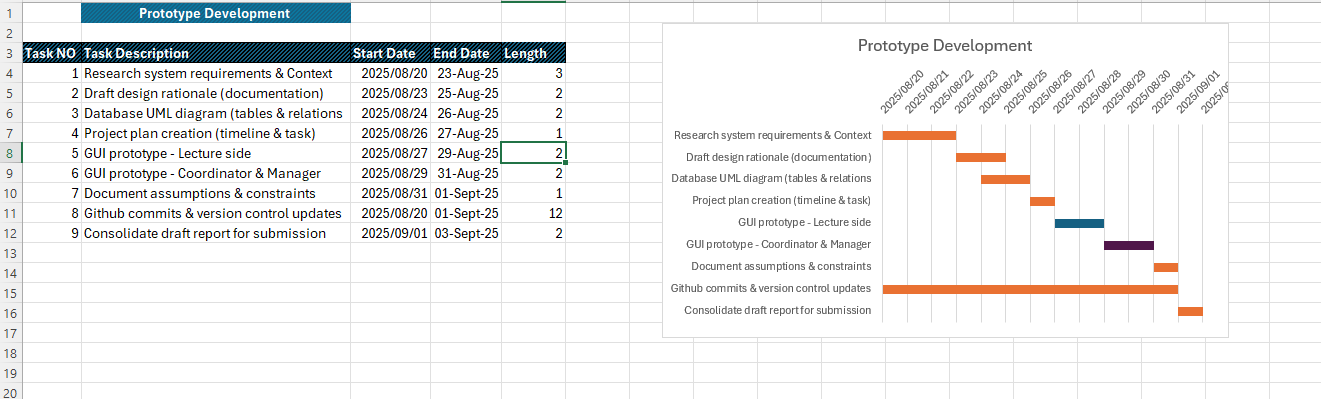
* Each role will have different permissions.
* Claims are submitted monthly and linked to uploaded documents.

## Constraints

# UML Diagram



# Project Plan



# GUI Screenshot

**Landing Page Lecture Dashboard**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a phone

AI-generated content may be incorrect.

**Coordinator Dashboard Manager Dashboard**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

# PART 2 – Improvements

In Part 2 of the project, I developed a fully functional version of the Contract Monthly Claim System (CMCS) using C# and WPF, connected to a SQL Server database. The main goal was to add real features to the system for lecturers, coordinators, and managers while maintaining a colourful and user-friendly interface. Lecturers can now submit their monthly claims by entering details such as module name, hours worked, hourly rate, and notes, with the total amount being calculated automatically. They can also upload supporting documents like PDFs or Excel files and later view all their submitted claims with updated statuses. Coordinators have their own dashboard that allows them to review lecturer claims, approve or reject them, and communicate directly with lecturers through an internal messaging system. Managers are responsible for verifying approved claims and generating professional PDF reports that summarize all processed claims, including totals, approved, and rejected claims. A messaging feature was also added to allow coordinators and managers to communicate easily about claims. The entire system was styled using a consistent color palette and enhanced with animations, glow effects, and modern dialog boxes for better user experience. The database was built using Entity Framework Core to manage users, claims, and messages efficiently with proper relationships and validation. Finally, all progress was tracked using GitHub with regular commits showing each development stage. Overall, Part 2 transformed CMCS from a prototype into a fully functional, reliable, and visually appealing system that simplifies claim submission, review, and verification for all users.