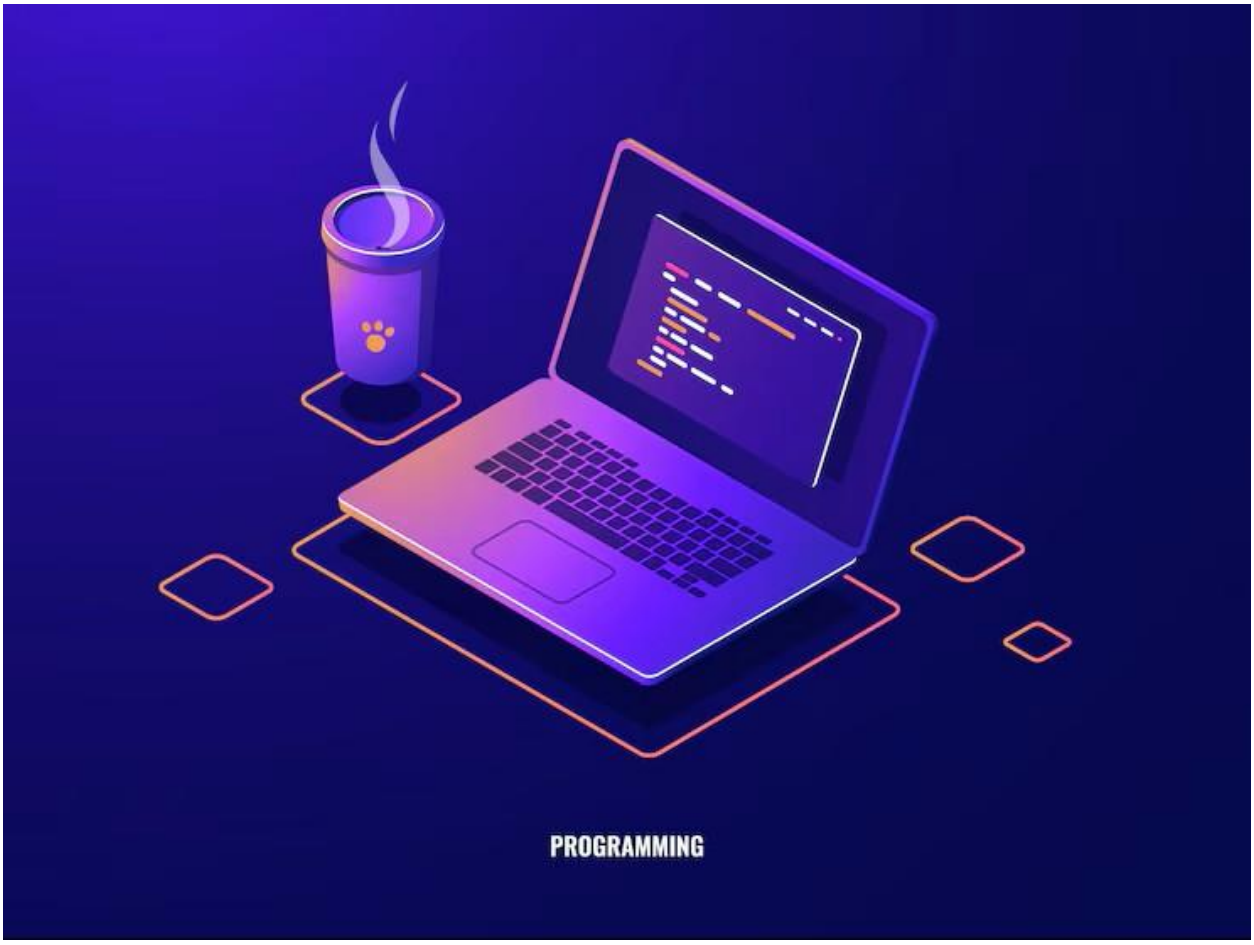
PROG6212 POE

ST10439398

Morgan Gibbon



Content

Part 1

Contract Monthly Claim System (CMCS) Prototype Report

Database Design

The database (Figure 1) has been created and developed to allow for control and responsibility of users with roles and claims. The User table includes role-based permissions (Lecturer, Coordinator, Manager, HR) to allow for proper system access while keeping data consistent through a single user.

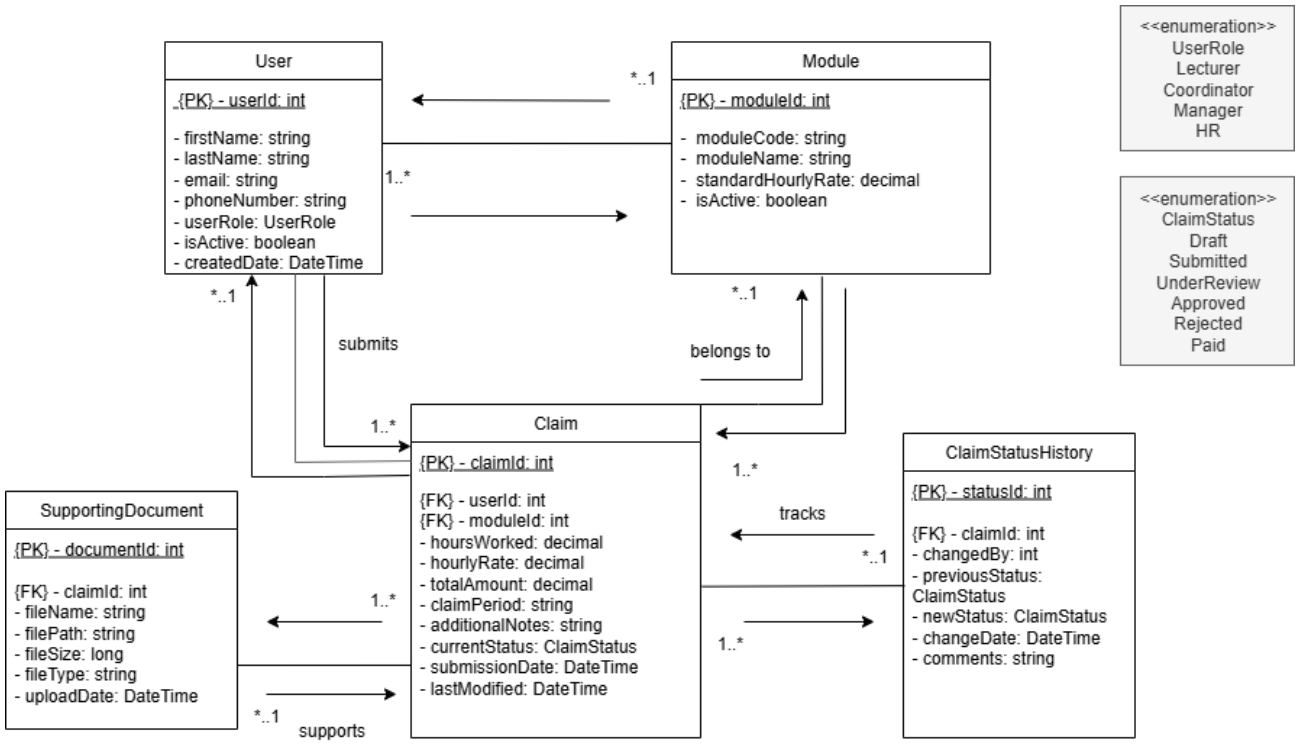
Lecturers are linked to specific modules they teach, with each module containing standard hourly rates for automated validation. This improves the user experience by limiting module options on claim forms to only those lecturers who are allowed to teach, while giving coordinators context during approvals.

The central Claim table gets all necessary information including hours worked, hourly rates, and calculated totals, with foreign key relationships to both User and Module tables

ClaimStatusHistory provides tracking by recording every status change with timestamps and user id. This creates clarity in the approval workflow while supporting accountability requirements through historical records of claim progression from submission to payment.

SupportingDocument Management allows lecturers to upload claim documentation while having safe file handling through database storage and file path references. This separation supports many storage solutions while ensuring proper document-claim associations.

FIGURE 1



Graphical User Interface (GUI)

Theme

For the CMCS interface, I selected a blue gradient scheme (#1e3a8a to #3b82f6) . The primary blue conveys an institutional feel, while the lighter blue creates visual hierarchy without overwhelming users. The accent green (#10b981) signals positive actions like approvals, and amber (#f59e0b) draws attention to pending items requiring action.

The font 'Segoe UI' allows for proper readability across devices while keeping the clean, professional appearance expected in academic institutional software's. This choice allows for functionality, supporting extended reading sessions during claim review processes.

The appearance created through drop shadows displays click ability while keeping the visual hierarchy clear. Color-coded status badges (yellow for pending, green for approved, red for rejected) enable instant recognition of claim states.

Login

All users access the system through a single login interface ensuring simplicity and role clarity. The centered design focuses attention on essentials while the role selector buttons allow instant identification of access levels. This vertical flow guides users logically through authentication.

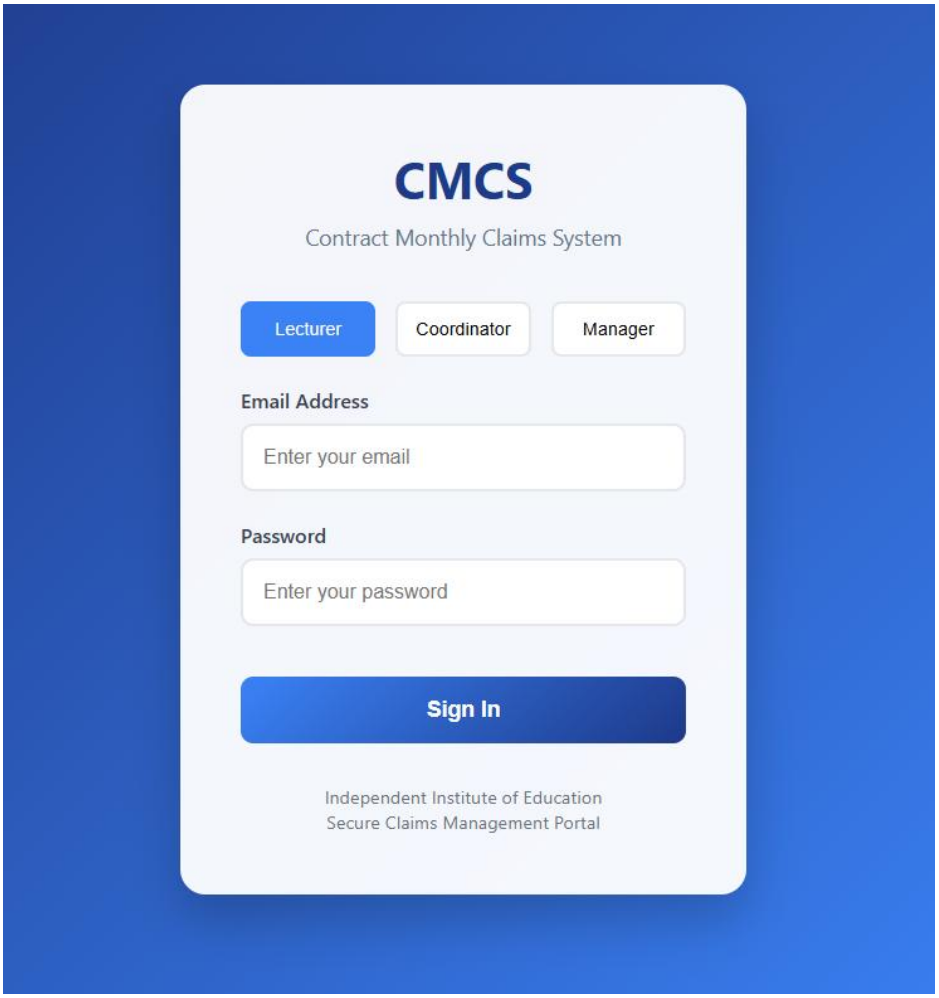


FIGURE 2 LOGIN PAGE

Lecturer Dashboard

The lecturer's dashboard displays and allows for task completion with action cards featuring clear icons and descriptions. The dashboard statistics (pending claims, monthly totals) provide context for what needs to be done. The card-based layout allows for easy scanning.

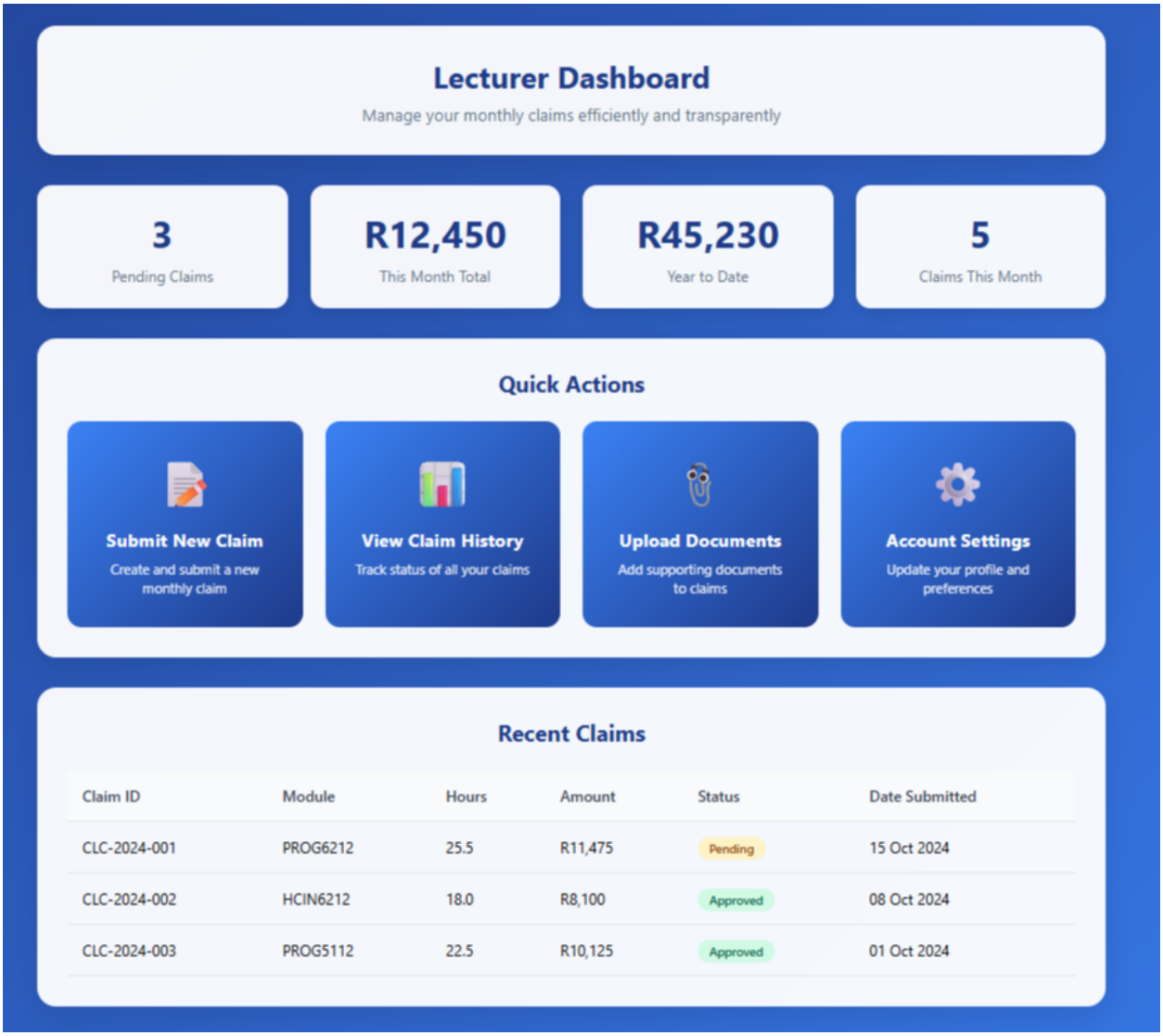


FIGURE 3 LECTURER DASHBOARD

Programme Coordinator Dashboard

The coordinator interface allows for efficiency through information along with readability. Statistics provide immediate data, while the filterable claims table displays essential information (lecturer, module, amount, urgency) in a clear format. Color-coded priority indicators (red borders for urgent items) enable the user to see what needs are urgent, while the action buttons support efficient processing of multiple claims. They can access the claim history, upload documents and navigate to the settings page when needed.

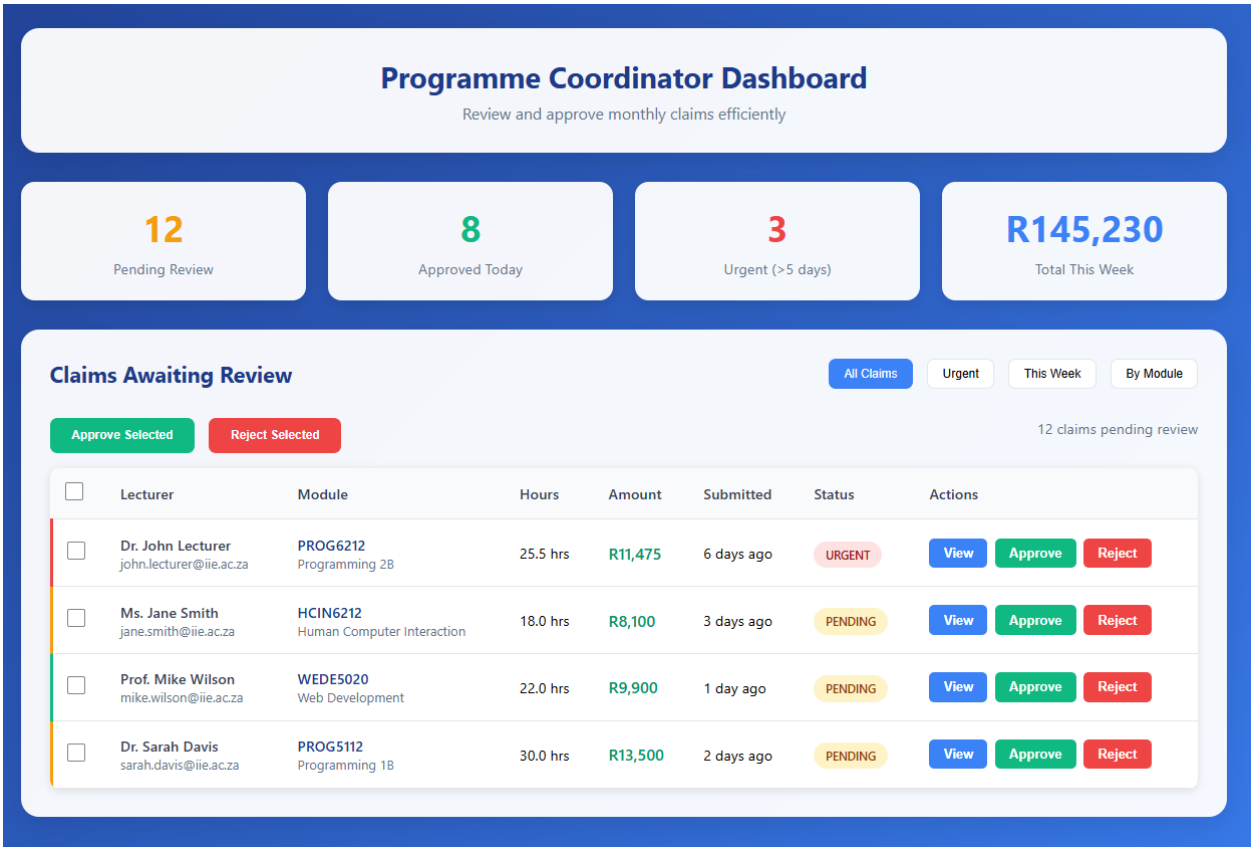
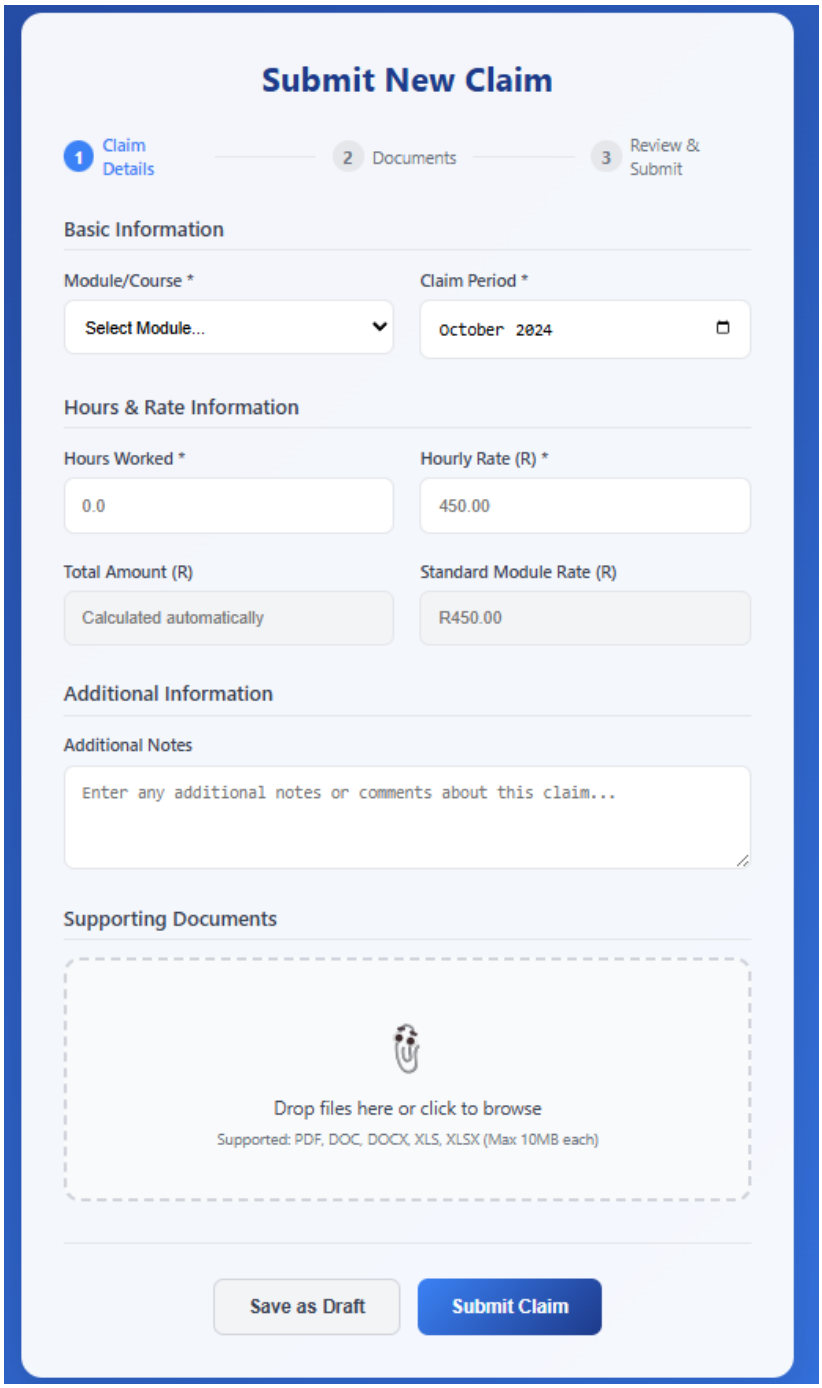


FIGURE 4 COORDINATOR DASHBOARD

Claim Submission Form

The progress indicator at the top establishes clear progress. The auto-calculation feature for total amounts provides immediate feedback, while the drag-and-drop document upload area uses visual cues to communicate functionality clearly.

FIGURE 5 CLAIM FORM



Status Tracking Interface

The timeline shows visual progress. The numbered circles create a clear sense of the claim journey for the user. Supporting information is organized easily to understand what is related to the claims being handled.

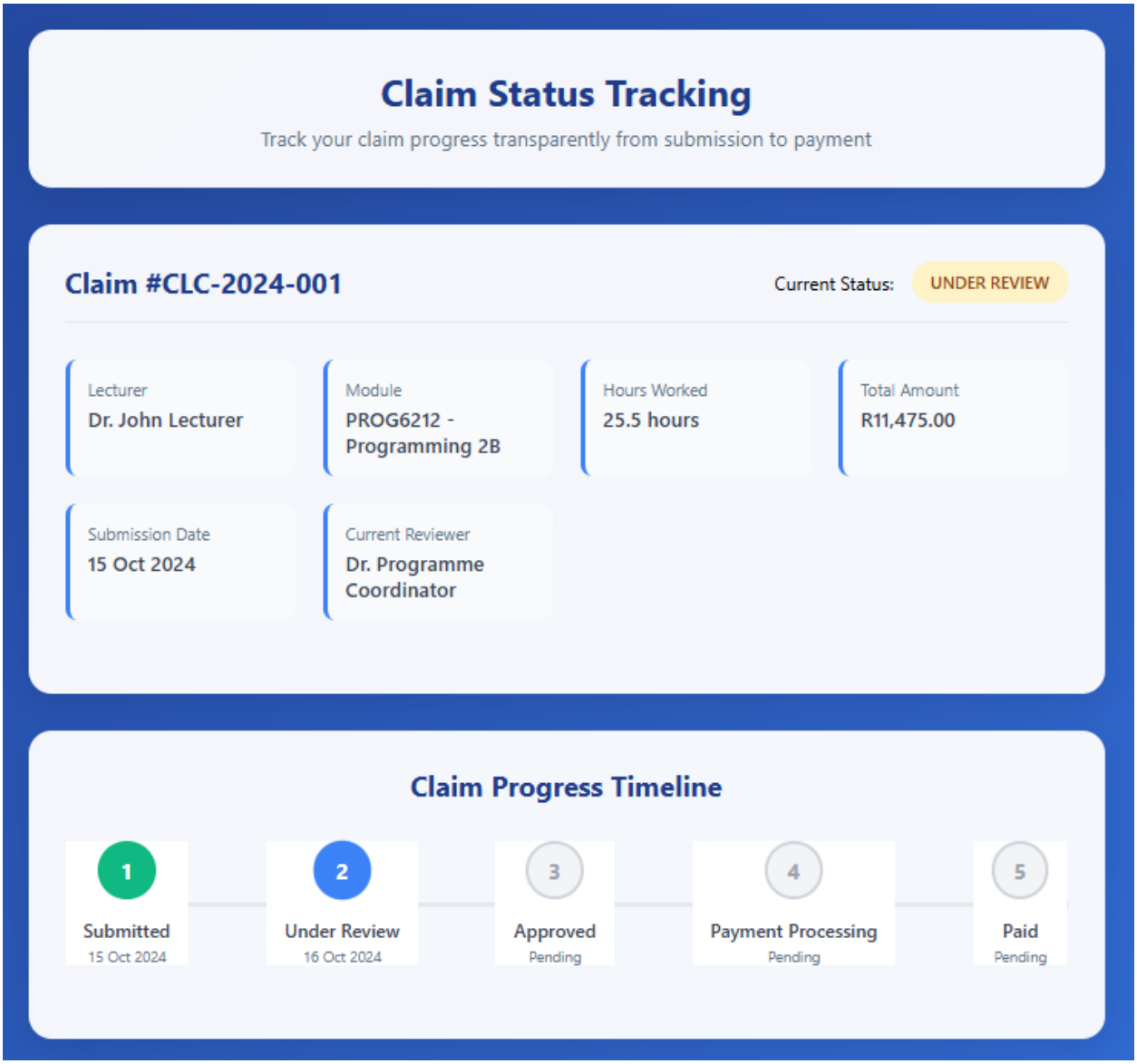


FIGURE 6 CLAIM TRACKING

Account Settings page

Accounts Settings page shows the user account view. Lecturers can change their personal information and banking details here.

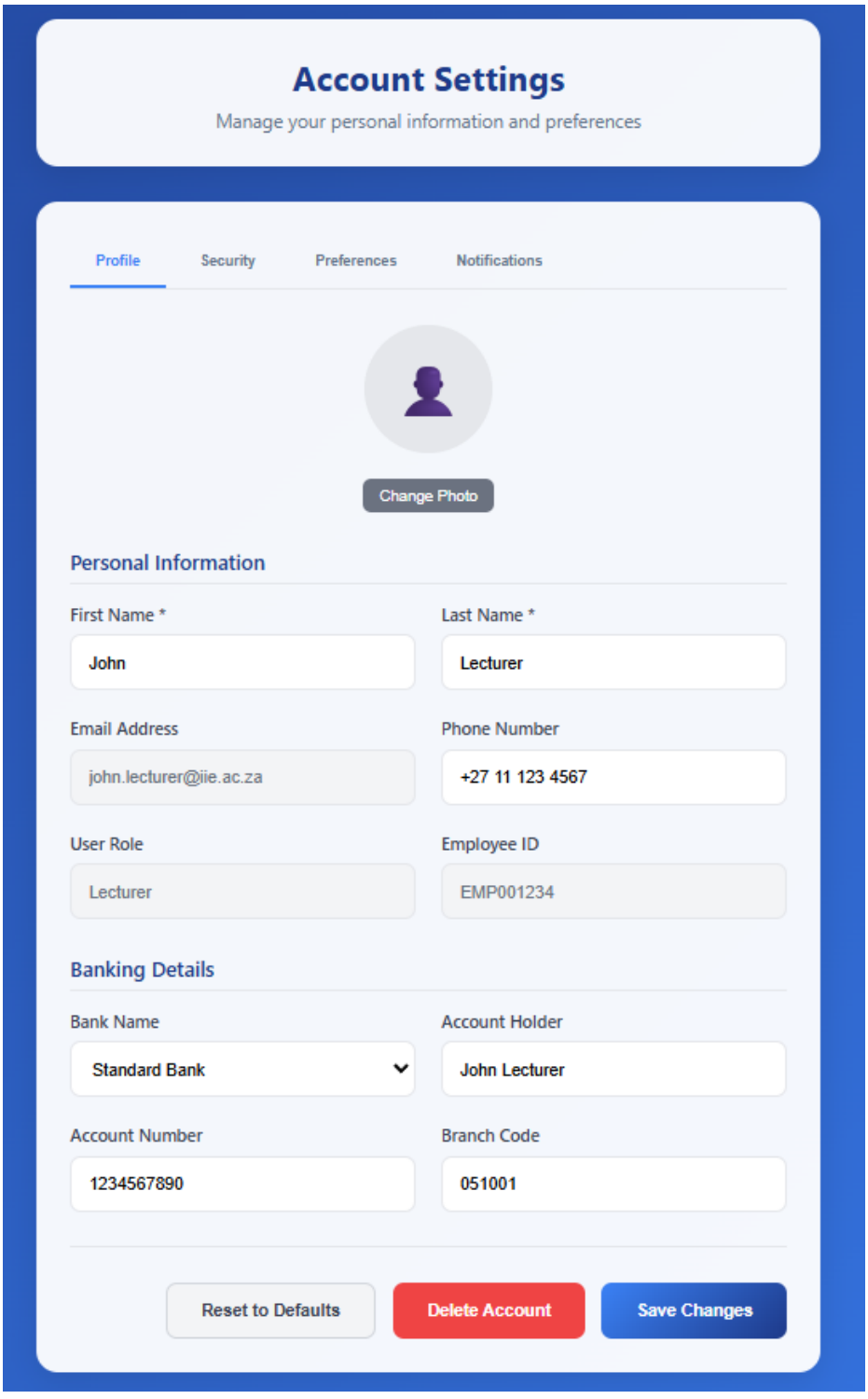


FIGURE 7 SETTINGS

Claim History Page

This page shows lecturers a complete list of all the claims they've ever submitted to the system.

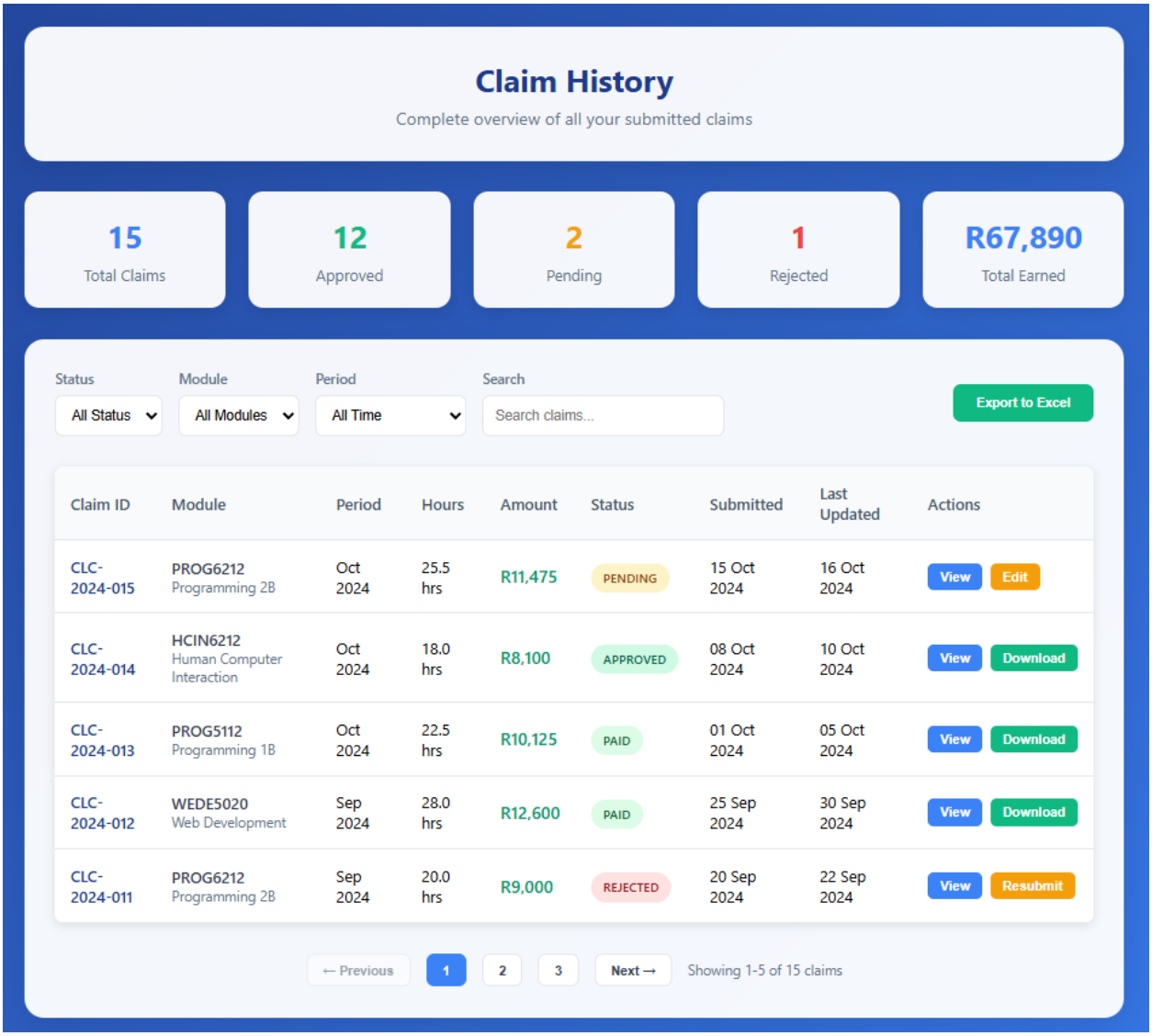
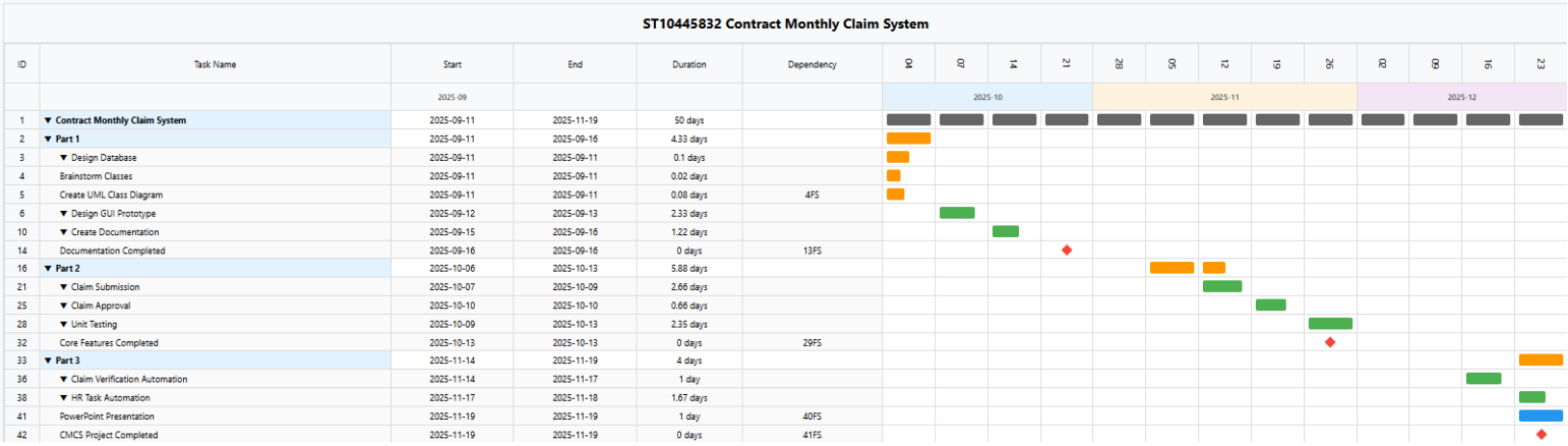


FIGURE 8 CLAIM HISTORY

Project Plan



ID	Task Name	Start	End	Duration	Dependency
		2025-09			
1	▼ Contract Monthly Claim System	2025-09-11	2025-11-19	50 days	
2	▼ Part 1	2025-09-11	2025-09-16	4.33 days	
3	▼ Design Database	2025-09-11	2025-09-11	0.1 days	
4	Brainstorm Classes	2025-09-11	2025-09-11	0.02 days	
5	Create UML Class Diagram	2025-09-11	2025-09-11	0.08 days	4FS
6	▼ Design GUI Prototype	2025-09-12	2025-09-13	2.33 days	
10	▼ Create Documentation	2025-09-15	2025-09-16	1.22 days	
14	Documentation Completed	2025-09-16	2025-09-16	0 days	13FS
16	▼ Part 2	2025-10-06	2025-10-13	5.88 days	
21	▼ Claim Submission	2025-10-07	2025-10-09	2.66 days	
25	▼ Claim Approval	2025-10-10	2025-10-10	0.66 days	
28	▼ Unit Testing	2025-10-09	2025-10-13	2.35 days	
32	Core Features Completed	2025-10-13	2025-10-13	0 days	29FS
33	▼ Part 3	2025-11-14	2025-11-19	4 days	
36	▼ Claim Verification Automation	2025-11-14	2025-11-17	1 day	
38	▼ HR Task Automation	2025-11-17	2025-11-18	1.67 days	
41	PowerPoint Presentation	2025-11-19	2025-11-19	1 day	40FS
42	CMCS Project Completed	2025-11-19	2025-11-19	0 days	41FS

FIGURE 9 PROJECT PLAN

This project plan has been created with realistic time restraints in mind to remain achievable

Assumptions and Constraint

System Assumptions

- The CMCS will be used by Independent Contractor lecturers, Programme Coordinators, and Academic Managers at educational institutions and should be designed with that user base in mind.
- Users' technical ability will vary from basic to advanced, requiring intuitive interface design.
- The CMCS will be a .NET Core MVC web application accessed primarily on desktop and laptop computers.
- Data storage will utilize both relational database systems and secure file storage for supporting documents.
- Claims require approval from either a Programme Coordinator OR Academic Manager, not both.
- Coordinators and Managers are only authorized to approve claims for modules within their departmental authority.
- Standard hourly rates are pre-established for each module and maintained by system administrators.
- Lecturer accounts will be created by administrators, but personal and banking information will be modifiable by lecturers themselves.

- The system will generate claim summaries for payment processing but will not handle actual financial transactions.
- All data processing must comply with POPIA (Protection of Personal Information Act) requirements.

Project Constraints

- Single developer working on the project with limited resources for extensive testing phases.
- Stakeholder engagement is restricted to post-submission feedback sessions with lecturers.
- Internet connectivity required for all system functionality as web-based application.
- File upload restrictions limited to PDF, DOCX, and XLSX formats with maximum 10MB file size.
- Claims must be submitted within 30 days of the claim period end date.
- System requires modern web browsers with JavaScript enabled for full functionality.
- Database backup and recovery procedures must be established before production deployment.

Development Timeline

- Prototype development phase: 4 days (Part 1 completion)
- Functional implementation phase: 6 days (Part 2 completion)
- Enhancement and automation phase: 4 days (Part 3 completion)
- Total project duration: 14 days as specified in POE requirements

(Claude.ai, 2025)

References

Claude.ai. (2025, September 17). Retrieved from https://claude.ai/

Disclosure of AI use

Part 1 Assumptions & Constraints

Purpose of use: to expand and understand the question to provide a more detailed answer than just by myself.

The tool I used was Claude.ai and accessed it on September 17th, 2025, as I needed help with understanding what should be included in the assumptions and constraints.

Link : https://claude.ai/chat/38d0a817-ebe5-4248-ae4f-0797fb42ce65