COMP 3005 Final Project Report

ER MODEL LINK
- https://online.visual-paradigm.com/share.jsp?id=333131383230342d38

RELATIONAL DATABASE SCHEMA LINK
- https://online.visual-paradigm.com/share.jsp?id=333135323634362d32

Mapping and Assumption Table
Dotted line represent partial participation
Solid line represents total participation

Representation in ER Model	Extra Information	Assumptions
User relation: A User relation used to store the user_id(PK), username, password and the role-used to differentiate between the roles and responsibilities if the different users	This will mainly be for the sign in page, in which we will use it to delegate certain accessibility features depending on their role	
Member relation: stores member_id (PK), first_name, last_name, gender, email, date_of birth, address, phone_number, start_date, Payment_status.	This keeps track of member information. The payment status is used to keep track of if a member paid the starting fee. Members are the only users allowed to create an account.	
Member_Health relation: Stores member_id (PK), start_weight, current_weight, height, age, fitness_goal(fitness_id, weight_goal, time_goal, diet_goal, form_of_excercise)	This keeps track of individual member goals and current health. Members are allowed to update these, and based on this we can show them their progress based on how far along they are. The form of exercise includes things like cardio, strength training, etc.	
Exercise_Routine relation: Stores Routine_id (PK), Member_id (FK), Routine_name, description, duration, date_created	This allows member to create a personalised workout routine so that they can stay on track for their goals	
Admin Relation: Stores admin_id (PK), first_name, last_name, email, phone_number		
Trainer relation:	Store trainers information. The	

Stores trainer_id (PK), first_name, last_name, email, phone_number, specialisations	specialisations is a multi-valued attribute that will allow us to pair the right trainers with the right members	
Personal_training_session relation: Stores: Session_id (PK), Trainer_id (FK), Member_id (FK), session_date, session_time, duration, Room_id (FK), payment(Price, payment_status), Booking Status.	For personal training sessions, personal trainers go in and put in their availability. Member id is set to a default value null until a member signs up and pays. Admins reserve the right to cancel anything, and update room sessions. Members must pay before they are approved for the session	
Group_fitness_classes relation: Stores Class_id (PK), Trainer_id (FK), Room_id, class_name, description, Session_date, Session_time, Member_ids, price	Similar to the idea of personal training, trainers go in and put the time they are available for classes, plus what they will be teaching, the classes are all free, the are included in the membership fee, Members must sign up and a list of who ha signed up is kept in the member_ids attribute which us multi-valued	
Room_bookings relation: Stores Room_id (PK), Room_name, Room_location, booked(timing, duration, date, session_id/class_id)	For room bookings, all available rooms for 3 days from the time 4pm-9pm with intervals of 1 hour, and the status of their booking. Trainers can pick a room for their classes from the available slots.	
Equipment_maintenance relation: Stores equipment _id (PK), Equipment_name, maintained_date, next_maintenace, performed_by		
Equipment maintenance relationship:	Admin can update, delete and create new equipment and manage the maintenance of the equipment.	

Equipment is maintained by admin, and all equipment must be maintained, but not all admin must maintain equipment. Its and N:M relationship because an employee can maintain many pieces of equipment and a piece of equipment can be maintained by many employees.		
Room management relationship: Admins can manage and oversee room bookings. It's an N:M relationship and total participation on the side of the rooms, because all rooms must be managed but not all admin have to manage them.	Admins can free up a room by cancelling a class, or switch the rooms in which classes are taking place	
Trainer book relationship: Trainers book rooms for their personal and group sessions. It's a 1:N relationship, with total participation on both sides. Only one trainer per room, but a trainer can book multiple rooms.		
Members register relationship: Members register for personal training/group class sessions. Not all members have to register and all classes have to be registered for or will be cancelled. One member can sign up for many personal fitness/group training sessions, but only one person can sign up for a personal training session and many people can sign up for one group fitness class..		
Member create excercise_routine relationship:		

Each member can create many exercise routines and each exercise routine is created by one member, thus, it's a 1:N relationship. Also all exercise must be created be members, but not all members must create an exercise routine, thus a partial participation on the members		
Member update member_health relationship: 1:1 relationship and total participation on both sides, because every member must update their health, and goals and can only do one "profile", which consists of a goal, and all health is updated by one member and must be created by a member.		
Trainer lead group_fitness_class: Every trainer can lead many group fitness classes and every fitness class is led by one trainer, thus a 1:N relationship. Total participation on both sides		
Trainer conduct personal_fitness_classes: Every trainer can conduct many personal fitness classes and every fitness class is led by one trainer, thus a 1:N relationship. Total participation on both sides.		
Admin updates classes: Admin has the power to update the classes as they see fit, which usually just means cancelling, and changing the location for the classes. Not all classes have to		

be updated and not all admins have to update, but only one admin can cancel a class and they can cancel multiple classes		