

# COMP 3005A - Project: Health and Fitness Club Management System

**Instructor:** Abdelghny Orogat

Due: Dec. 10, 2023 (11:59 PM)

## 1 Problem Statement

Design and implement an application for a "Health and Fitness Club Management System". This system will serve as a comprehensive platform catering to the diverse needs of club members, trainers, and administrative staff.

Members should be able to **register** and **manage their profiles**, **setting personal fitness goals** and **inputting health metrics**. Once registered, they will gain access to a **personalized dashboard** that **tracks their exercise routines**, **fitness achievements**, and **health statistics**. The platform will also enable members to **schedule**, **reschedule**, or **cancel** personal **training sessions** with **certified trainers**. Furthermore, members can **register for group fitness classes**, **workshops**, and **other events**, ensuring they always **stay updated with their schedules** and **receive timely reminders for their sessions**.

On the other side, the system should empower **trainers** with tools **to manage their schedules**, **view member profiles**, and **input progress notes after each training session**. **Administrative staff**, the backbone of the club's operations, should **have features** that allow them to **oversee club resources effectively**. This includes **managing room bookings**, **monitoring fitness equipment maintenance**, and **updating class schedules**. Additionally, they should have a robust system to **oversee billing**, **process payments** for **membership fees**, **personal training sessions**, and **other services**, and **monitor club activities for quality assurance**. The club's unique selling point is its **loyalty program**; **every transaction earns members loyalty points**, which can be **redeemed for future services**.

## 2 Project Report

You need to submit one report file that contains the following sections. You can add other sections, but the following sections (except Bonus Features) must be in the report:

### 2.1 Conceptual Design

Explain the **conceptual design of the database**: the ER-diagram for the Health and Fitness Club, and **assumptions made** regarding cardinalities and participation types. Ensure that assumptions align with the problem statement in Section 1.

### 2.2 Reduction to Relation Schemas

Reduce your **ER-diagram** into **relation schemas and list them**.

### 2.3 Normalization of Relation Schemas

Given the problem statement and your design, write functional dependencies. Show that your schemas are in a good normal form or demonstrate their decomposition into one.

### 2.4 Database Schema Diagram

Show the final schema diagram for the club. It should resemble the schema diagram of a university database, as covered in this course.

## 2.5 Implementation

Your application can be web or desktop-based. Describe your application's architecture, including **module interactions**. Include **a diagram and explain application usage scenarios and workflows**. Ensure you use a relational database.

Your application should have two user interfaces: one for members and another for club managers and staff. Include screenshots of your interfaces in different scenarios.

## 2.6 Bonus Features (Optional)

Add bonus features to your system for extra marks. For instance, a recommendation system or integrating wearable devices to track workouts.

## 2.7 GitHub Repository

Upload your source code to a public GitHub repository. Include a comprehensive README and a directory titled "SQL" with all the SQL statements and queries used. List the repository URL here.

# 3 Instructions for Submission

Adhere to these guidelines:

- **Video Demonstration:** Submit a 10-minute video that thoroughly demonstrates your application. The video should:
  - Highlight the key functionalities of the application.
  - Walk through the primary use cases for both members and staff interfaces, illustrating the user experience and responsiveness.
  - Provide a brief overview of the code structure, emphasizing critical components and design choices.
  - Ensure clarity and focus, especially when presenting the user interfaces.

Make sure that the video is well-paced, with a clear voiceover, and covers all essential aspects of your project.

- **GitHub Repository:** Upload all source code to a public GitHub repository. Organize your repository with distinct folders for project components. Include a detailed README.
- **Code Structure:** Maintain a logical code structure with clear naming conventions, function separation, and in-code documentation.
- **Video Submission:** Upload your video to platforms like YouTube or Vimeo. Ensure the video is accessible to the TAs.

TAs will grade based on the video demonstration. Ensure your application functions as described and that the video is of high quality.

# 4 Group Rules

## 4.1 Group Composition

Collaboration is encouraged, but groups should have no more than three students. All members must participate in both DB design and implementation.

## **4.2 Bonus Marks for Smaller Groups**

To recognize the additional effort required for smaller groups:

- For individual projects (1 student), a bonus of 20% of the total possible marks will be added.
- For groups of two students, a bonus of 10% of the total possible marks will be added.

Note: Groups of three students will receive no bonus as they have the full collaborative advantage.

## **4.3 Video Guidelines for Groups**

Detail each member's contribution in the video, with each member explaining their specific roles in the project's development.