COMP3005
Final Project
Sam Drennan
101185231

Conceptual Design

Requirements:

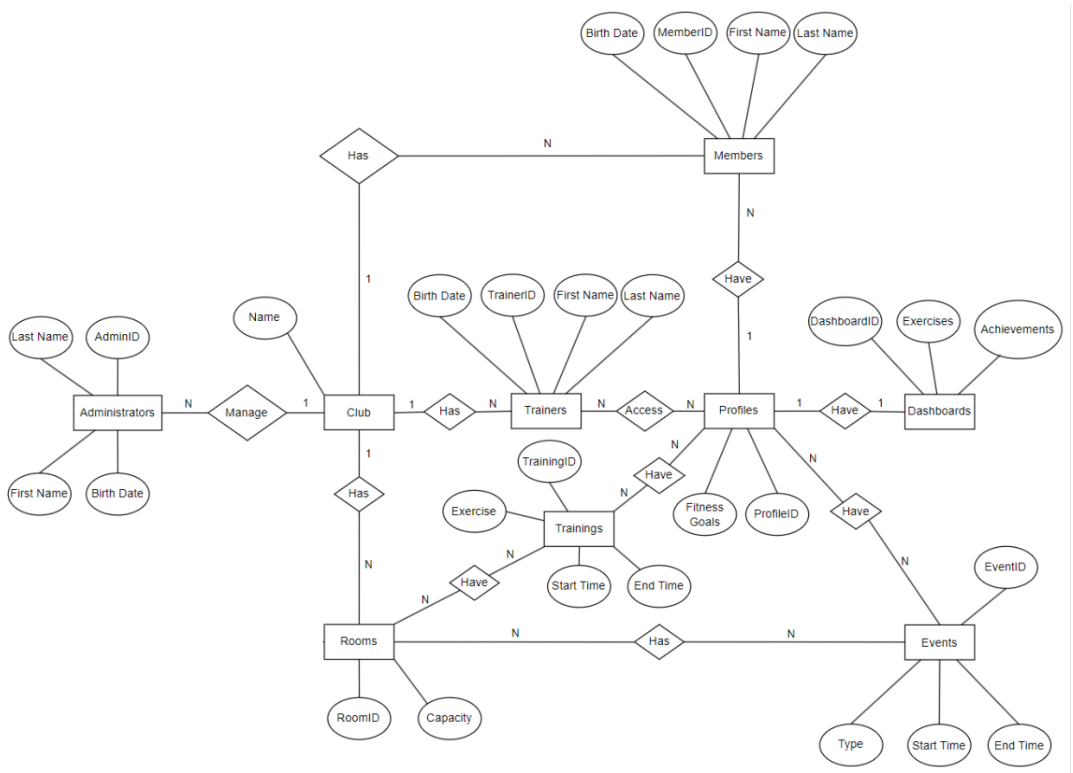
- Have members, trainers and admins
- Be able to create events and personal trainings for specific rooms
- Have exercise tracking and goals

Tables Needed:
Club
Members
Trainers
Administrators
Profiles
Dashboards
Trainings
Events
Rooms

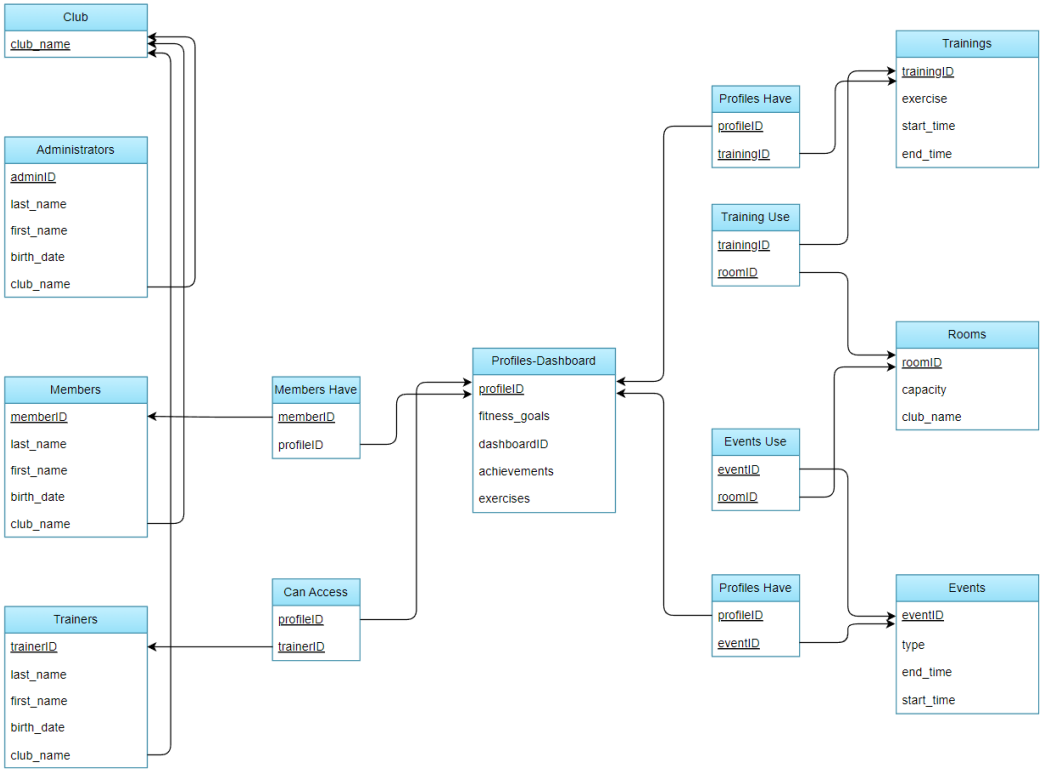
Assumptions Regarding Cardinality and Participation types:

The relationship between training and rooms assumes there is availability checking to make sure the room is available at the time associated with the training. The same goes for the relationship between events and rooms.

This design also assumes that the database will be used for only one Club.



Reduction to Relational Schemas



Normalization of Relational Schemas

Dependencies:

2NF:

Administrators:
- adminID → {last_name, first_name, birth_date, club_name}

Members:
- memberID → {last_name, first_name, birth_date, club_name}

Trainers:
- trainerID→ {last_name, first_name, birth_date, club_name}

Trainings:
- trainingID → {exercise, start_time, end_time}

Rooms:
- roomID → {capacity, club_name}

Events:
- eventID→ {type, start_time, end_time}

Profiles-Dashboard:
- profileID → {fitness_goals, dashboardID, achievements, exercises}

Since no part of X which is only adminID can be removed to cause the dependency to not hold all of the dependencies in Administrators are full functional dependencies meaning it is in 2NF.

This is true for all relations that only have 1 primary key therefore the rest of the relations in this schema are also in 2NF.

3NF:

Checking for 3NF there are no issues with the other relations however there may be an issue with the Profiles-Dashboard relation since:

Administrators:
- profileID → {achievements, exercises}
- profileID → dashboardID
- dashboardID → {achievements, exercises}

However because I know that the relationship between the dashboards and the profiles is a one to one relationship meaning this example cannot exist:

profileID	fitness_goals	dashboardID	achievements	exercises
1	“Lose 10 pounds”	1	“Lost 3 pounds”	“Push-ups”
2	“Lose 5 pounds”	1	“Lost 3 pounds”	“Push-ups”

This can be fixed with decomposition however it would still cause redundancy since profileID will always equal dashboardID.

Therefore the best fix would be to just get rid of the dashboardID.

Database Schema Diagram

Normalized Relation Diagram: