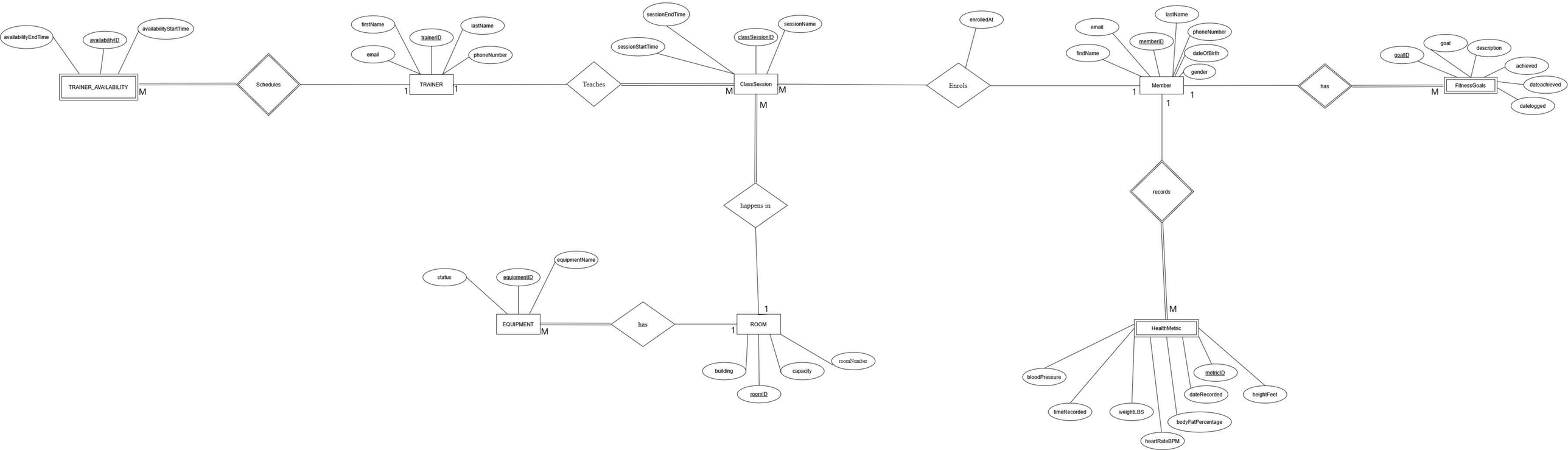
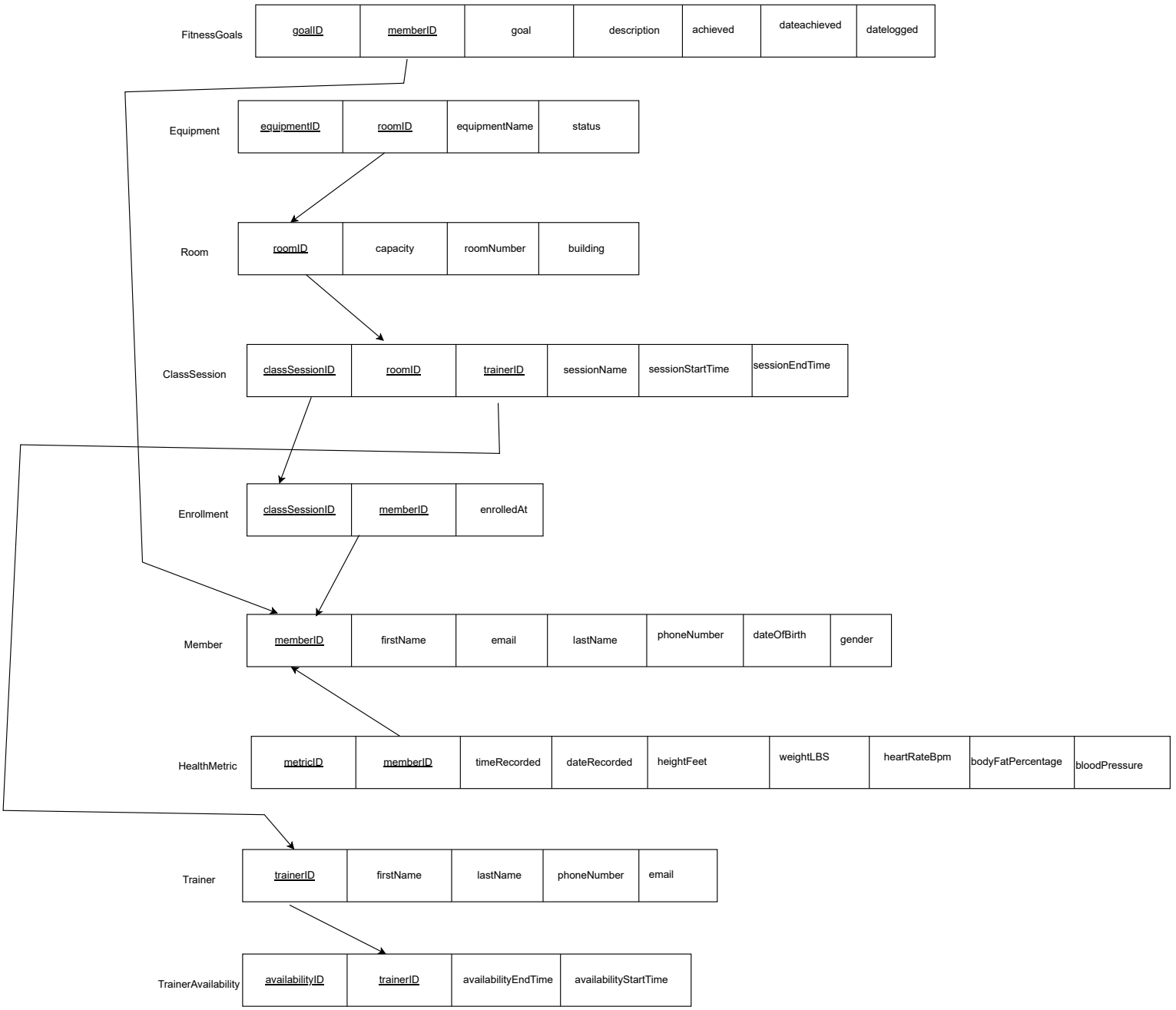
ER Model Diagram



Final Project Database Schema Diagram



Note: The data is already in Third Normal Form (3NF). All tables have an attribute primary key upon which all non-primary attributes fully rely. There are no transitive dependencies, and all columns are atomic.

Mapping Table

Requirement	Assumption/Reasoning	Representation in ER Model / Relational Schema
System retains Member information such as: firstName, lastName, phoneNumber, gender, and dateOfBirth.	Members self-register. Each email is unique.	**Entity: Member** — Attributes: memberID (PK), firstName, lastName, email,phoneNumber,gender, dateOfBirth.
A member can record many health metrics.	Each health metric belongs to exactly one member.	**Entity: HealthMetric** — Attributes: metricID (PK), memberID (FK), dateRecorded, timeRecorded, heightFeet, weightLBS, heartRateBpm, bodyFatPercentage, bloodPressure. **Relationship:** Member 1 to M HealthMetric.

Members create Fitness Goals and update status when completed.	A member has many fitness goals. A goal belongs to one member. Only the member can update goal status.	**Entity: FitnessGoals** — Attributes: goalID (PK), memberID (FK), goal, description, achieved, dateLogged, dateAchieved. **Relationship:** Member 1 to M FitnessGoals.
System stores Trainer information such as firstName, lastName, phoneNumber, email.	Trainers are created by Admin.	**Entity: Trainer** — Attributes: trainerID (PK), firstName, lastName, phoneNumber, email.

Trainers creates availability.	Trainer availability cannot overlap with an availiabiliy they have already set for themselves.	**Entity: TrainerAvailability** — Attributes: availabilityID (PK), trainerID (FK), availabilityStartTime, availabilityEndTime. **Relationship:** Trainer 1 to M TrainerAvailability.
Members can enroll into group classes.	A member cannot enroll twice into the same class. Class capacity is based on room capacity.	**Entity: Enrollment** — Attributes: memberID (FK), classSessionID (FK), enrolledAt. **Relationships:** Member M to M ClassSession via Enrollment.

Rooms are stored in the system containing building name, room number, and capacity.	Room capacity must be > 0.	**Entity: Room** — Attributes: roomID (PK), building, roomNumber, capacity.
Equipment's are in a room.	Equipment can exist on their own but should remain connected to a room.	**Entity: Equipment** — Attributes: equipmentID (PK), roomID (FK), equipmentName, status. **Relationship:** Room 1 to M Equipment.
Group Class Sessions (ClassSession) are created with a trainer, an available room, and a start/end time.	ClassSession cannot overlap another class in the same room or with the same trainer.	**Entity: ClassSession** — Attributes: classSessionID (PK), sessionName, sessionStartTime, sessionEndTime, trainerID (FK), roomID (FK). **Relationships:** Trainer 1 to M ClassSession; Room 1 to M ClassSession; Member M to M ClassSession (via Enrollment).
Admin can create trainers, add rooms, schedule classes, update classes, cancel classes, and manage equipment.	Admin only exists as a role and is not an entity stored in the database. It acts upon the database but does not need to be represented within it. I chose not to include Admin as an entity because entities represent persistent data, and the admin role does not have attributes that necessarily connect to other entities.	