ST10163788 **PEACE SALOMY PHIRI**

PROG6212 PART 1

PROGRAMMING 2B

SEPTEMBER 10, 2024

IIEMSA RUIMSIG CAMPUS

POE PART 1

Table of Content

ACADEMIC HONESTY DECLARATION

Please complete the Academic Honesty Declaration below.
Please note that your assessment will not be marked, and you will receive 0% if you have not completed ALL aspects of this declaration.

Declaration

I have read the assessment rules provided in this declaration.	Yes
This assessment is my own work.	Yes
I have not copied any other student's work in this assessment.	Yes
I have not uploaded the assessment question to any website or App offering assessment assistance.	Yes
I have not downloaded my assessment response from a website.	Yes
I have not used any AI tool without reviewing, re-writing, and re-working this information, and referencing any AI tools in my work.	Yes
I have not shared this assessment with any other student.	Yes
I have not presented the work of published sources as my own work.	Yes
I have correctly cited all my sources of information.	Yes
My referencing is technically correct, consistent, and congruent.	Yes
I have acted in an academically honest way in this assessment.	Yes

Documentation

My Design Choices

My design of the CMCS prototype revolves around three key principles mainly, usability, data integrity, and efficiency. The systems development prioritizes user experience design while ensuring that data and information transfer between Lectures, Program Coordinators, and Academic Managers remains protected and effective.

1. **Usability**

- The Lecturer interface is designed for simplicity, allowing easy claim submissions by entering work hours, hourly rates, and uploading documents. This minimizes errors and streamlines the process.
- The Programme Coordinator and Academic Manager dashboard provides a clear, filtered view of submitted claims, enabling quick approval or rejection, making decision-making faster and more efficient**.**

2. **Data Integrity**

- The relational database design uniquely identifies each lecturer, ensuring claims are tied to their individual ID and properly tracked. Supporting documents are required for legitimacy, with file size limits in place to manage storage and prevent misuse.

3. **Efficiency**

- The system calculates claims automatically, saving time and reducing errors. It ensures a smooth approval process by notifying coordinators and managers about pending claims. The role-based dashboard helps users focus on their tasks, making the process more efficient and reducing admin work.

Assumptions and Constraints

A. **Assumptions**
   - Each lecturer is identified by a unique LecturerID, ensuring claims are linked to the right person. All claims must include a supporting document in PDF or image format.
   - Programme Coordinators and Academic Managers can approve or reject claims based on the provided evidence before payments are processed.

B. **Constraints**

   - **File Size Limitation:** Supporting document uploads are limited to a maximum of 5MB per file to prevent storage overload and ensure faster upload times. This ensures that the system can handle a large number of claims efficiently without being burdened by excessive data.
   - **Data Integrity:** system requires each field (hours worked, hourly rate, supporting document) to be filled out completely before a claim can be submitted. This prevents incomplete submissions and ensures that all necessary information is provided, thereby maintaining the quality and accuracy of the data.

Database Structure

The CMCS database is designed using a relational structure to ensure data consistency, integrity and scalability. Every entity in the system - such as Lecturer, Program Coordinator, Academic Manager, Claim or Supporting Document- is represented uniquely enabling clear relationships and

management of information.

I. **Entities and Relationships**

- **Lecture:** Holds details about lecturers (lecturerID, first name, last name, hourly rate) necessary for calculating claims.
- **Claim:** Records information related to claims (claimID, hours worked, total amount, status) and connects to lecturers, Programme Coordinators, and Academic Managers for the review and approval process.
- **Programme Coordinator:** Contains information (coordinatorID, first name, last name, email) about coordinators who initially assess claims.
- **Academic Manager:** Maintains details (managerID, first name, last name, email) of managers responsible for making the final decisions on claims.
- **Supporting Document:** Ensures that each claim includes a document (documentID, linked to claimID) for validation and auditing purposes.

GUI Layout

The graphical user interface (GUI) for CMCS has been designed to be intuitive and user-friendly, with a focus on ease of use for both lecturers and administrators.

a. Lecture View

- **Purpose:** Enables lecturers to submit their claims with minimal effort.
- **Design:** The lecturer's interface includes fields for "hours worked," "hourly rate," and a file upload section for attaching supporting documents. Once the form is completed, the user can submit their claim with just one click.
- **Rationale:** This simple and clear interface reduces user errors and ensures that lecturers can quickly complete their claims, thereby reducing friction in the submission process.

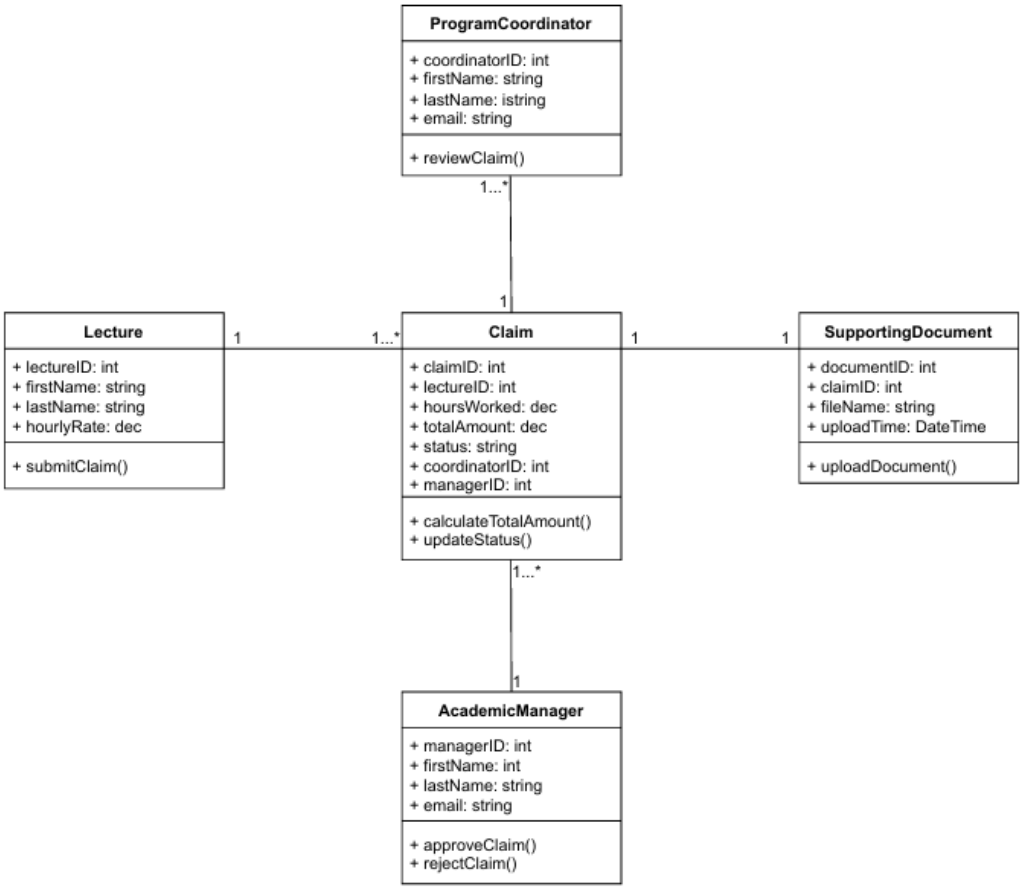
b. Programme Coordinator/Academic Manager Dashboard

- **Purpose:** This feature enables administrators to review, approve or reject claims.
- **Design:** The dashboard presents a list of all claims along with their current status, whether Pending, Approved or Rejected. The administrator can click on any claim to view its details including supporting documents and make decisions either approving or rejecting it.
- **Rationale:** The dashboard has been optimized to allow coordinators and managers to focus on verifying and processing claims without the need for extensive navigation. Filter options are available to help with sorting the claims.

Rationale Behind Design Decisions

- ✓ Every decision made while designing CMCS is motivated by the necessity to offer a system that is efficient and user-friendly. The unique separation of entities in the database ensures that the system can expand and adjust itself to future modifications, like incorporating additional roles or new claim categories.
- ✓ The user interface has been designed to reduce confusion and keep users focused on the task at hand. The inclusion of automated calculations, like determining total claim amounts, helps eliminate manual errors which improves accuracy and reliability within the system.

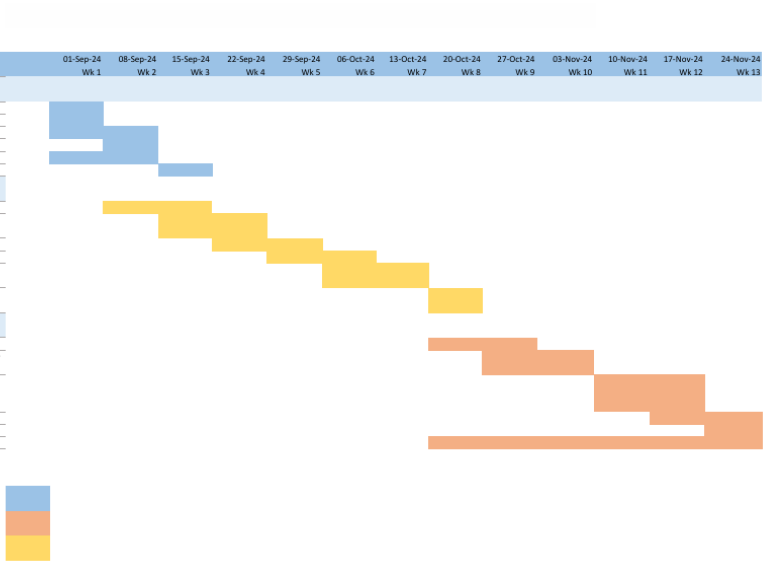
UML Class Diagram



Project Plan

Project Name:	Prog POE
Project Start:	01-Sep-24
End Date:	22-Nov-24
Weeks In Progress:	Wk 13

Task	Start Date	End Date	Duration	Dependencies
Part 1: Project Planning and Prototype Development				
Documentation	2024/09/01	2024/09/02	2 days	-
UML Class Diagram	2024/09/03	2024/09/05	3 days	Documentation
Project Plan	2024/09/06	2024/09/08	3 days	UML Class Diagram
GUI Design	2024/09/09	2024/09/10	2 days	Project Plan
Version Control (Ongoing	2024/09/01	2024/09/10	Ongoing	-
Final Submission	2024/09/10	2024/09/10	1 day	GUI Design
Part 2: Implement a Prototype Web Application				
Claim Submission	2024/09/11	2024/09/17	7 days	Final Submission
Claim Verification and Approval	2024/09/18	2024/09/24	7 days	Claim Submission
Document Upload	2024/09/25	2024/09/30	6 days	Claim Submission
Claim Status Tracking	2024/10/01	2024/10/06	6 days	Claim Submission
Consistent and Reliable Information	2024/10/07	2024/10/13	7 days	Claim Status Tracking
Final Submission	2024/10/18	2024/10/18	1 day	Consistent and Reliable Information
Part 3: Automation and Finalization				
Automation of Lecturer View	2024/10/19	2024/10/28	10 days	Final Submission
Automation of Coordinator and Manager View	2024/10/29	2024/11/07	10 days	Automation of Lecturer View
Automation of HR View	2024/11/08	2024/11/17	10 days	Automation of Coordinator and Manager View
PowerPoint Presentation	2024/11/18	2024/11/21	4 days	Automation of HR View
Final Submission	2024/11/22	2024/11/22	1 day	PowerPoint Presentation
Version Control (Ongoing)	2024/10/19	2024/11/22	Ongoing	-



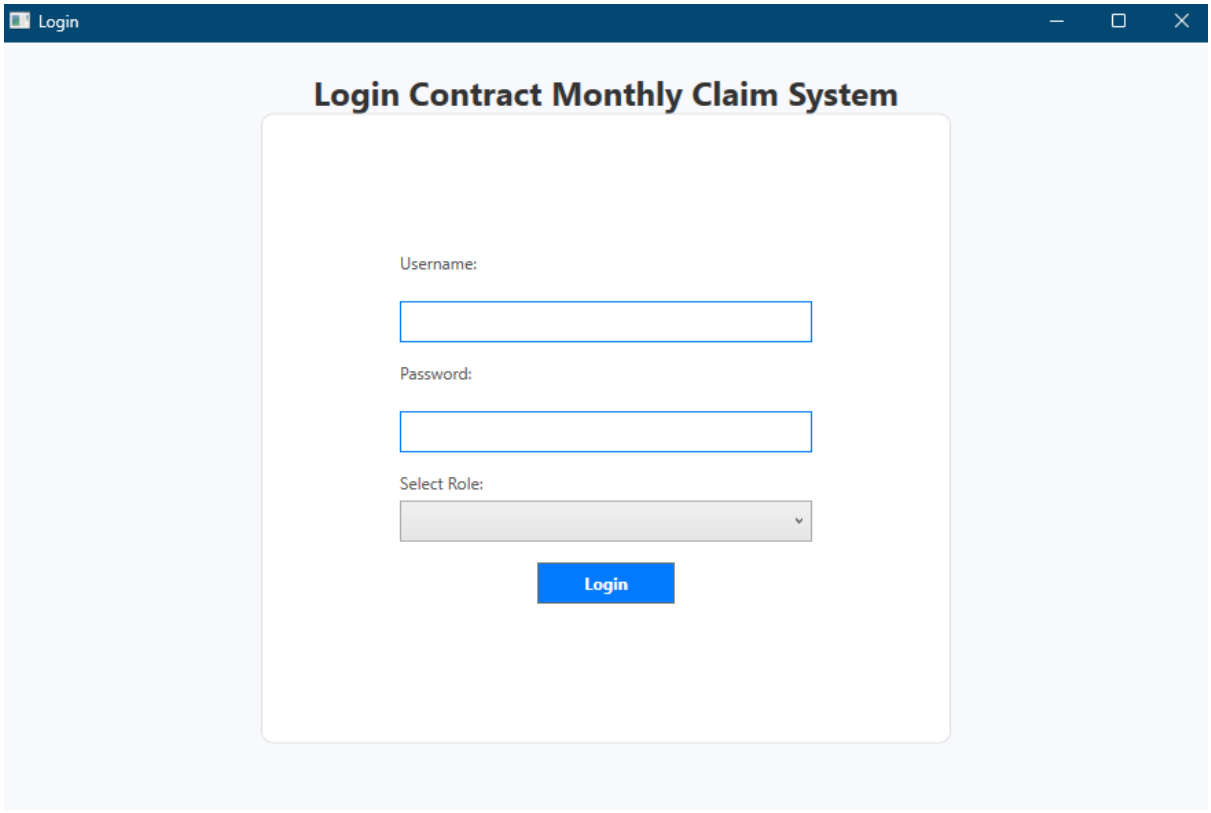
KEY

Part 1 Duration on Project Parts

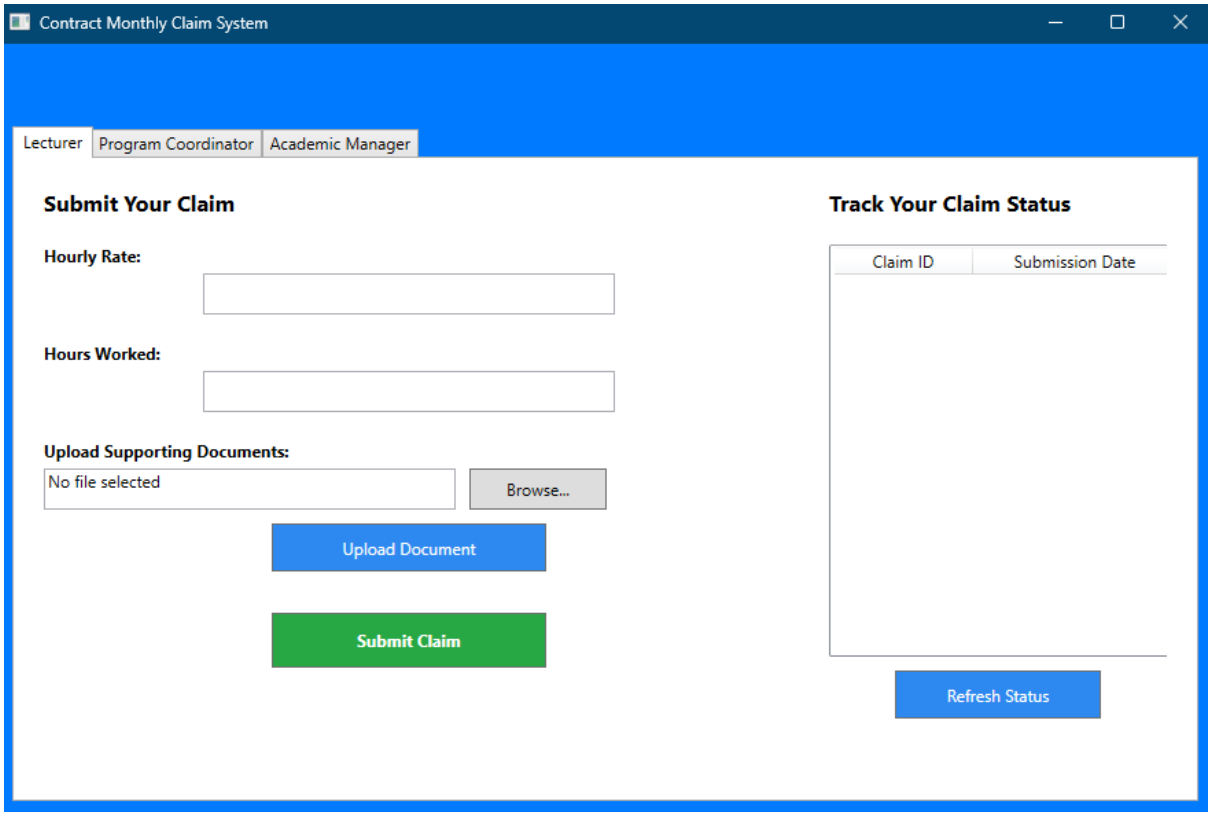
Part 2 Duration on Project Parts

Part 3 Duration on Project Parts

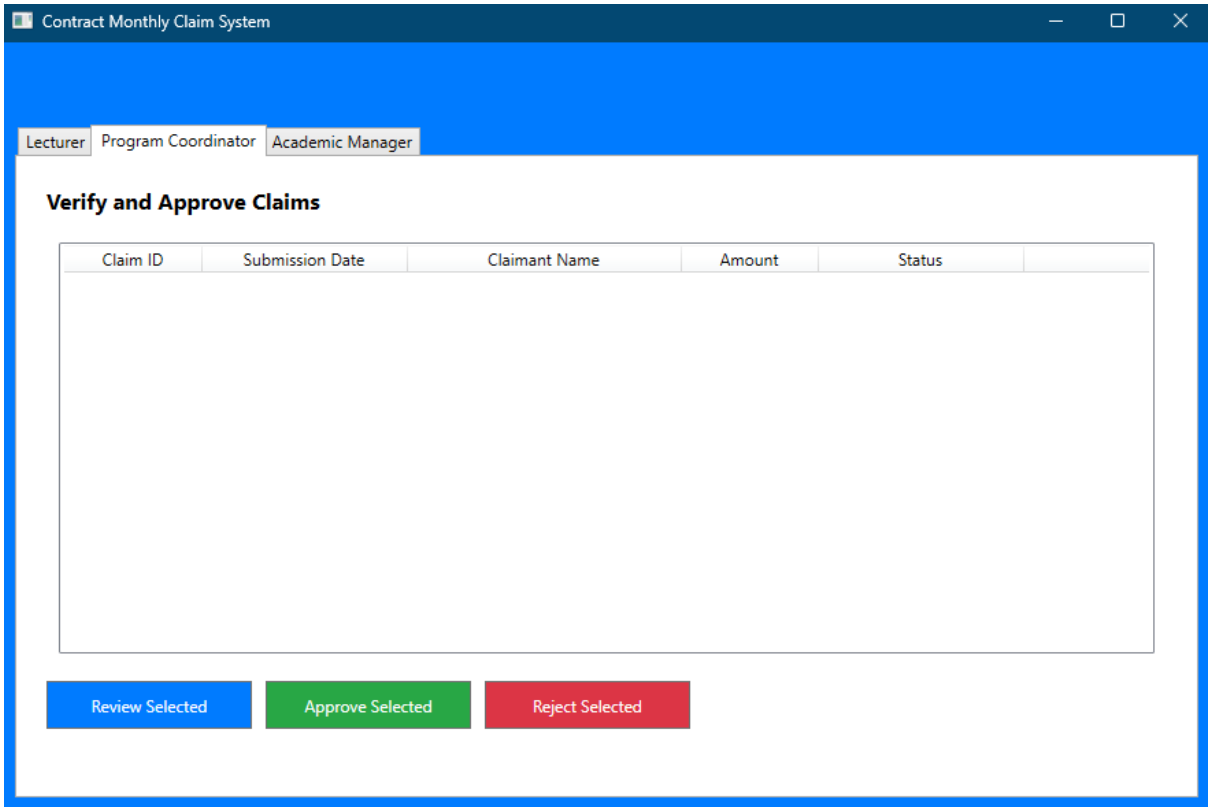
GUI

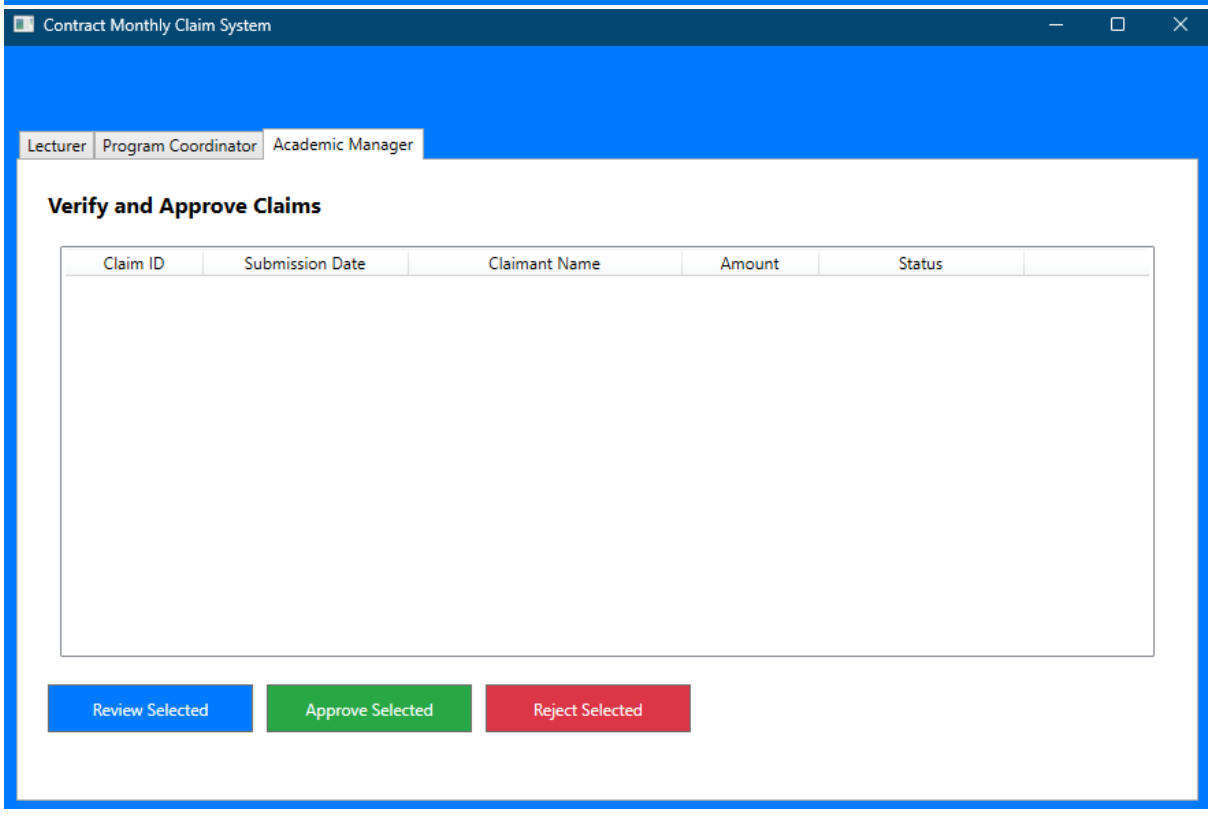


This is the login page where any of the users either, Lecture, Program Coordinator, or the Academic Manager can login using their username and password as well as selecting which role best describes them. When they log in it takes them to the role dashboard they selected.



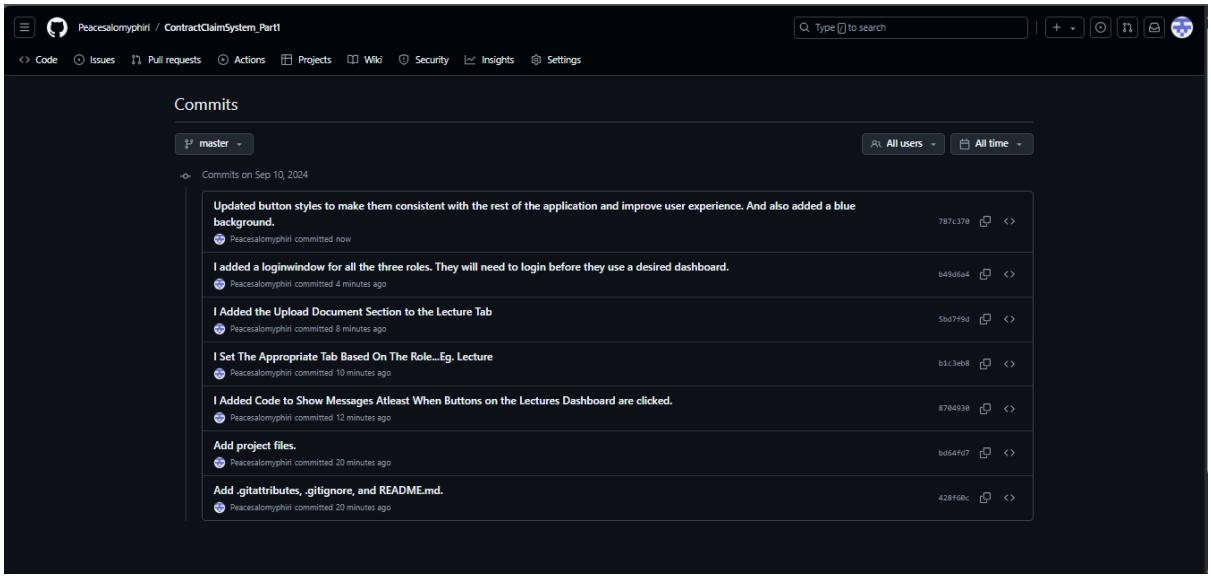
This is the lecture dashboard, it allows the user to enter the hours worked, the hourly rate, upload supporting documents, submit claims as well as track the status of the claims they submitted already.





The above two shots are the interfaces for the program coordinator and the academic manager. They can both review, approve or reject the selected claim. The submitted claims will show in the grid above the decision buttons.

Commits



GitHub Links

https://github.com/Peacesalomyphiri/ContractClaimSystem_Part1.git **Personal Account**

https://github.com/IIEMSA/poe-part-1-Peacesalomyphiri.git **School Provisioned Account**

THE END OF PART 1 PROGRAMMING 2B