Cover Page

Module: PROG6212

Programming 2B

Student Number: ST10439309

Name: Mofenyi Kubheka

Table of Contents

Report

Purpose & Scope

The Contract Monthly Claim System (CMCS) allows part-time lecturers to submit monthly teaching claims with supporting documents. Programme Coordinators (PCs) and Academic Managers (AMs) verify/approve claims, while lecturers can track status until settlement. The prototype in this phase is front-end only; workflows and persistence will be implemented later.

Key Design Choices

I selected **ASP.NET Core MVC** for a clean separation of concerns and easy evolution into a full web app later. The data model centers on **Claim** and **ClaimLineItem** to capture multiple sessions per month, with **SupportingDocument** for uploads. **User** carries a **Role** (Lecturer, PC, AM) to drive UI visibility and future authorization. **ClaimStatus** is an enum (Draft, Submitted, UnderReview, Approved, Rejected, Settled). An **AuditLog** entity records state changes for transparency.

Database Structure (Conceptual)

Entities: User, LecturerProfile, Claim, ClaimLineItem, SupportingDocument, AuditLog. Relationships:

- User 1—1 LecturerProfile (only for lecturer role)
- User 1—* Claim (a lecturer owns claims)
- Claim 1—* ClaimLineItem
- Claim 1—* SupportingDocument
- Claim 1—* AuditLog

GUI Layout (Prototype)

- **Lecturer**: Dashboard (claim list + status chips), "New Claim" form (month, module, totals), Line-items table (date, hours, rate, amount), "Upload Documents" (drag-and-drop), "Submit" button.
- **PC/AM**: Review queue, claim detail view with read-only items, approve/reject actions (disabled in prototype), history panel showing audit entries.
- **Global**: Consistent header, breadcrumbs, and status badges.

Assumptions & Constraints

- Lecturers submit one claim per module per month.
- Files are limited to PDFs/images; virus scanning and storage quotas are future work.

- Authentication/authorization and integrations (HR/Finance) are out of scope for this stage.

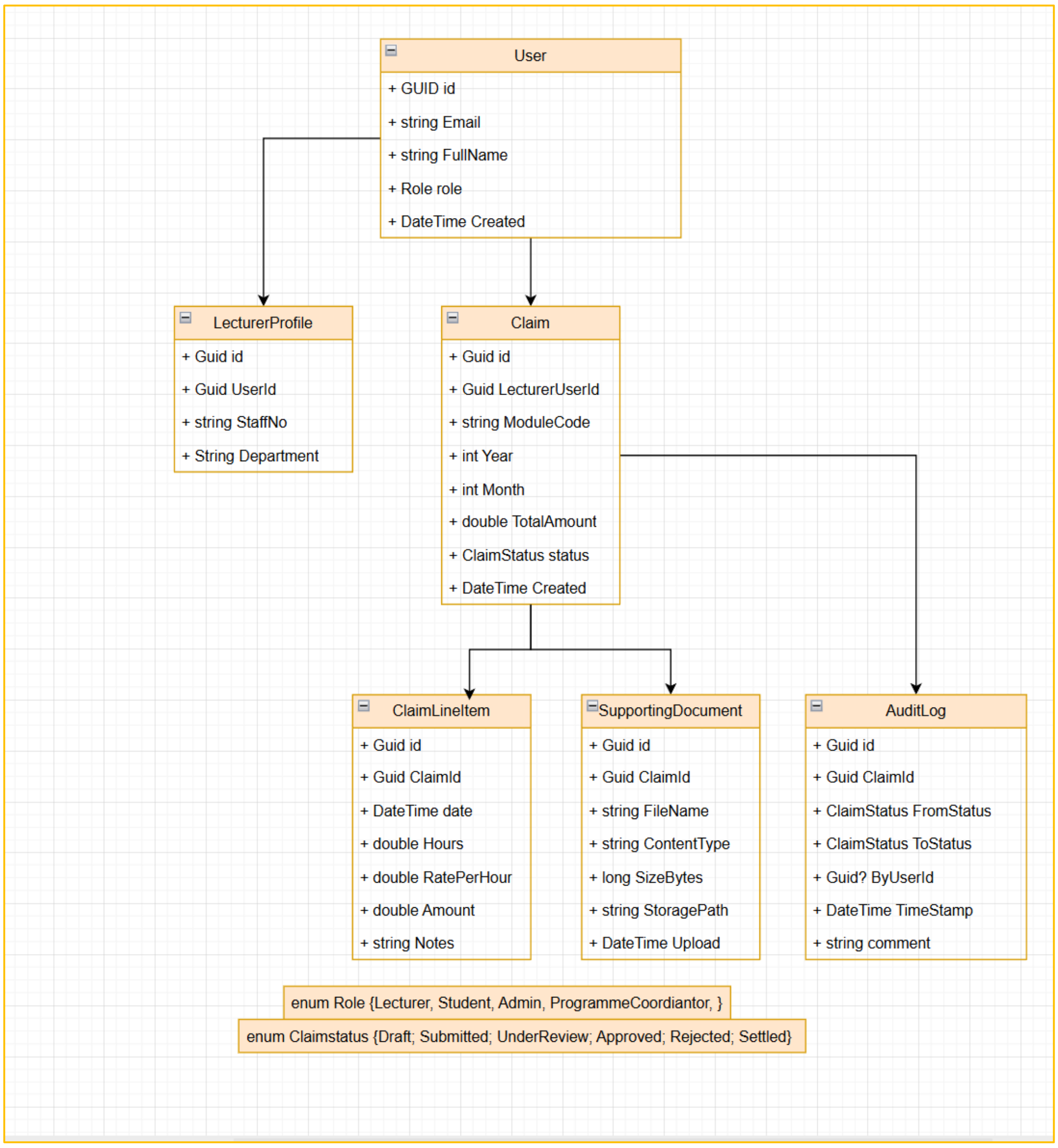
Why this design?

The model normalizes repeating data (line items), supports attachments, and provides a clear approval lifecycle. MVC lets us ship a believable UI now and wire up controllers/repositories later with minimal rework.

Version Control

Frequent small commits with descriptive messages (see plan below) ensure traceability of design decisions and UI iterations.

UML Class Diagram



Project Plan

Timeline (2 weeks suggestion):

- Day 1–2: Finalize requirements & assumptions; draft UML and GUI wireframes.
- Day 3–4: Create ASP.NET Core MVC project; scaffold layout, header, nav, status chips.
- Day 5–6: Build Lecturer pages: Dashboard, New Claim (static form), Line-items (client-side only).
- Day 7: Build Uploads page (no backend; fake list UI).
- Day 8–9: Build PC/AM review screens (read-only views).
- Day 10: Add placeholder controllers/models, sample seed JSON, and mock data service.
- Day 11: Polish UI, accessibility pass, responsiveness.
- Day 12: Prepare report, screenshots, and README; ensure min. 5 descriptive commits.

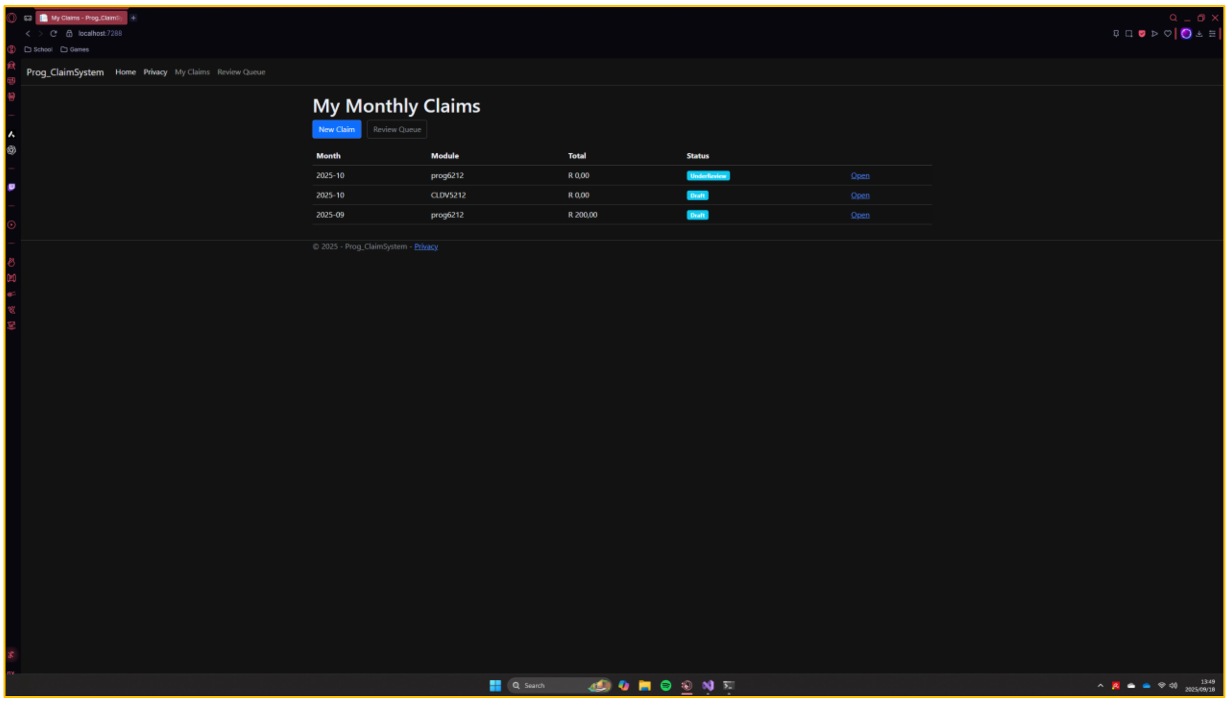
Dependencies: Finish UML before UI fields; base layout before sub-pages; mock data service before listing pages.

Risks & Mitigation: Scope creep → stick to non-functional UI; time loss → reuse Bootstrap/TagHelpers; unclear rules → document assumptions.

MVC front-end scaffold (static, non-functional)

The screenshots provided are in order of succession of how the system will operate on a basic level.

- This is the screen anyone will see when they launch the programme.

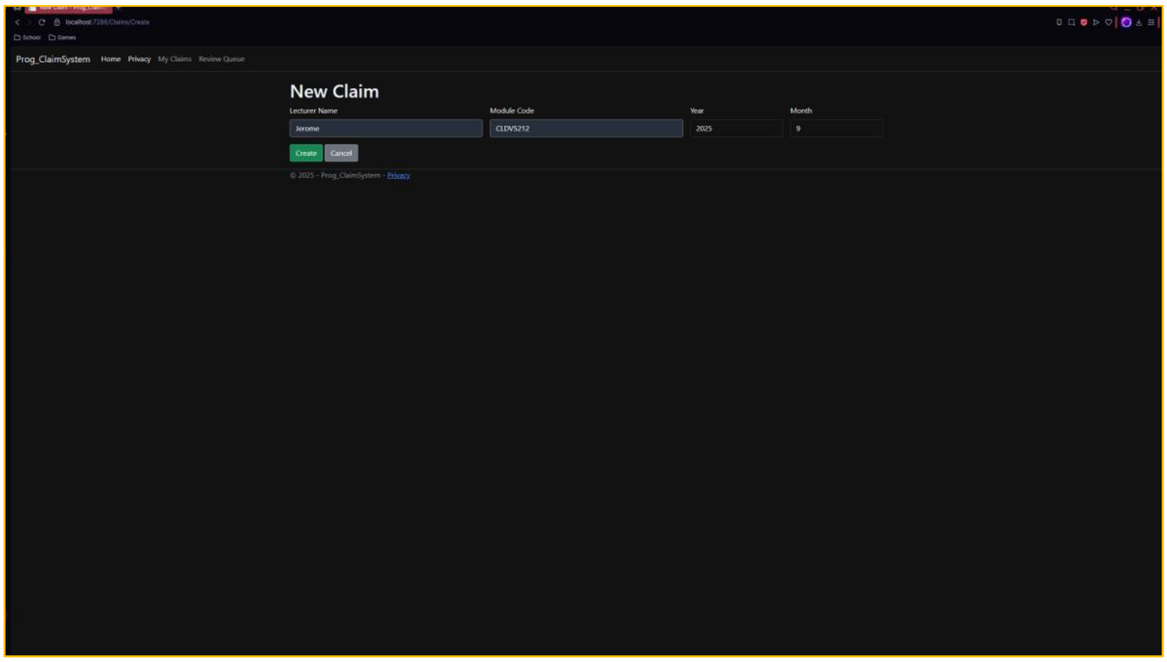


My Monthly Claims

Month	Module	Total	Status	
2025-10	prog6212	R 0,00	UnderReview	Open
2025-10	CLDV5212	R 0,00	Draft	Open
2025-09	prog6212	R 200,00	Draft	Open

© 2025 - Prog_ClaimSystem - Privacy

- What the user will see once they decide to create a claim, prompting for the name of the lecturer, module code and date of the class they will be teaching.


localhost:7288/Claims/Create

Prog_ClaimSystem Home Privacy My Claims Review Queue

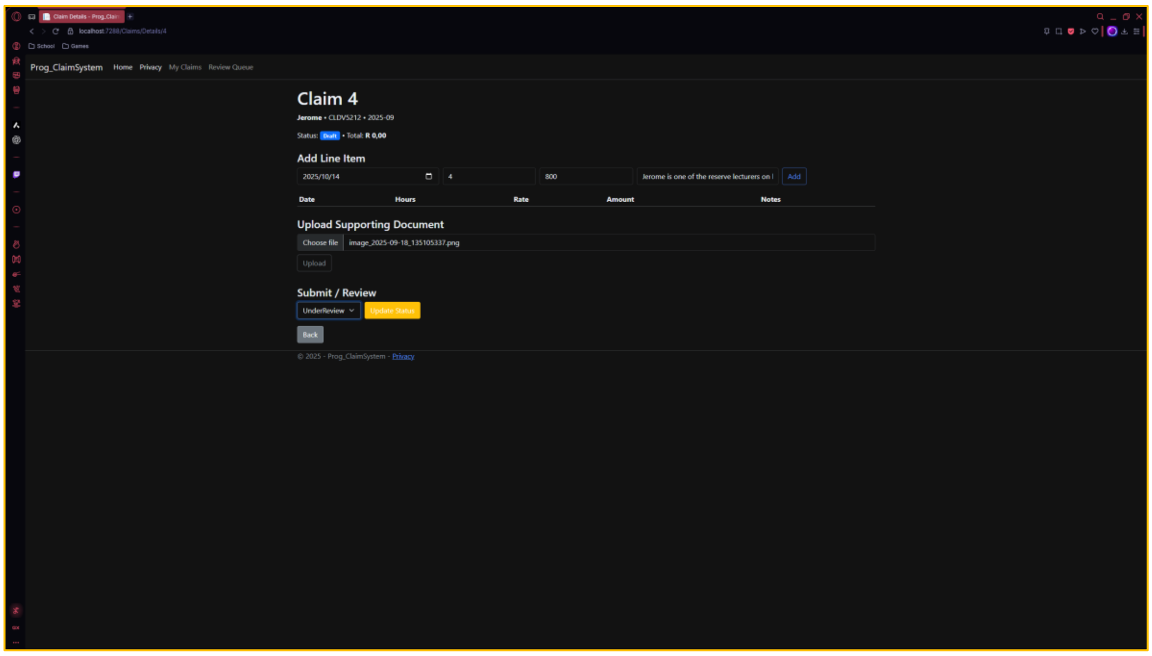
New Claim

Lecturer Name	Module Code	Year	Month
Jerome	CLDV5212	2025	9

Create Cancel

© 2025 - Prog_ClaimSystem - Privacy

- The user will then be prompted to fill in the details of the lecture as well as any notes and any supporting documents. Each new claim begins as a draft and can be modified later.


localhost:7288/Claims/Details/4

Prog_ClaimSystem Home Privacy My Claims Review Queue

Claim 4

Jerome • CLDV5212 • 2025-09

Status: Draft • Total: **R 0,00**

Add Line Item

2025/10/14 | 4 | 800 | Jerome is one of the reserve lecturers on t | Add

Date	Hours	Rate	Amount	Notes

Upload Supporting Document

Choose file image_2025-09-18_135105337.png

Upload

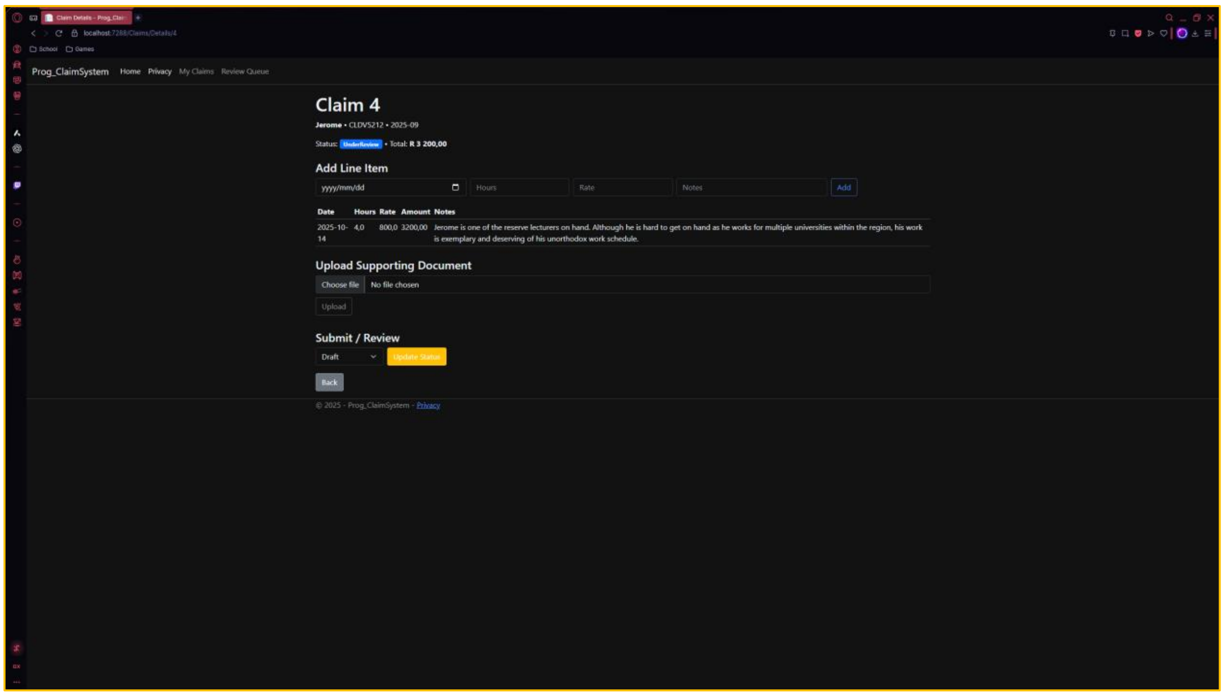
Submit / Review

UnderReview | Update Status

Back

© 2025 - Prog_ClaimSystem - Privacy

- The user may add as many different times and lectures as they please to a specific lecturer they are all saved under their name.



Claim 4

Jerome • CLDV5212 • 2025-09

Status: UnderReview • Total: **R 3 200,00**

Add Line Item

yyyy/mm/dd | Hours | Rate | Notes | Add

Date	Hours	Rate	Amount	Notes
2025-10-14	4,0	800,0	3200,00	Jerome is one of the reserve lecturers on hand. Although he is hard to get on hand as he works for multiple universities within the region, his work is exemplary and deserving of his unorthodox work schedule.

Upload Supporting Document

Choose file No file chosen

Upload

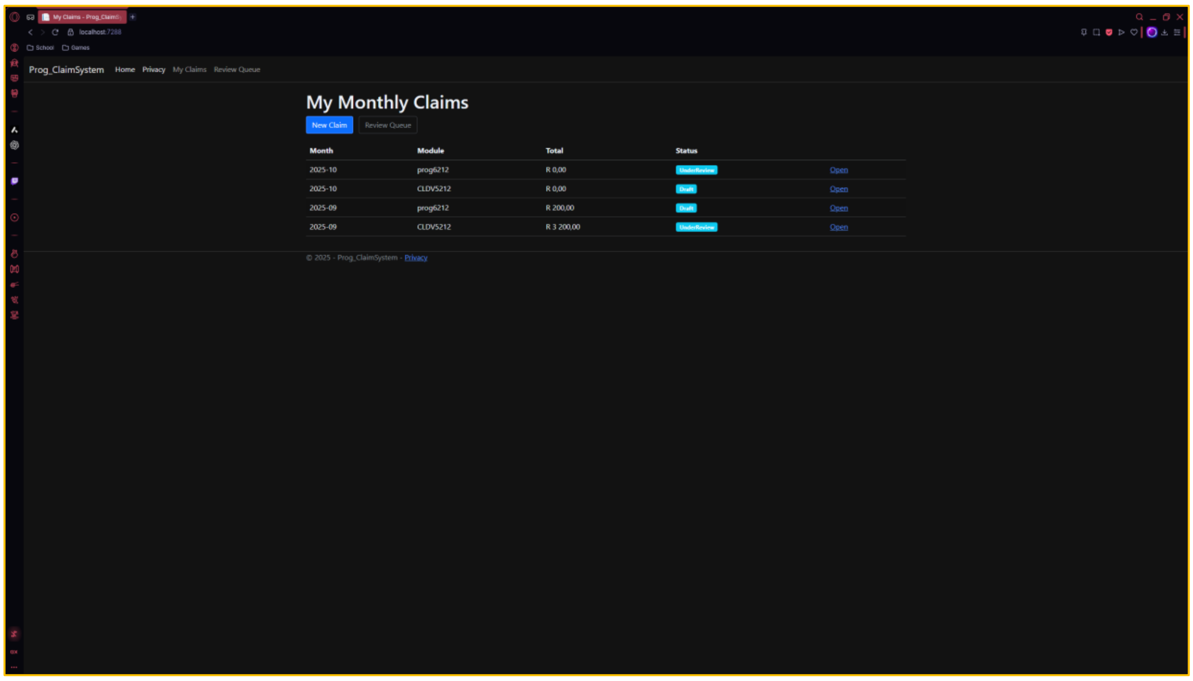
Submit / Review

Draft | Update Status

Back

© 2025 - Prog_ClaimSystem - Privacy

- Once the user decides to go back, they will see the list updated with the new entry as well as the status of said entry. Each entry may be opened at any time to add more or change something.



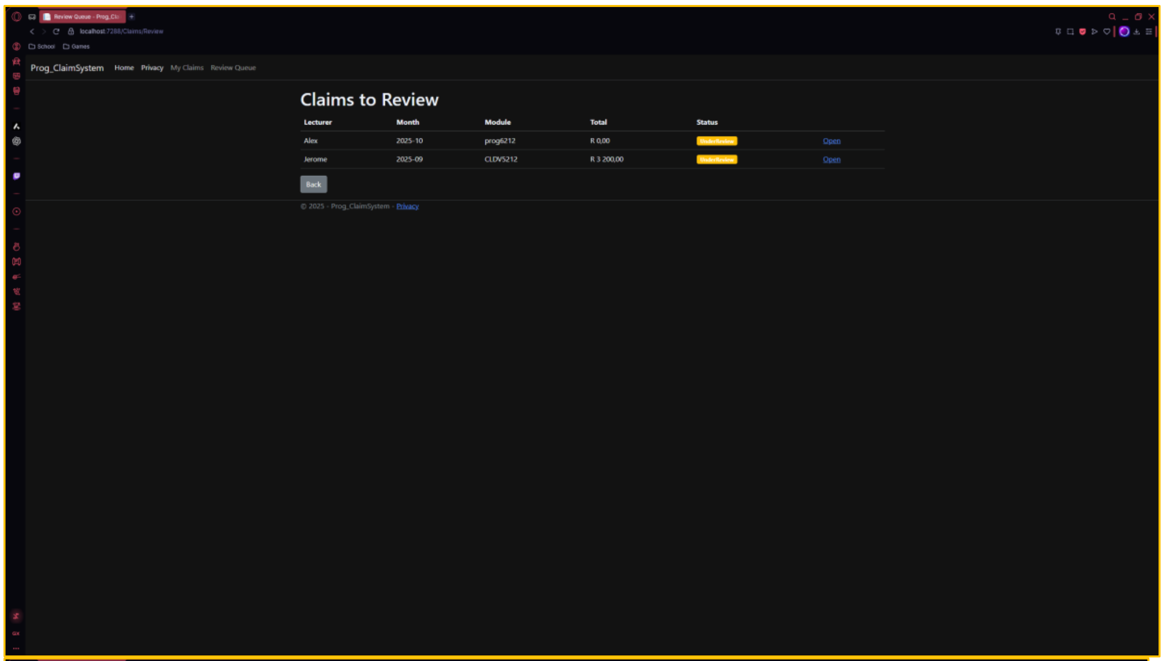
My Monthly Claims

New Claim | Review Queue

Month	Module	Total	Status	
2025-10	prog6212	R 0,00	UnderReview	Open
2025-10	CLDV5212	R 0,00	Draft	Open
2025-09	prog6212	R 200,00	Draft	Open
2025-09	CLDV5212	R 3 200,00	UnderReview	Open

© 2025 - Prog_ClaimSystem - Privacy

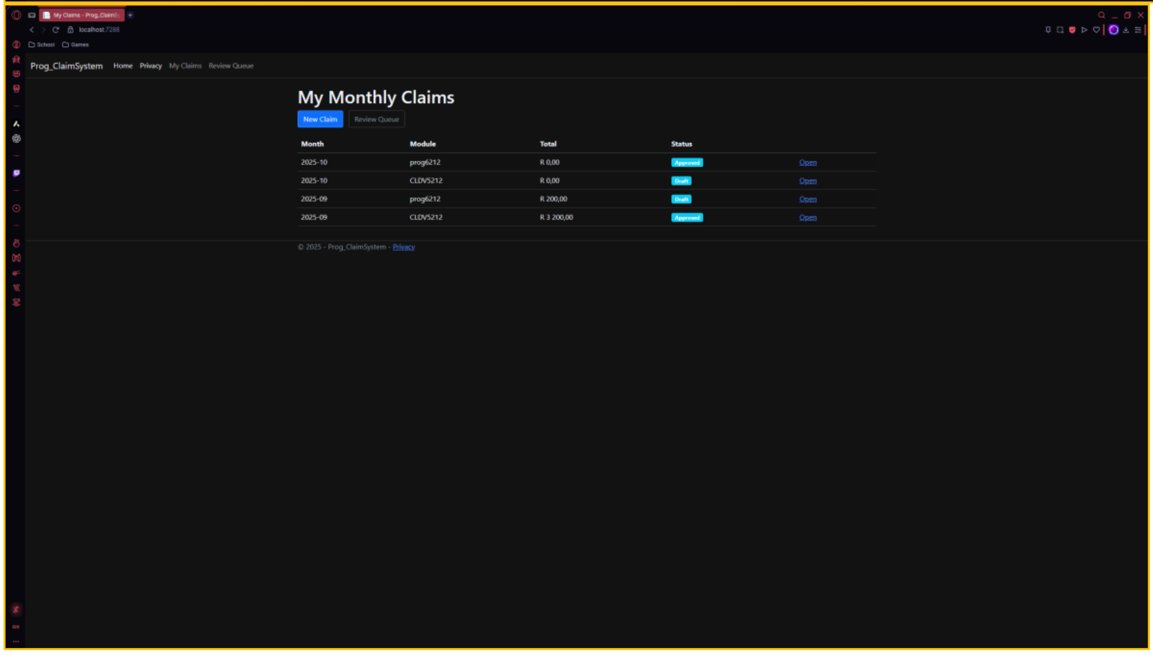
- A reviewer would be able to click on the review tab and go see anything that they may need to approve, they can check on any information, documents added to the entry and deny, approve or leave be is up to their discretion.



Claims to Review

Lecturer	Month	Module	Total	Status	
Alex	2025-10	prog6212	R 0,00	UnderReview	Open
Jerome	2025-09	CLDV5212	R 3 200,00	UnderReview	Open

© 2025 - Prog_ClaimSystem - Privacy



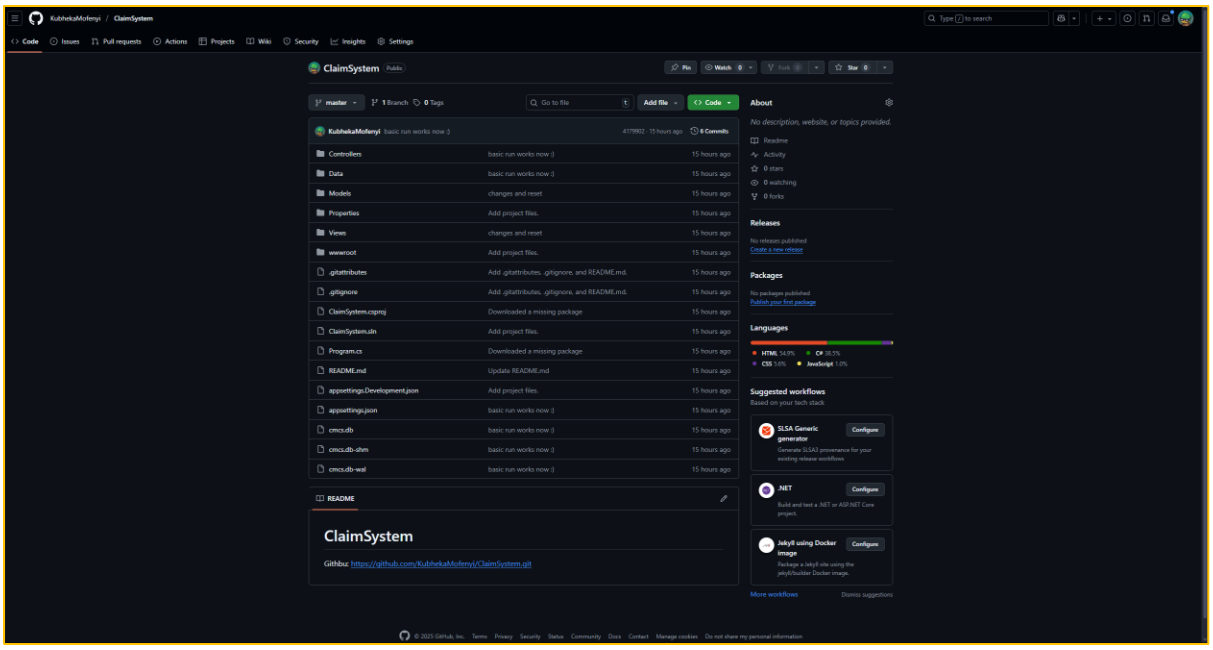
My Monthly Claims

Month	Module	Total	Status	
2025-10	prog6212	R 0,00	Approved	Open
2025-10	CLDV5212	R 0,00	Draft	Open
2025-09	prog6212	R 200,00	Draft	Open
2025-09	CLDV5212	R 3 200,00	Approved	Open

© 2025 - Prog_ClaimSystem - Privacy

Github

Github commits as of time of submission:



KubhekaMofenyi / ClaimSystem

ClaimSystem Public

master · 1 Branch · 0 Tags

KubhekaMofenyi basic run works now :) 4179902 · 15 hours ago · 6 Commits

Controllers	basic run works now :)	15 hours ago
Data	basic run works now :)	15 hours ago
Models	changes and reset	15 hours ago
Properties	Add project files.	15 hours ago
Views	changes and reset	15 hours ago
wwwroot	Add project files.	15 hours ago
.gitattributes	Add .gitattributes, .gitignore, and README.md.	15 hours ago
.gitignore	Add .gitattributes, .gitignore, and README.md.	15 hours ago
ClaimSystem.csproj	Downloaded a missing package	15 hours ago
ClaimSystem.sln	Add project files.	15 hours ago
Program.cs	Downloaded a missing package	15 hours ago
README.md	Update README.md	15 hours ago
appsettings.Development.json	Add project files.	15 hours ago
appsettings.json	basic run works now :)	15 hours ago
cmcs.db	basic run works now :)	15 hours ago
cmcs.db-shm	basic run works now :)	15 hours ago
cmcs.db-wal	basic run works now :)	15 hours ago

README

ClaimSystem

Githbu: https://github.com/KubhekaMofenyi/ClaimSystem.git

About

No description, website, or topics provided.

Readme
Activity
0 stars
0 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

● HTML 54.9% ● C# 38.5% ● CSS 5.6% ● JavaScript 1.0%

Suggested workflows
Based on your tech stack

SLSA Generic generator — Generate SLSA3 provenance for your existing release workflows

.NET — Build and test a .NET or ASP.NET Core project.

Jekyll using Docker image — Package a Jekyll site using the jekyll/builder Docker image.

© 2025 GitHub, Inc. Terms Privacy Security Status Community Docs Contact Manage cookies Do not share my personal information

Link: https://github.com/KubhekaMofenyi/ClaimSystem.git