|  |  |
| --- | --- |
| A picture of a winding road and trees  Prototype of monthly claim system  PROG6212 | This is Part 1, and a prototype of the Monthly Claim System. In this document is a brief explanation of the structure and design of the system, a UML class Diagram of the system, and a link to the GUI of the System.  Sizwe Thandaza Majola  ST101143043 |

**Documentation:**

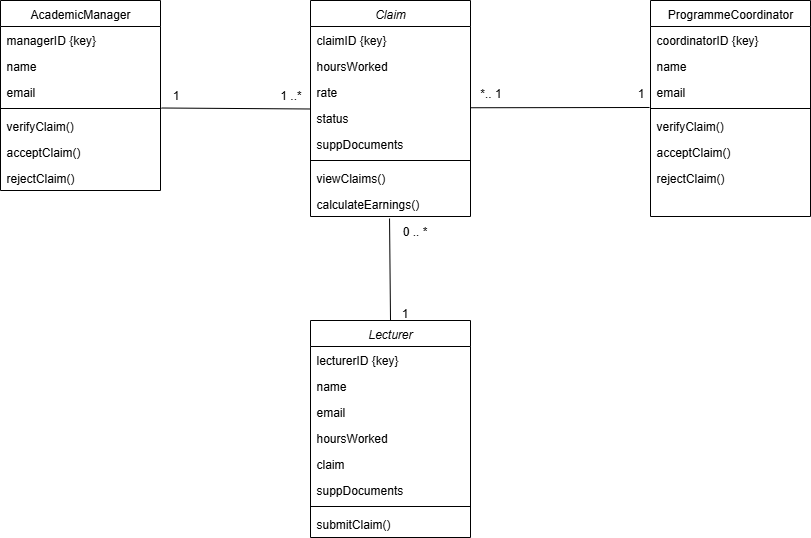
Based on the requirements of this system, the type of database that will be used is a relational database, due to the structured and repetitive nature of the system, which involves the lecturer submitting their claims of the hours they have worked and calculated, along with any necessary supporting documentation, and the programme coordinator, as well as the academic manager, reviewing and verifying the claims to approve, or disapprove of them.

This type of database supports this type of system, due to the unchanging nature of the system and the need for integrity. As claims by lecturers must be accurate and remain consistent, they can be paid according to the hours they have honestly and earnestly worked. To ensure consistent data storage, the same structure must be maintained and followed, guaranteeing transparency and simplicity in storing claims, supporting documents, approvals, and disapprovals. This ensures that all necessary information is readily available to relevant lecturers, coordinators, or managers (reference). Another reason for the choice of this sort of database is that it is better suited to storing and repeating the stored data, which is better for consistency, reliability, security, and assurance, with only authorized coordinators and managers having access to approving and viewing the claims of the lecturers. This consistency makes it easier for the database to keep track of claims, documents, and information, as well as make it more able to make hourly rate calculations for the lecturers (Satzinger, Jackson, and Burd, 2012).

The limitations in terms of the required structure make it even more suitable for the chosen type of database, as a specific input is required, considering the claims and their supporting documents. Should the system grow large over time and accommodate more contractor lecturers, or should the system request to have a messaging function on the app, it won't be feasible for the system, as it would require a database that is unstructured in nature, for more space for the data, and for any new sorts of docs, or files that will be shared on it.

The structured database lacks the capacity for growth and innovation due to these limits. Storage on this database will also have to be managed well over time, as it doesn’t have the ability to handle large amounts of data, unlike the NoSQL relational, or non-structured database.

**UML Class Diagram for Monthly Claim System:**



**Project Plan of Monthly Claims System:**

* Project Plan for Monthly Customer Claims System
* The development of the Contract Monthly Claim System (CMCS) prototype will follow a structured six-week plan, combining iterative waterfall methodology with agile-inspired feedback loops.
* In the first week, the project will begin with initiation tasks including scope definition, stakeholder identification, and requirements gathering.
* In the second week, the project will focus on system design, producing UML Class Diagrams, ERDs, and a database schema aligned with textbook principles.
* In the third week, the frontend development will be initiated using .NET GUI and CH interface tools to build the lecturer claim submission form and document upload features.
* In the fourth week, the plan will focus on backend development, implementing claim logic, database integration, and approval workflows for Programme Coordinators and Academic Managers.
* The fifth week will involve system integration and testing, validating calculations, user interactions, and data flow. Finally,
* The last week will be dedicated to refinement, user feedback, and POE documentation, culminating in the presentation of a fully functional prototype. This plan ensures accuracy, transparency, and repeatability while enhancing user satisfaction through a seamless and interactive experience.

**Reference List:**

Satzinger, J.W., Jackson, R.B., and Burd, S.D, 2012. Systems Analysis and Design in a Changing World. 6th ed. Boston: Cengage Learning. Available at:

<<https://archive.org/details/systemsanalysisd0000satz>>

[Accessed 16 September 2025]