Final Project V2 Report
COMP 3005
Health and Fitness Club Management System

Harishan Amutheesan, 101154757
Ali Abdollahian, 101229396

Conceptual Design

The conceptual design of the Health and Fitness Club Management System centers on an Entity-Relationship (ER) diagram that effectively models the relationships between various entities such as members, trainers, administrative staff, and other components vital to the club's operations. This diagram is instrumental in visualizing data structures and their interconnections, thereby facilitating a systematic approach to database design.

Entities and Relationships

Members:

Represents individuals who join the club to achieve personal fitness goals.
Attributes: Member ID, Name, Date of Birth, Fitness Goals, Weight Goal, Time Goal.
Relationships: Members can enroll in multiple classes and schedule sessions with trainers.

Trainers:

Represents fitness professionals who provide personal training and lead fitness classes.
Attributes: Trainer ID, Name, Availability.
Relationships: Trainers lead various classes and have scheduled sessions with members.

Administrative Staff:

Manages operational aspects such as room bookings, class schedules, equipment maintenance, and financial transactions.
Attributes: Admin ID, Name, Email.
Relationships: Oversee room bookings, class schedules, and equipment maintenance.

Classes:

Fitness classes offered by the club, ranging from yoga to high-intensity interval training.
Attributes: Class ID, Name, Schedule, Trainer ID.
Relationships: Each class is assigned a trainer and has enrollments from multiple members.

Room Bookings:

Management of spaces within the club for various activities and classes.
Attributes: Booking ID, Room ID, Booking Date, Start Time, End Time.
Relationships: Admins handle the scheduling, ensuring rooms are available for classes and events.

Equipment Maintenance:

Ensures all fitness equipment is in optimal working condition.
Attributes: Maintenance ID, Equipment ID, Date, Type of Maintenance.
Relationships: Linked to specific pieces of equipment, each potentially having multiple maintenance records.

Billing and Payments:

Handles all financial transactions within the club, including membership fees and payments for classes.
Attributes: Transaction ID, Member ID, Date, Amount, Status.
Relationships: Each member has associated billing records, which are crucial for financial management.

Assumptions

Cardinality:

- Members can enroll in multiple classes, but each class at a specific time is led by one trainer.
- Trainers can lead multiple classes; however, each class session is tied to a single trainer.
- Equipment maintenance records are specific to each piece of equipment, with potential for multiple records per equipment.
- Room bookings are specific to individual rooms, which can be booked multiple times.

Participation Types:

- All members are linked to at least one billing record but are not required to enroll in classes.
- Each trainer is required to manage or lead at least one class.
- Each class must have a room booked to ensure space allocation is managed efficiently.

Reduction to Relational Schemas

Refer to Github for ER-Diagram and Relational Schemas

DDL & DML File

Refer to GitHub

Implementation

We implemented this application using Java, adhering to the Model-View-Controller (MVC) design pattern. This architectural choice allows for modularity, which is vital for maintaining the system's scalability and manageability. In our implementation, the model layer directly interacts with a PostgreSQL database through JDBC, handling all data-related operations including the CRUD functionalities—creating, reading, updating, and deleting data relevant to club members, trainers, and administrative staff. This ensures efficient data management and integrity across the system.

For the view component, we used Java Swing because of our familiarity with it and its mature library for building GUIs. Java Swing provides a variety of components that enhance user experience and make the application's interface user-friendly. We designed the GUI to ensure smooth navigation and responsiveness to user inputs. Each GUI component, from forms for data entry to tables for displaying schedules, is dynamically linked to the underlying model data, for real-time updates and changes.

The controller serves as the intermediary, processing user inputs from the view, invoking model operations, and updating the view. This setup allows us to decouple the application's logic from its presentation layer, simplifying enhancements and maintenance. We incorporated Maven to manage project dependencies and simplify our build processes, ensuring that all team members worked in a consistent environment.

Our use of PostgreSQL along with PgAdmin for database management, supports SQL queries for handling data interactions required by a club management system. The database schema is designed to efficiently store and retrieve all necessary data, from member registrations to scheduling and billing, supporting the system's extensive functionality.
Through planning and implementation, we made a system that not only meets the specific needs of this project but also offers flexibility for future expansions, such as integrating new member services or enhancing existing functionalities.

Bonus Features

We implemented several bonus features that enhance the user experience and provide additional functionality beyond the basic requirements. These features are designed to increase engagement and provide valuable feedback to both the users and the club's administration.

Leaderboards
This feature ranks members based on their achievements and activity levels within the club. By leveraging the data captured through member activities and achievements, we dynamically generate leaderboards that are displayed within the member's dashboard. This motivates members by fostering a sense of competition and community, encouraging them to engage more actively with various classes and trainers. Leaderboards are updated in real-time, reflecting each member's progress and ensuring that all achievements are promptly recognized.

Feedback System
This feature allows members to provide feedback on various aspects of the club, including personal training sessions, group fitness classes, and the overall club. Feedback is crucial for continuous improvement, and by implementing this system, we enable the club to gather insights directly from its members. The feedback data is seen by administration to identify areas of strength and potential improvement.

GitHub Repository

https://github.com/AliAbdollahian/Health-and-Fitness-Club-Management-System