CPSC 304 Project Cover Page

Milestone #: 2

Date: October 17, 2023

Group Number: 27

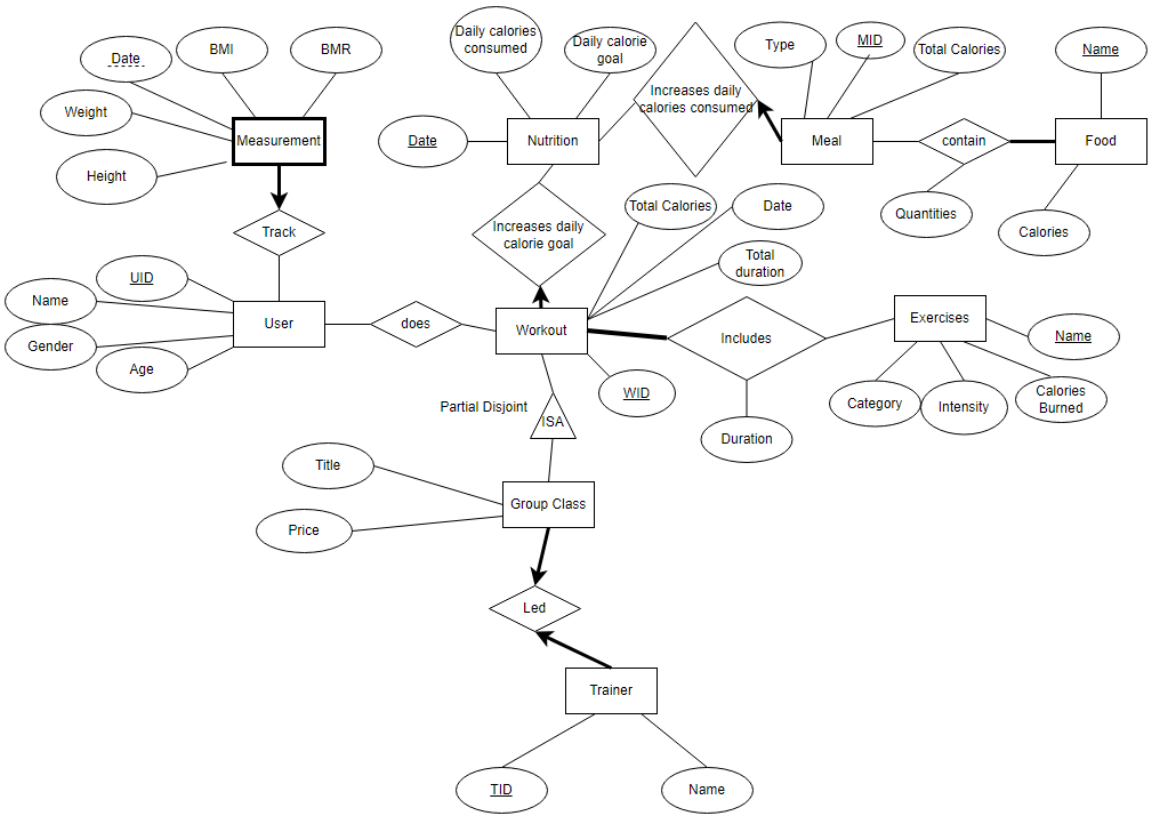
Name	Student Number	CS Alias (Userid)	Preferred Email Address
Clare Pan	95483459	f2l5o	clarepan0@gmail.com
Dizhe Xiang	565742	b5h9t	dizhexiang@gmail.com
Vincent Lee	84258847	i6z1i	vinlee1208@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your email address, and then let us assign you to a TA for your project supervisor.)
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

2. Project Description

a) This application is a user-focused fitness application that allows users to track their workouts, nutrition, and body measurements. The domain of this application is fitness/body management.
b) This project models the tracking of users' body measurements, daily workout and their daily nutrition.

3. ER Diagram



Graph explanation:

- Now the user can have multiple workout in a day since we change the primary key of Workout to WID (workoutID), and Group Class has a "ISA" relationship to Workout, which means Group Class also has its own WID, and a user can either do workout or Group Class (or both) on the same date. Also, for every workout(including Group Class), it can contain multiple exercises(like jogging).
- Eating meals will increase the total amount of calories consumed on the given day while doing workouts will increase the calories necessary to be eaten in order to achieve the daily goal.

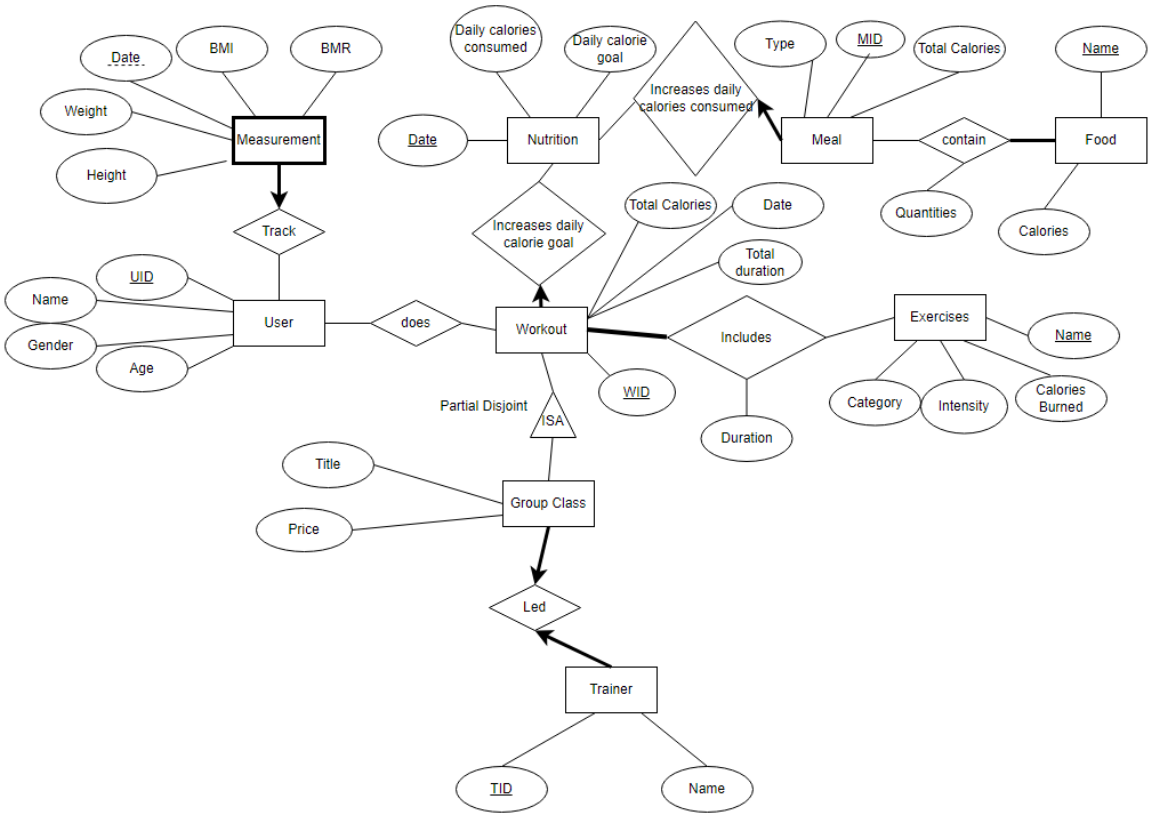
Changes:

- Changed workout & exercises relationship to 'includes' for uniqueness
- Removed relationship between user and class
- Changed class name to group class
- Added 'price' attribute to group class
- Removed 'type' attribute from group class since we can track exercises done during class through workout
- Since workout must partake in exercises, then we can track consumed calories that way. And now since group class in a workout, it also contains exercises, so removed 'consumed calories' attribute from group class
- Changed names of relationship between 'meal', 'nutrition', and 'workout' to better reflect the relationship
- Changed 'goal' attribute of nutrition to 'daily calorie goal' to clarify attribute
- Added a 'Total duration' attribute to workout
- Changed 'coach' entity name to 'trainer'
- Changed Measurement key to be ‘Date’ and User ‘ID’
- Added 'Calories' to entity Meal as non-primary key
- Changed Workout primary key from 'Date' to 'WID'
- Changed Meal primary key to ‘MID’
- Changed Group Class and trainers to many-to-one relationship so one trainer can teach multiple classes

4. Schema derived from the ER diagram

- User(UID, userName UNIQUE NOT NULL, Gender, Age)
    - CK: userName
- TrackMeasurement(**UID**, MDate NOT NULL, Weight, Height, BMI, BMR, **Age**, **Gender**)
    - Weight is in Kg, Height is in centimeter, and BMR is in calories/day
- Workout(**NDate**,WID ,Total Calories Burned, workoutDate NOT NULL, Total Duration NOT NULL)
- GroupClass(**WID**, Title, Price,**TID, TrainerName**)
- Exercises(exerciseName, Category,Calories Burned, Intensity)
    - Calories Burned is in calories/30 mins
- Trainer(TID, TrainerName NOT NULL)
- Nutrition(NDate, Daily Consumed Calories NOT NULL, Daily Calories Goal NOT NULL,**MID**,**WID**)
- Meal(**NDate**, MID, Type, Total Calories NOT NULL)
- Food(FoodName, FoodCalories NOT NULL)
- DoesWorkout(**UID**, **Date**, **WID**) (record workout did by user)
- Contain(**MID, FoodName**, Quantity)
- Includes(**WID**, **ExerciseName**, Duration NOT NULL)
    - Duration in minutes

5. Functional Dependencies



UID -> userName, Gender, Age, WID
userName -> UID, Gender, Age, WID
UID, MDate -> userWeight, userHeight, BMI, BMR, Age, Gender
NDate, WID ->Total Calories Burned, workoutDate, Total Duration
WID -> Title, Price,TID, TrainerName
NDate -> Daily Consumed Calories, Daily Calories Goal
ExerciseName -> Category, Calories Burned, Intensity
TID -> TrainerName
FoodName -> FoodCalories
NDate, MID -> Total Calories, Type
MID, FoodName -> FoodCalories, Quantities
WID, ExerciseName -> Duration

UserWeight, UserHeight -> BMI
UserWeight, UserHeight, Age, Gender -> BMR, BMI
GroupClassTitle -> TrainerName, ClassPrice
GroupClassTitle -> Price, TrainerName, TID
TID -> TrainerName, GroupClassTitle
ExercisesName -> Category
ExerciseIntensity, ExerciseName -> Consumed Calories

6. Normalization

UID -> userName, Gender, Age, WID
CK: userName -> UID, Gender, Age, WID
UID, MDate -> userWeight, userHeight, BMI, BMR, Age, Gender
NDate, WID ->Total Calories Burned, workoutDate, Total Duration
WID -> Title, Price,TID, TrainerName
NDate -> Daily Consumed Calories, Daily Calories Goal
ExerciseName -> Category, Calories Burned, Intensity
TID -> TrainerName
FoodName -> FoodCalories
NDate, MID -> Total Calories, Type
MID, FoodName -> FoodCalories, Quantities
WID, ExerciseName -> Duration

Non-PK/CK FDs:
~~userWeight, userHeight -> BMI~~
userWeight, userHeight, Age, Gender -> BMR
userWeight, userHeight, Age, Gender -> BMI
GroupClassTitle -> TrainerName
GroupClassTitle -> TID
GroupClassTitle -> ClassPrice
~~ExercisesName -> Category~~
~~ExerciseIntensity, ExerciseName -> Calories Burned~~

Removed Redundant Non-PK/CK FDs by minimal cover

We decompose using 3NF:
TrackMeasurement(UID,MDate,UserWeight, UserHeight, Age, Gender, BMR, BMI)
FD userWeight, userHeight, Age, Gender -> BMR violates **3NF**:

- R1(userWeight, userHeight, Age, Gender, BMR), R2(UID, MDate, userWeight, userHeight, Age, Gender, BMI)

FD UserWeight, UserHeight, Age, Gender -> BMI violates R2 BCNF:

- R3(userWeight, userHeight, Age, Gender, BMI), R4(UID, MDate, userWeight, userHeight, Age, Gender)

Since we have UID -> userName, Gender, Age, WID, we can get UID -> Gender, Age, which violates R4

- R5(UID, Age, Gender), R6(UID, MDate, userWeight, userHeight)

GroupClass(Total Calories, Total Duration, WID, WorkoutDate, Title, Price, TID, TrainerName)
FD GroupClassTitle -> TrainerName violates **3NF**:

- R7(GroupClassTitle, TrainerName), R8(GroupClassTitle, TotalDuration, WID, WorkoutDate, Price, TID)

FD GroupClassTitle -> ClassPrice violates R8 for BCNF:

- R9(GroupClassTitle, Price), R10(GroupClassTitle, TotalDuration, WID, WorkoutDate, TID, TrainerName)

FD GroupClassTitle -> TID violates R10 for BCNF:

- R11(GroupClassTitle, TID), R12(GroupClassTitle, TotalDuration, WID, WorkoutDate)

So, instead TrackMeasurement, and Group Class, now we have:
R1: CalculateBMR(userWeight, userHeight, Age, Gender, BMR),
R3: CalculateBMI(userWeight, userHeight, Age, Gender, BMI),
R5: UserInfo(UID, Age, Gender),
R6: UserMeasureRecord(UID, MDate, userWeight, userHeight),
R7: ClassTrainer(GroupClassTitle, TrainerName),
R9: ClassPrice(GroupClassTitle, Price),
R11: ClassTrainerID(GroupClassTitle, TID)
R12: ClassWorkoutInfo(GroupClassTitle, TotalDuration, WID, WorkoutDate)

And other tables keeps the same:
User(UID, userName UNIQUE NOT NULL, Gender, Age)
- CK: userName

TrackMeasurement(**UID**, MDate NOT NULL, Weight, Height, BMI, BMR, **Age**, **Gender**)
- Weight is in Kg, Height is in centimeter, and BMR is in calories/day

Workout(**NDate**,WID ,Total Calories Burned, workoutDate NOT NULL, Total Duration NOT NULL)
GroupClass(**WID**, Title, Price,**TID, TrainerName**)
Exercises(exerciseName, Category,Calories Burned, Intensity)
- Calories Burned is in calories/hour

Trainer(TID, TrainerName NOT NULL)
Nutrition(NDate, Daily Consumed Calories NOT NULL, Daily Calories Goal NOT NULL,**MID**,**WID**)
Meal(**NDate**, MID, Type, Meal Calories Consumed NOT NULL)
Food(Name, FoodCalories NOT NULL, Quantity NOT NULL)
DoesWorkout(**UID**, **Date**, **WID**) (record workout did by user)
Contain(**MID, FoodName**)
Includes(**WID**, **ExerciseName**, Duration)
- Duration is in minutes

7. SQL DDL Statement

- CREATE TABLE User(
  UID VARCHAR(10) PRIMARY KEY
  userName VARCHAR(10) UNIQUE
  NOT NULL
  Gender VARCHAR(8)
  Age INTEGER
)
- CREATE TABLE Workout(
  NDate DATE
  WID VARCHAR(20)
  Total Calories Burned INTEGER DEFAULT 0
  workoutDate DATE NOT NULL
  Total Duration INTEGER NOT NULL
  PRIMARY KEY(NDate, WID)
  FOREIGN KEY(NDate) REFERENCES
  Nutrition(NDate)

```
                ON DELETE CASCADE
    )
-   CREATE TABLE GroupClass(
        WID VARCHAR(10) PRIMARY KEY REFERENCES
            Workout(WID)
            ON DELETE CASCADE
        Title VARCHAR(10),
        Price INTEGER
        TID VARCHAR(10) UNIQUE
        TrainerName VARCHAR(10)
        FOREIGN KEY (TID) REFERENCES
            Trainer(TID)
            ON DELETE CASCADE
    )
-   CREATE TABLE Exercises(
        ExerciseName VARCHAR(10) PRIMARY KEY,
        Category VARCHAR(10),
        Calories Burned INTEGER DEFAULT 0,
        Intensity VARCHAR(10)

    )
-   CREATE TABLE Trainer(
        TID VARCHAR(10) PRIMARY KEY,
        trainerName VARCHAR(10) NOT NULL
    )
-   CREATE TABLE Nutrition(
        NDate DATE PRIMARY KEY,
        Daily Consumed Calories INTEGER NOT NULL,
        Daily Calories Goal INTEGER NOT NULL
        MID VARCHAR(10) UNIQUE
        WID VARCHAR(10)
        FOREIGN KEY (NDate, WID)
            REFERENCES Meal(NDate, MID)
            ON DELETE CASCADE
        FOREIGN KEY (NDate, MID)
            REFERENCES Workout(NDate, WID)
            ON DELETE CASCADE
    )
-   CREATE TABLE Meal(
        NDate DATE
        MID VARCHAR(20)
        Type VARCHAR(10)
        Meal Calories Consumed INTEGER NOT NULL DEFAULT 0
        PRIMARY KEY(NDate,MID)
        FOREIGN KEY(NDate) REFERENCES
            Nutrition(NDate)
            ON DELETE CASCADE
    )
-   CREATE TABLE Food(
```

FoodName VARCHAR(10) PRIMARY KEY
FoodCalories INTEGER NOT NULL DEFAULT 0
)

- CREATE TABLE TrackMeasurement(
  UID VARCHAR(10)
  MDate DATE NOT NULL
  Weight FLOAT
  Height FLOAT
  BMI FLOAT
  BMR INTEGER
  Age INTEGER
  Gender VARCHAR(8)
  PRIMARY KEY(UID, MDate)
  FOREIGN KEY (UID) REFERENCES
  User(UID)
  ON DELETE CASCADE
)
- CREATE TABLE DoesWorkout(
  UID VARCHAR(10)
  WID VARCHAR(10)
  workoutDate DATE
  PRIMARY KEY(UID, WID)
  FOREIGN KEY (sid) REFERENCES
  User(UID)
  ON DELETE CASCADE
  FOREIGN KEY (WID) REFERENCES
  Workout(WID)
  ON DELETE CASCADE
)
- CREATE TABLE Contain(
  MID VARCHAR(10)
  FoodName VARCHAR(10)
  Quantity NOT NULL INTEGER DEFAULT 0
  PRIMARY KEY(MID,FoodName)
  FOREIGN KEY(MID) REFERENCES
  Meal(MID)
  ON DELETE CASCADE
  FOREIGN KEY (FoodName) REFERENCES
  Food(FoodName)
  ON DELETE CASCADE
)
- CREATE TABLE Includes(
  WID VARCHAR(10)
  ExerciseName VARCHAR(10)
  Duration INTEGER
  PRIMARY KEY(WID, ExerciseName, Duration)
  FOREIGN KEY (WID) REFERENCES
  Workout(WID)
  ON DELETE CASCADE

```
        FOREIGN KEY (ExerciseName) REFERENCES
            Exercises(ExerciseName)
            ON DELETE CASCADE
    )
```

8. Insert Statement

```
INSERT
INTO        User(UID, userName, Gender, Age)
VALUES      ('1', 'userA', 'female', 42),
            ('2', 'userB', 'male', 41),
            ('3', 'userC', 'female', 18),
            ('4', 'userD', 'male', 19),
            ('5', 'userE', 'female', 30)

INSERT
INTO        Workout(NDate, WID, Total Calories Burned, workoutDate, Total Duration)
VALUES      ('2022-01-01', '1', 0, '2022-01-01', 0),
            ('2022-05-20', '2', 3005, '2022-05-20', 120),
            ('2022-07-05'', '3', 1928, '2022-07-05', 36),
            ('2022-10-09', '4', 2394, '2022-10-09', 65),
            ('2022-12-31', '5', 230, '2022-12-31', 15),

INSERT
INTO        GroupClass(WID, Title, Price, TID, TrainerName)
VALUES      ('10', 'classA', 20, '35', 'trainerA')
            ('2', 'classB', 30, '12', 'trainerB')
            ('8', 'classC', 65, '2', 'trainerC')
            ('10', 'classD', 12, '33', 'trainerD')
            ('20', 'classE', 25, '8', 'trainerE')


INSERT
INTO        Exercises(exerciseName, Category, Calories Burned, Intensity)
VALUES      ('Jogging, 'Aerobics', 20, '250', 'Low'')
            ('SpinBiking', 'Cardio', 20, '600', 'High')
            ('Tango', 'A', 20, '600', 'High')
            ('Volleyball', 'A', 20, '40', 'Medium')
            ('Swimming', 'A', 20, '350', 'Low')


INSERT
INTO        Trainer(TID, TrainerName NOT NULL)
VALUES      ('1', 'trainerA'),
            ('2', 'trainerB'),
            ('3', 'trainerC'),
            ('4', 'trainerD'),
            ('5', 'trainerE')
```

```
INSERT
INTO        Nutrition(NDate, Daily Consumed Calories, Daily Calories Goal , MID, WID)
VALUES      (‘2022-01-01’, 1800, 2000 , ‘1’, ‘1’)
            (‘2022-01-01’, 1556, 1800 , ‘2’, ‘1’)
            (‘2022-01-01’, 1900, 1600 , ‘3’, ‘1’)
            (‘2022-01-01’, 2500, 2300 , ‘4’, ‘1’)
            (‘2022-01-01’, 2600, 2800 , ‘5’, ‘1’)

INSERT
INTO        Meal(NDate, MID, Type, Meal Calories Consumed)
VALUES      (‘2022-01-01’, ‘1’, ‘Breakfast’, 700),
            (‘2022-01-01’, ‘2’, ‘Snack’, 200),
            (‘2022-01-01’, ‘3’, ‘Lunch’, 900),
            (‘2022-01-01’, ‘4’, ‘Snack, 200),
            (‘2022-01-01’, ‘5’, ‘Dinner’, 600)

INSERT
INTO        Food(Name, FoodCalories, Quantity)
VALUES      (‘Medium Green Apple’, 90, 1)
            (‘Banana’, 89, 2),
            (‘Church’s Chicken Fried Thigh’, 230, 2),
            (‘Dominos Deluxe Pizza Medium Slice’, 220, 4),
            (‘Starbucks Venti Black Coffee’, 5, 1)

INSERT
INTO        TrackMeasurement(UID, MDate, Weight, Height, BMI, BMR)
VALUES      (‘1’, ‘2023-01-01’, 52.0,167.6, 18.5, 1197),
            (‘2’, ‘2023-05-20’, 73.4, 178.2, 23.1,1649),
            (‘3’, ‘2023-07-05’’, 60.5,172.3, 20.4,1431),
            (‘4’, ‘2023-10-09’, 86.0,180.9, 26.3,1906),
            (‘5’, ‘2023-12-31’, 100.0, 200.0, 24.75, 2039)

INSERT
INTO        DoesWorkout(UID, Date, WID)
VALUES      (‘1’, ‘2022-01-01’, ‘1’),
            (‘1’, ‘2022-05-20’, ‘2’),
            (‘2’, ‘2022-07-05, ‘3’),
            (‘3’, ‘2022-10-09’, ‘4’),
            (‘5’, ‘2022-12-31’, ‘5’)

INSERT
INTO        Contain(MID, FoodName)
VALUES      (‘1’, ‘Medium Green Apple’)
            (‘2’, ‘‘Banana’’)
            (‘3’, ‘Church’s Chicken Fried Thigh’)
            (‘4’, ‘Dominos Deluxe Pizza Medium Slice’)
```

('5', 'Starbucks Venti Black Coffee')

```
INSERT
INTO        Includes(WID, ExerciseName, Duration)
VALUES      ('1', 'Jogging, 30)
            ('2', 'SpinBiking', 45)
            ('3', 'Tango', 50)
            ('4', 'Volleyball', 80)
            ('5', 'Swimming', 60)
```