Gym & Padel Management System

Team ID: **GEN\_074**

TA: Engy Abdallah

Team members:

2023170714 يوسف السيد حسنين السيد

2023170698 ياسمين محمد عبيد محمد

2023170258 سلمى خالد محمود البهائي

2023170695 ياسمين ربيع محمد محمد

2023170570 مروان صبحي محمد مصيلحي

2023170580 مريم أسامه جعفر ابو الحمد

* Main Idea

The system simulates the management of subscriptions and reservations for gym members and organizes the reservation system for the gym's padel courts.

* Structure of system
  + User Type: Each user has an email and password to help him log in to his account, he also has a name for the account, his date of birth, data of his subscription , and a variable that stores the location of the account’s profile picture and contains some custom data type such as the **slot** he has reserved, **notifications**, classes he is waiting for, and **classes** that have been reserved.
  + Staff Type: His responsibilities differ based on his role. There are three possible roles: manager, receptionist, and coach. Each staff member has a password, email, username, a variable for the profile picture location, and a variable indicating their position. If the staff member is a coach, this also includes the class ID they are assigned to.
  + Gym Class type : Each gym class has class Id , class name , instructor , instructor Id , start date , end date, price ,type of subscription allowed to enter this class , max number of member and session added by manager , it also includes a waitlist and a membership list, which are updated based on actual users who join.
  + Padel Court type : Each court has a court ID, name, location, price, and a list of reserved slots.
  + Credit Card type : Each card includes an ID , CVV , Name , Expiration date and amount
  + Workout type: Each workout has Id , type , duration and count of calories burned
  + Subscriptions type: Each subscription includes a type, start date, end date, price, and two variables to indicate whether the account is activated and whether it has VIP status.
  + Slot type : Each slot includes an Id ,court name , date of slot and start time.
  + Notification type: Each notification includes an ID, message, the date and hour it was sent, and a Boolean value indicating whether the message has been read.
* Program logic
  + Login page : The user enters their email and password, and if the credentials are correct, they're directed to their personal account page based on their account, either a staff or a regular user.  
    If the user doesn't have an account, they can create a new one if they're a regular user. However, if they're a staff member, they must enter a specific code generated by the manager. This code is valid for one-time use only and is required to successfully create a new account.
  + User page : There are several options or actions that the user can perform, including:
    - Booking padel courts: The user can choose the desired day and time for booking, and the system will display all available courts. If no courts are available, the user can select a different location, time, or day. Once they find a suitable court and time slot, they can proceed with the booking and will be redirected to the payment page.
    - Viewing the courts they 've booked and the corresponding times : All courts and time slots that the user has booked are displayed, and they have the option to cancel a booking provided it is done at least 3 hours before the scheduled time.
    - Renewing their subscription: When the renewal page opens, the user's current subscription type and duration are automatically selected, so if they want to renew the same plan, they can go straight to the payment page without extra steps. However, they also have the option to modify any details of the subscription. The new subscription will begin once the current one ends.  
      Subscription renewal becomes available up to 15 days before the current one expires.
    - Viewing the messages they've received : There are three types of messages:
      * The first type appears automatically when the subscription is close to expiring and it pops up as soon as the user opens their account.
      * The second type is sent manually by a staff to remind the user that their subscription is nearing expiration. This message appears in the user's direct messages (DM).
      * The third type is an automatic message sent when a user on the waiting list gets accepted into a class, notifying them that they have officially joined.

* + - Subscribing to gym classes : Each class is available only to users with specific subscription types. Every class has a set number of participants, a designated coach, and consists of multiple sessions. Users can either subscribe to a class or join its waiting list within 15 days from the class's start date. This same 15-day window also applies to cancellations.

When a user cancels their class subscription, one of the users from the waiting list is automatically added to the class, and a notification is sent to inform them of the update.

* **Waiting List Management System:**
  + - * Regular users are added to the end of the waiting list in order they join.
      * VIP users are given priority and are added to the beginning of the waiting list.
* **Workout filter:** 
  + - * This function enables the user to filter his workout history based on the type of workout selected from a predefined list (e.g., Cardio, Strength, Yoga, etc.). To ensure reliable filtering regardless of user input formatting, a normalization strategy is used.
      1. User Selection Normalization

The workout type selected from the combo box is:

* + - * + Converted to lowercase
        + Stripped of all whitespace
        + Handle the misspelling
      1. Workout Data Access

The program retrieves the list of workouts for the currently logged-in user from a data structure (unordered\_map<string, vector<Workout>>).

* + - 1. Workout Type Normalization

Each workout’s stored type is also:

* + - * + Converted to lowercase
        + Stripped of whitespace

This ensures comparisons are case- and space-insensitive.

* + - 1. Filtering Logic
* The normalized selected type is compared with the normalized workout type for each workout. If a match is found, the workout is displayed.
  + - 1. Result to User
* If matches are found, their details (date, type, duration, calories burned) are displayed.
* If no matches are found, a message box notifies the user.
* **Add Workout Logic**
* This function handles the process of adding a new workout entry for a user.
  + - 1. Input Validation
* The function reads user input values: date, type, duration, and weight.
* It checks if any field is empty.
* Then, it converts duration and weight to integers and ensures both are positive numbers.

2. Subscription Validity Check

* + - * + It retrieves the user's subscription start and end dates, converts them to time\_t, and compares them to the current date.
        + If the current date is outside the valid subscription period, the user is not allowed to add a workout.

3. Calorie Calculation

* It calls calcCalories() in WorkoutManager, passing the user, workout duration, type, and body weight.
* This function returns the estimated calories burned during the workout.

4. Workout Creation and Recording

* + - It creates a new Workout object using the provided values and calculated calories.
    - The new workout is then saved under the current user's ID using recordWorkout().
* **Track Progress Logic**

This function calculates the user’s overall workout progress based on historical data

1.Fetch User's Workout Records:

* + - * Retrieves the list of workouts for the current user from the workout data map.
      * If no workouts are found, the function exits.

2.Calculate Total Metrics:

* + - * Initializes totalCalories and totalDuration counters.
      * Loops through each workout and accumulates:
      * Total calories burned
      * Total workout duration
* **View History Logic**

This function retrieves and displays all stored workouts for the currently logged-in user.

1.Identify User's Workout Records:

* + - * The function searches for the current user’s ID in the workout data map.
      * If no entry exists or the user has no workouts, the process stops.

2.Iterate and Display Workouts:

* If workouts exist, the function loops through each workout associated with the user.
* For every workout, the following details are accessed
* **User Account :**
* **User Details Retrieval**: The system retrieves user information (name, email, ID, birthday) from the user list data structure. If the user is not found, an error message is generated. The password is not displayed for security reasons.
* **Subscription Details:** The system checks the user's subscription status. If the subscription is inactive or expired, a message prompts the user to renew. If the subscription is active but set to expire within 5 days, a warning message is generated, including details like VIP status, subscription type, start date, end date, and price. If the end date cannot be retrieved, an error message is generated.
* **Enrolled Classes:** The system retrieves the user's enrolled classes from (user. My Classes) to fetch class details (Class ID, Class Name, Session Date, Session Time). If no classes are enrolled or the subscription is expired, an appropriate message is generated. If an error occurs during data retrieval, an error message is produced.
* **Manager Page:**
* The manager page allows the gym manager to oversee and control gym operations through various administrative tasks. It displays the manager’s name and profile picture and provides options to manage users, staff, classes, and reports.
* **The manager can perform the following actions**:
* **Search User by ID:** The manager enters a user ID to view a detailed profile, including the user’s full name, date of birth, subscription type, VIP status, number of classes entered, wallet balance, profile picture path, enrolled classes, waitlisted classes, and booked padel courts with slot details (court name, date, start time, court ID). The system checks if the ID is valid and exists in the database. If the ID is empty, invalid, or not found, an error message appears. The manager can use this information to monitor user activity, such as class participation, court bookings, or account status**.**
* **Delete User:** The manager inputs a user ID to remove a user from the system. The system first validates the input to ensure the ID is not empty, then checks if the user exists. If the user has an active subscription, the system blocks the deletion, preserving data consistency. If the subscription is inactive, the user is deleted, and a confirmation message is displayed confirms the user’s removal, the system should also clear all associated user data such as (court bookings, class enrollments, and wallet information) when the user is removed. If the ID is invalid or not found, an appropriate error message is shown. This feature helps the manager maintain a clean and up-to-date user database by removing inactive or erroneous accounts.
* **Display Coach Classes:** The manager enters a coach’s ID to see all classes assigned to that coach, including class names, IDs, dates, start times, and end times for each session. The system also shows the coach’s name, email, and phone number. The system verifies that the ID belongs to a coach and checks for assigned classes. If the coach has no classes, a message indicates no classes are assigned. If the ID is invalid or does not belong to a coach (e.g., a receptionist or manager), an error message appears. The manager can review this information to monitor coach schedules and ensure classes are properly assigned.
* **Create Class:** The manager can create a new gym class by entering details such as the class name, instructor name, instructor ID, start date (day, month, year), start and end times, maximum number of participants, allowed subscription types, and recurring days of the week (e.g., Sunday, Monday). The system validates the input, ensuring the class name contains only letters, the start date is valid and between 2025 and 2030, The end Date automatically calculates a month from the Start Date, the times are in HH:MM format with the end time after the start, the maximum number of participants is positive, the instructor is a valid coach in the system, at least one subscription type and one day are selected, and the coach has no scheduling conflicts for the session dates and times. The system generates a unique class ID and calculates an end date for the class. It creates sessions for each recurring day from the first session date (the earliest selected day on or after the start date) to the end date, with each session including the date, start time, and end time. The system ensures no scheduling conflicts by checking the coach’s existing classes for overlapping times on session dates. Once created, the class is added to the system with its sessions and allowed subscription types, and users can subscribe or join its waitlist. A confirmation message displays the class ID upon successful creation, or an error message appears if any details are invalid, such as a scheduling conflict or invalid coach. This allows the gym to expand its class offerings with properly scheduled sessions.
* **Generate Monthly Report:** The manager selects a month from a dropdown menu to generate a report summarizing gym activities for that month and the current year. The report includes a header with the month and year, a list of up to 10 most active members (showing their name, ID, number of classes attended, and VIP status), a summary of subscriptions by type (showing each plan type, number of subscribers, and revenue per type), and the total subscription revenue in EGP. The system retrieves data for users with active subscriptions in the selected month. If no active users are found, an error message appears. After generation, the month selection resets for the next report. The manager can use this report to review member activity and subscription revenue **(“Bonus”).**
* **Generate Staff Code:** The manager enters a unique code to create a one-time-use code for new staff registration. The system saves the code and copies it to the clipboard for easy sharing with the new staff member. A confirmation message instructs the manager to share the code securely. If the entered code is empty or invalid, an error message appears. This ensures only authorized staff can create accounts, maintaining system security.
* **Log Out:** The manager can log out of the system. The system displays a loading screen with a “Logging out…” message, and returns to the login page. This ensures the manager’s account is secure and prevents unauthorized access.
* **Display Classes for a specific month:** this function is responsible for showing all gym class sessions that take place in a selected month. It starts by clearing any previously displayed data and then converts the selected month from a string to an integer. It sets up a styled Data Grid View with columns like Class ID, Name, Instructor, and session details. The function then loops through all gym classes stored in a global list, and for each class, it checks each session's date. If the session’s date matches the selected month, that session is added to the table. If no sessions match, a message is shown saying there are no classes for that month. Otherwise, the table is made visible, displaying the matched results. This ensures users get a clear view of all scheduled classes in the chosen month.
* **Staff page:**
* This page allows logged-in staff members to carry out tasks based on their role (coach or receptionist). Core functionalities include identifying users with expiring subscriptions, registering new users, searching for users by ID, and displaying gym class details associated with coaches. The logic is centered around validating input, filtering global data structures, and showing context-specific information depending on staff role and action.
* **Search User by ID:** Onlyreceptionists can search for existing users by entering their ID using a two-step process. First, the search mode is activated by clicking the "Search User" button, which resets the system’s state by clearing previous notifications, class displays, and search results. Then, when the staff enters an ID and clicks the "Search" button, the system trims and checks the input to ensure it’s not empty. It then goes through the user List to find a matching ID. If a match is found, the system displays the user’s full details, including name, date of birth, subscription type, VIP status, wallet balance, enrolled and waitlisted classes, and padel court bookings (with court name, date, start time, and court ID). If the ID is invalid or not found, an error message is shown. This search feature helps staff quickly access and manage user information while ensuring only valid inputs are processed.
  + **Send Notifications:** Staff can see a list of users whose subscriptions will end in the next seven days. When this feature is used, the system clears any old data and gets the screen ready for new information. It goes through all users and checks how many days are left in each person’s subscription using “get Remaining Days”. If a user has less than seven days left, their details—like name, ID, subscription type, end date, and account type (VIP or Normal) are added to a list. This helps staff know which users need reminders so they can ask them to renew their subscriptions on time.
  + **Register New User**: Staff can add new users by clicking the "Add New User" button, which begins the signup process. When this happens, the system clears any previous search results, flags, or class displays to make sure the environment is clean and ready for new data. Then, the sign-up form appears, allowing staff to enter the new user's details. The system checks that the email is unique and properly formatted, the name contains only letters and spaces, the date of birth is valid, and a subscription type is selected. Once everything is verified, the payment is processed based on the selected plan. This smooth and reliable process helps receptionists register new gym members without confusion or errors. (For full details, see the **Signup Page** section.)
  + **Display Classes:** it works differently based on the staff member's role. If the logged-in staff member is a coach, the system automatically retrieves and displays only their own class schedule without asking for any input. It uses the coach’s login ID to fetch their classes from “gym Classes List”, showing details like class ID, name, date, start time, and end time. If the coach has no assigned classes, the system displays a message indicating that. On the other hand, if the logged-in staff member is a receptionist, they must first enter a coach’s ID to search for that coach’s class schedule. The system validates the entered ID, checks if it belongs to a coach, and then retrieves the coach’s information (name, email, phone number) along with their assigned classes. If the ID is invalid, not a coach, or the coach has no classes, the system shows an error or informative message. This role specific logic ensures coaches can view only their own schedule, while receptionists can assist users by checking class schedules for any coach as needed.
* **Sign up page:**
* The signup page is the entry point for account creation in the Gym & Padel Management System. It handles two account types:

1. **Regular User:** For gym members who book padel courts, subscribe to classes, or track their workout.
2. **Staff:** Employees with roles (receptionists, or coaches), with restricted access requiring a manager-generated code (managers can’t sign up they are already defined on the files).

* **For regular users:** the signup process requires a unique email address with a valid format containing “@.com”, a secure password, a name that contains only letters and spaces, and a valid date of birth. Uploading a profile picture is optional. Users must also select a subscription type from the available options (1 month, 3 months, 6 months, or 1 year), each of which has its own pricing, duration, and class access limits. If the VIP option is selected, an additional fee of 100 EGP is added to the base subscription price. To complete the registration, users must proceed through a payment process where they enter their credit card information. The form includes real-time input validation to ensure correct formatting—only digits are allowed in the card number (exactly 16 digit), expiration date, and CVV fields (exactly 3 digit), while only letters and spaces are accepted in the cardholder name field. If the payment fails, registration is halted and an error message is displayed. Upon successful payment, a user object is created and the system attempts to register the user through the “register Member” function, which validates all provided data. If any validation fails, the process is canceled and any uploaded profile image is removed. If successful, a unique user ID is generated based on existing records, and the user’s data, including their profile picture, is securely stored—ensuring a complete and validated registration process.
* **For staff:** The staff registration process ensures that only valid and authorized personnel can create accounts. The system first collects essential input: name, email, password, phone number, and a registration code, along with selecting a staff role (Reception or Coach). Input validation checks are applied: all fields must be filled, the email must be properly formatted containing “@.com”, the phone number must be more than 11 digits and contain only numeric characters, and the registration code must match a predefined passcode created by the manager. If any validation fails, the registration is aborted with an error message. Once validated, a temporary profile picture may be set if provided, using a placeholder ID. The “register Staff” function then verifies name and email validity, checks for email uniqueness, assigns an ID based on the role (prefixing with 'R' for Reception or 'C' for Coach), and stores the new staff object. If successful, the temporary image (if present) is renamed to match the actual staff ID and linked to the account. Finally, the new staff is added to the system with a success message. If the process fails, the image is deleted and an error message is shown. This approach ensures secure, verified, and well-structured staff account creation.