UNIVERSITY of INFORMATION TECHNOLOGY and MANAGEMENT
in Rzeszow, POLAND

Documentation

LIGHTWEIGHT GYM

Web application

Team members:
Brian Abel Timothy (w66760)
Zhanarys Malik (w65398)
Nurtas Mukashev (w66257)
Rustam Izbassov (w65395)

Supervisor:
Knap Maksymilian

Project Overview

The Lightweight Gym is planning to create a web application that is a comprehensive fitness tool that offers a range of functionalities to help gym members and non-members achieve their fitness goals. The users can download the application for free and sign up, or sign in if they have registered, to get into the application and see the application's features and provided gym services.

Design Description

The app has three functionalities in terms of user management, which are:

1. User registration to the web application
2. Log in to the web application
3. Password resetting/recovery

There are seven features that users can use. These are:

1. Viewing the list of activities and the schedule.
2. Viewing the diet plan
    a. Bulking
    b. Cutting
    c. Maintaining
3. Viewing workout plans
    a. Upper body
        i. Arms
        ii. Backs
        iii. Grip
    b. Lower body
        i. Calves
        ii. Thighs
4. Viewing sauna schedule
5. Upgrade membership if your current one is either bronze or silver
   This feature only works for members.
6. Downgrade membership if your current one is either silver or gold
   This feature only works for members.
7. Unsubscribe from the gym
    a. Ask for feedbacks

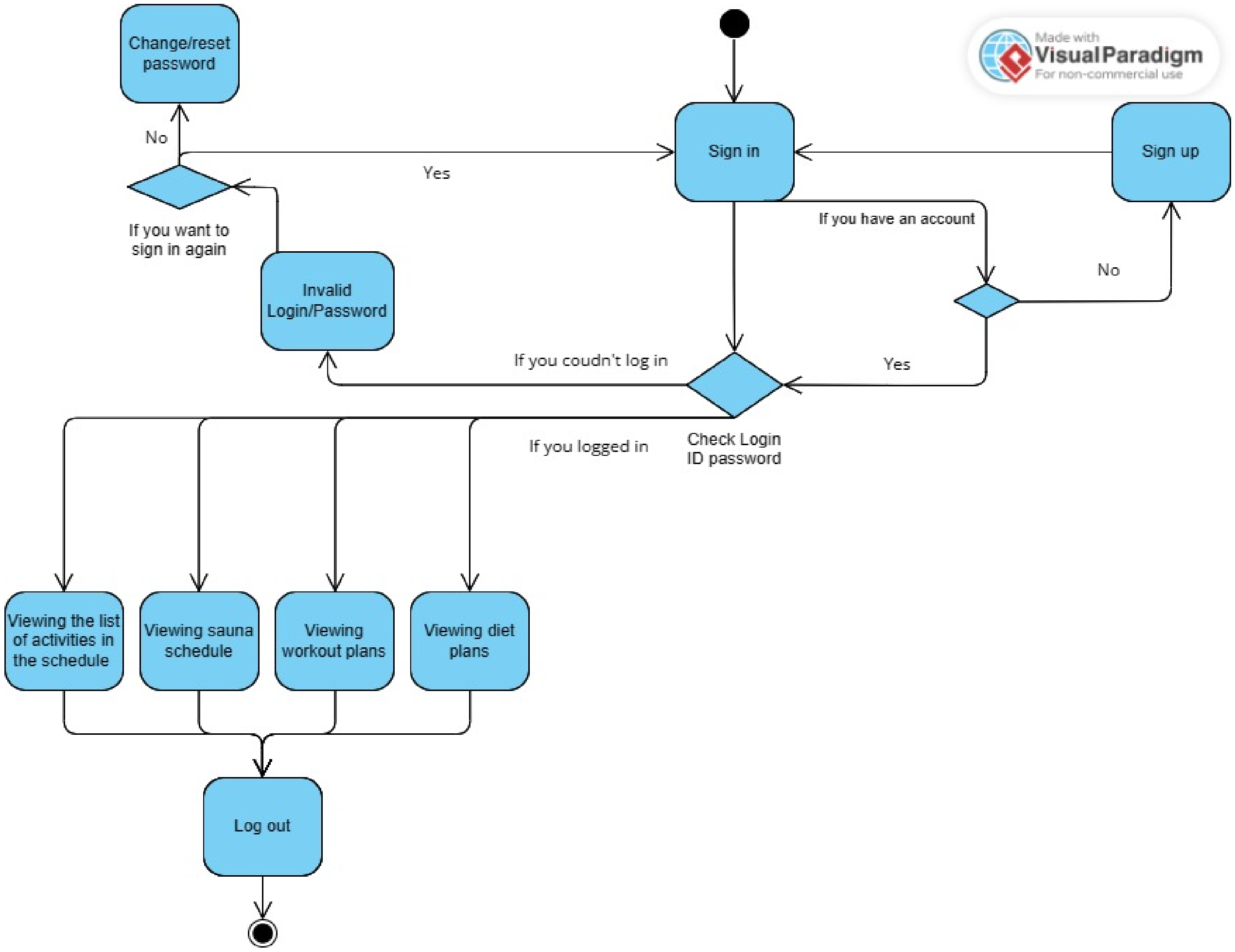
   This feature only works for members.

User Stories

1. As a new user, I would like to register for the Lightweight gym web application.
2. As a user, I would like to log in to my profile.
3. As a user, I would like to change my password or recover my password if I forget my password.
4. As a user, I would like to view the list of activities and the schedule of those activities.
5. As a user, I would like to view the diet plan for bulking, cutting, and maintaining my body.
6. As a user, I would like to view the workout plans for upper body and lower body.
7. As a user, I would like to view the sauna schedule.
8. As a user, I would like to upgrade membership if my current one is either bronze or silver.
9. As a user, I would like to downgrade membership if my current one is either silver or gold.
10. As a user, I would like to unsubscribe from the gym and share my feedback.

Diagrams

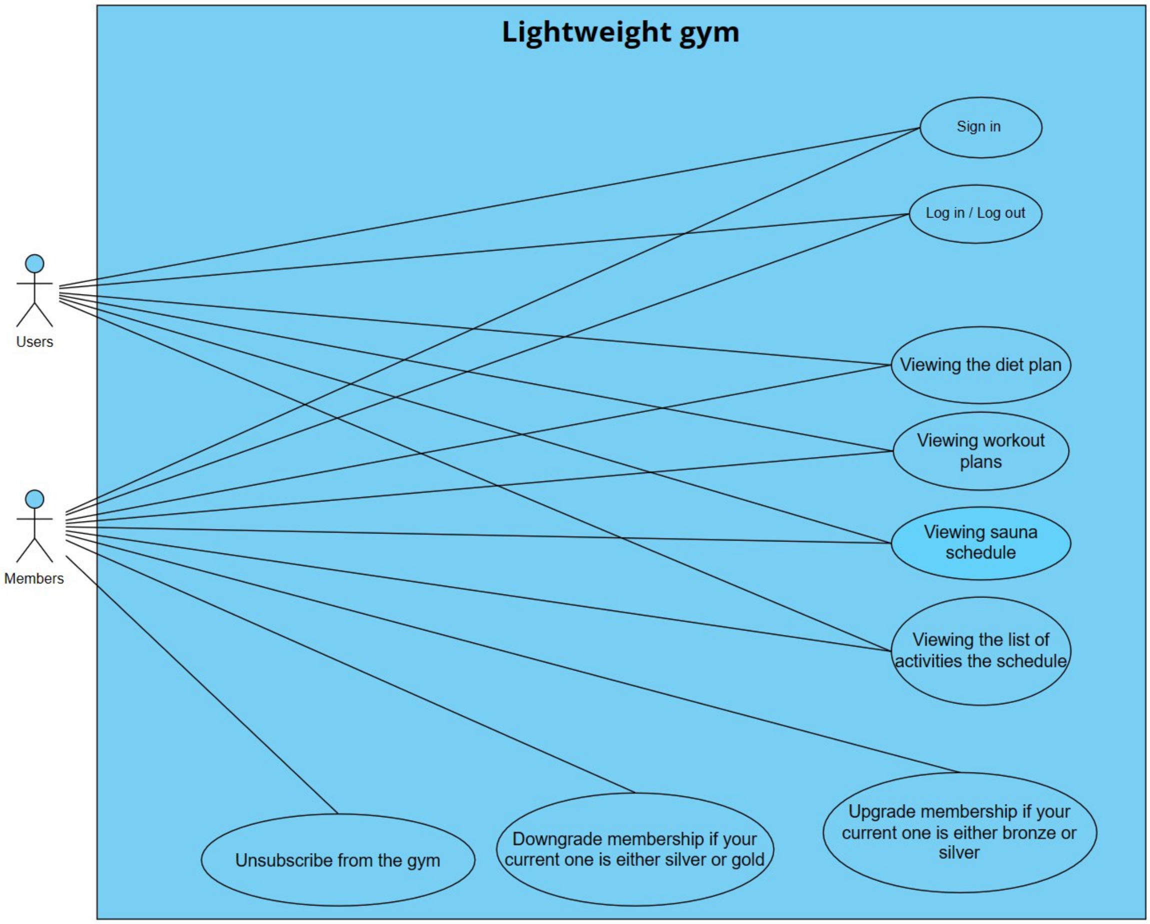
Activity diagram



This activity diagram illustrates the sequential flow of activities and processes within a system. The user initiates the process by accessing the "Sign In" page. In the case of new users, they have the option to first complete the "Sign Up" process before proceeding to the "Sign In" page. In the event that users encounter difficulties signing in, they can utilize the password recovery functionality to reset their password and attempt signing in once again.

Once successfully signed in, users gain access to various website features, including the activity schedule, sauna schedule, workout plans, and diet plans. Alternatively, users have the option to log out of the system, thereby concluding their session.

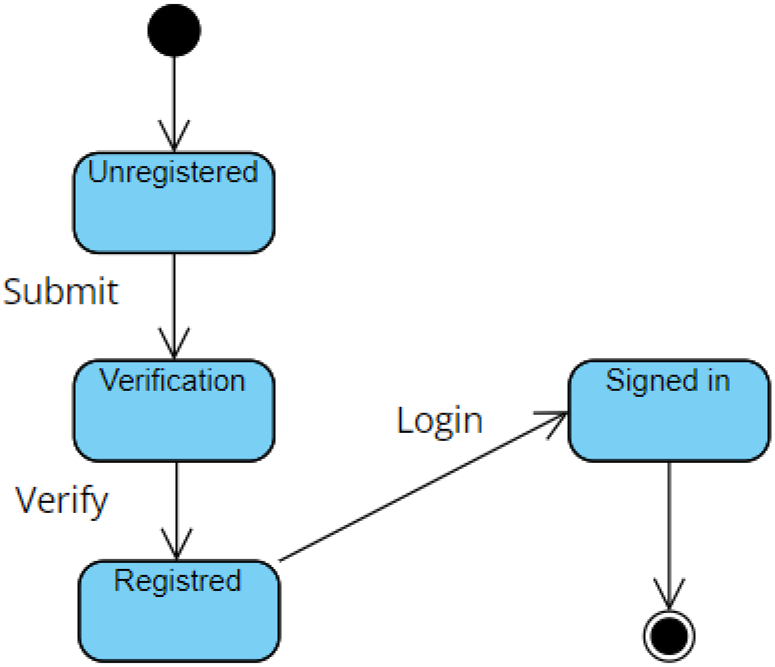
Use Case Diagram


Lightweight gym
Users
Members
Sign in
Log in / Log out
Viewing the diet plan
Viewing workout plans
Viewing sauna schedule
Viewing the list of activities the schedule
Unsubscribe from the gym
Downgrade membership if your current one is either silver or gold
Upgrade membership if your current one is either bronze or silver

We have distinguished our consumers into two distinct roles: Users and Members. Members refer to those consumers who have subscribed to Lightweight Gym, while Users encompass the general visitors of our website. At this initial stage of development, the primary distinction between these roles lies in the Members' ability to modify their subscription type and unsubscribe from the gym, subsequently becoming Users.

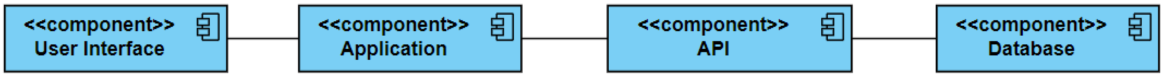
Our system comprises nine key features, six of which are accessible to both roles. These features include the options to sign up and sign in, view the diet plan, access workout plans, review the sauna schedule, and explore the activity schedule.

State diagram



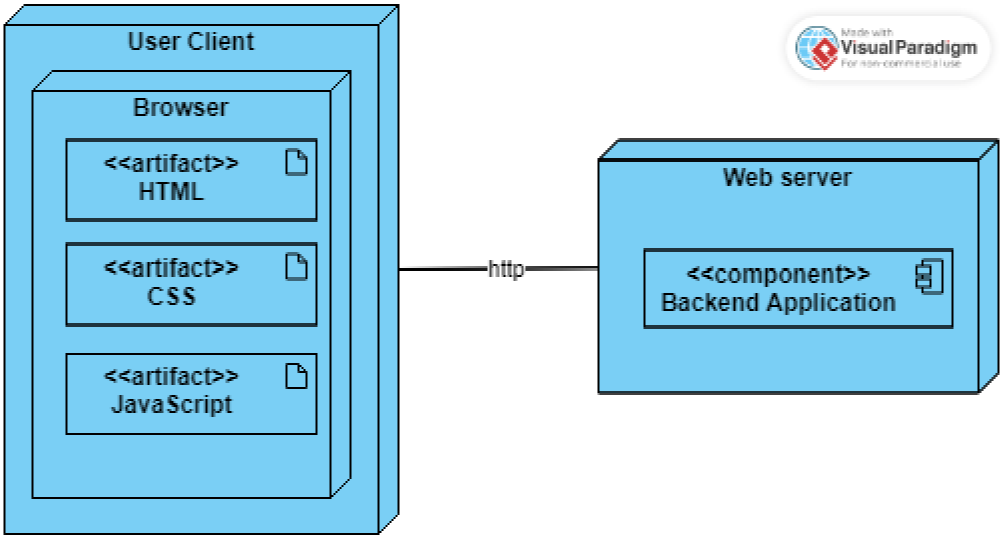
The state diagram represents the user journey in the gym web application, starting from the "Unregistered" state. After registering, the user continues to the "Verification" state, where their account is verified. Once the account is verified, the user transitions to the "Registered" state. From here, they can sign in to the application, moving to the "Signed in" state.

Component diagram



The gym web application consists of four main components. The "User Interface" handles pages for login, home, profile, and schedule. The "Application" includes modules for authentication, user management, and schedule management. The application integrates with “API” to fetch data from the database, such as user’s data and the list of the activities. The "Database" stores data in tables for users, schedules, activities, etc.

Implementation diagram



The web server node serves as the hosting environment for the backend application. It provides the necessary infrastructure for users to serve the web application on the user client server.

"User Client" node represents the user interface or client-side application that users interact with. It handles the presentation layer and user interaction.

"Web Server" node represents the server that consists of the backend application component. It provides the necessary infrastructure of the REST API architecture (controllers, dbcontext, etc.) including the LightweightGymApi.db (SQLite).

Overall, this deployment diagram illustrates the physical arrangement and relationship between the web server and user client nodes in the gym web application.

Test Cases

- Use Case: Sign Up

  Success Conditions:
    - User provides valid and unique registration details.
    - User's information is successfully stored in the database.

  Failure Conditions:
    - User enters incomplete registration details.
    - User enters invalid password guidelines
    - User enters different password (password mismatch)
    - User attempts to sign up with an already existing email.

- Use Case: Sign In

  Success Conditions:
    - User provides valid credentials (email and password).
    - User's credentials match an existing user in the database.
    - User is successfully authenticated and granted access.

  Failure Conditions:
    - User enters incorrect or invalid credentials.
    - User's credentials do not match any existing user in the database.

- Use Case: See List of Activities

  Success Conditions:
    - User is authenticated and authorized to access the list of activities.
    - List of activities is successfully fetched from the server.
    - Activities are displayed to the user.

  Failure Conditions:
    - User is not authenticated and unauthorized to access the list of activities.

- Use Case: Upgrade Member

  Success Conditions:
  - User is authenticated and authorized.
  - Member's member type is successfully upgraded.
  - Updated member information is stored in the database.

  Failure Conditions:
  - User is not authenticated or authorized.

- Use Case: Downgrade Member Type

  Success Conditions:
  - User is authenticated and authorized.
  - User selects a member to downgrade their member type.
  - Member's member type is successfully downgraded.
  - Updated member information is stored in the database.

  Failure Conditions:
  - User is not authenticated or authorized.
  - Selected member does not has a membership.

- Use case:The diet plan

  Success conditions:
  - User is authenticated and authorized to access the list of activities.
  - The diet plan is successfully fetched from the server.
  - The diet plan is displayed to the user.

  Failure conditions:
  - User is not authenticated and unauthorized to access the diet plan.

- Use case:The workout plans

  Success conditions:
  - User is authenticated and authorized to access the workout plans.
  - The diet plan is successfully fetched from the server.
  - The diet plan is displayed to the user.

  Failure conditions:
  - User is not authenticated and unauthorized to access the workout plan.

- Use case:Password recovery
  - Success conditions:
  - User provides an existing email
  - User has a mail for password recovery.

  Failure conditions:
  - User's credentials do not match any existing user in the database.

- Use case:Sauna schedule

  Success conditions:
  - User is authenticated and authorized to access the workout plans.
  - The diet plan is successfully fetched from the server.
  - The diet plan is displayed to the user.

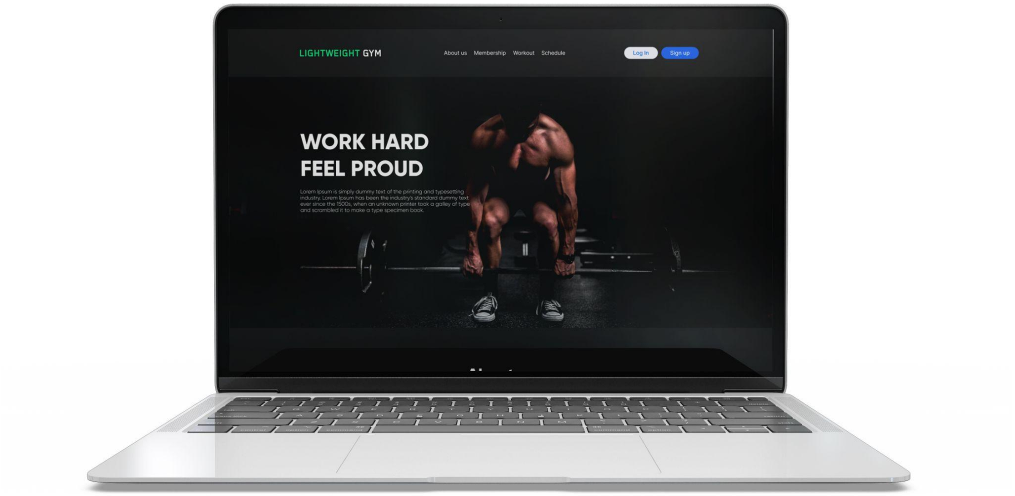
  Failure conditions:
  - User is not authenticated and unauthorized to access the workout plan.

- Use case:Unsubscribe from the gym

  Success conditions:
  - User is authenticated and authorized to unsubscribe from the gym.
  - User unsubscribes from the gym.
  - User is successfully unsubscribed from the gym
  - Updated information is stored in the database

  Failure conditions:
  - User isn't authenticated and unauthorized to unsubscribe from the gym.
  - User isn't subscribed to the gym

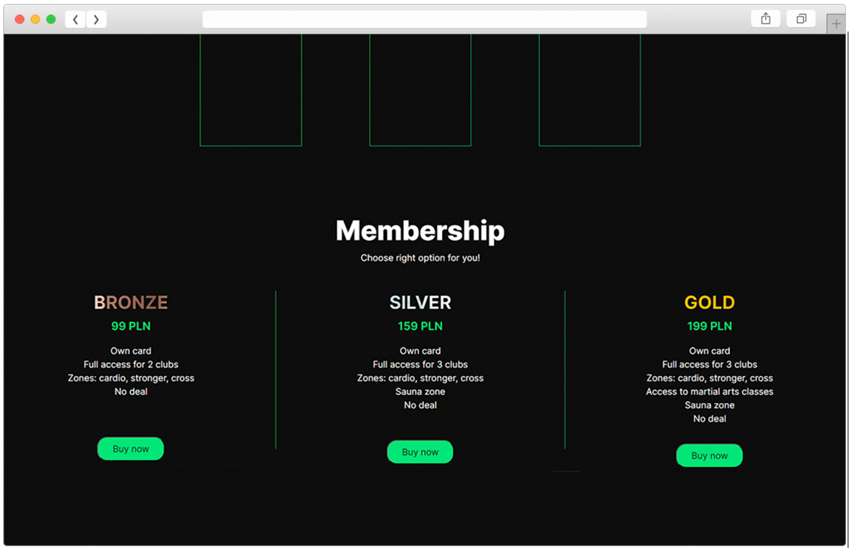
Web Design



WORK HARD
FEEL PROUD

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

The main section of our website includes a header. The header has a logo, navigation bar and authentication buttons. In main banner placed main text with tag <h1></h1>.



Membership

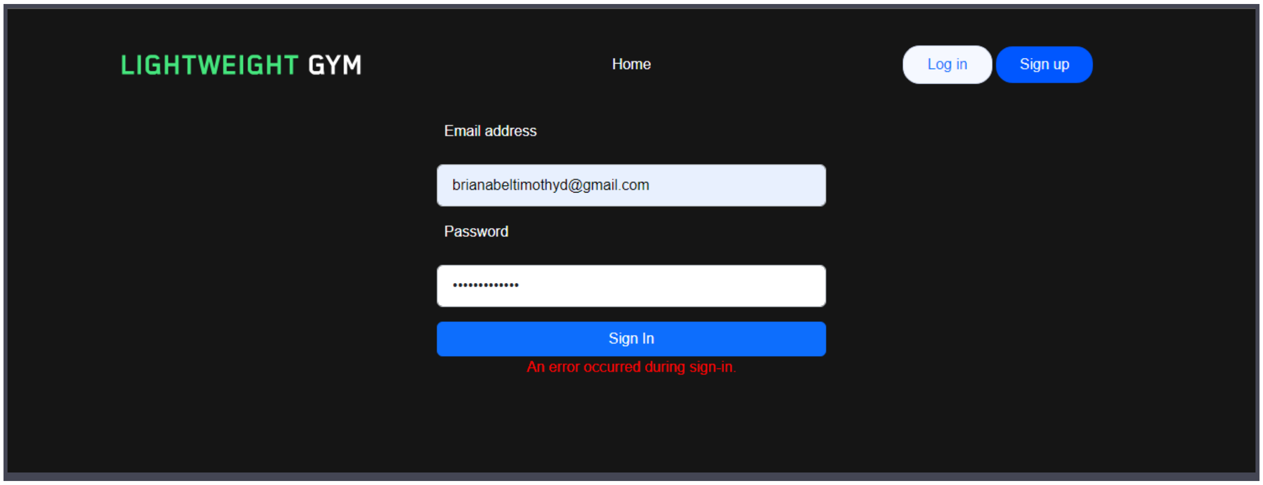
Choose right option for you!

BRONZE
99 PLN
Own card
Full access for 2 clubs
Zones: cardio, stronger, cross
No deal

SILVER
159 PLN
Own card
Full access for 3 clubs
Zones: cardio, stronger, cross
Sauna zone
No deal

GOLD
199 PLN
Own card
Full access for 3 clubs
Zones: cardio, stronger, cross
Access to martial arts classes
Sauna zone
No deal

In the section Membership, there are three options available for the users to choose from. Users are able to choose the most suitable subscription and buy it.



Email address

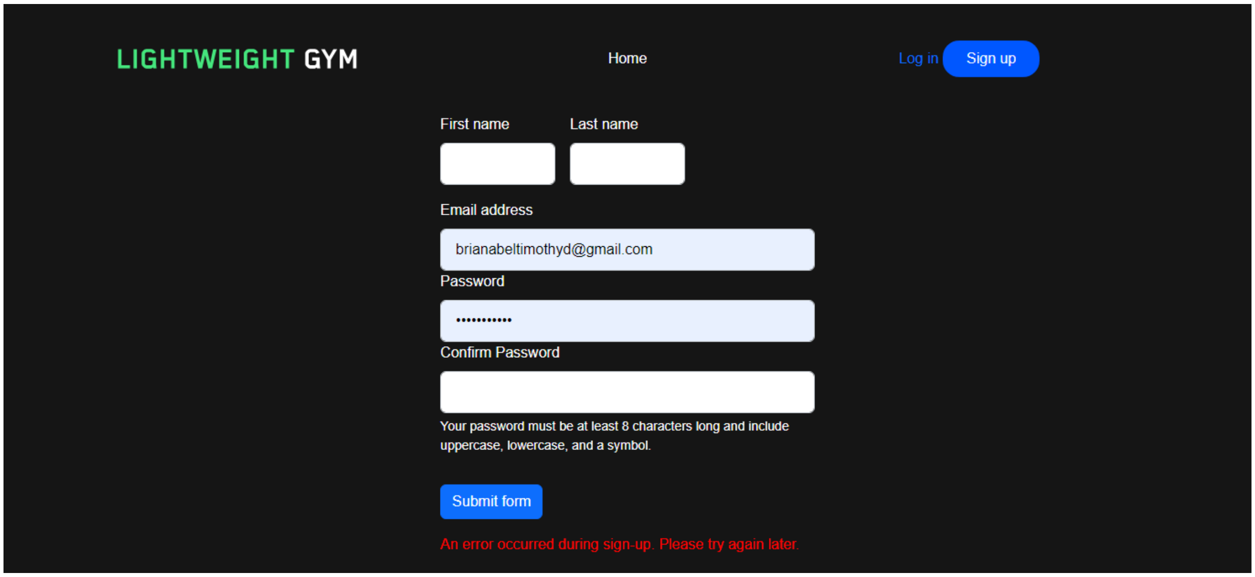
brianabeltimothyd@gmail.com

Password

•••••••••••••

An error occurred during sign-in.

This is a page where members and users can log in to the Lightweight Gym website.



First name

Last name

Email address

brianabeltimothyd@gmail.com

Password

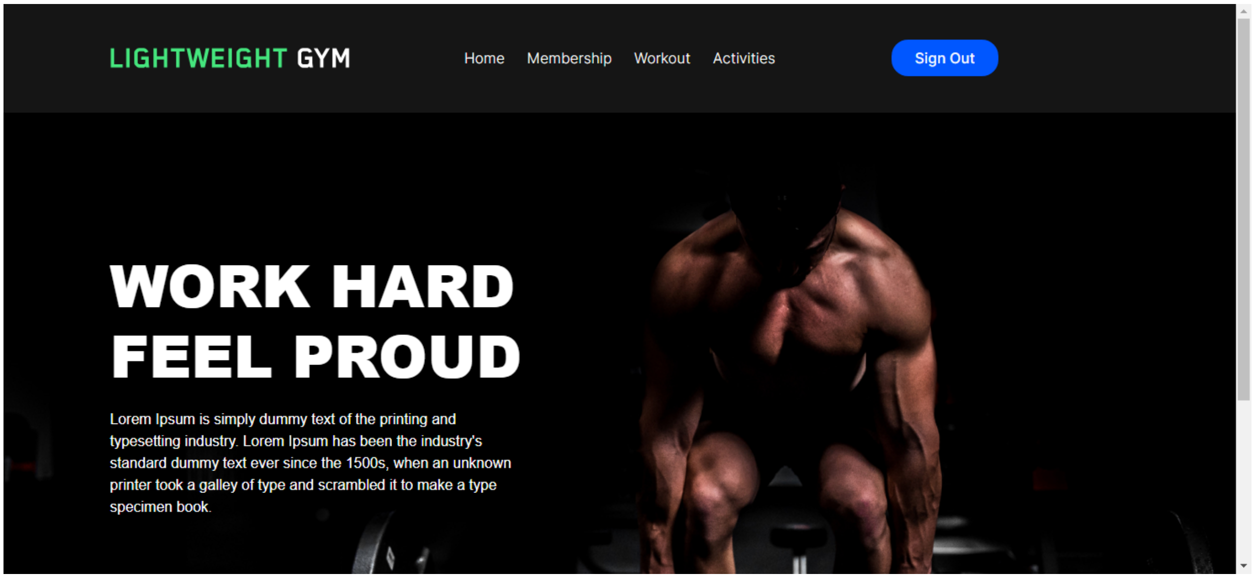
•••••••••••

Confirm Password

Your password must be at least 8 characters long and include uppercase, lowercase, and a symbol.

An error occurred during sign-up. Please try again later.

This is a page where non-registered users go to sign up to see the whole website and unlock more functionalities.



WORK HARD
FEEL PROUD

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

The webpage after users login to the website. The navigation bar is the functionalities available.

Frontend Technology

HTML

HTML (Hypertext Markup Language) is the standard markup language for creating web pages and web applications. It provides the structure and defines the content of a web page. HTML consists of a series of elements, which are represented by tags, that define different parts of a webpage's structure and content.

CSS

CSS (Cascading Style Sheets) is a style sheet language used to describe the presentation and formatting of HTML (and XML) documents. It provides a way to control the appearance of web pages, including elements like layout, colors, fonts, and other visual aspects.

CSS works by applying styles to HTML elements using selectors. Selectors target specific elements or groups of elements on a webpage, and the associated styles define how those elements should be presented.

JavaScript

JavaScript (JS) is a versatile scripting language used for adding interactivity and dynamic functionality to web pages. It allows developers to manipulate page content, respond to user actions, and interact with web APIs, making web pages more engaging and interactive.

Backend Technology

ASP .NET

ASP.NET is a web development framework developed by Microsoft. It is a part of the larger.NET framework and provides a programming model, tools, and libraries for building web applications and services. ASP.NET allows developers to create dynamic and interactive websites, web applications, and web services using various programming languages, such as C#, which will be used for developing the system.

ASP.NET offers different approaches for building web applications, and the Web API is one of them. The reason for choosing APIs is to promote a clear separation of concerns between the backend and the frontend. The backend can focus on providing data and functionality through the API endpoints, while the frontend can consume and present the data in the desired format for the user interface.

REST API

For fetching data to the frontend, we are using the REST API architectural style for designing networked applications. REST APIs are widely used for building web services, enabling systems to interact and exchange data in a flexible and standardized manner.

By following the principles of REST, APIs can be designed to be stateless, scalable, and easily consumed by different clients, including web browsers, mobile applications, and other servers for scalability.

Data Design

Data design provides efficient data exchange between clients and servers. They can be optimized for performance and scalability, allowing the application to handle a large number of requests and efficiently manage resources.

This section outlines the design of the database management system (DBMS) files associated with the system. Changes to the logical data model may occur due to software requirements and data traffic.

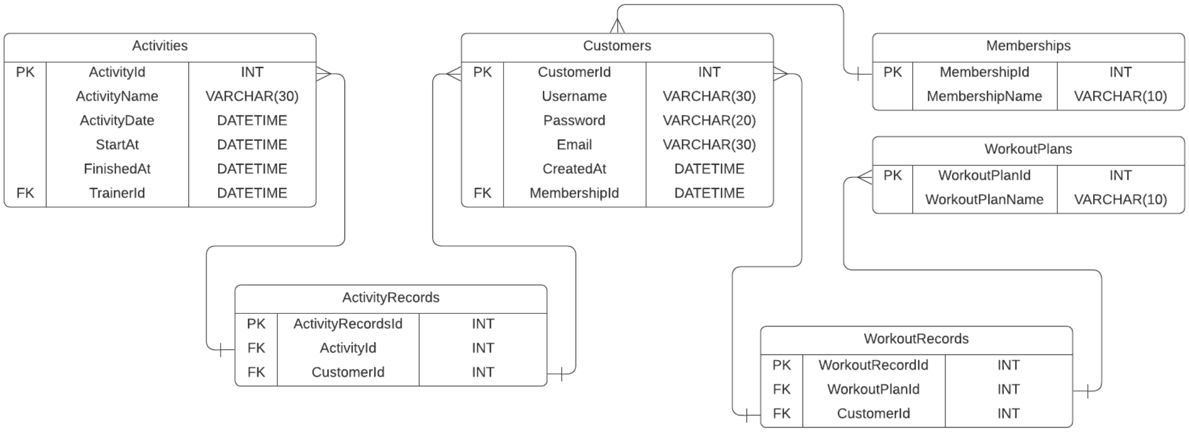
SQLite

SQLite is a popular, lightweight, embedded relational database management system (RDBMS). It is a software library that provides a self-contained, serverless, zero-configuration, and transactional SQL database engine. Unlike traditional client-server database systems, SQLite is embedded directly into the application that uses it rather than running as a separate process.

SQLite has been chosen since ASP.NET includes built-in support for working with SQLite databases. The library can be downloaded from NuGet Packages under the name of Microsoft. EntityFrameworkCore.Sqlite and using the AddDbContext and UseSqlite methods.

SQLite is right for early development to create a database locally on the project to test every functionality that requires getting data from the database, such as user authentication and personal data.

ERD



Activities

PK	ActivityId	INT
	ActivityName	VARCHAR(30)
	ActivityDate	DATETIME
	StartAt	DATETIME
	FinishedAt	DATETIME
FK	TrainerId	DATETIME

Customers

PK	CustomerId	INT
	Username	VARCHAR(30)
	Password	VARCHAR(20)
	Email	VARCHAR(30)
	CreatedAt	DATETIME
FK	MembershipId	DATETIME

Memberships

PK	MembershipId	INT
	MembershipName	VARCHAR(10)

WorkoutPlans

PK	WorkoutPlanId	INT
	WorkoutPlanName	VARCHAR(10)

ActivityRecords

PK	ActivityRecordsId	INT
FK	ActivityId	INT
FK	CustomerId	INT

WorkoutRecords

PK	WorkoutRecordId	INT
FK	WorkoutPlanId	INT
FK	CustomerId	INT

The ERD above describes relationships between entities. **Customers** entity has one foreign key, which is **Memberships** entity as the dictionary for membership types (bronze, silver, gold). There are two tables of features, the **Activities** and **WorkoutPlans** tables. These tables have many-to-many relationships with the **Customers** table, and use records tables as mediators of both relationships with particular names.

API

There are two controllers that handle specific requests from the client side, which are accounts controller and activities controller. Account controller responsible for user management functionalities, such as sign in, sign up, and sign out. Activities controller responsible for activities management. For instance, to get the list of the activities as well as the details. The following are the endpoints, grouped by controllers:

- Accounts controller:
  - http://localhost:5296/api/accounts
    This is an http get method, to get all the accounts, including the user's personal data.
  - http://localhost:5296/api/accounts/user
    This is an http get method, to get the details of a personal account. This endpoint gets called for 'view profile' functionality.
  - http://localhost:5296/api/accounts/signup
    This is an http post method, for new users to sign up before sign in to the website and get full access to the web application.
  - http://localhost:5296/api/accounts/signin
    This is an http post method, for existing users to sign in to get full access to the web application.
  - http://localhost:5296/api/accounts/signout
    This is an http post method, for existing users to sign out after finishing using the web application, to clean the cache
- Activities controller:
  - http://localhost:5296/api/Activities
    This is an http get method, to retrieve all of the activities to display in the frontend.
  - http://localhost:5296/api/Activities
    This is an http post method, to add a new activity.
  - http://localhost:5296/api/Activities/{activityId}
    This is an http get method, to get details of a specific activity.

User’s Manual

Introduction

The Gym Application Website is a web-based application that allows users to log in, sign up, and access various functionalities related to gym activities. This user's manual provides an overview of the implemented application and guides users on how to effectively navigate and utilize its features.

System Requirements

To access and use the Gym Application Website, users will need the following:

- A computer or mobile device with an internet connection
- A compatible web browser (e.g., Google Chrome, Mozilla Firefox, Safari)
- Accessing the Application:
  To access the Gym Application Website, follow these steps:
  1. Open your preferred web browser.
  2. Enter the URL of the application's website in the address bar.
  3. Press Enter or click the Go button to navigate to the website.

User Registration:

If you are a new user, you need to register an account to access the application's functionalities. Follow these steps to register:

1. On the application's home page, locate and click on the Sign-Up button.
2. Fill in the required registration details, such as your name, email address, and password.
3. Click the Submit or Register button to create your account.
4. Upon successful registration, you will receive a confirmation email with further instructions.

User Login:

If you are a registered user, follow these steps to log into the application:

1. On the application's home page, locate and click on the Sign In button.
2. Enter your registered email address and password in the provided login fields.
3. Click the Login or Sign In button to access your account.
4. Application Dashboard:
   a. Once logged in, you will be directed to the application's dashboard. The dashboard provides available functionalities. The specific features and

options may vary based on the implementation of the Gym Application Website.

Exploring Functionalities

Within the Gym Application Website, you will find various functionalities that enhance your experience. These may include:

- Viewing the list of available gym activities, classes, or programs.
- Enrolling in specific activities or classes.
- Tracking users progress and attendance.
- Accessing personalized workout plans or diet recommendations.
- Interacting with other users or trainers through messaging or forums.

Navigating the Website

To navigate through the Gym Application Website, use the provided menu or navigation bar. These elements typically include links or buttons that direct the users to different sections or pages within the application.

Data Privacy and Security

The Gym Application Website prioritizes the privacy and security of user data. All personal information, including passwords, are handled with utmost care and security measures. The website employs password encryption techniques to ensure the confidentiality of user passwords. This means that when users create an account or update their password, the passwords are encrypted before being stored in the database.

Additionally, the Gym Application Website follows best practices for password security, such as enforcing password complexity requirements (e.g., minimum length, combination of uppercase and lowercase letters, numbers, and special characters). These measures further enhance the security of user accounts and mitigate the risk of unauthorized access.

Rest assured that the Gym Application Website is committed to safeguarding user privacy and adhering to applicable data protection regulations, including the secure handling and storage of user passwords.

Support and Assistance

If users encounter any difficulties or require assistance while using the Gym Application Website, refer to the provided Help or Support section within the application. Users may find FAQs, contact information, or a user forum where users can seek further guidance or report issues.

Conclusion

The Gym Application Website aims to provide users with a convenient platform to access gym-related functionalities and enhance their fitness journey. This user's manual serves as a guide to help users navigate and utilize the application effectively. Enjoy your experience and make the most of the available features to achieve users fitness goals.

Prototype

Database and API

https://github.com/brianabeltimothy/LightweightGym

Frontend

https://github.com/zhvnvrys/lightweight_gym_frontend/tree/main/Lightweight_Gym_Frontend