

PROG6212

# POE PART 1

ST10445866 - Botshelo Koketso  
Sekwena

IIE MSA MONASH RUIMSIG

## Table of Contents

<b><i>Documentation:</i></b> .....	<b>2</b>
<b>Design Choices:</b> .....	<b>2</b>
<b>Database Structure:</b> .....	<b>2</b>
<b>GUI/UI Layout:</b> .....	<b>3</b>
<b>Assumptions:</b> .....	<b>4</b>
<b>Constraints:</b> .....	<b>4</b>
<b><i>UML Class Diagram for Databases:</i></b> .....	<b>5</b>
<b><i>Project Plan:</i></b> .....	<b>6</b>
<b><i>GUI/UI:</i></b> .....	<b>9</b>
<b><i>Version Control:</i></b> .....	<b>17</b>
<b><i>Bibliography</i></b> .....	<b>18</b>

# Documentation:

## Design Choices:

The Contract Monthly Claim System (CMCS) was created with the primary workflow in mind: lecturers submit monthly claims, which academic managers and program coordinators assess and approve. The primary design decisions were informed by the necessity of:

- Clarity:
  - To prevent confusion and closely link the system to real-world procedures, entities were divided into **Lecturer**, **Claim**, **ClaimLine**, **SupportingDocument**, and **Approval**.
- Extensibility:
  - Roles like **Lecturer**, **Program Coordinator**, and **Academic Manager** can inherit shared characteristics (such authentication details and profile information) while maintaining role-specific activities separate by introducing a **User** base class.
- Granularity:
  - Accuracy and ambiguity are improved by separating claims into Claim and ClaimLine entities, which enable thorough computations of work sessions.
- Accountability:
  - To provide an auditable trail, the **Approval** entity was incorporated to monitor each decision made throughout the workflow.
- Supporting Documentation:
  - A distinct **SupportingDocument** entity guarantees that assertions are backed up by proof (such as contracts or time logs), which can be consulted later in the event that disagreements emerge.

## Database Structure:

A normalised relational model serves as the foundation for the suggested schema:

- Users Table:
  - central role management and authentication, featuring role-based access attributes (Manager, Coordinator, Lecturer).

- Lecturers Table:
  - Adds lecturer-specific information, like employee number and bank details, to user details.
- Claims table:
  - Which is associated with a single lecturer and may contain several claim lines and supporting documentation, is a representation of the monthly claim submission.
- ClaimLines Table:
  - Keeps track of the daily and hourly breakdown of claims. This guarantees that varied hourly rates or work sessions can be captured with flexibility.
- SupportingDocuments Table:
  - Associated with file storage metadata claims.
- Approvals Table:
  - Keeps track of comments, timestamps, and approvers for decisions made at each stage of the review process.

PLEASE NOTE: ADDITIONAL ROLES, PROCEDURES, OR PERMISSION LAYERS CAN BE INTRODUCED WITHOUT REORGANISING THE DATABASE THANKS TO THIS FLEXIBLE AND SCALABLE FRAMEWORK.

## GUI/UI Layout:

Role-based navigation is incorporated into the GUI's design. Only options pertinent to their duties will be displayed to each role:

- Lecturer Displays:
  - Dashboard: Claims list (approved, pending, and submitted).
  - Claim form submission: Features included such as a month selector, a table for claim lines (date, hours, rate, and amount), and a document upload section.
  - Claim Details: Check the status and any feedback from reviewers.
- Screens for Program Coordinators
  - Allegations Inbox: A list of unresolved allegations from lecturers.
  - Review Screen: Claim information, documents uploaded, comment section, and options to accept or deny.

- Academic Manager Screens
  - Collective Claims See: List of all claims that coordinators have authorised.
  - Approval Interface: The capacity to reach a final judgement while providing feedback.

PLEASE NOTE: WITH ITS PRIMARY CONTENT PANEL FOR FORMS AND DATA, SIDE NAVIGATION MENUS FOR ROLE-BASED TASKS, AND UNIFORM HEADERS FOR SYSTEM IDENTIFICATION, THE GUI IS PURPOSEFULLY MADE TO BE EASY TO USE AND STRAIGHTFORWARD.

## Assumptions:

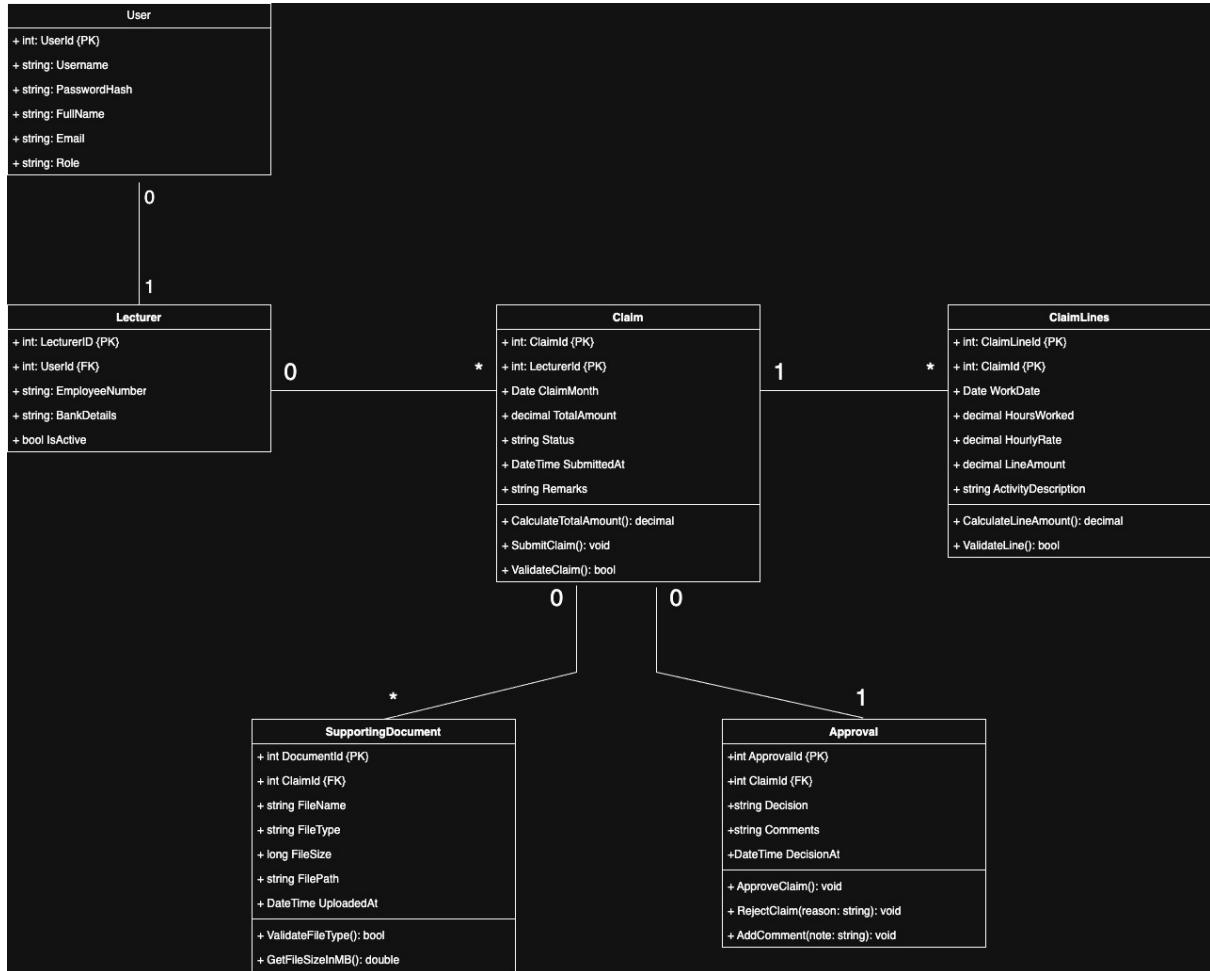
- It is expected that lecturers operate at predetermined hourly rates, which are either stored as part of each claim line or fixed per lecturer.
- Although several claim lines (per workday/session) are permitted, each professor may submit one claim per month.
- Claims must be supported by supporting documentation, such as contracts or attendance records.
- The approval process consists of two steps: the Academic Manager comes after the Program Coordinator.
- Security and authentication are made simpler during the prototype stage and will be put into practice during the functional stage.

## Constraints:

- Limitation of the prototype: This stage only produces a non-working prototype (no true authentication, no database persistence).
- Time constraints: Only necessary roles and procedures are represented due to the PoE submission cycle's scope restrictions.
- Technical scope: WPF or ASP.NET Core MVC must be used in the prototype; backend services must not yet be fully operational.
- File handling: In the prototype, documents will only be represented by metadata; upload storage will be added later.

(Jack Gido, Jim Clements, Rose Baker, Nishani Harinarain, Chukuakadibia Eresia-Eke, 2023)

# UML Class Diagram for Databases:



(John Satzinger, Robert Jackson, Stephen Burd, 2024)

# Project Plan:

## Project Objectives:

- Create a CMCS prototype that isn't working.
- Deliverables consist of WPF/MVC GUI wireframes, database schemas, UML class diagrams, and documentation of design decisions.
- Make that the timeline can be met within the normal time frame for academic submissions.

## Tasks, Dependancies, and Timeline

Task ID	Task Description	Duration	Dependancies	Deliverables	Notes
T1	Requirement Analysis	1 Day	None	Roles, procedures, and scope are described in the requirement summary document.	Determine the roles, workflows, and scope.
T2	Database Design	1 Day	T1	Draft database structure (relationships, PK/FK, tables)	Can begin the same day as T1 is finished.
T3	UML Class Diagram Design	2 Day	T2	Classes, properties, connections, and methods in a UML class diagram	Describe the relationships, classes, characteristics, and methods.

T4	GUI Design	3 Day	T3	WPF/MVC GUI wireframes for the Manager, Coordinator, and Lecturer screens	Lecturer, Coordinator, and Manager (WPF/MVC) screens
T5	Prototype Documentation	1 Day	T2	Written records of the constraints, assumptions, and design choices	Document design decisions, presumptions, and limitations
T6	Review & Refinement	1 Day	T4, T5	UML, database schema, GUI, and documentation have all been validated and improved.	Verify the documentation, database, GUI, and UML. queue up
T7	Compile Final Prototype Package	1 Day	T6	Full prototype bundle ready for submission	Compile all deliverables into a bundle that is ready for submission.

## Timeline Visualization

Day	Tasks
Day 1	Requirement Analysis
Day 2	Database Design
Day 3-4	UML Class Diagram Design
Day 5-7	GUI Design
Day 5	Prototype Documentation
Day 8	Review & Refinement
Day 9	Compile Final Prototype Package

## Dependancies

- T1 – T2:
  - Clear criteria are necessary before database design can begin.
- T2 – T3:
  - The database schema (PK/FK information) determines the UML class diagram.
- T3 – T4:
  - The class structure is necessary for layouts in GUI design.
- T2 – T5:
  - After the database structure is drafted, design choices can be documented, sometimes overlapping the GUI.
- T4 and T5 – T6:
  - Verification of consistency in UML, GUI, and documentation.
- T6 – T7
  - Following review, the final package is compiled.

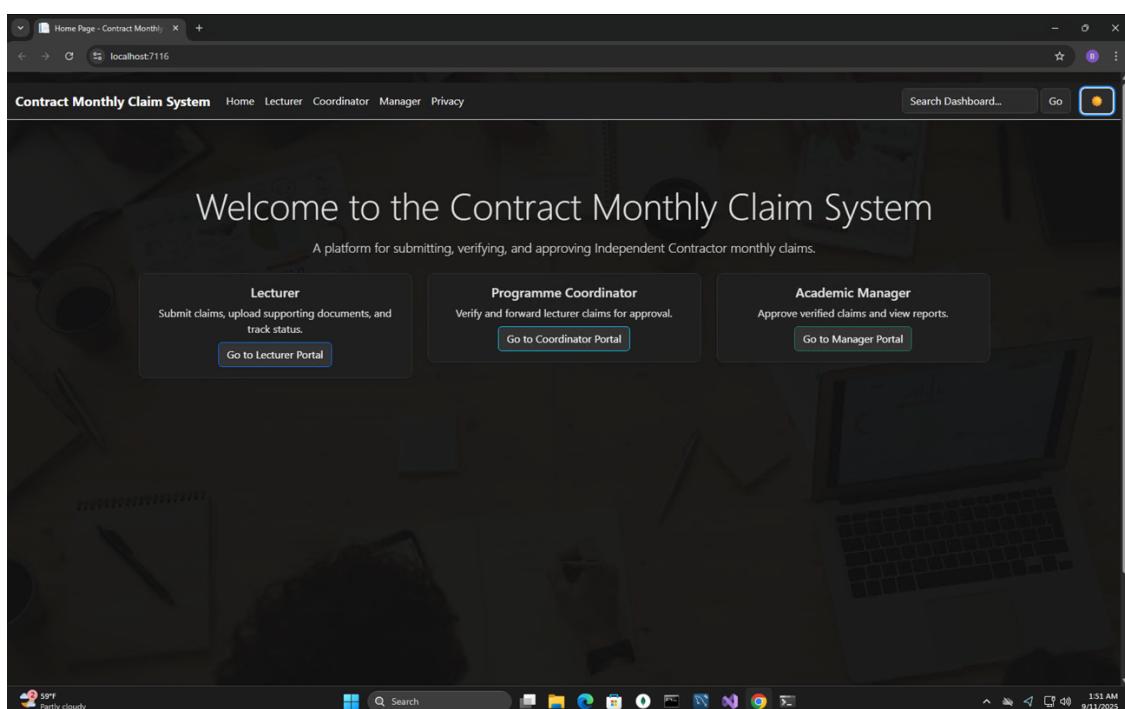
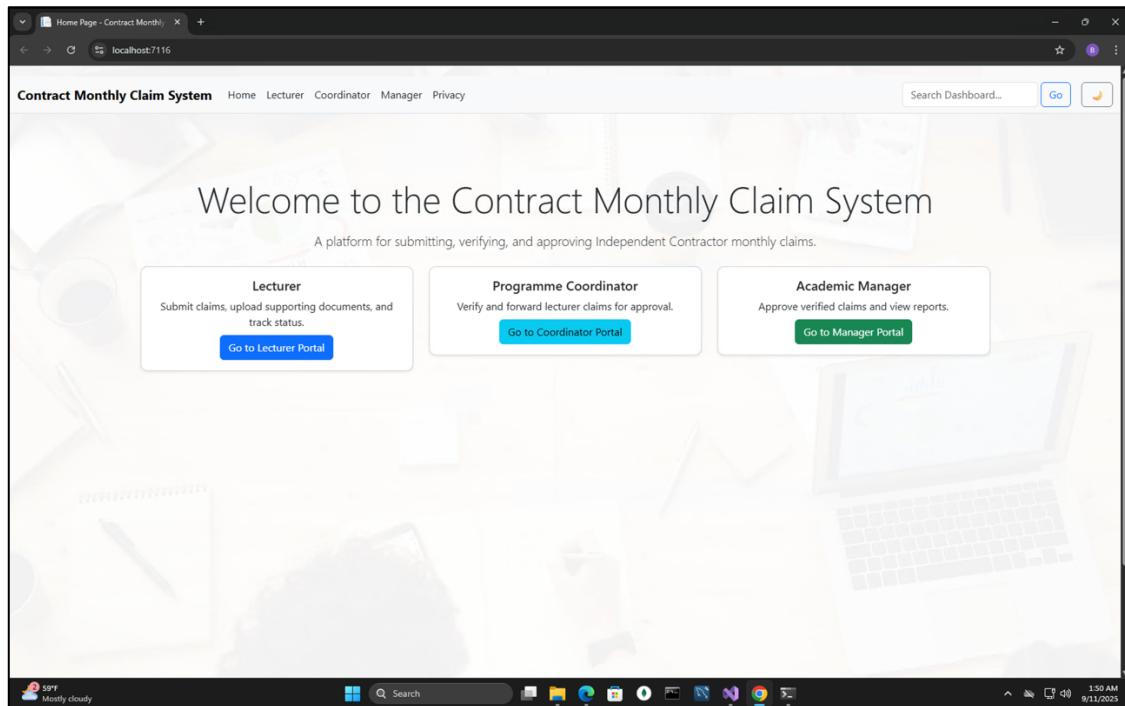
(Jack Gido, Jim Clements, Rose Baker, Nishani Harinarain, Chukuakadibia Eresia-Eke, 2023)

# GUI/UI:

## GitHub Link:

<https://github.com/SekwenaBotshelo/PROG6212-POE-ST10445866.git>

## Screenshot of the Home Page:



## Screenshot of the Lecture Page:

This screenshot shows the Lecturer Dashboard of the Contract Monthly Claim System. The interface is in light mode. At the top, there is a navigation bar with links for Home, Lecturer, Coordinator, Manager, and Privacy. A search bar labeled "Search Dashboard..." and a "Go" button are also present. Below the navigation, the title "Lecturer Dashboard" is displayed. There are three main buttons: "Submit New Claim" (green), "Upload Supporting Documents" (blue), and "View Submitted Claims" (grey). A section titled "Submitted Claims" contains a table with columns: Claim ID, Month, Total Hours, Hourly Rate, Status, and Actions. A message "No claims submitted yet." is shown below the table. A "Guidance" section with the message "Use the 'Submit New Claim' button to add claims once the backend is implemented." is also present. The desktop taskbar at the bottom shows various icons and the date/time as 1:51 AM 9/11/2025.

This screenshot shows the same Lecturer Dashboard as above, but in dark mode. The overall theme is darker, with the background being a dark grey or black. The interface elements, including the navigation bar, buttons, and sections, are styled to match the dark theme. The "Guidance" message remains the same: "Use the 'Submit New Claim' button to add claims once the backend is implemented." The desktop taskbar at the bottom shows various icons and the date/time as 1:52 AM 9/11/2025.

## Screenshot of the Co-ordinator Page:

This screenshot shows the Coordinator Dashboard of the Contract Monthly Claim System. The page has a light blue header with the title 'Coordinator Dashboard' and a sub-header 'Contract Monthly Claim System'. Below the header is a navigation bar with links for Home, Lecturer, Coordinator, Manager, and Privacy. A search bar is located in the top right corner. The main content area features three summary boxes: 'Claims Pending Verification' (0), 'Claims Verified' (0), and 'Total Claims' (0). Below these boxes are two buttons: 'Verify Claims' (blue) and 'View Details' (grey). A section titled 'Claims Overview' contains a table with columns for Claim ID, Lecturer, Month, Total Hours, Status, and Actions. A message at the bottom of this section states 'No claims available yet.' The background of the dashboard features a blurred image of a laptop, a smartphone, and some papers.

This screenshot shows the same Coordinator Dashboard as above, but in dark mode. The overall theme is dark grey/black with white text. The header, navigation bar, and search bar are identical to the light mode version. The summary boxes and the 'Verify Claims' button are also present. The 'Claims Overview' table and its message are identical. The background image is also present but appears darker due to the theme.

## Screenshot of the Manager Page:

This screenshot shows the Manager Dashboard of the Contract Monthly Claim System. The interface is in light mode. At the top, there is a navigation bar with links for Home, Lecturer, Coordinator, Manager, and Privacy. A search bar labeled 'Search Dashboard...' is also present. Below the navigation, the title 'Manager Dashboard' is displayed. Underneath the title are three buttons: 'Approve Claims' (blue), 'Reject Claims' (red), and 'View Details' (grey). A section titled 'Claims Pending Approval' follows, featuring a table with columns for Claim ID, Lecturer, Month, Status, and Actions. The table is currently empty, showing the message 'No claims pending approval.' The dashboard is set against a background of blurred office-related images like a laptop, papers, and a calculator.

This screenshot shows the same Manager Dashboard as above, but in dark mode. The overall appearance is darker, with the background images of office equipment appearing in grayscale or muted colors. The navigation bar, title, and buttons are visible but have a darker aesthetic. The 'Claims Pending Approval' section and its table remain identical to the light mode version, showing no pending claims.

# Screenshot of the Privacy Page:

**Contract Monthly Claim System** Home Lecturer Coordinator Manager Privacy Search Dashboard... Go

## Privacy Policy

The Contract Monthly Claim System (CMCS) is designed to streamline the submission, verification, and approval of monthly claims for Independent Contractor lecturers. This Privacy Policy outlines how we handle user data, even at the prototype stage, to demonstrate best practices in secure software development. For more details about the system's purpose and scope, visit the [About Page](#).

### 1. Data Collection

At this prototype stage, all data entered into the system is fictional and used solely for demonstration purposes. In a production environment, the system would collect the following data:

- Lecturer details (name, ID, contact information)
- Claim information (hours worked, hourly rate, claim month)
- Supporting documents uploaded for claims

### 2. Data Usage

Data collected is intended solely for:

- Submitting and tracking monthly claims
- Verification and approval by Programme Coordinators and Academic Managers
- Maintaining transparency and accountability in the claims process

### 3. Data Security

In a fully implemented system, all sensitive information would be stored securely and access restricted to authorized personnel only. During this prototype stage, no real personal information is stored.

### 4. Data Retention

In a production environment, claim records and associated documents would be retained only for the duration required by institutional policy and legal compliance. In this prototype, no actual data is stored beyond the demonstration session.

### 5. User Responsibilities

Users should ensure that any real data used in a production environment is accurate and compliant with institutional privacy standards. Prototype data is for demonstration purposes only.

### 6. Contact

If you have any questions about the privacy practices of CMCS, please visit our [Contact Page](#) or reach out to the system administrator or project supervisor.

© 2025 - Contract Monthly Claim System - ST10445866 - Botshelo Koketso Sekwena | [Privacy Policy](#)

**Contract Monthly Claim System** Home Lecturer Coordinator Manager Privacy Search Dashboard... Go

## Privacy Policy

The Contract Monthly Claim System (CMCS) is designed to streamline the submission, verification, and approval of monthly claims for Independent Contractor lecturers. This Privacy Policy outlines how we handle user data, even at the prototype stage, to demonstrate best practices in secure software development. For more details about the system's purpose and scope, visit the [About Page](#).

### 1. Data Collection

At this prototype stage, all data entered into the system is fictional and used solely for demonstration purposes. In a production environment, the system would collect the following data:

- Lecturer details (name, ID, contact information)
- Claim information (hours worked, hourly rate, claim month)
- Supporting documents uploaded for claims

### 2. Data Usage

Data collected is intended solely for:

- Submitting and tracking monthly claims
- Verification and approval by Programme Coordinators and Academic Managers
- Maintaining transparency and accountability in the claims process

### 3. Data Security

In a fully implemented system, all sensitive information would be stored securely and access restricted to authorized personnel only. During this prototype stage, no real personal information is stored.

### 4. Data Retention

In a production environment, claim records and associated documents would be retained only for the duration required by institutional policy and legal compliance. In this prototype, no actual data is stored beyond the demonstration session.

203 AM 9/11/2025

SPF Partly cloudy

Search

203 AM 9/11/2025

following data:

- Lecturer details (name, ID, contact information)
- Claim information (hours worked, hourly rate, claim month)
- Supporting documents uploaded for claims

### 2. Data Usage

Data collected is intended solely for:

- Submitting and tracking monthly claims
- Verification and approval by Programme Coordinators and Academic Managers
- Maintaining transparency and accountability in the claims process

### 3. Data Security

In a fully implemented system, all sensitive information would be stored securely and access restricted to authorized personnel only. During this prototype stage, no real personal information is stored.

### 4. Data Retention

In a production environment, claim records and associated documents would be retained only for the duration required by institutional policy and legal compliance. In this prototype, no actual data is stored beyond the demonstration session.

### 5. User Responsibilities

Users should ensure that any real data used in a production environment is accurate and compliant with institutional privacy standards. Prototype data is for demonstration purposes only.

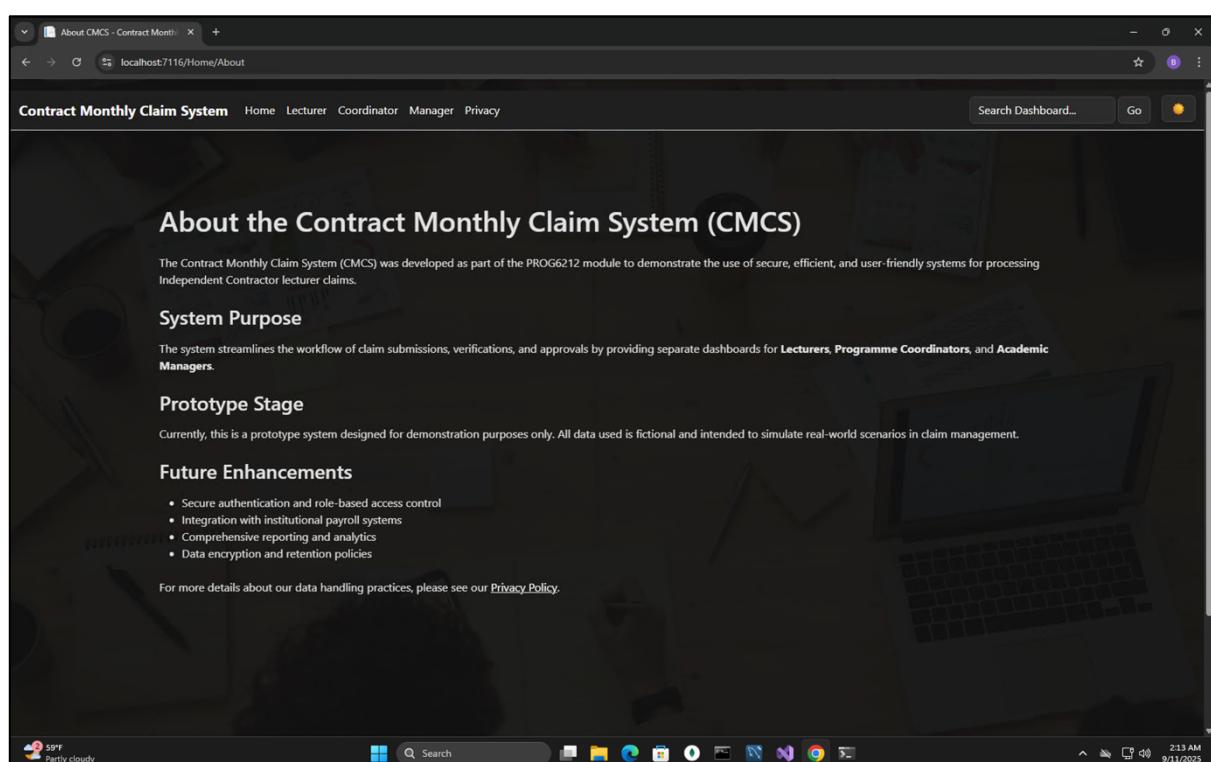
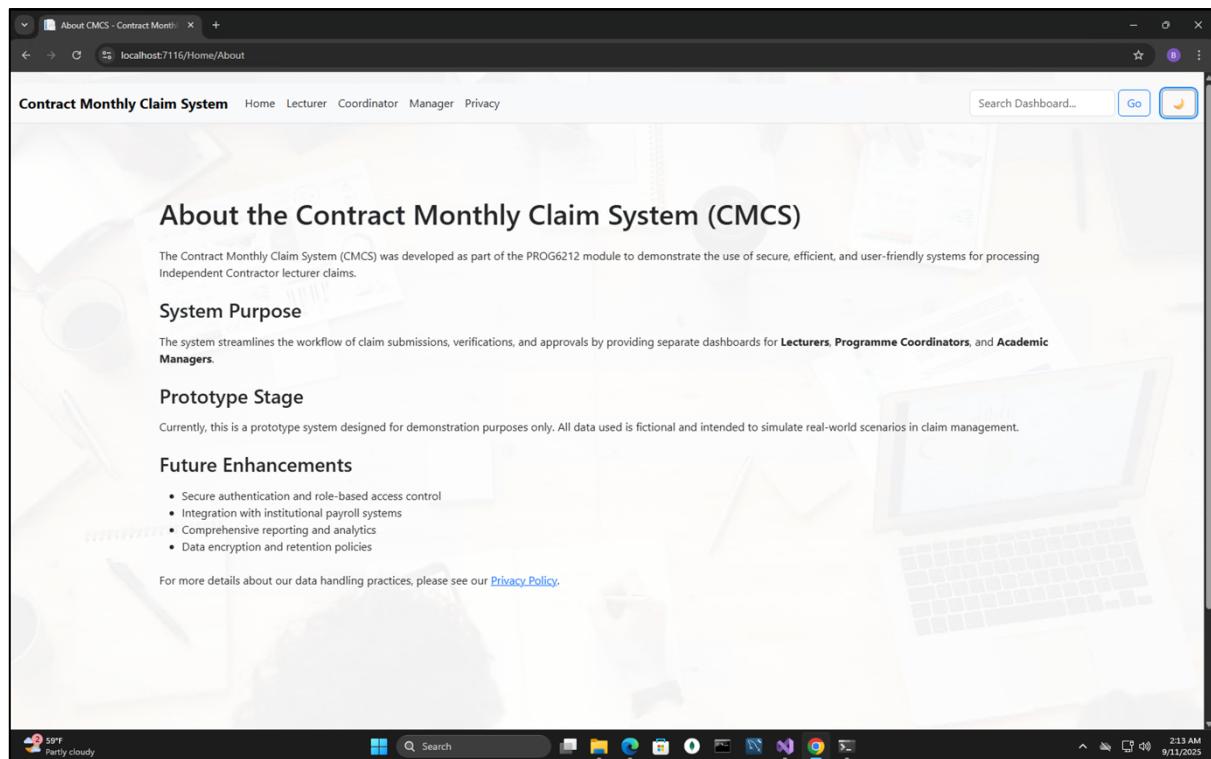
### 6. Contact

If you have any questions about the privacy practices of CMCS, please visit our [Contact Page](#) or reach out to the system administrator or project supervisor.

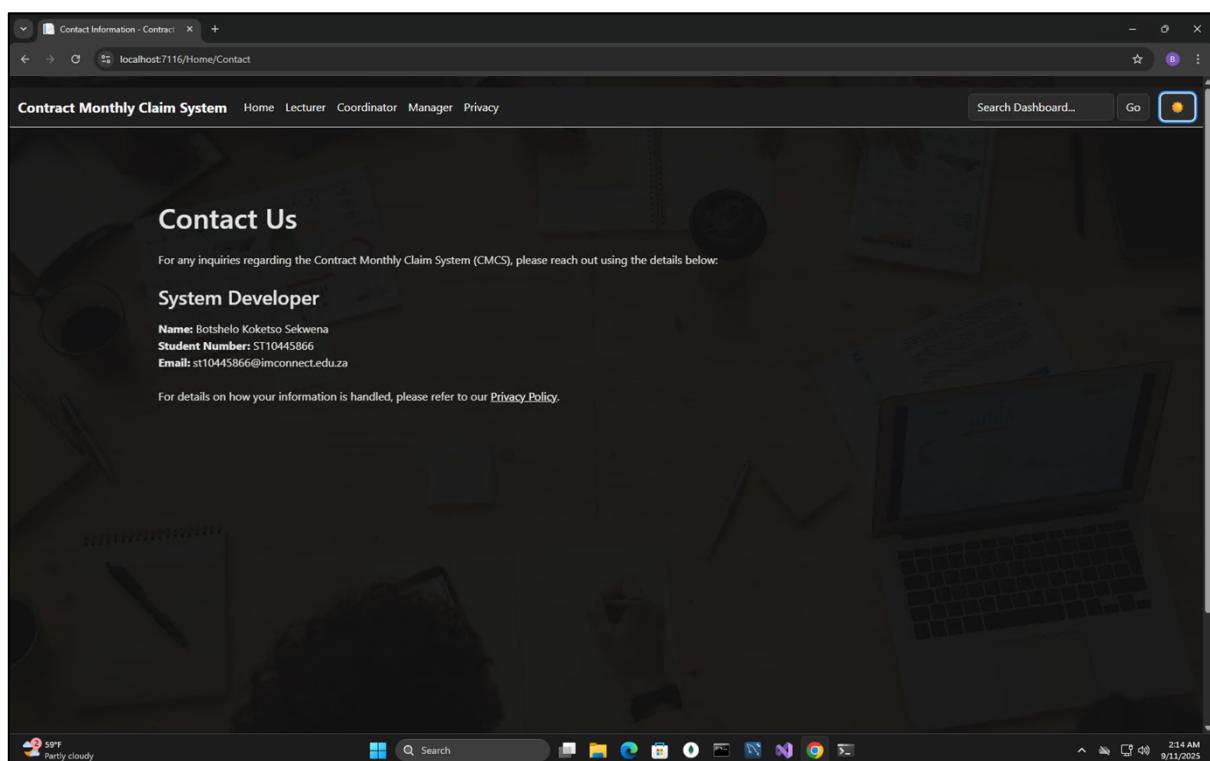
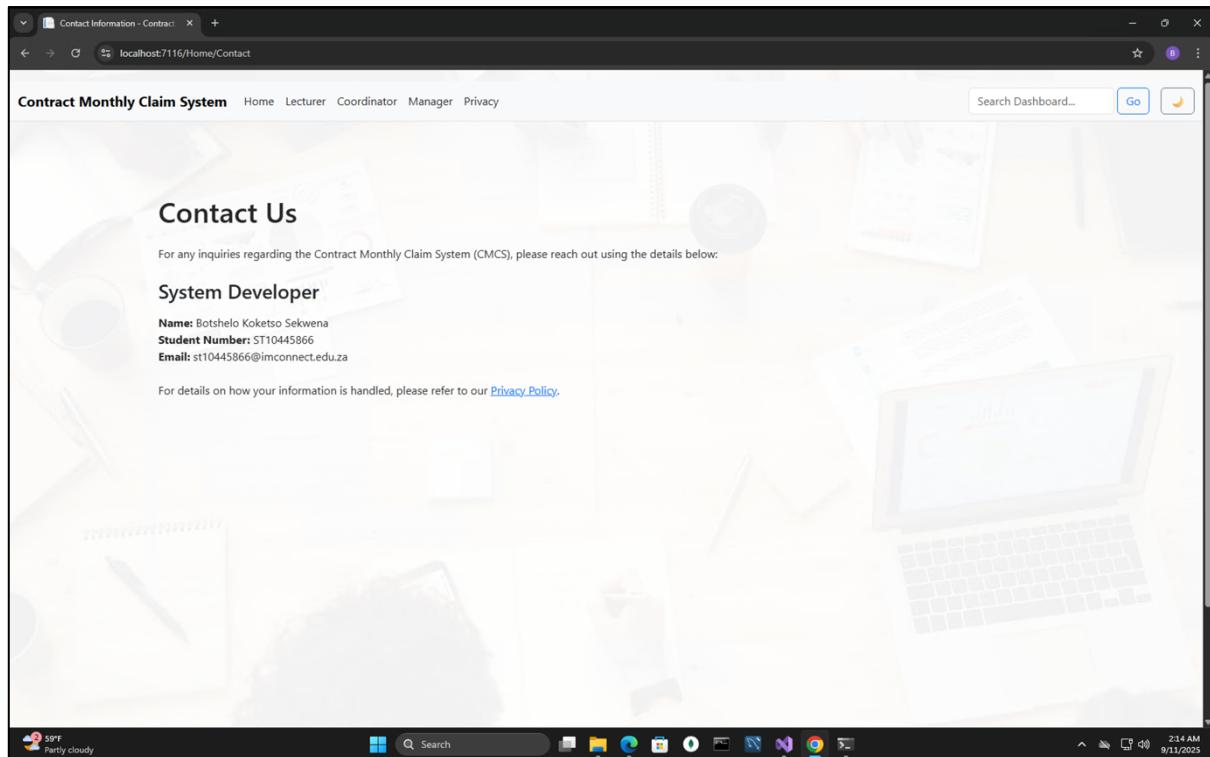
© 2025 – Contract Monthly Claim System - ST10445866 - Botshelo Koketsu Sekwena | [Privacy Policy](#)

204 AM 9/11/2025

## Screenshot of the About Page:



# Screenshot of the Contact Page:



(fauxels, 2019) (Andrew Troelsen, Phil Japikse, 2022) (ChatGPT, 2025)

# Version Control:

The screenshot shows the GitHub interface for a repository named PROG6212-POE-ST10445866. The user is viewing the 'Commits' page for the 'master' branch. The commits are listed in chronological order from September 11, 2025, to September 15, 2025. Each commit includes the author's name (SekwenaBotshelo), the date, a brief description, and a green 'Verified' badge with a unique commit hash.

Date	Description	Author	Hash
Sep 15, 2025	Update README.md	SekwenaBotshelo	7b62120
Sep 15, 2025	Update README.md	SekwenaBotshelo	1fe56fc
Sep 15, 2025	Update README.md	SekwenaBotshelo	762deb8
Sep 14, 2025	Adding UML Diagram To Git Repository	SekwenaBotshelo	a814af5
Sep 14, 2025	Adding Word/PDF Document To Git Repository	SekwenaBotshelo	3edfb41
Sep 11, 2025	Adding the Background Watermark JPEG for application	SekwenaBotshelo	f91d824
Sep 11, 2025	Updating the site.css	SekwenaBotshelo	fc96395
Sep 11, 2025	Updating the Layout cshtml	SekwenaBotshelo	ed63cef
Sep 11, 2025	Creating and Updating the Manager File and cshtml's	SekwenaBotshelo	ffff9f8e

This screenshot shows the same GitHub commit history for the 'master' branch of the repository PROG6212-POE-ST10445866. The commits are identical to the ones shown in the first screenshot, indicating no new commits were made between the two snapshots.

Date	Description	Author	Hash
Sep 15, 2025	Update README.md	SekwenaBotshelo	7b62120
Sep 15, 2025	Update README.md	SekwenaBotshelo	1fe56fc
Sep 15, 2025	Update README.md	SekwenaBotshelo	762deb8
Sep 14, 2025	Adding UML Diagram To Git Repository	SekwenaBotshelo	a814af5
Sep 14, 2025	Adding Word/PDF Document To Git Repository	SekwenaBotshelo	3edfb41
Sep 11, 2025	Adding the Background Watermark JPEG for application	SekwenaBotshelo	f91d824
Sep 11, 2025	Updating the site.css	SekwenaBotshelo	fc96395
Sep 11, 2025	Updating the Layout cshtml	SekwenaBotshelo	ed63cef
Sep 11, 2025	Creating and Updating the Manager File and cshtml's	SekwenaBotshelo	ffff9f8e

# Bibliography

fauxels. (2019, October 22). *Pexels*. Retrieved from Pexels:

<https://www.pexels.com/photo/photo-of-people-having-meeting-3183186/>

Andrew Troelsen, Phil Japikse. (2022). *Pro C# 10 with .NET 6*. Chambersburg:  
Apress.

John Satzinger, Robert Jackson, Stephen Burd. (2024). *Systems Analysis and Design In A Changing World*. Boston: Cengage.

Jack Gido, Jim Clements, Rose Baker, Nishani Harinarin, Chukuakadibia Eresia-Eke. (2023). *Project Management in South Africa*. Hampshire: Cengage.

ChatGPT. (2025). *ChatGPT*. Retrieved from ChatGPT: <https://chatgpt.com>