**FGMS: Fitness Gym Management System**

**A red and blue logo

Description automatically generated**

In Partial Fulfilment of the Requirements of the Course:

CC105 – Information Management

Federio, Rodgie A.

**EBRAHIM E. DIANGCA**

Course Instructor

**May 2025**

**I. Problem Statement**

Many fitness gyms today still use manual methods or old systems to manage their members. This can make things slow and confusing, like when signing up new members, checking their details, or updating their records. It’s also hard to know if a member is still active or when their membership ends. Without a proper system, gym staff may struggle to find the right information quickly. This causes delays and can be frustrating for both the workers and the members.

Another common issue is assigning members to trainers or fitness programs. Most of the time, this is done by hand, which can lead to confusion or unfair scheduling. Some trainers may get too many members while others don’t have enough. Also, it’s not easy to track which member is assigned to which trainer or if the member is even attending their sessions. A better system could help with tracking, balancing trainer schedules, and making sure members are following their fitness plans.

Gyms also find it hard to handle payments and keep track of money properly. If done by hand, payments can be forgotten, recorded wrong, or missed completely. Owners and staff often don’t have a clear view of how much money is coming in or how many members are paying on time. Without this, it's harder to plan and improve the business. A good system should help monitor payments, send reminders, and show useful statistics like monthly income or how many active members there are.

**II. Logical Design**

**NORMALIZATION**

**1NF table: members( All fields are atomic)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Member\_id** | **First\_name** | **Last\_name** | **Gender** | **Birthdate** | **phone** | **Assigned trainer** |
| **1** | **Jirod** | **Burod** | **male** | **2004-11-10** | **09008297939** | **1111** |

|  |
| --- |
| **Status** |
| **Active** |

**2NF table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Member\_id** | **First\_name** | **Last\_name** | **Gender** | **Birthdate** | **phone** | **Trainer\_id** |
| **1** | **Jirod** | **Burod** | **male** | **2004-11-10** | **09008297939** | **1111** |

|  |
| --- |
| **Status** |
| **Active** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **trainer\_id** | **First\_name** | **Last\_name** | **Specialization** | **phone** | **Email** | **Hire\_date** |
| **1111** | **Jirod** | **Burod** | **Muscular** | **09008297939** | **shaur@gmail.com** | **2022-11-22** |

|  |
| --- |
| **Status** |
| **Active** |

**Assign tariaenr**

|  |  |  |
| --- | --- | --- |
| **Member\_id** | **Name** | **Trainer\_id** |
| **1** | **Jirod Burod** | **1111** |
|  |  |  |

**Payment stats**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Payment\_id** | **Member\_id** | **Payment\_date** | **amount** | **method** | **paymentFor** |
| **11** | **1** | **2025-10-19** | **12000** | **Cash** | **VIP** |
|  |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Payment\_id** | **Member\_id** | **Names** | **Amount** |
| **11** | **1** | **Jirod Burod** | **12000.00** |

**REQUIREMENT ANALYSIS**

Many fitness gyms still have a hard time managing their daily tasks because they use manual methods or old systems. This makes it difficult to keep member records, assign trainers properly, and handle payments correctly. Without a good system, signing up new members, updating their details, and checking their membership status can be slow and confusing. It also becomes hard to divide members fairly among trainers and to avoid scheduling problems. Payments can be missed or recorded wrong, making it hard for gym owners to know how much money is coming in. Based on the normalization process, we will look deeper into the system needs by identifying the main parts of the system like Members, Trainers, and Payments, along with their details and how they are connected.

**ENTITIES**

1. Members: People who get added at the gym info. Includes their name, gender, birthday, contact, membership type, and if they are active.
2. Trainers: Gym staff who help members. Includes their name, contact, skill, hire date, and status.
3. Payments: Records of member payments. Includes date, amount, how they paid, and what it’s for.
4. Trainer Assignments: Shows which trainer is helping which member. Helps keep track of trainer-member pairs.

**ATTRIBUTES**

1. Members

* member\_id: Unique ID for each member
* first\_name: Member’s first name
* last\_name: Member’s last name
* gender: Male or Female
* birth\_date: Member’s birthdate
* phone: Contact number
* email: Email address
* address: Home address
* join\_date: Date the member joined the gym
* membership\_type: Type of membership (e.g., Regular, VIP)
* status: Active or Inactive
* trainer\_id: Trainer assigned to the member

2. Trainers

* trainer\_id: Unique ID for each trainer
* first\_name: Trainer’s first name
* last\_name: Trainer’s last name
* specialization: Area of expertise (e.g., strength, cardio)
* phone: Contact number
* email: Email address
* hire\_date: Date the trainer was hired
* status: Active or Inactive

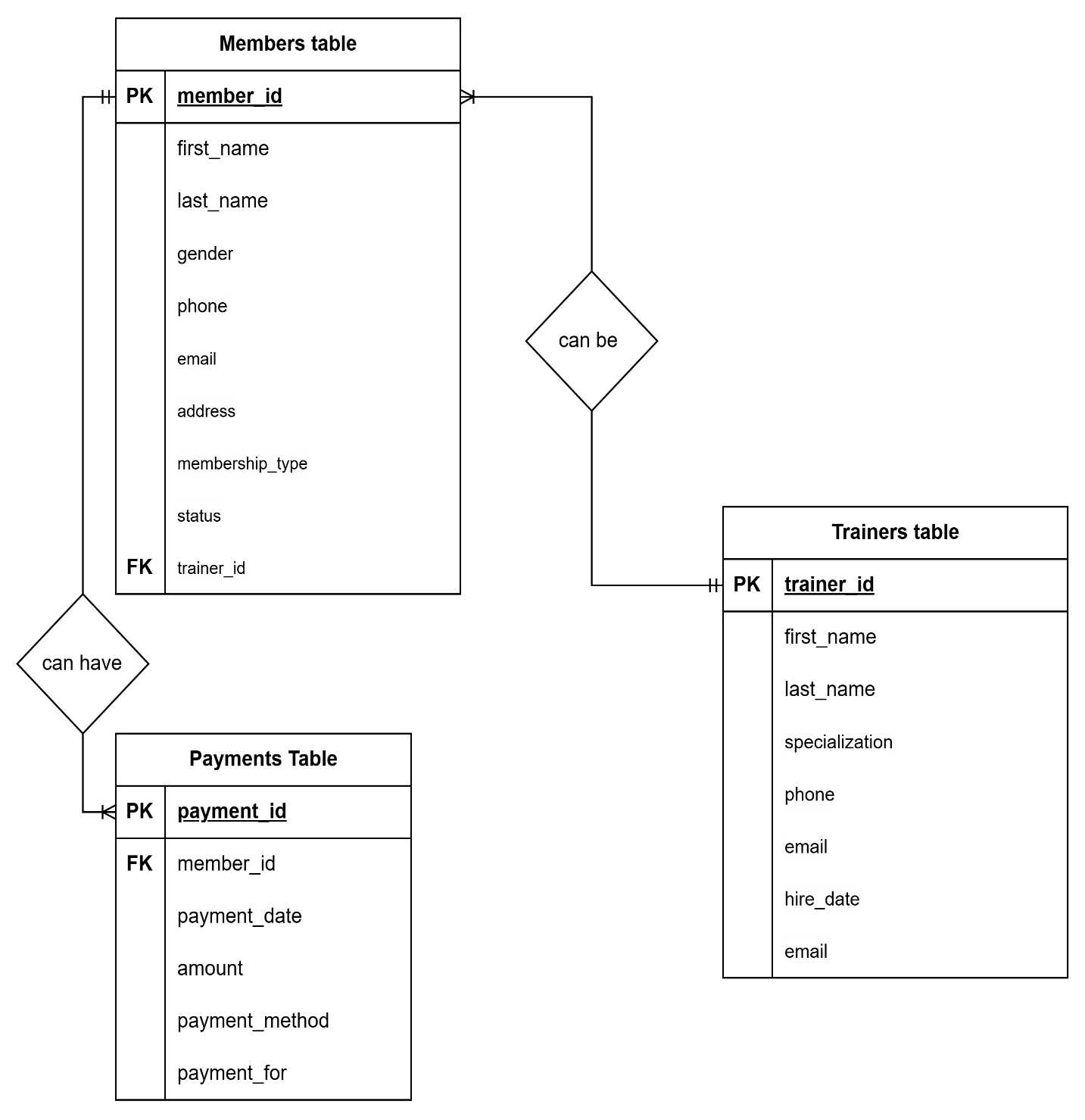
3. Payments

* payment\_id: Unique ID for each payment
* member\_id: Member who made the payment
* payment\_date: Date of payment
* amount: Amount paid
* payment\_method: How the payment was made (e.g., Cash, Card)
* payment\_for: What the payment is for (e.g., VIP membership)

**RELATIONSHIP & CARDINALITY**

* Members-Trainers:  
  *Many-to-One:* Each member is assigned to one trainer, but one trainer can have many members.
* Members-Payment:  
  *One-to-Many:* A member can have multiple payment records, but each payment belongs to only one member.

**ERD DIAGRAM**

****

**RELATIONAL ALGEBRA**

Projection - Find active members

Selection - Show payment amounts and methods

Join - Payment details with member names

**III. Physical Design**

**Create Table**

CREATE TABLE members (

member\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

gender VARCHAR(10),

birth\_date DATE,

phone VARCHAR(15),

email VARCHAR(100),

address TEXT,

join\_date DATE,

membership\_type VARCHAR(50),

status VARCHAR(20) DEFAULT 'Active',

trainer\_id INT,

FOREIGN KEY (trainer\_id) REFERENCES trainers(trainer\_id)

);

CREATE TABLE trainers (

trainer\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

specialization VARCHAR(100),

phone VARCHAR(15),

email VARCHAR(100),

hire\_date DATE,

status VARCHAR(20) DEFAULT 'Active'

);

CREATE TABLE payments (

payment\_id INT(11) NOT NULL AUTO\_INCREMENT,

member\_id INT(11) NOT NULL,

payment\_date DATE NOT NULL,

amount DECIMAL(10,2) NOT NULL CHECK (amount > 0),

payment\_method ENUM('Cash', 'GCash', 'Credit Card', 'Bank Transfer') NOT NULL,

payment\_for VARCHAR(100) NOT NULL,

PRIMARY KEY (payment\_id),

FOREIGN KEY (member\_id) REFERENCES members(member\_id)

) ;

**INSERT INTO MEMBERS**

INSERT INTO members (first\_name, last\_name, gender, birth\_date, phone, email, address, join\_date, membership\_type, status, trainer\_id) VALUES

('Maria', 'Clara', 'Female', '1995-08-22', '09182345678', 'maria.clara@email.com', '456 Oak Ave, Quezon City', '2024-02-15', 'VIP', 'Active', 12),

('Pedro', 'Penduko', 'Male', '1988-11-30', '09193456789', 'pedro.p@email.com', '789 Pine Rd, Cebu', '2024-03-05', 'Basic', 'Active', NULL),

('Sofia', 'Reyes', 'Female', '1992-04-18', '09204567890', 'sofia.r@email.com', '321 Elm St, Davao', '2024-01-25', 'Premium', 'Active', 11),

('Miguel', 'Santos', 'Male', '1985-07-12', '09215678901', 'miguel.s@email.com', '654 Maple Dr, Makati', '2023-12-01', 'VIP', 'Inactive', 12),

('Isabel', 'Chua', 'Female', '1996-12-03', '09260123456', 'isabel.c@email.com', '369 Palm Ct, Alabang', '2024-01-05', 'Premium', 'Active', 11);

**INSERT INTO PAYMENT**

INSERT INTO payments (member\_id, payment\_date, amount, payment\_method, payment\_for) VALUES

(16, '2024-05-01', 1500.00, 'GCash', 'Monthly Membership Fee'),

(17, '2024-05-02', 2500.00, 'Credit Card', 'Premium Membership Fee'),

(19, '2024-05-03', 1000.00, 'Cash', 'Basic Membership Fee'),

(16, '2024-05-05', 500.00, 'PayMaya', 'Personal Training Session'),

(17, '2024-05-10', 800.00, 'Bank Transfer', 'Locker Rental'),

(19, '2024-05-12', 1200.00, 'GCash', 'Monthly Membership Fee');

**INSERT INTO TRAINERS**

INSERT INTO trainers (first\_name, last\_name, specialization, phone, email, hire\_date, status) VALUES

('Michael', 'Johnson', 'Weight Training, Bodybuilding', '09171234567', 'michael.j@fitgym.com', '2022-01-15', 'Active'),

('Sarah', 'Gonzales', 'Yoga, Pilates', '09182345678', 'sarah.g@fitgym.com', '2022-03-10', 'Active'),

('David', 'Santos', 'Boxing, MMA', '09193456789', 'david.s@fitgym.com', '2022-05-22', 'Active'),

('Jennifer', 'Lim', 'Zumba, Aerobics', '09204567890', 'jennifer.l@fitgym.com', '2022-07-18', 'Active'),

('Robert', 'Tan', 'CrossFit, Functional Training', '09215678901', 'robert.t@fitgym.com', '2022-09-05', 'Active'),

('Elizabeth', 'Reyes', 'Postnatal Fitness, Senior Fitness', '09226789012', 'elizabeth.r@fitgym.com', '2023-01-12', 'Active'),

('Christopher', 'Chen', 'Swimming, Aqua Fitness', '09237890123', 'chris.c@fitgym.com', '2023-04-20', 'On Leave');

**UPDATE MEMBERS**

UPDATE members

SET

first\_name = 'Maria',

last\_name = 'Clara',

gender = 'Female',

birth\_date = '1995-08-22',

phone = '09182345678',

email = 'maria.clara@email.com',

address = '456 Oak Ave, Quezon City',

join\_date = '2024-02-15',

membership\_type = 'VIP',

status = 'Active'

WHERE member\_id = 1;

**UPDATE TRAINERS**

UPDATE TRAINERS

SET

first\_name = 'Michael',

last\_name = 'Johnson',

specialization = 'Weight Training, Bodybuilding',

phone = '09171234567',

email = 'michael.j@fitgym.com',

hire\_date = '2022-01-15',

status = 'Active'

WHERE

trainer\_id = 1;

**DELETE MEMBERS**

DELETE FROM members WHERE member\_id = 2

**DELETE TRAINERS**

DELETE FROM trainers WHERE trainer\_id = 4

**DELETE PAYMENTS**

DELETE FROM payments WHERE payment\_id = 8

**will return the list of members sorted by their last names in ascending order**

SELECT member\_id, first\_name, last\_name

FROM members

ORDER BY last\_name;

**will return all active trainers sorted by their last names**

SELECT trainer\_id, first\_name, last\_name

FROM trainers

WHERE status = 'Active'

ORDER BY last\_name;

**IV. INFORMATION MANAGEMENT SYSTEM**

**Features:**

* 1. **Member Management**
* Add, edit, delete, and view member records.
* Track personal information, membership type, join date, and status (Active/Inactive).
* Assign trainers to members.
  1. **Trainer Management**
* Add, edit, delete, and view trainer records.
* Manage trainer specializations, hire date, and current employment status.
  1. **Trainer Assignment**
* Assign trainers to multiple members.
* Prevent unbalanced workload by tracking member-trainer assignments.
  1. **Payment Management**
* Record payment transactions with date, method, and purpose (e.g., VIP Membership).
* View payment history of each member.
  1. **Database Normalization**
* Ensures efficient storage and retrieval with 1NF and 2NF applied.
* Data is broken into logical entities: members, trainers, payments.

**Objectives**

1. **Make Gym Work Easier**
   * Replace manual records with a system that is faster and more organized.
2. **Keep Records Safe and Complete**
   * Store member and trainer info properly.
3. **Track Payments**
   * Help gym staff know who paid and when.
4. **Help with Decisions**
   * Show useful info like number of active members or total payments.

**Scope and Limition;**

**Scope:**

The system can register members, manage trainer records and assignments, and record payments using a MySQL database. It uses relational design with foreign key connections for organized data.

**Limitations:**

The system works well for its purpose but has no visual interface (SQL-only). It doesn’t support search or filtering, is limited to a single gym location, and lacks features like payment reminders and class scheduling.