1. **Documentation**

1.1 **Design Choices**

* + 1. Overview: Explain the goals and parameters of the CMCS. Describe the advantages it will have for the administrative personnel and Independent Contractor (IC) lecturers. Emphasise the system's primary features, such as the ability to submit claims, upload documents, and initiate approval procedures.
    2. **System Architecture:**
* **Architecture Diagram**: Provide a high-level diagram that displays the system's components and how they interact. The database, data access layer, business logic, and user interface (UI) may all be examples of this.

**•** Technology Stack: List the technologies that are being used (e.g., SQL Server for database, MVC for UI, and.NET Core for backend).

* 1. **Database Design**

**1.2.1 UML Class Diagram:**

* **Entities**: List the principal entities, such as the lecturer, the claimant, the program coordinator, and the academic manager.
* **Attributes**: List the essential characteristics of every entity, such as the Lecturer's ID, name, and email; the Claim's ID, LecturerID, hours worked, hourly rate, and status.
* **Relationships**: Describe the connections between the entities (e.g., Lecturer and Claim are one-to-many).

1.2.2 **Database Schema:**

* Tables: List the columns in the tables that correspond to each entity.
* Primary Keys: Specify each table's primary keys.
* Foreign Keys: To ensure data integrity, specify foreign key associations.

1.3 **GUI Design**1.3.1 **User Interface Layout:**

* Wireframes/Mockups: Feature visuals of important screens (such as the Claim Approval Dashboard and Claim Submission Form).
* Components: Describe the primary UI elements, such as buttons, forms, and navigation menus.

1.3.2 **User Interaction Flow**:

* Flowcharts: Describe user interactions with the system, highlighting important actions and replies from the system.
* Navigation: Explain the process by which users will switch between screens.

1.4 **Assumptions and Constraints**

1.4.1 **Assumptions**:

* User Base: Assume academic managers, program coordinators, and lecturers will use the system.
* System Requirements: Explain any suspicions you may have regarding the technology stack or other prerequisites.

1.4.2 **Constraints:**

* Budget and Time: Describe any restrictions pertaining to project budgets or schedules.
* Technical Constraints: Indicate any restrictions pertaining to the technologies employed or the range of the prototype.  
    
  **Example Outline:**

1. **Introduction**

* Purpose of CMCS
* Key functionalities

1. **System Architecture**

* Architecture Diagram
* Technology Stack

1. **Database Design**

* UML Class Diagram
* Database Schema

1. **GUI Design**

* Wireframes/Mockups
* User Interaction Flow

1. **Assumptions and Constraints**

* Assumptions
* Constraints