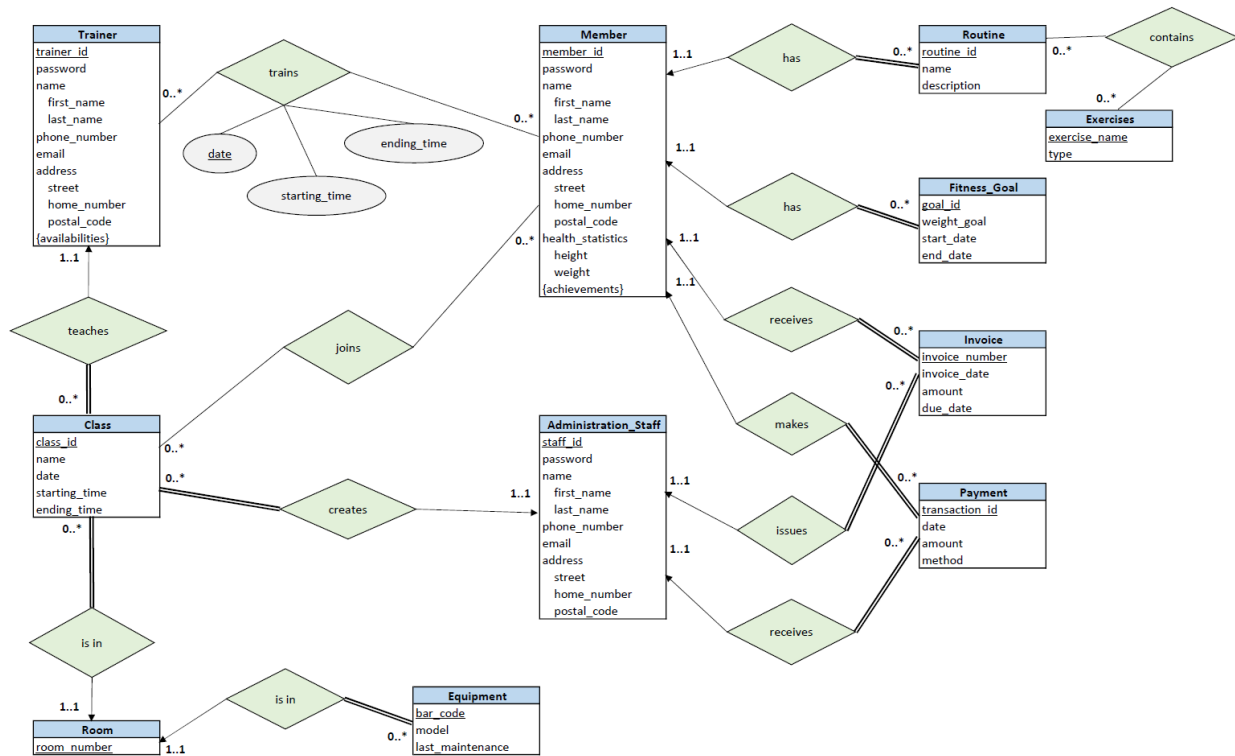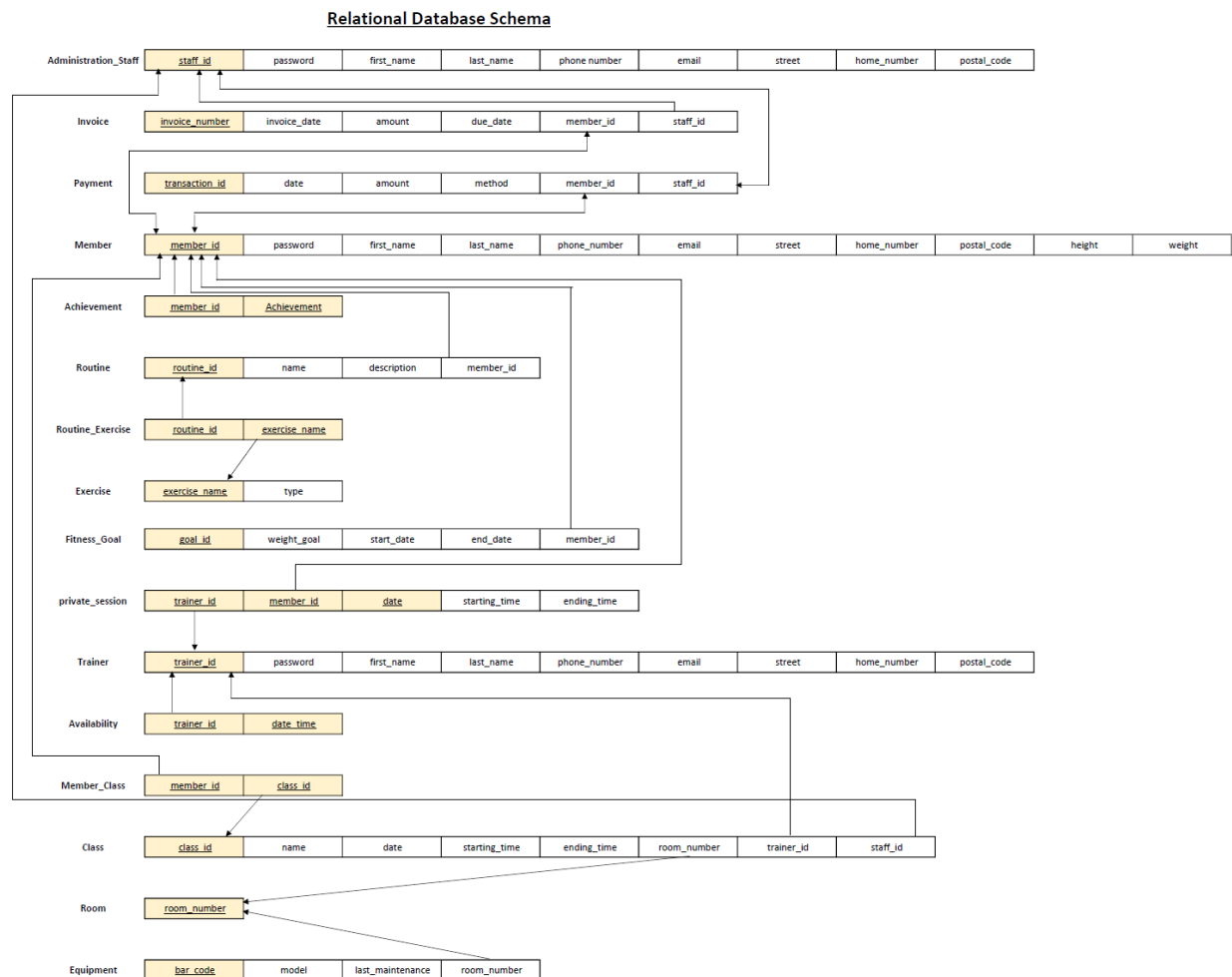# COMP3005 Project_V2 Report

Conceptual Design

ER Model (UML-Like Notation)

Assumptions

- Trainers' availabilities (multi-value field) are in blocks of one hour for simplicity (i.e. if there's block of availability that starts at 8:00, it will end at 9:00). A block is removed from the availabilities when a member books for that block so no other member might also book it.
- A class always has one trainer. A trainer can teach multiple classes.
- A class is always located in a room. A room does not necessarily need to have a class.
- An equipment is always located in a room. A room does not need to have any equipment.
- Room booking is only made for classes.
- A class is always created by an administration staff. Not all administration staff need to create classes.
- A class can have many members, and members can join multiple classes.
- A trainer can train multiple members (but only one at a time). He can also train the same member more than once (assumed that the maximum amount of time a trainer can train a member in one day is only one). A member can be trained by multiple trainers (but only one at a time). He can also be trained by the same trainer more than once (assumed that the maximum amount of time a member can be trained by a trainer in one day is only one).
- A member can have multiple routines. A routine is always associated with a member.
- A routine can be empty or have 1 or more exercises. An exercise does not need to be attached to a routine.
- A member does not need to have any fitness goals. A fitness goal must always be linked to a member.
- A member can have zero, one or more achievements (multi-value field).
- An invoice is always created by an administration staff and assigned to a member. Not all administration staff need to create invoices, and not all members need to have an invoice (for example, new members). The same applies to payments.

Reduction to Relation Schemas

Relational Database Schema

Administration_Staff	staff_id	password	first_name	last_name	phone number	email	street	home_number	postal_code

Invoice	invoice_number	invoice_date	amount	due_date	member_id	staff_id

Payment	transaction_id	date	amount	method	member_id	staff_id

Member	member_id	password	first_name	last_name	phone_number	email	street	home_number	postal_code	height	weight

Achievement	member_id	Achievement

Routine	routine_id	name	description	member_id

Routine_Exercise	routine_id	exercise_name

Exercise	exercise_name	type

Fitness_Goal	goal_id	weight_goal	start_date	end_date	member_id

private_session	trainer_id	member_id	date	starting_time	ending_time

Trainer	trainer_id	password	first_name	last_name	phone_number	email	street	home_number	postal_code

Availability	trainer_id	date_time

Member_Class	member_id	class_id

Class	class_id	name	date	starting_time	ending_time	room_number	trainer_id	staff_id

Room	room_number

Equipment	bar_code	model	last_maintenance	room_number

Explanation

- There is no 1-to-1 relationship set.
- Most of the relationship sets are a 1-to-Many relationship.
- All the 1-to-Many relationships are also Total Participation. Therefore, no extra tables were needed to map them.
- There are two Many-to-Many relationship sets (between Trainer and Member, and between Member and Class). A table was required for each one of them.
- There are 2 multi-value fields. A table was required for each one of them.

DDL File

https://github.com/MathCU/COMP3005_Project/blob/main/SQL/DDL.sql

Explanation

- Most of the attributes are required, therefore the "NOT NULL" constraint has been added to most of them.
- Tables with Foreign Keys are created after the tables they refer to.

DML File

https://github.com/MathCU/COMP3005_Project/blob/main/SQL/DML.sql

Explanation

- Values were added to testing purposes.
- Attributes categorized as "SERIAL" in the DDL don't need any data input since they will be automatically generated.

Implementation

https://github.com/MathCU/COMP3005_Project/blob/main/Health_and_Fitness_Club_System.py

Explanation

- The navigation consists of 4 menus. There is the main menu, the menu for existing users, the menu for trainers and the menu for the administration staff.
- Users just need to follow the prompts to navigate around the system.
- New users can be created from the main menu.
- To access their menu/account, the user will need to select is role (existing user, trainer, administration staff) on the menu then enter their credentials (ID and password).
- Users can logout when they are done using the system.
- Existing users can see their personal information, see their dashboard, see their bookings, schedule/cancel personal training sessions, join/leave classes, update their personal information/health metrics, create/delete goals, add achievements, lookup/create//modify/delete routines and see the list of exercises they can add to their routines.

- Trainers can search for members by their names. They can also show/add/delete their own availabilities.
- Staff members can search for members by their names, create invoices, receive payments, see a list of equipment, update the maintenance date of the equipment, see a list of classes, create/delete classes.

Bonus Features (Optional)

- Accounts have passwords. Users can login with their credentials. They can also logout when they are done using the systems.

GitHub Repository

https://github.com/MathCU/COMP3005_Project

Video

https://youtu.be/VHnA76F2sWQ