

COMP 3005 - Project Version 2 - Project Plan

Legend:

Bold: Primary Key

Red: Foreign Key

Blue: Relations made for multi-value Variables

Possible Relations:

Profiles: (**Username**, First_Name, Last_Name, Password, user_type) DONE

Members: (**Username**, Credit_Card, birthday, Height, Weight, Diet_Plan, Goal_Weight, Goal_Speed, Goal_Lift, Weight_Deadline, Speed_Deadline, Lift_Deadline, BMI, systolic_bp, diastolic_bp, Heart_Rate, Blood_Sugar) DONE

Exercise_Routines: (**Members**, Exercise) DONE

Weight_Statistics(**Username**, Weight) DONE

Speed_Statistics(**Username**, Speed) DONE

Lift_Statistics(**Username**, Lift) DONE

Classes: (**Class_ID**, **Trainer_ID**, room_id, Class_Type, Class_Description, Class_Date, Time_Of_Day) DONE

Dates_Trainer_Available: (Schedule_ID, **Trainer_ID**, start_trainer_date, start_time_of_day, end_trainer_date, end_time_of_day) DONE

Dates_Trainer_Unavailable: (**Trainer_ID**, Trainer_Date, Time_Of_Day) DONE

Class_Members: (**Class_ID**, **Username**) DONE

Bookings: (Booking_ID, Room_ID, Class_Date, Time_Of_Day, Event_Info) DONE

Equipment: (**Equipment_id**, Room_ID, Equipment_Status) DONE

Bills: (**Bill_ID**, **Username**, Price, Date_Issued) DONE

Program:

1. **Main()** - start-up
 - Menu function that can take you to:
 - userRegistration()**
 - userLogin()**
 - userLogout()**
 - **Member Menu:**
 - profileInformation()**
 - memberScheduleManagement()**
 - **Trainer Menu:**
 - trainerScheduleManagement()**
 - viewMemberProfile()**
 - **Admin Menu:**
 - roomBookingManagement()**
 - manageEquipment()**
 - manageClassSchedule()**
 - viewBills()**
2. **userRegistration()** - (NONE)
 - Prompt user for username, password, first_name, last_name (if username already taken, try again)
 - Set user_type to TYPE_MEMBER
 - Set bmi, blood_pressure, heart_rate, cholesterol, blood_sugar to random variables
3. **userLogin()** - (NONE)
 - Prompts user for username and password
 - If username and password non-existent, try again
4. **userLogout()** - (**MEMBER**, **TRAINER**, **ADMIN**)
 - Asks user if they're sure to log out
 - If yes, go back to **main()** as user_type NONE
 - If no, go back to **main()** as user_type **MEMBER**, **TRAINER**, or **ADMIN**
5. **profileInformation()** - (**MEMBER**)
 - Menu function that can take you to:
 - personalInformation()**
 - fitnessGoals()**
 - healthMetrics()**
 - healthActions()**
 - healthStatistics()**
6. **personalInformation()** - (**MEMBER**)
 - Displays user's first_name, last_name, credit_card, birthday, height, weight

- If No information found, go to **setPersonalInformation()**
- Asks if the user wants to update information.
If yes, go to **setPersonalInformation()**
If no, go back to **profileInformation()**
7. **setPersonalInformation()** - (MEMBER)
 - Prompt user for first_name, last_name, credit_card, birthday, height, weight
 - After confirming changes, go to **personalInformation()**
 8. **fitnessGoals()** - (MEMBER)
 - Display's user's diet_plan, goal_weight, goal_speed, goal_lift, weight_deadline, speed_deadline, lift_deadline
 - Asks if the user wants to update information.
If yes, go to **setFitnessGoals()**
If no, go back to **profileInformation()**
 9. **setFitnessGoals()** - (MEMBER)
 - Ask user if they want to set values for diet_plan, goal_weight, goal_speed, goal_lift, weight_deadline, speed_deadline, lift_deadline
 - Note: These values don't have to be filled out, so null values are allowed
 - If goal_weight set, then set weight_loss to random value
 - If goal_speed set, then set max_speed to random value
 - If goal_lift set, then set max_lift to random value
 - Grabs max value from one of either weight_loss table,
 - After confirming changes, go to **fitnessGoals()**
 10. **healthMetrics()** - (MEMBER)
 - Displays user's bmi, blood_pressure, heart_rate, blood_sugar
 11. **healthActions()** - (MEMBER)
 - Displays user's diet plan and exercises
 - Asks if the user wants to update information.
If yes, go to **setHealthActions()**
If no, go back to **profileInformation()**
 12. **setHealthActions()** - (MEMBER)
 - Menu function that can take you to:
newRoutine()
changeDietPlan()
 - Prompts user for diet plan and exercises
 - After confirming changes, go to **healthActions()**
 13. **newRoutine()** - (MEMBER)
 - Prompts user for exercise

- Asks if the user wants to add another exercise
If yes, go to **newRoutine()**
If no, go to **healthActions()**
14. **changeDietPlan()** - (MEMBER)
- Prompts user for diet plan
 - After confirming changes, go to **healthActions()**
15. **healthStatistics()** - (MEMBER)
- Displays user health statistics based off of aggregate queries on the Health_Statistics table
 - Truly random
 - **So random statistics or proper statistics on random data? (IMPORTANT)**
 - **Current implementation has max, min, and average on all data (lift, speed, weight) for any given member (based on who's logged in)**
16. **memberScheduleManagement()** - (MEMBER)
- Menu function that can take you to:
personalTraining()
groupTraining()
17. **personalTraining()** - (MEMBER)
- Prompts user for date and time_of_day for session
 - Displays trainers available at that time of day
 - User selects a trainer for the session
 - Entry added to the Dates_Trainer_Unavailable
 - Bill is added to the user for the personal session
 - Asks user if they want to schedule another personal session
If yes, repeat function
If no, go back to **memberScheduleManagement()**
18. **groupTraining()** - (MEMBER)
- Displays classes available
 - User selects a class they wish to participate in
 - Bill is added to the user for the group session
 - Member added to the Class_Members table
 - Asks user if they want to schedule another group session
If yes, repeat function
If no, go back to **memberScheduleManagement()**
19. **trainerScheduleManagement()** - (TRAINER)
- Menu function that can take you to:

addSchedule()
viewSchedule()
changeSchedule()

20. **addSchedule()** - (TRAINER)
 - Asks user to enter start_trainer_date, end_trainer_date, start_time_of_day, end_time_of_day
 - Returns error if schedule conflicts with trainer's existing schedules
21. **viewSchedule()** - (TRAINER)
 - Display's trainer's schedule_id, start_trainer_date, end_trainer_date, start_time_of_day, end_time_of_day
22. **changeSchedule()** - (TRAINER)
 - Prompts user for trainer_id and schedule_id of schedule to be changed
 - Menu function that can take you to:
changeStartDate()
changeEndDate()
changeStartTime()
changeEndTime()
23. **changeStartDate()** - (TRAINER)
 - Asks user for new start date of schedule
24. **changeEndDate()** - (TRAINER)
 - Asks user for new end date of schedule
25. **changeStartTime()** - (TRAINER)
 - Asks user for new start time of schedule
26. **changeEndTime()** - (TRAINER)
 - Asks user for new end time of schedule
27. **viewMemberProfile()** - (TRAINER)
 - Asks for username of member and displays member profile
28. **roomBookingManagement()** - (ADMIN)
 - Menu function that can take you to:
bookRoom()
viewRoomBooking()
cancelRoomBooking()
29. **bookRoom()** - (ADMIN)
 - Asks for Room ID of the room to be booked
 - Returns error if booking already exists for that room
 - Asks for Date, Time_of_day, and Event taking place in the room
30. **viewRoomBookings()** - (ADMIN)
31. **cancelRoomBooking()** - (ADMIN)
 - Asks for Room Id of the room booking to be canceled
 - Returns error if the booking does not exist for that room
 - NOTE: introduce booking_id to the bookings table for easier cancellation
32. **manageEquipment()** - (ADMIN)

- Menu function that takes you
viewEquipments()
updateEquipmentStatus()
- 33. **viewEquipments()** - (ADMIN)
 - Displays all equipments, the room they're in, and their statuses
- 34. **updateEquipmentStatus()** - (ADMIN)
 - Asks for the Equipment_id of the equipment that the admin wants to update the status on
- 35. **manageClassSchedule()** - (ADMIN)
 - Menu function for:
viewClassSchedule()
bookClass()
cancelClass()
- 36. **viewClassSchedule()** - (ADMIN)
 - Display all classes, room_id of the class, class_type, date, and time_of_day
- 37. **bookClass()** - (ADMIN)
 - Prompts user for room_id, Class_Type, Date, Time_Of_Day
 - Return error if room not available for that time of day on that date
 - Return error if class_type invalid
 - Displays trainers available at that time of day
 - User selects a trainer for the session
 - Entry added to the Dates_Trainer_Unavailable
 - Asks user if they want to schedule another class
 - If yes, repeat function
 - If no, go back to **manageClassSchedule()**
- 38. **cancelClass()** - (ADMIN)
 - Ask user for the class_id of the class they want to cancel
 - Return error if the class does not exist
- 39. **viewBills()** - (ADMIN)
 - View all bills and properties of the bills

Documentation:

1. ER Diagram
2. Assumptions made regarding cardinalities and participation types
3. Reduction to Relational Schemas
4. DDL File
5. DML File

6. Video Demonstration

Functionality