**Documentation**

Part 1 – Prog 6212

Project Planning and Prototype Development

1. **Documentation**

Design Choices:

Architecture: MVC (Model-View-Controller) pattern chosen to separate data, user interface, and control logic for better maintainability and scalability.

**Database**

SQL-based relational database with normalized tables to efficiently store lecturer, claim, approval, and document data.

**-- Lecturers Table**

CREATE TABLE Lecturers (

LecturerID INT PRIMARY KEY,

Name VARCHAR(100),

Email VARCHAR(100),

HourlyRate FLOAT

);

**-- Claims Table**

CREATE TABLE Claims (

ClaimID INT PRIMARY KEY,

LecturerID INT,

DateSubmitted DATE,

HoursWorked FLOAT,

ClaimStatus VARCHAR(20),

ApprovedBy VARCHAR(100),

FOREIGN KEY (LecturerID) REFERENCES Lecturers(LecturerID)

);

**-- Supporting Documents Table**

CREATE TABLE SupportingDocuments (

DocumentID INT PRIMARY KEY,

ClaimID INT,

FilePath VARCHAR(255),

DateUploaded DATE,

FOREIGN KEY (ClaimID) REFERENCES Claims(ClaimID)

);

**-- Approvals Table**

CREATE TABLE Approvals (

ApprovalID INT PRIMARY KEY,

ClaimID INT,

ApprovedBy VARCHAR(100),

ApprovalStatus VARCHAR(20),

ApprovalDate DATE,

FOREIGN KEY (ClaimID) REFERENCES Claims(ClaimID)

);

*This SQL code defines the Lecturers, Claims, Supporting Documents, and Approvals tables and their relationships*

**GUI Design**

User-friendly interface using .NET Core for lecturers to submit claims, and for administrators to review and approve them. The design includes a clear form layout and intuitive controls for easy navigation.

**Assumptions**

Lecturers submit claims monthly, and claims are processed within a week.

Only .pdf, .docx, and .xlsx file formats are accepted for document uploads.

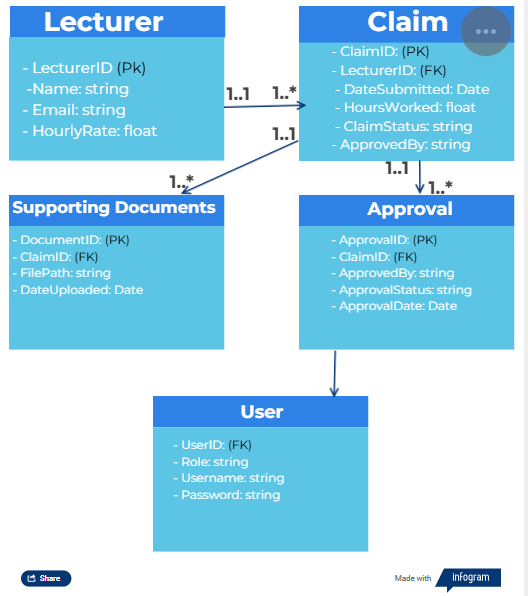
Claims will be approved hierarchically, first by the Programme Coordinator and then by the Academic Manager.

**Constraints**

File size limits for supporting documents to optimize system performance.

No payroll integration claims approval and payment processing are separate.

1. **UML Diagram**

****

This diagram outlines the key entities in the CMCS system:

* Lecturer has a one-to-many relationship with Claim (a lecturer can submit multiple claims).
* Claim has a one-to-many relationship with SupportingDocuments (a claim can have multiple supporting documents).
* Claim has a one-to-one relationship with Approval (each claim can have one approval status).
* User approves/reviews the claims.

1. **Project Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Duration** | **Dependencies** | **Assigned To** |
| Requirements Gathering | 1 day | None | Project Manager |
| Database Design | 3 days | Requirements Gathering | Database Admin |
| Prototype GUI Design | 6 days | Database Design | Front-End Dev |
| UML Class Diagram Development | 2 days | Prototype GUI Design | Architect |
| Front-End Design (MVC) | 5 days | UML Class Diagram | Front-End Dev |
| GUI Prototype Review | 1 days | Front-End Design | QA Team |
| Report Writing | 1 days | GUI Prototype Review | Documentation |

1. **GUI Design (Prototype)**

The front-end prototype is designed using .NET Core MVC. The GUI contains the following:

**Lecturer Claim Submission**

A form that includes fields for:

* HoursWorked
* HourlyRate
* Supporting document upload (.pdf, .docx, .xlsx)
* Submit button for claim submission.

**Admin View**

* A list of submitted claims for Programme Coordinators and Academic Managers.
* Buttons for Approve and Reject.
* Status updates displayed as Pending, Approved, or Rejected.

**REFERENCE**

GeeksforGeeks 2017. Unified Modeling Language (UML) | An Introduction - GeeksforGeeks. GeeksforGeeks. Available at: <https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/>.

Infogram 2018 Create Infographics, Reports and Maps - Infogram. online Infogram. Available at: <https://infogram.com/>.

‌

**‌**