ST10180342/ PROG6212/ POE\_PART\_1

[Company name] | [Company address]

CONTRACT MONTHLY CLAIMS SYSTEM

Ntembeko Ndayi

2025

**Documentation**

**Design Choices, Database Structure, GUI Layout, and Project Plan for Contract Monthly Claim System**

**Design Choices**

The Contract Monthly Claim System uses ASP.NET Core MVC for its separation of Models (data), Views (UI), and Controllers (logic), ideal for a beginner friendly, small scale system. Code is organized into namespaces (ContractMonthlyClaimSystem, ContractMonthlyClaimSystem.Controllers), with HomeController handling pages like Error, and ClaimController managing actions (Submit, Verify, Approve, Track). Error handling uses ErrorViewModel to display debug infomation in development, protecting sensitive data in production. The Index view’s role dropdown (Lecturer, Programme Coordinator, Academic Manager) supports role based access using ASP.NET, restricting actions like verification to authorized users(once project is completed). The Claim model (Id, LecturerName, SubmissionDate, Status, SupportingDocuments) supports file uploads and status tracking, making MVC extensible for these features.

**Database Structure**

The prototype lacks a database, but the Claim model implies a relational structure:

* **Claims Table**: Id (int, PK), UserId (int, FK), SubmissionDate (DateTime), Status (string), HoursWorked (double), HourlyRate (decimal), TotalAmount (decimal), Notes (string).
* **Users Table**: Id (int, PK), Username (string, unique), PasswordHash (string), RoleId (int, FK), FullName (string).
* **Roles Table**: Id (int, PK), Name (string, unique).
* **SupportingDocuments Table**: Id (int, PK), ClaimId (int, FK), FilePath (string), UploadDate (DateTime).
* **Relationships**: One-to-many (User to Claims, Claim to SupportingDocuments), User to Role (one-to-one). File paths prevent database bloat.

**GUI Layout**

Bootstrap 4.5.2 in \_Layout.cshtml ensures a responsive UI with a blue navbar (Home, Submit, Verify, Approve, Track), content container, and 2025 footer. The Index view offers a centered login form for username, password, and role. Submit includes fields for hours, rate, and file uploads; Verify/Approve lists claims with action buttons; Track displays a status dashboard, all using Bootstrap’s grid for usability.

**Assumptions and Constraints**

Assumes a small system, basic features (submission, verification, tracking), and simple security (hashed passwords). Constraints include limited ASP.NET skills, POPIA compliance, and small file uploads stored in wwwroot/uploads.

**Project Plan**

**Phase 1: Initiation (15-19 Sep 2025, 1 week, 40 hours)**

* **Tasks**: Document user stories, set up Visual Studio/Git, assess risks (e.g., file security).
* **Milestones**: Requirements, repo, risk register.
* **Dependencies**: None.

**Phase 2: Design (22 Sep-03 Oct 2025, 2 weeks, 80 hours)**

* **Tasks**: Design database schema, UI for Submit/Verify/Track, workflow.
* **Milestones**: ER diagram, workflow document.
* **Dependencies**: Phase 1.

**Phase 3: Development (06 Oct-07 Nov 2025, 5 weeks, 200 hours)**

* **Tasks**: Implement authentication, claim submission, verification/approval, tracking, and integration.
* **Milestones**: Functional features, integrated application.
* **Dependencies**: Phase 2.

**Phase 4: Testing (10-21 Nov 2025, 2 weeks, 80 hours)**

* **Tasks**: Unit tests, system/stakeholder testing, security checks.
* **Milestones**: 80% code coverage, bug fixes, security audit.
* **Dependencies**: Phase 3.

**Phase 5: Deployment (24 Nov-06 Dec 2025, 2 weeks, 80 hours)**

* **Tasks**: Deploy to Azure, train users, handover.
* **Milestones**: Live app, training, closure report.
* **Dependencies**: Phase 4.

**Timeline**: 12 weeks (15 Sep-06 Dec 2025).

**Resources**: Lead Developer, Junior Developer (full-time), Tester (part-time).

**Monitoring**: Weekly stand-ups.

|  |
| --- |
| **Claim** |
| ID:int(pk) |
| UserID:int(fk) |
| SubmissionDate:DateTime |
| HoursWorked |
| HourlyRate |
| TotalAmount |
| Status |
| Description |

UML Diagram

|  |
| --- |
| **Role**  1  0..\*  1  1  1  0..\*  Is\_a  Is\_a  Is\_a |
| ID:int (pk) |
| Name:String |
|  |

|  |
| --- |
| **User** |
| ID:int (pk) |
| Username:string |
| Passwordhash:string |
| RoleID:int(fk) |
| FullName: String |

|  |
| --- |
| **SupportingDocuments** |
| ID:int(pk) |
| ClaimID: int(fk) |
| FilePath:string |
| UploadDate |

|  |
| --- |
| **Lecture** |
| ID:int(pk) |
| Department |
|  |

|  |
| --- |
| **ProgrammeCoordinator** |
| ID:int(pk) |
| ProgrammeCode |
|  |

|  |
| --- |
| **AcademicManager** |
| ID:int(pk) |
| Faculty: string |
|  |