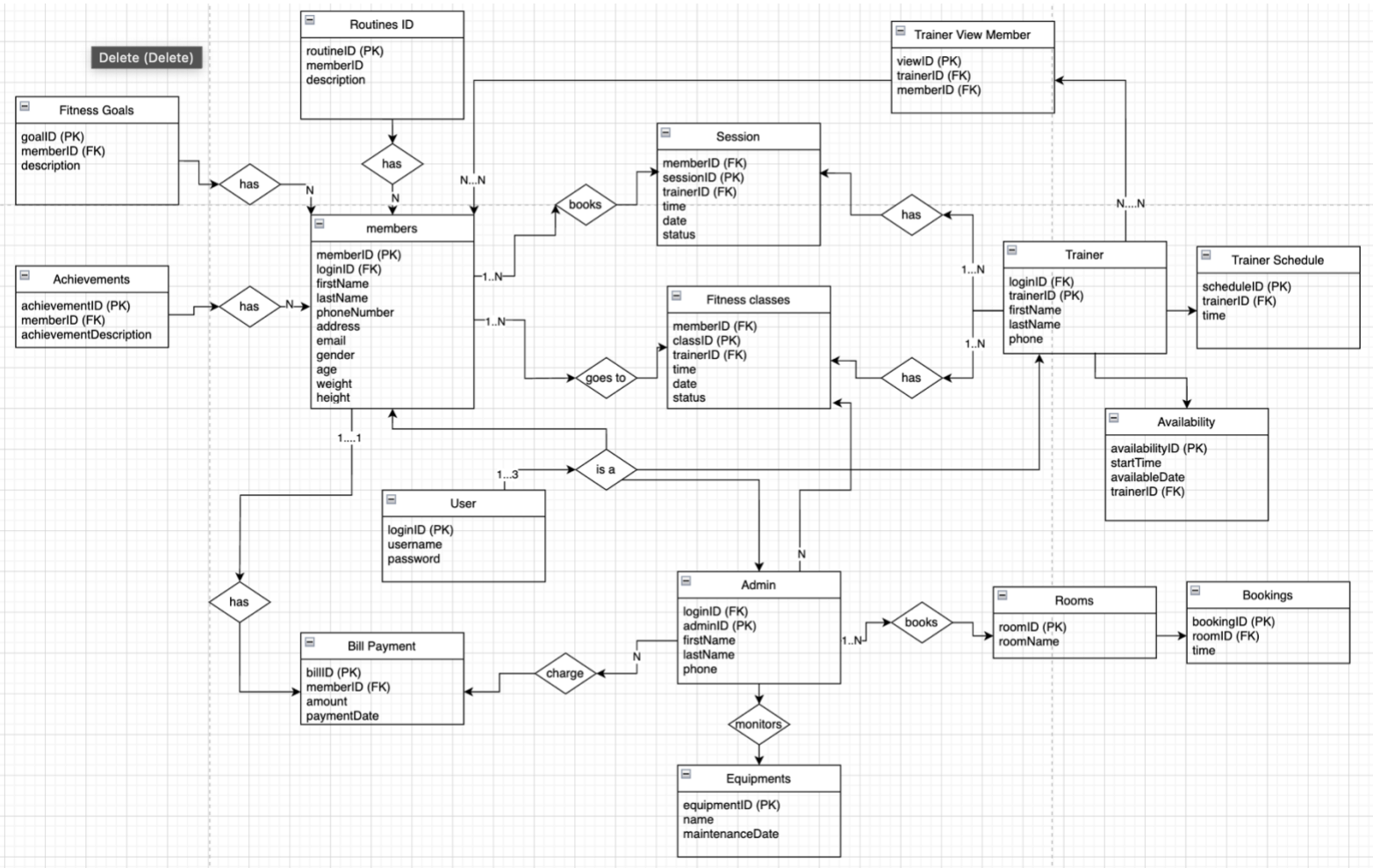
COMP 3005 Final Project Report

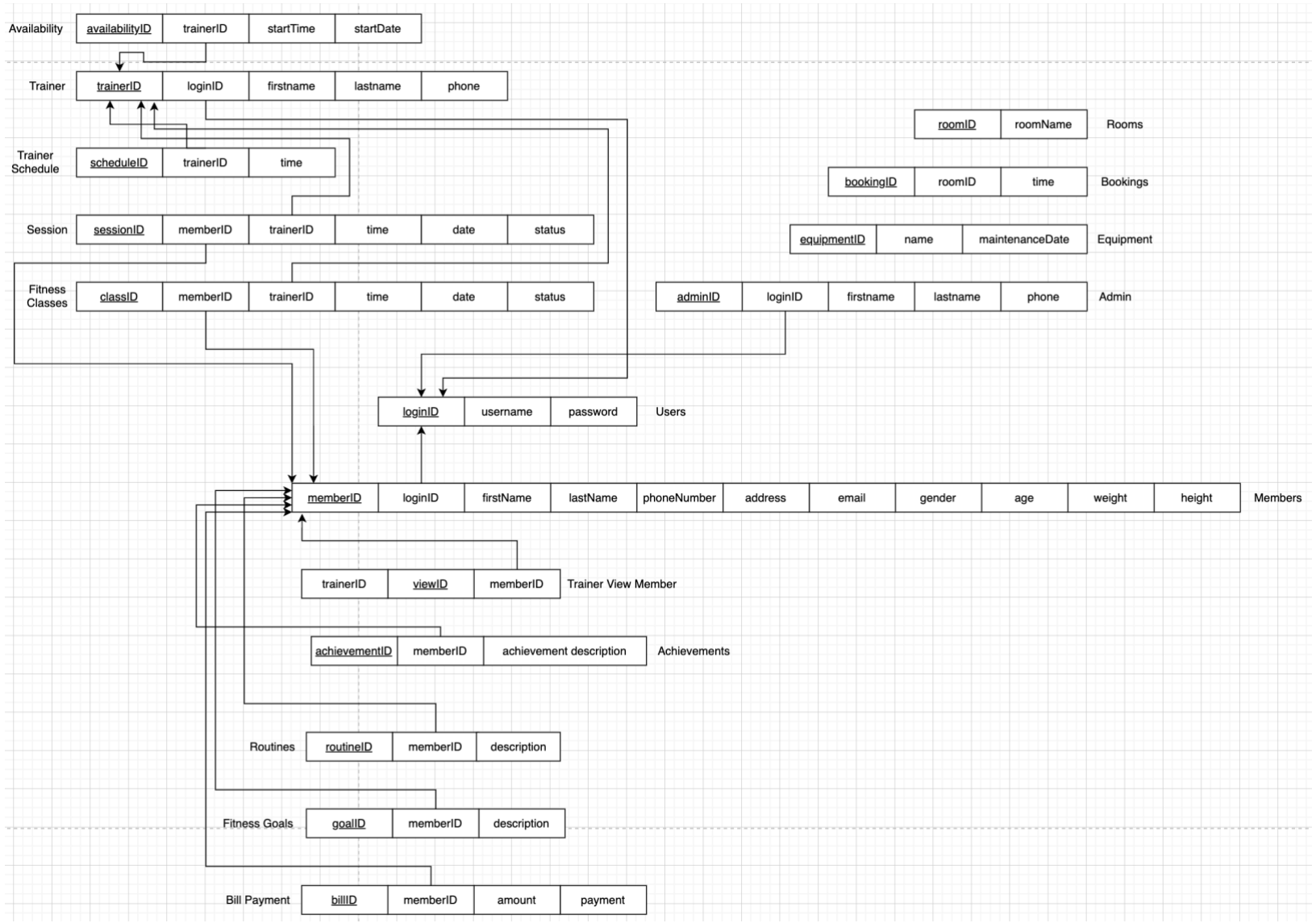
By Khun Thu Rein and Daniel Samchenko (Group 84)

Health and Fitness Management CLI Application

Conceptual Design
ER Diagram



Relational Schema



DDL Statements

```
-- User table
CREATE SEQUENCE user_login_seq;

CREATE TABLE UserLogin (
    loginID INT DEFAULT nextval('user_login_seq') PRIMARY KEY,
    username VARCHAR(50) NOT NULL,
    password VARCHAR(50) NOT NULL
);

-- Member table
CREATE SEQUENCE member_login_seq;

CREATE TABLE Member (
    memberID INT DEFAULT nextval('member_login_seq') PRIMARY KEY,
    loginID INT,
    firstname VARCHAR(50) NOT NULL,
    lastname VARCHAR(50) NOT NULL,
    phone VARCHAR(20),
    address VARCHAR(100),
    email VARCHAR(100),
    gender CHAR(1),
    age INT,
    weight DECIMAL(5,2),
    height DECIMAL(5,2),
    FOREIGN KEY (loginID) REFERENCES UserLogin(loginID)
);

-- Trainer table
CREATE TABLE Trainer (
    trainerID INT PRIMARY KEY,
    loginID INT,
    firstname VARCHAR(50) NOT NULL,
    lastname VARCHAR(50) NOT NULL,
```

```
    phone VARCHAR(20),
    FOREIGN KEY (loginID) REFERENCES UserLogin(loginID)
);

-- TrainerMemberAccess table
CREATE TABLE TrainerViewMember (
    trainerID INT,
    memberID INT,
    PRIMARY KEY (trainerID, memberID),
    FOREIGN KEY (trainerID) REFERENCES Trainer(trainerID),
    FOREIGN KEY (memberID) REFERENCES Member(memberID)
);


-- Achievements table
CREATE TABLE Achievements (
    achieveID INT PRIMARY KEY,
    memberID INT,
    achieveDescription TEXT,
    FOREIGN KEY (memberID) REFERENCES Member(memberID)
);

-- FitnessGoals table
CREATE TABLE FitnessGoals (
    goalID INT PRIMARY KEY,
    memberID INT,
    description TEXT,
    FOREIGN KEY (memberID) REFERENCES Member(memberID)
);

-- Routines table
CREATE TABLE Routines (
    routineID INT PRIMARY KEY,
    memberID INT,
    description TEXT,
    FOREIGN KEY (memberID) REFERENCES Member(memberID)
);
```

```sql
CREATE SEQUENCE session_id_seq;

-- Alter the Session table to use the sequence
CREATE TABLE Session (
    sessionID INT DEFAULT nextval('session_id_seq') PRIMARY KEY,
    memberID INT,
    trainerID INT,
    time TIME,
    date DATE,
    status VARCHAR(20),
    FOREIGN KEY (memberID) REFERENCES Member(memberID),
    FOREIGN KEY (trainerID) REFERENCES Trainer(trainerID)
);

-- FitnessClasses table
CREATE SEQUENCE class_id_seq;

CREATE TABLE FitnessClasses (
    classID INT PRIMARY KEY DEFAULT nextval('class_id_seq'),
    memberID INT,
    trainerID INT,
    time TIME,
    date DATE,
    status VARCHAR(20),
    FOREIGN KEY (memberID) REFERENCES Member(memberID),
    FOREIGN KEY (trainerID) REFERENCES Trainer(trainerID)
);

-- TrainerSchedule table
CREATE TABLE TrainerSchedule (
    scheduleID INT PRIMARY KEY,
    trainerID INT,
    time TIME,
    FOREIGN KEY (trainerID) REFERENCES Trainer(trainerID)
);
```

```sql
-- Admin table
CREATE TABLE Admin (
    adminID INT PRIMARY KEY,
    loginID INT,
    firstname VARCHAR(50) NOT NULL,
    lastname VARCHAR(50) NOT NULL,
    phone VARCHAR(20),
    FOREIGN KEY (loginID) REFERENCES UserLogin(loginID)
);

-- Rooms table
CREATE TABLE Rooms (
    roomID INT PRIMARY KEY,
    roomName VARCHAR(50)
);

-- Bookings table
CREATE TABLE Bookings (
    bookingID INT PRIMARY KEY,
    roomID INT,
    time TIME,
    FOREIGN KEY (roomID) REFERENCES Rooms(roomID)
);

-- Equipments table
CREATE TABLE Equipments (
    equipmentID INT PRIMARY KEY,
    name VARCHAR(50),
    maintenanceDate DATE
);

-- BillPayment table
CREATE TABLE BillPayment (
    billID INT PRIMARY KEY,
    memberID INT,
    amount DECIMAL(10,2),
    paymentDate DATE,
```

```
    FOREIGN KEY (memberID) REFERENCES Member(memberID)
);

--Time Availability
CREATE TABLE Availability (
    availability_id SERIAL PRIMARY KEY,
    start_time TIME NOT NULL,
    available_date DATE NOT NULL,
    trainer_id INT NOT NULL,
    FOREIGN KEY (trainer_id) REFERENCES Trainer(trainerID)
);
```

DML Statements

```sql
-- Inserting data into the User table
INSERT INTO UserLogin (loginID, username, password)
VALUES
    (1, 'user1', 'password1'),
    (2, 'user2', 'password2'),
    (3, 'user3', 'password3'),
    (4, 'user4', 'password4'),
    (5, 'user5', 'password5'),
    (6, 'trainer', 'trainer'),
    (7, 'user7', 'password7'),
    (8, 'admin', 'admin'),
    (9, 'user9', 'password9');

-- Inserting data into the Member table
INSERT INTO Member (memberID, loginID, firstname, lastname, phone, address, email, gender, age, weight, height)
VALUES
    (1, 1, 'John', 'Doe', '1234567890', '123 Main St', 'john@example.com', 'M', 30, 180.00, 70.00),
    (2, 2, 'Alice', 'Smith', '9876543210', '456 Elm St', 'alice@example.com', 'F', 25, 150.00, 65.00),
    (3, 3, 'Bob', 'Johnson', '5551234567', '789 Oak St', 'bob@example.com', 'M', 35, 160.00, 75.00);

-- Inserting data into the Trainer table
INSERT INTO Trainer (trainerID, loginID, firstname, lastname, phone)
VALUES
    (1, 4, 'Amad', 'Diallo', '1112223333'),
    (2, 5, 'Diogo', 'Dalot', '4445556666'),
    (3, 6, 'Bruno', 'Fernandes', '7778889999');

-- Inserting data into the TrainerMemberAccess table
INSERT INTO TrainerViewMember (trainerID, memberID)
VALUES
    (1, 1),
    (2, 2),
    (3, 3);

-- Inserting data into the Achievements table
```

```sql
INSERT INTO Achievements (achieveID, memberID, achieveDescription)
VALUES
    (1, 1, 'Ran a marathon'),
    (2, 2, 'Lost 20 pounds'),
    (3, 3, 'Completed 100 push-ups challenge');

-- Inserting data into the FitnessGoals table
INSERT INTO FitnessGoals (goalID, memberID, description)
VALUES
    (1, 1, 'Run 5 miles in under 30 minutes'),
    (2, 2, 'Achieve 15% body fat'),
    (3, 3, 'Squat double body weight');

-- Inserting data into the Routines table
INSERT INTO Routines (routineID, memberID, description)
VALUES
    (1, 1, 'Morning jog'),
    (2, 2, 'Weightlifting circuit'),
    (3, 3, 'Yoga practice');

-- Inserting data into the Session table
INSERT INTO Session (sessionID, memberID, trainerID, time, date, status)
VALUES
    (1, 1, 1, '08:00:00', '2024-04-10', 'Scheduled'),
    (2, 2, 2, '10:00:00', '2024-04-12', 'Completed'),
    (3, 3, 3, '15:00:00', '2024-04-15', 'Cancelled');

-- Inserting data into the FitnessClasses table
INSERT INTO FitnessClasses (classID, memberID, trainerID, time, date, status)
VALUES
    (1, 1, 1, '10:00:00', '2024-04-15', 'Scheduled'),
    (2, 2, 2, '12:00:00', '2024-04-17', 'Completed'),
    (3, 3, 3, '17:00:00', '2024-04-20', 'Cancelled');

-- Inserting data into the TrainerSchedule table
INSERT INTO TrainerSchedule (scheduleID, trainerID, time)
VALUES
```

```
    (1, 1, '08:00:00'),
    (2, 2, '10:00:00'),
    (3, 3, '15:00:00');

-- Inserting data into the Admin table
INSERT INTO Admin (adminID, loginID, firstname, lastname, phone)
VALUES
    (1, 7, 'Erik', 'Tenhag', '1234567890'),
    (2, 8, 'Pep', 'Guardiola', '9876543210');

-- Inserting data into the Rooms table
INSERT INTO Rooms (roomID, roomName)
VALUES
    (1, 'Yoga Room'),
    (2, 'Ball Room'),
    (3, 'Hype Room');

-- Inserting data into the Bookings table
INSERT INTO Bookings (bookingID, roomID, time)
VALUES
    (1, 1, '08:00:00'),
    (2, 2, '10:00:00'),
    (3, 3, '15:00:00');

-- Inserting data into the Equipments table
INSERT INTO Equipments (equipmentID, name, maintenanceDate)
VALUES
    (1, 'Treadmill', '2024-04-01'),
    (2, 'Dumbbells', '2024-04-05'),
    (3, 'Yoga Mats', '2024-04-10');

-- Inserting data into the BillPayment table
INSERT INTO BillPayment (billID, memberID, amount, paymentDate)
VALUES
    (1, 1, 50.00, '2024-04-01'),
    (2, 2, 75.00, '2024-04-05'),
    (3, 3, 100.00, '2024-04-10');
```

```
-- Time Availability
INSERT INTO Availability(availability_id, start_time, available_date, trainer_id)
VALUES
    (1, '10:00', '2024-04-23', 1),
    (2, '13:00', '2024-04-23', 1),
    (3, '9:00', '2024-04-26', 2),
    (4, '17:00', '2024-04-25', 3);
```

2.5 Implementation

Implementation file aka python files are includes in the submission folder.

Diagrams can also be viewed in terms of .png or .drawio file.

Link to the demonstration video: https://youtu.be/6C3HTzLdNXY