NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

MILITARY COLLEGE OF SIGNALS

(CAMPUS OF NUST)

CS-220 DATABASE SYSTEM

PROJECT REPORT

COURSE: BESE – 28

SECTION: C

SUBMITTED BY:

1 - Nimrah
2 - Kashaf
3 - Amal
4 - Aiffa

SUBMITTED TO:

Dr. Ayesha Naseer

REPORT TO BE SUBMITTED ON :

FATMAN GYM MANAGEMNET SYSTEM

100% original work

FATMAN GYM

Only for the fat peeps!!

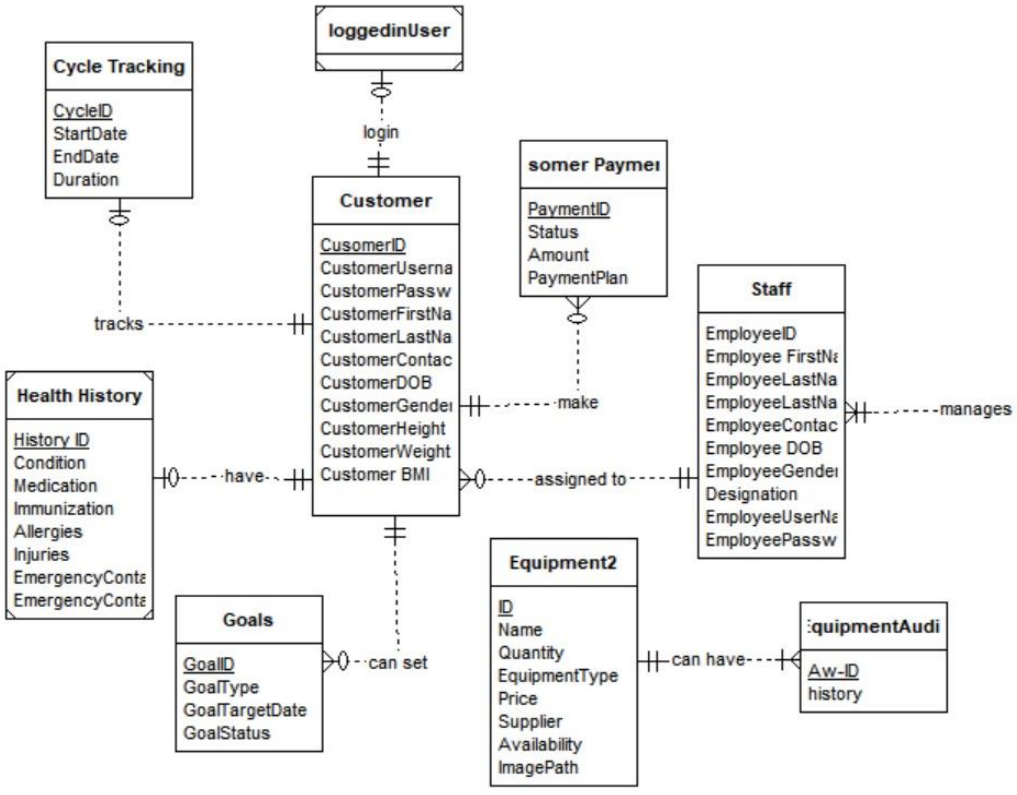
Introduction:

Fatman Gym Management System - a comprehensive solution for managing all aspects of your fitness center. Our innovative platform streamlines gym operations and enhances member experiences by offering a suite of tools tailored to the fitness industry. From membership management to equipment maintenance to personal trainer coordination, our system is designed to optimize efficiency and elevate your gym's performance. Fatman Gym Management System will revolutionize the way gym business is managed online.

Software used:

- VS Code
- SQL Server Management Studio 19
- SQL Server Express 19
- Azure Database
- ER Assistant

ERD:



Relational Schema:

The detailed relational schema of these tables is also given with their attributes and primary keys mentioned in it to clear any ambiguity and to make it feasible for the database designer to draw the ERD. The relational schema is given:

DB Tables:

a. **Staff:** Stores the information of staff. Staff ID is the unique attribute given to it as the primary key. Other attributes are designation, date of birth and contact.
b. **Customers:** This table stores the customer information. The primary key is the customer id which is unique for every customer. Other than that date of birth, gender, contact, username, and password are some of the attributes.
c. **Cycle Tracking:** Cycle id is the primary key of this table. The start date, end date and duration are the attributes.
d. **Health History:** History id is the unique attribute, so it is the primary key. Medication, allergies, injuries, emergency contact number are the others attribute.
e. **Customer Payments:** Payment id is the primary key. Other attributes are payment plans.
f. **Customer Goals:** Goal id being the unique attribute is the primary key. Other attributes are goal start date, end date and duration.
g. **Equipment:** Equipment id being the unique attribute gains the status of primary key. Other attributes are equipment name, equipment price, equipment seller etc.
h. **Equipment audit:** Audit id is deemed to be the primary key. It uses the equipment id as the foreign key and tells us when any equipment was added, removed, or updated.

Conversion of ERD to relational schema was considered effectively by keeping in mind all the essential factors for conversion of ERD to relational model.

Functionalities Of Project

Customer Functions:

a. **Create account:** Customer creates account based on their Email-address. They chose their password, set Date of birth and gender.
b. **Update Information:** Customers can update their information.
c. **Delete Information:** Customers can delete information they want.
d. **Select Payment Plan:** Customer can select any payment plan they want from the given plans.
e. **Health History:** Customer can add the record of their health history and access it however they want. The health history of each user is stored based on customer id.
f. **Cycle Tracking:** If the customer's gender is female, they can also track their menstruation cycle.
g. **Instructional Video links:** The instructional video links are given on the customer dashboard where they can access it easily.
h. **Customer Goals:** The customer can add their goals. An entry field is given in the customer dashboard for this.
i. **Fat-o-meter:** Fat-o-meter gives the customer an idea about their weight's fluctuations from the BMI.

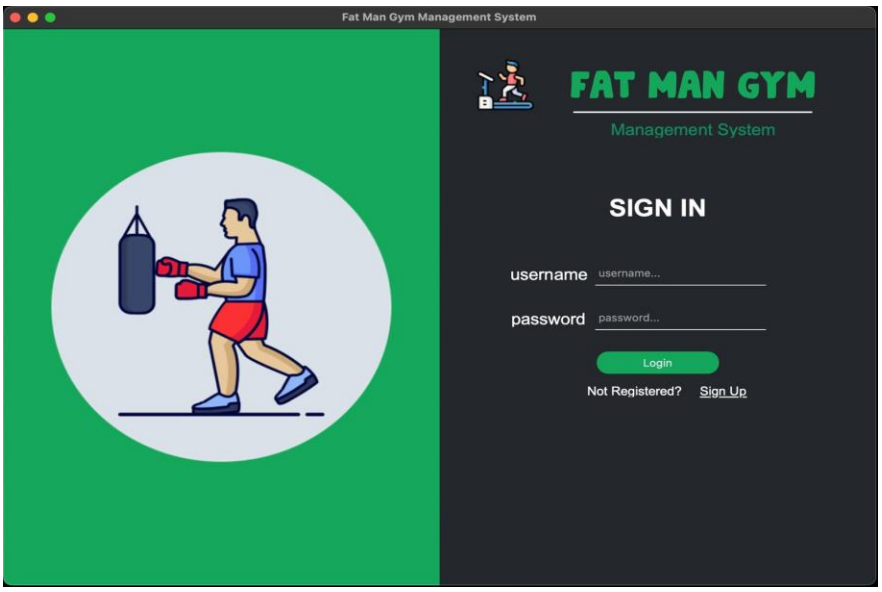
Admin Functions:

a. **Customer Management:** Admin can add, delete, or update the customer using the customer id.
b. **Staff Management:** Admin can add, delete, or update the staff members based on the staff id.
c. **Inventory Management:** Inventory is managed by the admin to add, update, or delete the equipment.
d. **Finances:** Admin has access to the gym expenditures and income.
e. **Payment:** Admin can make payments on user's behalf.
f. **Member's Data:** Admin can track the member's data and has a track of number of registered and active customers.

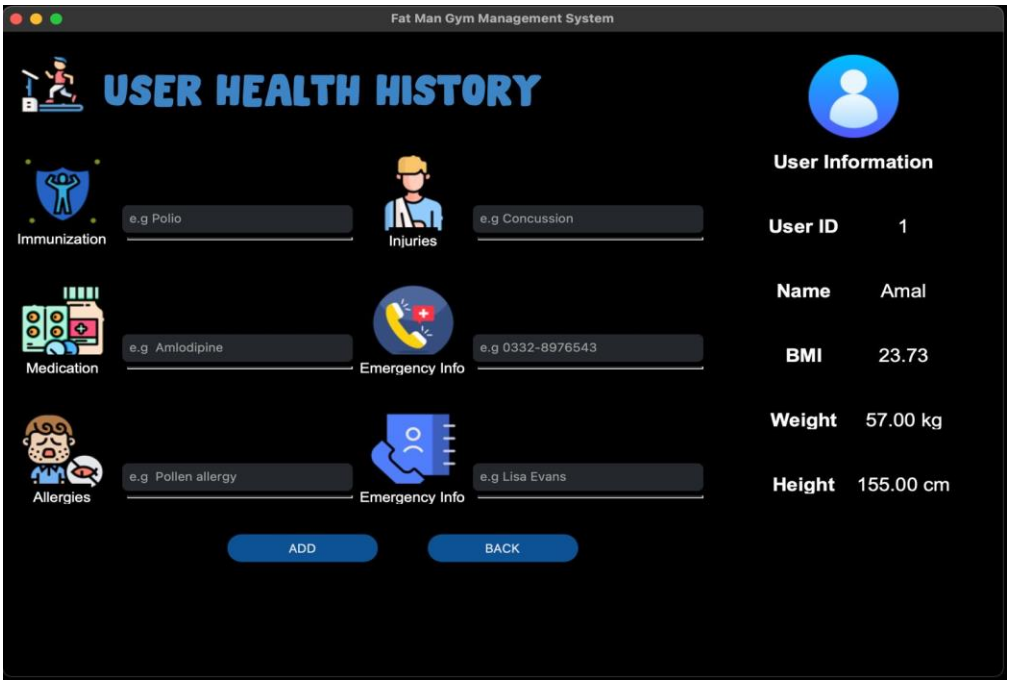
FRONT END

GUI is added to this management system to make it easy for the user to use it. Python is used to code the front end. For the GUI python GUI libraries tkinter and custom tkinter are added. Graphs are also added to make visuals better.

Sign In:

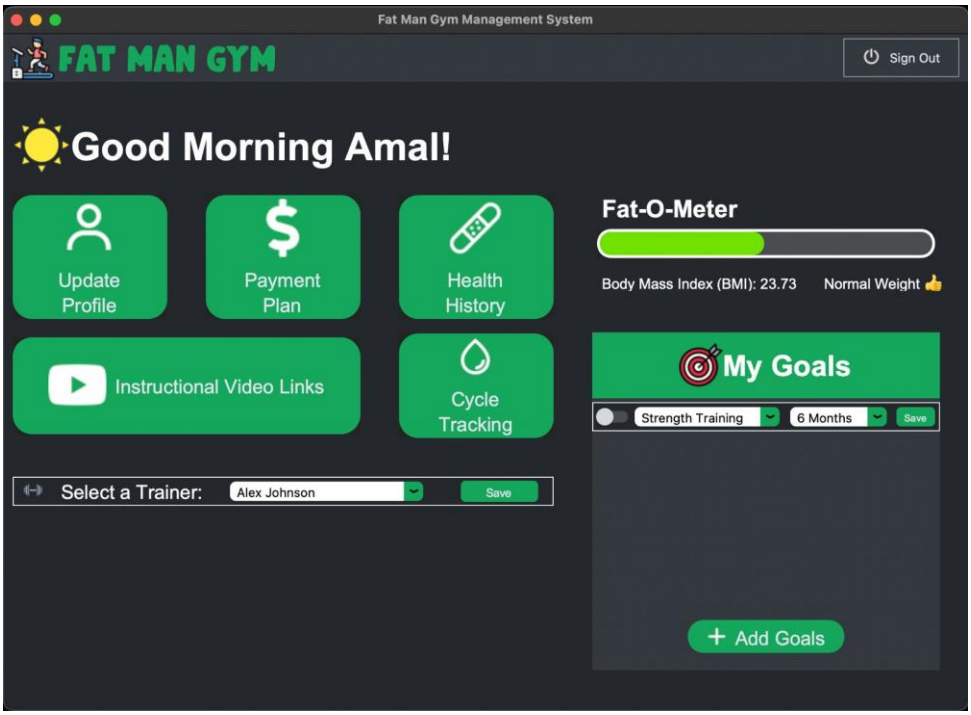


User Health History:

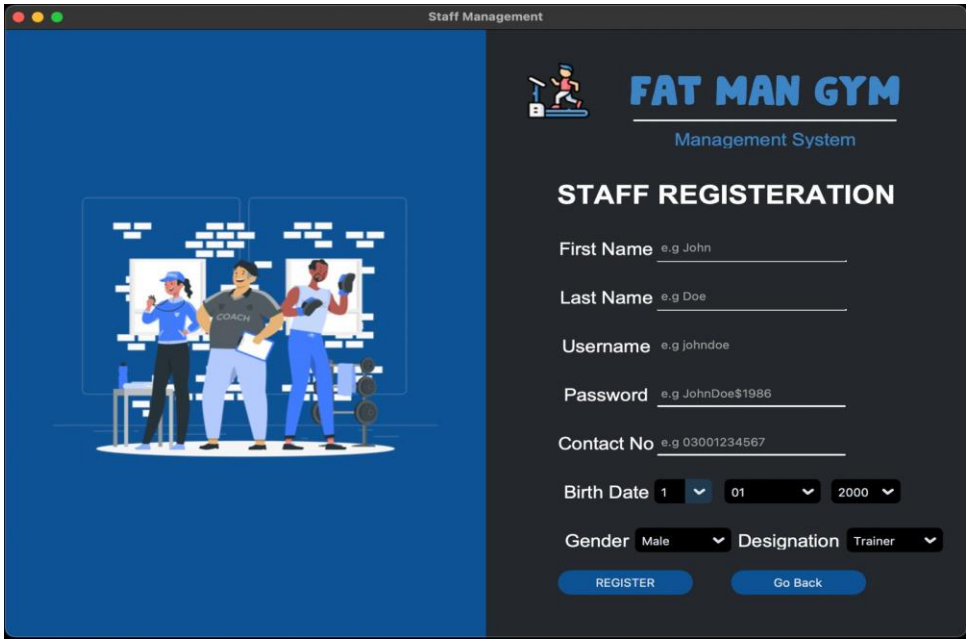


Admin Dashboard:

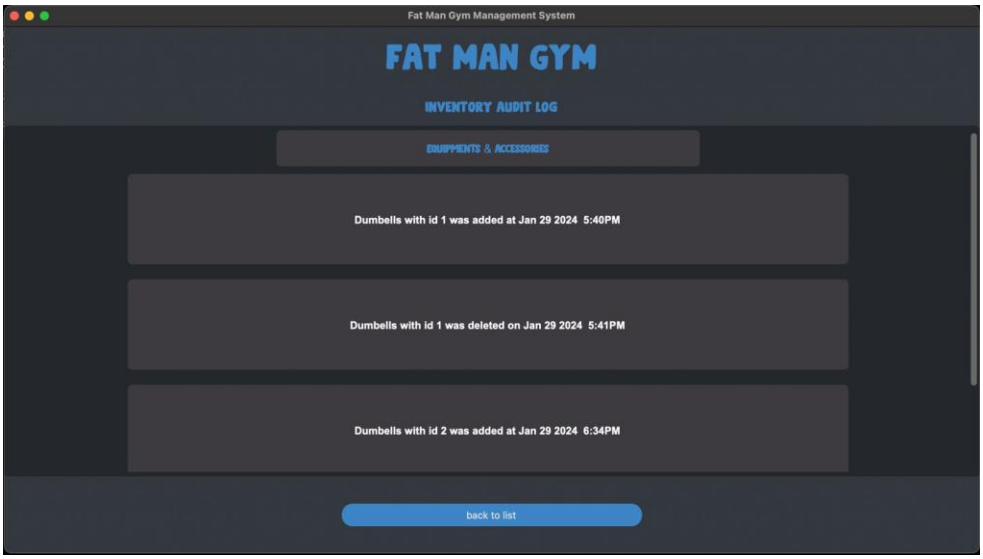
Customer Dashboard:



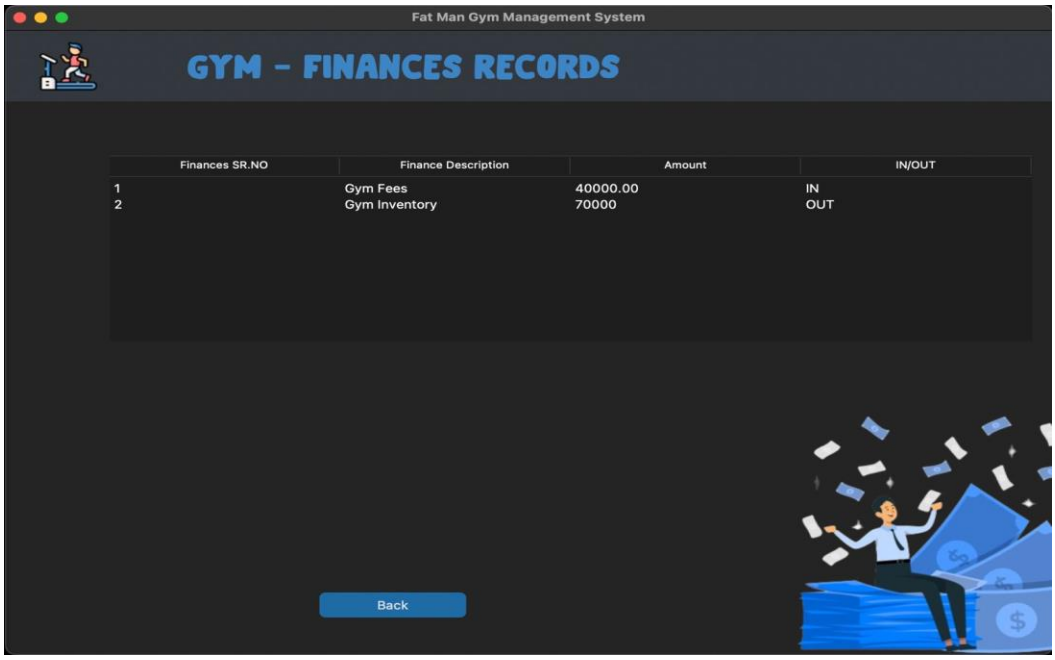
Staff Registration:



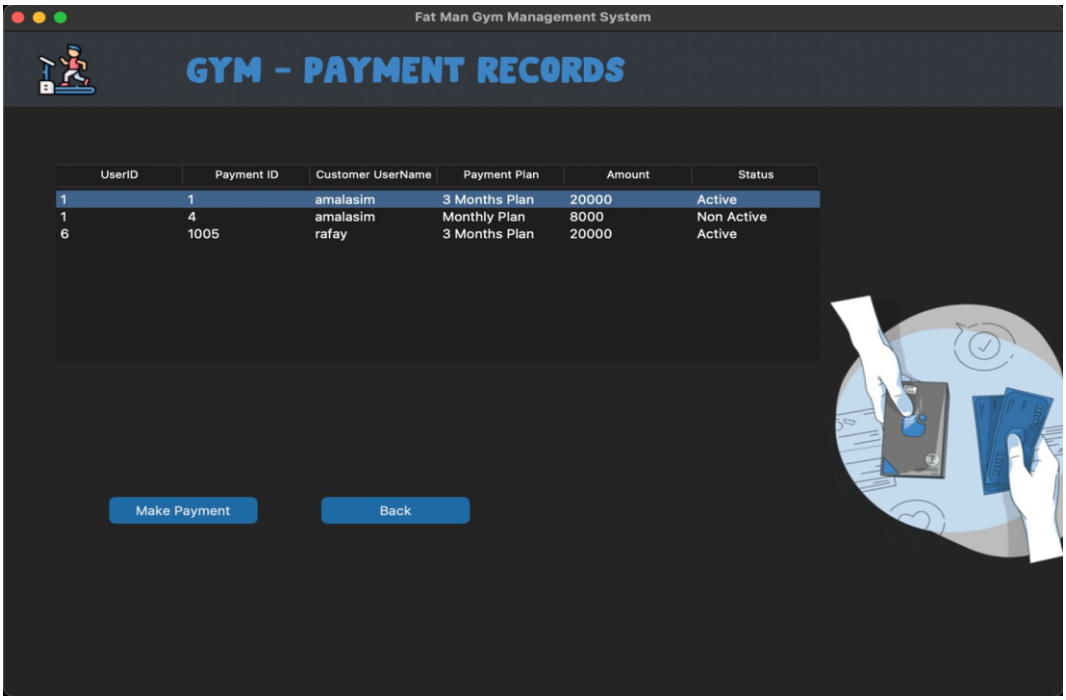
Inventory Audit Log:



Gym Finances Record:



Payments Record:



Back-end methods of SQL:

Joins:

Joins are used when data from multiple tables is required. Join are used at various points in this project.

```
--joining table to print data in payment table
SELECT CP.UserID, CP.payment_id, C.CustomerUserName, CP.payment_plan, CP.amount, CP.status
FROM CustomerPayments AS CP
JOIN Customers AS C ON CP.UserID = C.CustomerID;
```

The above join is used to get data from the user and payment table. Implementing this join has made it easy to get it in the DB query.

Procedures:

Procedures are just like the methods or functions in the front-end programming languages. Multiple procedures are used in this project to implement different functionalities.

```
CREATE PROCEDURE AddEquipmentToDatabase
   @name TEXT,
          @quantity INTEGER,
          @supplier TEXT,
          @price int,
          @equipment_type TEXT,
          @image_path TEXT
AS
BEGIN
   -- Insert data into the Equipment2 table
   INSERT INTO Equipment2 (name, quantity, supplier, price, equipment_type, image_path)
   VALUES (@name, @quantity, @supplier, @price, @equipment_type, @image_path);
END;

create table Equipment_audit(
     aud_id int primary key identity(1,1),
           history VARCHAR(max)
);
```

```
USE [Fat Man Gym];
GO
CREATE PROCEDURE DeleteUser
  @UserID INT
AS
BEGIN
  -- Check if the User ID is provided
  IF @UserID IS NULL
  BEGIN
    RAISERROR('Please provide a User ID.', 16, 1)
    RETURN
  END
```

Triggers:

Triggers are the stored procedures that are automatically called an event. Triggers are used in this project in the inventory system.

```
GO
CREATE TRIGGER equip_insertion
ON Equipment2
AFTER INSERT
AS
BEGIN
   DECLARE @name VARCHAR(MAX)
   DECLARE @id INT

   SELECT @name = name, @id = id FROM inserted

   INSERT INTO EquipmentAudit VALUES
   (CAST(@name AS VARCHAR(MAX)) + ' with id ' + CAST(@id AS VARCHAR(10)) + ' was
added at ' + CAST(GETDATE() AS VARCHAR(50)));
END;
```

Equipment insertion is implemented using triggers in this code.

Working:

The Fatman Gym Management System facilitates **user registration via email**, enabling **login and signup** functionality. Upon signup, users input their personal information such as age, birthdate, gender, weight, and height. Subsequently, the system generates a BMI report based on the provided data and assesses the user's BMI status using the Fat-o-meter. Users can set and track their fitness goals and are assigned a trainer accordingly. Payment plans are offered, and users can select from three available options. All user data is stored in a **Microsoft SQL Server database**, allowing for easy access and management.

Customers have the capability to **update or delete** their information as needed. Additionally, an admin interface is accessible for managing staff, customers, and inventory. Staff records can be updated, registered, or removed, while the **inventory module** maintains a comprehensive list of gym equipment. Admins can seamlessly add new equipment to the system as it becomes available. Financial records, including **payments and expenses**, are meticulously recorded within the gym management system, and presented to the admin via dedicated Python pages.

Conclusion:

In conclusion, the Fatman Gym Management System provides a comprehensive solution for managing all aspects of a fitness center. From user registration and BMI assessment to goal tracking and trainer assignment, the system streamlines operations and enhances user experiences. With flexible payment plans and seamless data storage in a Microsoft SQL Server database, the system ensures efficient management of user information and financial records.

In nutshell, the Fatman Gym Management System offers a user-friendly and efficient platform for both users and administrators, facilitating smooth gym operations and effective management of resources and finances.