COMP3005-A Database Management Systems

Final Project Report

Fomenko Natalia, 100909652

December 1, 2025

Conceptual Design

Entity Sets:

Entity set	Type
ADMIN	Regular entity
MEMBER	Regular entity
TRAINER	Regular entity
TRAINER_AVAIL	Regular entity
ROOM	Regular entity
SESSION	Regular entity
HEALTH_METRIC	Regular entity
FITNESS_GOAL	Regular entity
BILL	Regular entity

Regular (strong) entities:

- Independent: Can be fully identified by its own attributes
- Representation: In an Entity-Relationship Diagram (ERD), they are represented as a single rectangle

Weak entities:

- Dependent: Cannot be fully identified by its own attributes
- Representation: In an Entity-Relationship Diagram (ERD), they are represented as a double rectangle

See next page

Relationship Sets:

Relationship Set	Cardinality	Participation type
creates_trainer_profile	1:N ADMIN: Each admin can create many trainer profiles TRAINER: Each trainer profile is created by one admin	ADMIN: Partial participation (some admins do not create any trainer profiles) TRAINER: Total participation (each trainer profile is created by some admin)
creates_bill	1:N ADMIN: Each admin can create many bills BILL: Each bill is created by one admin	ADMIN: Partial participation (some admins do not create a bill) BILL: Total participation (each bill is created by some admin)
indicates_avail	1:N TRAINER: Each trainer can indicate many availabilities TRAINER_AVAIL: Each trainer availability belongs to one trainer	TRAINER: Partial participation (some trainers do not indicate trainer availability) TRAINER_AVAIL: Total participation (each trainer availability is indicated by some trainer)
teaches	1:N TRAINER: Each trainer can teach many sessions SESSION: Each session has only one trainer	SESSION: Total participation (each session has a trainer) TRAINER: Partial participation (some trainers do not teach any sessions)
schedules	1:N ADMIN: Each admin can schedule many sessions SESSION: Each session is scheduled by one admin	ADMIN: Partial participation (some admins do not schedule sessions) SESSION: Total participation (each session is scheduled by an admin)
has_room	1:N ROOM: Each room can be used for many sessions SESSION: A session happens in one specific room	ROOM: Partial participation (some rooms are not booked) SESSION: Total participation (each session has a room as a location)

registers	N:M MEMBER: Each member can register in many sessions SESSION: Each session can have many members	MEMBER: Partial participation (some members are not registered in a session) SESSION: Partial participation (some sessions do have any members registered)
has_goals	1:N MEMBER: Each member can have many goals FITNESS_GOAL: Each fitness goal belongs to one specific member	MEMBER: Partial participation (some members might not have any fitness goals) FITNESS_GOAL: Total participation (each fitness goal belongs to a member)
has_records	1:N MEMBER: Each member can have many health records HEALTH_METRIC: Each health metric belongs to one specific member	MEMBER: Partial participation (some members might not have any health records) HEALTH_METRIC: Total participation (each health metric belongs to some member)

Participation constraints (Total/Partial)

Total participation:

- Every instance of an entity must participate in the relationship
- Represented by a double line
- Foreign key in the related table cannot be NULL

Partial participation:

- Some instances of an entity may or may not participate in the relationship
- Represented by a single line
- Foreign key in the related table can be NULL

Attributes:

Attribute	Type
admin_id	Simple, single-valued, stored, PK
password	Simple, single-valued, stored
admin_name	Simple, single-valued, stored
email_id	Simple, single-valued, stored, PK
password	Simple, single-valued, stored
name	Simple, single-valued, stored
is_active	Simple, single-valued, stored
date_of_birth	Simple, single-valued, stored
trainer_id	Simple, single-valued, stored, PK
password	Simple, single-valued, stored
trainer_name	Simple, single-valued, stored
tr_specialization	Simple, single-valued, stored
rate_per_hour	Simple, single-valued, stored
admin_id	Simple, single-valued, stored, FK
avail_id	Simple, single-valued, stored, PK
trainer_id	Simple, single-valued, stored, FK
start_timestamp	Simple, single-valued, stored
end_timestamp	Simple, single-valued, stored
room_id	Simple, single-valued, stored, PK
capacity	Simple, single-valued, stored
session_id	Simple, single-valued, stored, PK
trainer_id	Simple, single-valued, stored, FK
room_id	Simple, single-valued, stored, FK
capacity	Simple, single-valued, stored
num_participants	Simple, single-valued, stored
start_timestamp	Simple, single-valued, stored
end_timestamp	Simple, single-valued, stored
admin_id	Simple, single-valued, stored, FK
hm_id	Simple, single-valued, stored, PK
email_id	Simple, single-valued, stored, FK
hm_timestamp	Simple, single-valued, stored,
hm_type	Simple, single-valued, stored
hm_measurement	Simple, single-valued, stored
fg_id	Simple, single-valued, stored, PK
email_id	Simple, single-valued, stored, FK
fg_timestamp	Simple, single-valued, stored,
fg_type	Simple, single-valued, stored
fg_target	Simple, single-valued, stored

Attribute types (simple vs composite, single vs multi-valued, simple vs derived)

Simple vs composite attributes

- Simple attributes cannot be further divided into smaller components
- Composite attributes can be broken down into smaller sub-attributes

Single-valued vs multi-valued attributes

- Single-valued attributes hold only one value for a particular entity instance
- Multi-valued attributes can hold multiple values for a single entity instance

Stored vs derived attributes

- Stored attributes: their values that are directly present in the database and do not change unless explicitly updated
- Derived attributes: their values that are not directly stored but are calculated or derived from other attributes

Primary Key (PK) and Foreign Key (FK)

- Primary Key: an attribute (or a combination of attributes) that uniquely identify each instance of an entity
- Foreign Key: an attribute (or a combination of attributes) that refers to another entity or table
- An attribute in a table may be neither PK nor FK

See next page

Reduction to Relation Schema (Based on the ER model)

Step 1 - Regular Entities:

Entity set	Type	Mapping
ADMIN	Regular entity	New table
MEMBER	Regular entity	New table
TRAINER	Regular entity	New table
TRAINER_AVAIL	Regular entity	New table
ROOM	Regular entity	New table
SESSION	Regular entity	New table
HEALTH_METRIC	Regular entity	New table
FITNESS_GOAL	Regular entity	New table
BILL	Regular entity	New table

Step 2 - Weak Entities:

There are no weak entities.

Step 3 - 1:1 Relationships:

There are no 1:1 relationships.

Step 4 - 1:N Relationships

Relationship Set	Mapping
creates_trainer_profile	I chose the FK approach. I added the PK of the ADMIN relation as a FK to the TRAINER relation (into the N-side).
bills_ a_member	I chose the FK approach. I added the PK of the ADMIN relation as a FK to the BILL relation (into the N-side).
indicates_avail	I chose the FK approach. I added the PK of the TRAINER relation as a FK to the TRAINER_AVAIL relation (into the N-side).
teaches	I chose the FK approach. I added the PK of the TRAINER relation as a FK to the SESSION relation (into the N-side).
schedules	I chose the FK approach. I added the PK of the ADMIN relation as a FK to the SESSION relation (into the N-side).
has_room	I chose the FK approach. I added the PK of the ROOM relation a FK to the SESSION relation (into the N-side).

has_goals	I chose the FK approach. I added the PK of the MEMBER relation as a FK to the FITNESS_GOAL relation (into the N-side).
has_records	I chose the FK approach. I added the PK of the MEMBER relation as a FK to the HEALTH_METRIC relation (into the N-side).

Step 5 - N:M Relationships:

Relationship Set	Mapping
registers	I had to create a new relation (no choice). The new table (registers) includes the PKs of both MEMBER entity and the SESSION entity as FKs. The combination of these FKs forms the PK of the new relation.

Step 6 - Multi-valued attribute:

There are no multi-valued attributes.

See next page

Normalization

1NF

There are no multivalued attributes. There are no composite attributes. Therefore, the schema satisfies 1NF.

2NF

In each relation, except REGISTERS, PK has only one attribute. Therefore, it is not possible to remove an attribute from the PK. Furthermore, the PK of the REGISTERS relation must maintain both attributes to be able to identify both the MEMBER and the SESSION. In each relation, it is not possible to remove one of the attributes from the PK. In each relation, every nonprime attribute is fully functionally dependent on the PK. Therefore, there are no partial dependencies. Therefore, the relation schema satisfies 2NF.

3NF

In each relation, there are no relationships between a nonprime attribute and another nonprime attribute. In each relation, there are no transitive dependencies. Therefore, the relation schema satisfies 3NF.

See next page

DDL (or ORM Equivalent)

DDL file submitted

Views

View	Functionality	Comment
upcoming_sessions	Member	included in DDL
assigned_sessions	Trainer	included in DDL
member_lookup	Trainer	This view is not included in the DDL file but rather included in Java program. This view requires user input for member's name during execution.

Indexes

Index	Functionality	Comment
index_name	Trainer	When trainer looks up a member by name, this index facilitates search by member name.

Triggers

Trigger	Functionality	Comment
remove_trainer_avail	Admin	When a new row is added to the SESSION table, the corresponding trainer availability is removed from the TRAINER_AVAIL table.
increase_num_participants	Member	When a new row is added to the REGISTERS table, the number of participants for the corresponding session is increased in the SESSION table.

DML (or ORM Seeding/Operations)

DML file submitted

Code Implementation

Category	Operation implemented	Java Functions
2.1 Member Functions	User Registration Profile Management Health History Dashboard Session Registration (PT/Group)	add_new_member() update_member_profile() add_fitness_goal() add_health_metric() view_upcoming_sessions() member_session_registration()
2.2 Trainer Functions	Set Availability Schedule View Member Lookup	add_availability() view_assigned_sessions() member_lookup()
2.3 Administrative Staff Functions	Class Management Billing & Payment	add_session() update_session() generate_bill()

YouTube Video Link:

https://youtu.be/1XgYQs3gqBE