

# Lecturer Claim Management System

## **PROG6212 – Portfolio of Evidence (Part 3)**

**Student Name:** Alonzo Buwa

**Student Number:** ST10447866

**Lecturer:** Matthew Prinsloo

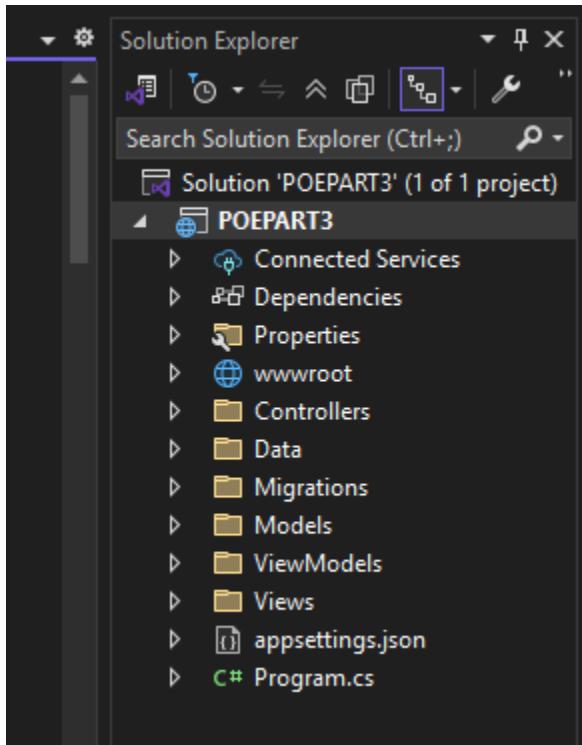
**Date:** 205/11/21

What This System Does .....	2
System Users.....	3
The System Has 4 User Roles:.....	3
Lecturer .....	3
Programmer Coordinator .....	3
Academic Manager .....	3
HR (Super User) .....	3
Changes from Part 2 to Part 3 .....	3
System Architecture .....	4
Database Schema.....	5
Database Tables.....	5
Relationships: .....	5
How to Run the Project .....	5
Lecturer: Submit a Claim .....	6
Claim Flow Diagram .....	7
Coordinator & Manager Dashboard .....	7
HR Dashboard .....	8
Code Snippets to Include.....	9
Project Hosting & Version Control.....	10
Testing the System .....	11
Challenges & Solutions.....	11
Conclusion .....	12

## **What This System Does**

- Allows lecturers to submit monthly claims
- Allows Programme Coordinator & Academic Manager to approve/reject

- Allows HR to manage lecturers and generate reports
- Uses SQLite for local data storage
- Includes authentication for security
- Follows the POE Part 3 requirements



“Project structure implemented according to MVC architecture.”

## System Users

**The System Has 4 User Roles:**

### *Lecturer*

- Submits claims
- Views their submitted claims

### ***Programmer Coordinator***

- First-level approval or rejection

### ***Academic Manager***

- Final approve or reject of claims

### ***HR (Super User)***

- Add / edit lecturers
- View HR report
- Total hours, total costs
- All data access

## **Changes from Part 2 to Part 3**

### **Major Enhancements**

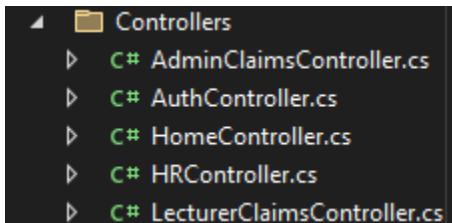
- Added login system with Sessions
- Added role-based page protection
- Auto-fill lecturer information (AJAX)
- Enforced hourly limits (max 180 hours)
- Improved document upload validation
- Added Admin Claims panel
- Added HR management panel
- Auto-calc total amount
- Created lecturer seeder
- Full database integration (SQLite)

- UI polishing + validation improvements

## System Architecture

### Built Using:

- ASP.NET Core MVC
- Entity Framework Core
- SQLite Database
- Bootstrap 5
- Session Authentication
- Strong MVC architecture
- Repository pattern (optional)



“MVC structure ensures clean separation of logic.”

## Database Schema

### Database Tables

- Users
- Lecturers
- Claims
- SupportingDocuments

## **Relationships:**

- Lecturer → Claims (One-to-Many)
- Claim → SupportingDocuments (One-to-Many)

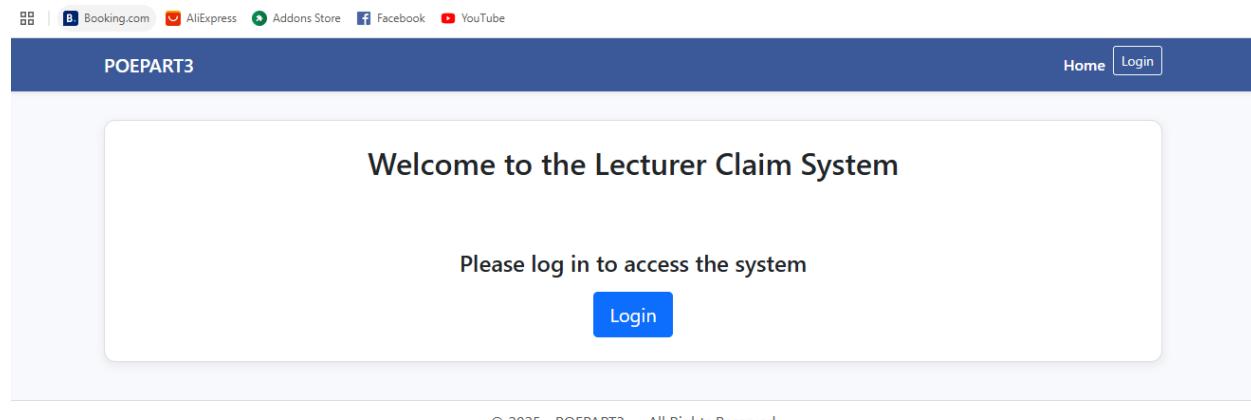
### SQL server code snippet

```
SELECT name FROM sqlite_master WHERE type='table';
```

“SQLite database containing all required tables.”

## **How to Run the Project**

Extract the zip file sent via Github link and open POEPART3.csproj. It should take you to visual studio. Once you’re in visual studio run the project and then it should take you to the login page.



After you then tap Login it will take you to another page where it asks for your username and password. Each user Lecturer, Coordinator, Manager, HR has their own specific page, username and password.

POEPART3

Home Login

## Login

Username

Password

Login

© 2025 - POEPART3 — All Rights Reserved

Lecturer Username: lecturer1

Lecturer Password: 123

Coordinator Username: moordinator

Coordinator Password: 123

Manager Username: manager

Manager Password: 123

HR Username: hr

HR Password: 123

## Lecturer: Submit a Claim

### Features:

- Auto-fill lecturer details
- Hours validation
- Document upload
- Total auto-calculated
- Enforces 180 hours per month rule

## Submit a New Claim

Please enter your **Lecturer ID**. If the lecturer exists, their details will auto-fill.

**Lecturer ID**

0

### Claim Details

**Hours Worked**

0,00

Notes (Optional)

Add notes about your claim

### Supporting Document

Accepted: PDF, DOCX, XLSX | Max size: 5MB

Select a lecturer ID by choosing and ID from 1-12

## Claim Flow Diagram

Lecturer → Submit Claim → Coordinator Review → Manager Review → HR Report

## Coordinator & Manager Dashboard

### Features:

- View pending claims
- Approve or reject
- View claim details
- View uploaded documents

The screenshot shows the POEPART3 Admin Panel with a dark blue header bar. On the left is the logo 'POEPART3'. On the right, it says 'Logged in as: manager (Manager)' and has a 'Logout' button. Below the header is a light gray content area with a title 'Pending Claims'. Underneath is a table with the following data:

Lecturer	Hours	Rate	Total	Status	Date	Document	Actions
John Mokoena	20,0	R 150,0	R 3000,0	Pending	2025/11/21	<a href="#">View</a>	<a href="#">Approve</a> <a href="#">Reject</a>

© 2025 - POEPART3 — All Rights Reserved

“Admin panel showing pending lecturer claims.”

## HR Dashboard

### What HR Can Do:

- Add new lecturer
- Update lecturer details
- View system-wide reports
- Filter by lecturer
- Filter by month

## Lecturer Management

[Add New Lecturer](#)

ID	First Name	Last Name	Email	Hourly Rate	Actions	
1	John	Mokoena	john.mokoena@example.com	R 150,0	<a href="#">Edit</a>	<a href="#">Delete</a>
2	Sarah	Daniels	sarah.daniels@example.com	R 180,0	<a href="#">Edit</a>	<a href="#">Delete</a>
3	Michael	Smith	michael.smith@example.com	R 200,0	<a href="#">Edit</a>	<a href="#">Delete</a>
4	Thabo	Nkosi	thabo.nkosi@example.com	R 160,0	<a href="#">Edit</a>	<a href="#">Delete</a>
5	Emily	Clark	emily.clark@example.com	R 170,0	<a href="#">Edit</a>	<a href="#">Delete</a>
6	Sipho	Zulu	sipho.zulu@example.com	R 155,0	<a href="#">Edit</a>	<a href="#">Delete</a>
7	Megan	Naidoo	megan.naidoo@example.com	R 165,0	<a href="#">Edit</a>	<a href="#">Delete</a>

## HR Claim Report

[Download PDF](#)

Filter by Month

December

Filter by Lecturer

Lerato Mabaso (12)

[Apply](#)

Total Payroll

R 3000,00

Approved Claims

0

Pending Claims

1

Rejected Claims

0

Lecturer	ID	Hours	Rate	Total	Status	Date	Document
John Mokoena	1	20,0	R 150,0	R 3000,0	Pending	2025/11/21	

# Code Snippets to Include

## Auto-fill Lecturer (AJAX)

Controller:

```
7  public JsonResult GetLecturerDetails(int id)
8  {
9      var lecturer = _context.Lecturers.FirstOrDefault(l => l.LecturerId == id);
10
11     if (lecturer == null)
12         return Json(new { exists = false });
13
14     return Json(new
15     {
16         exists = true,
17         firstName = lecturer.FirstName,
18         lastName = lecturer.LastName,
19         hourlyRate = lecturer.HourlyRate
20     });
21
22
23 // =====
24 // POST: Submit New Claim (Lecturer Only)
25 // =====
26 [HttpPost]
27 0 references
28 public IActionResult Create(ClaimCreateViewModel model)
29 {
30     if (HttpContext.Session.GetString("Role") != "Lecturer")
31         return RedirectToAction("Login", "Auth");
32
33     if (!ModelState.IsValid)
34         return View(model);
35
36 // =====
37 // VALIDATION: 180 hour monthly limit
38 // =====
39     var currentMonth = DateTime.Now.Month;
```

# Project Hosting & Version Control

## GitHub Used For:

- 10+ meaningful commits
- Clean repo
- Removed bin/obj
- Added README.md
- Included description of Part 2 → Part 3 changes

# Testing the System

## Test Cases Performed:

- Valid lecturer entering valid claim
- Auto-fill successful
- File upload validation tested
- Coordinator & Manager approval tested
- HR report filtering tested
- Wrong login credentials tested
- Unauthorized access blocked

# Challenges & Solutions

## Challenges:

- Fixing auto-fill issues
- Database resetting repeatedly

- Sessions not loading
- Coordinated seeding of users & lecturers
- UI issues due to outdated Layout
- SQLite decimal SUM bug

**Solutions:**

- Forced client-side SUM
- Rebuilt layout from scratch
- Added seeder block
- Cleaned project files
- Added proper session checks

## Conclusion

**The System Successfully Delivers:**

- Secure login & role-based access
- End-to-end claim workflow
- Automated validations & calculations
- HR management capabilities
- Clean and professional MVC architecture
- Fully functioning POE Part 3 requirements