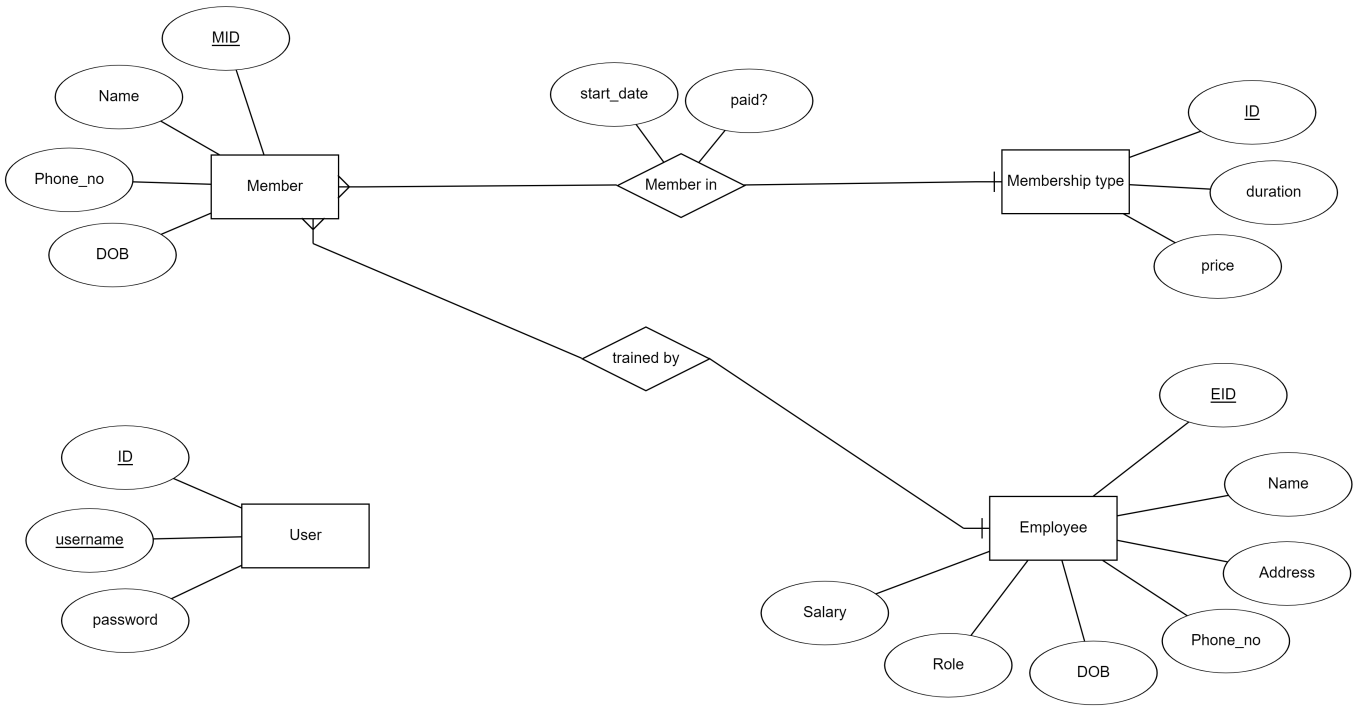
Design document

This project is build using on a **Client-server Architecture** (discussed in the previous section).

OOP was used in this project, since it's the most suitable for our application. which consists of: Member, employee, user and membership type.

using **OOP** made it easier to model the database tables to our application using **ORM**(object relational model) and **DAO**(data access object). also made it possible to use **UML diagrams**to help in the design process.

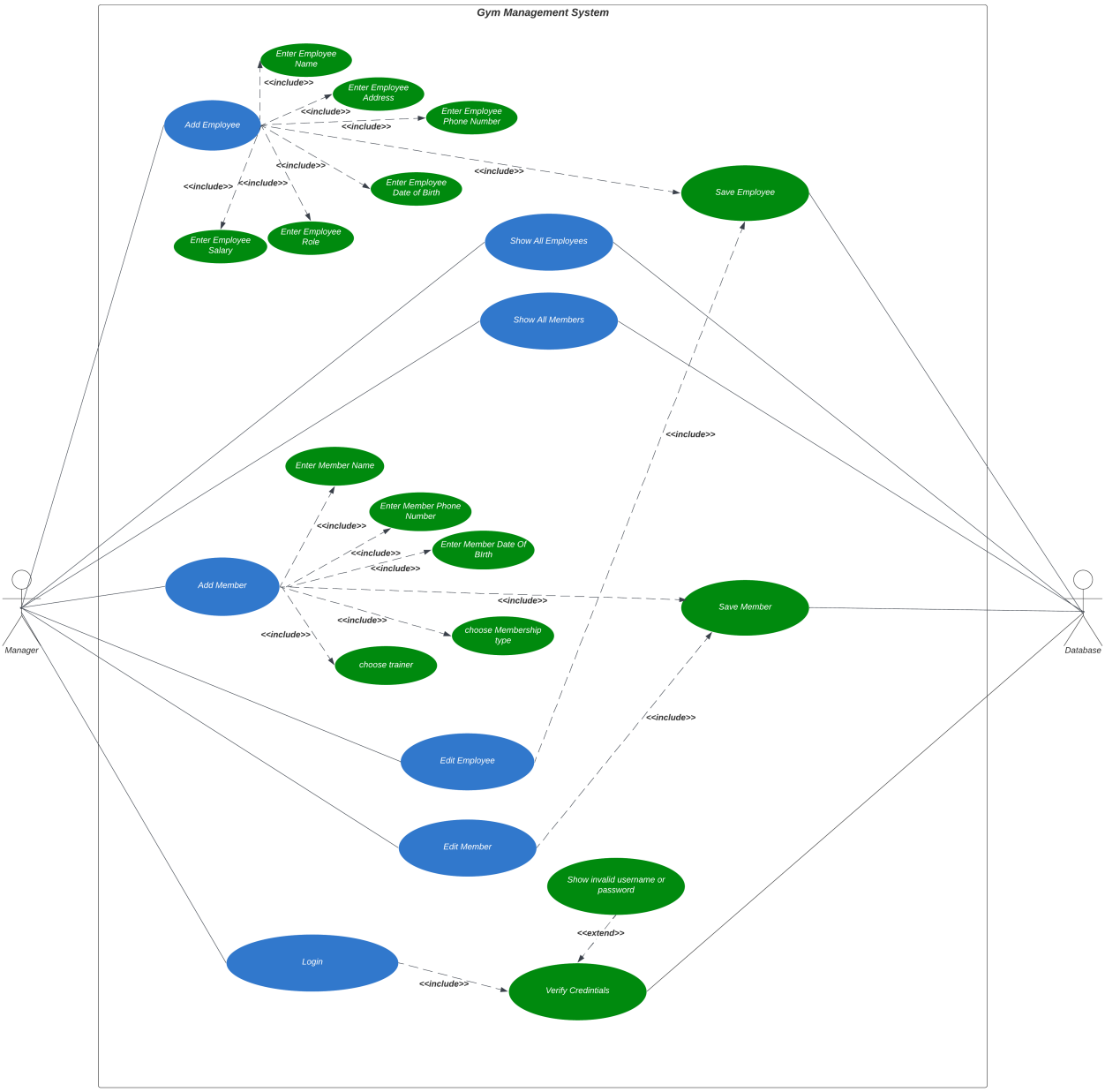
Entity relationship diagram



UML diagrams

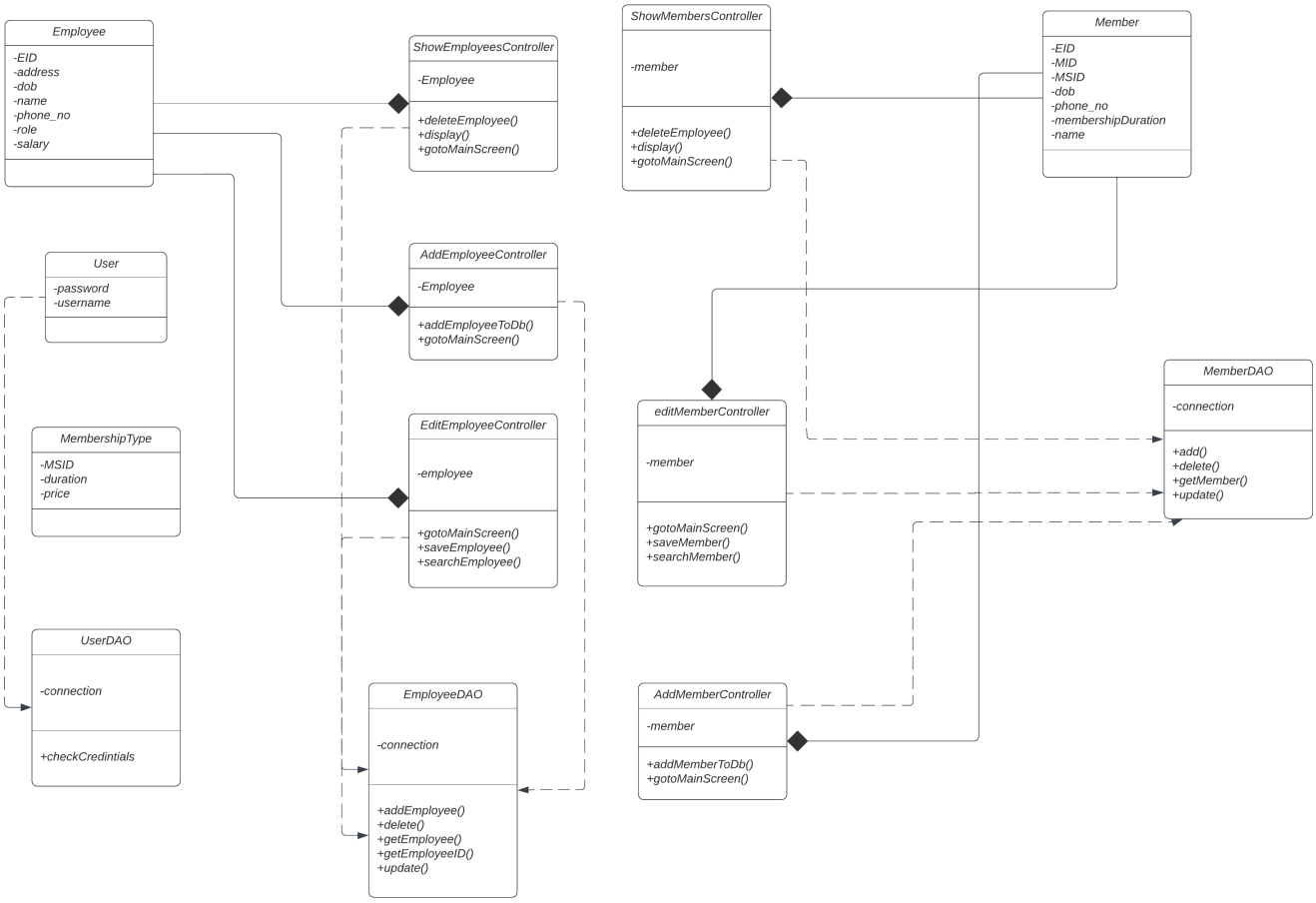
use case diagram:

A **use case diagram** is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.



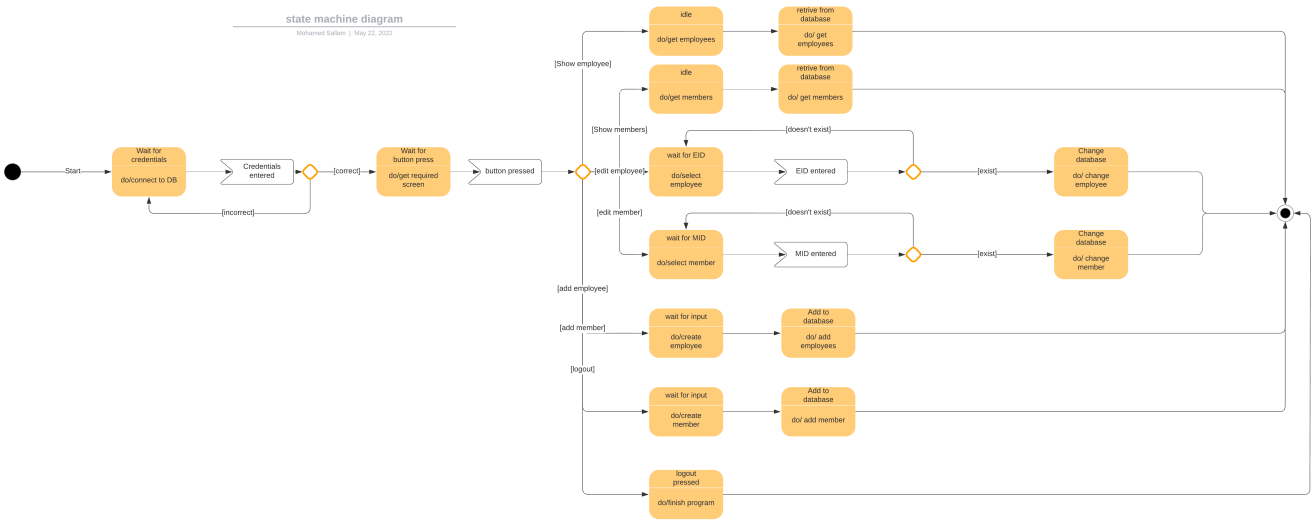
Class diagram:

A type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.



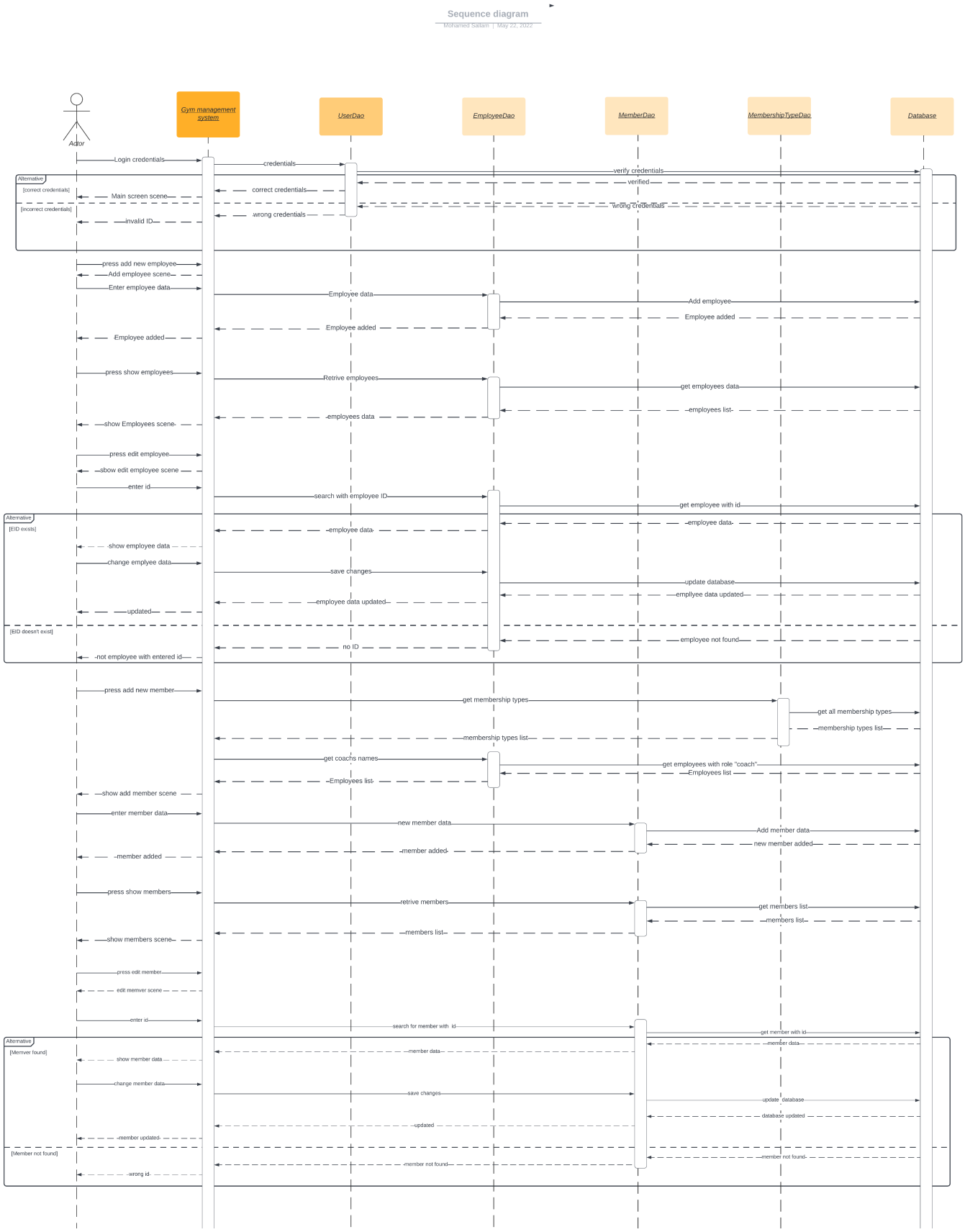
State machine diagram:

models the behaviour of a single object.



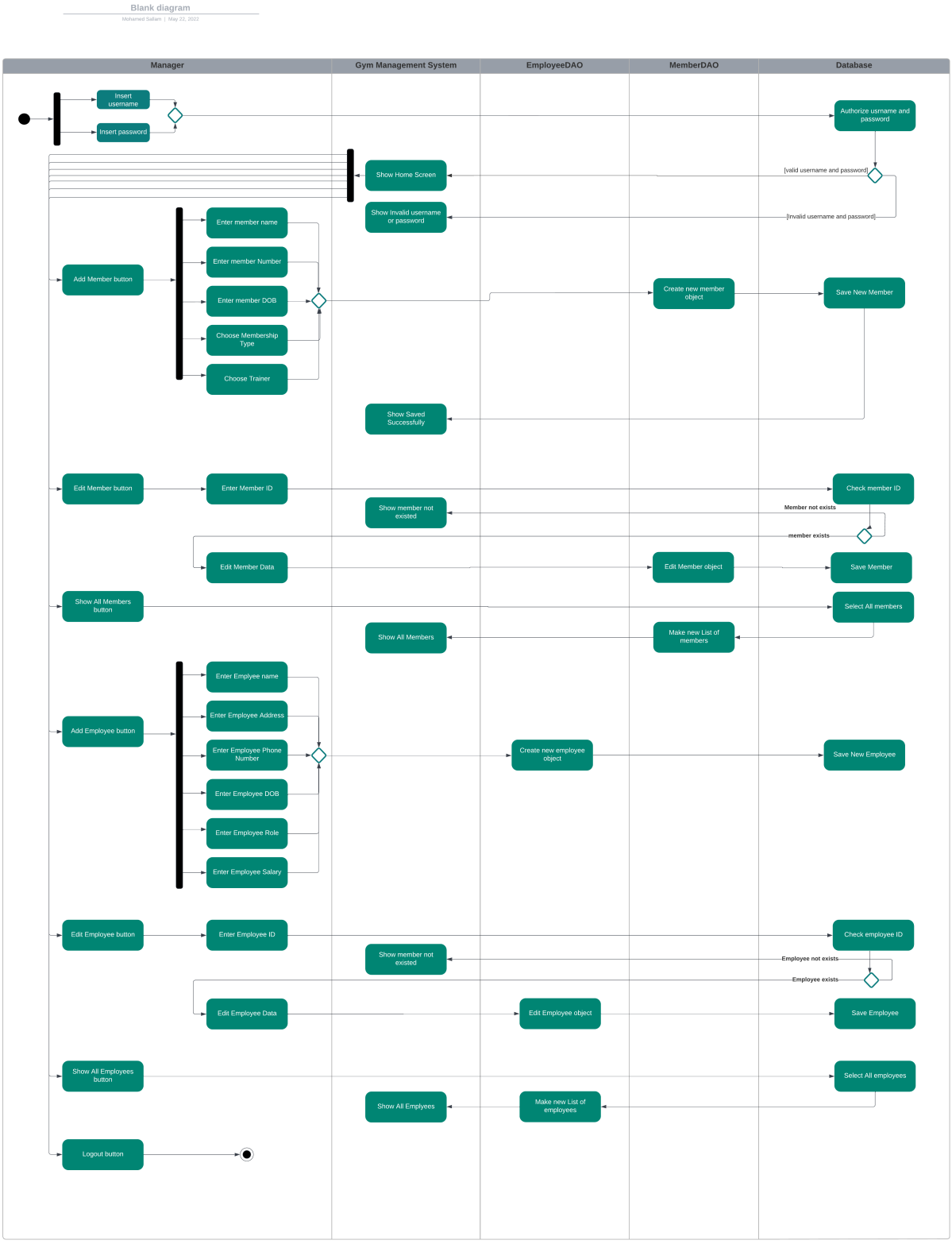
Sequence diagram:

shows process interactions arranged in time sequence.



Activity diagram:

graphical representations of workflows of stepwise activities and actions.



Design description

With focuse on ease of use and good GUI the system was fairly system with system working as follow:

1. Entering credentials in the login screen.

2. Login screen transfers you to the main menu screen.

3. In the main menu, several buttons exists with each button describing what it does.

Implementation

enviroment and tools used by all the team:

- MySQL as a **relational database management system**.
- Java was the language of choice.
- [IntelliJ IDEA](IntelliJ IDEA: The Capable & Ergonomic Java IDE by JetBrains) was used by the whole team for writing **Java** code.
- **JDK 8.1.3** was used.
- Visual Studio Code was used for manipulating database **DDL**.

> This document was supposed to be written according to the [IEEE 1016-2009](1016-2009 - IEEE Standard for Information Technology--Systems Design--Software Design Descriptions | IEEE Standard | IEEE Xplore), but since it requires a paid access we haven't been able to follow its guidelines, so this a design document based on what we have studied.