Capstone Project : Coach-Space

Introduction

Purpose

What is the problem or the opportunity that the project is investigating?

The problem this project is investigating revolves around making exercise planning and management more efficient for coaches working with a diverse range of clients. Coaches often handle multiple clients simultaneously, including individuals, pairs, groups, and classes. This project aims to streamline the process of assigning weekly exercises, scheduling future plans, and presenting these exercise routines to clients in a user-friendly manner.

Why is this problem valuable to address?
Addressing this problem is essential because it enhances the exercise experience for clients. By minimising distractions, it becomes easier for clients, especially children and competitive individuals, to stay engaged and committed to their exercise routines. The current state of exercise management often results in clients procrastinating or losing motivation, leading to missed goals and slow progress.

What is the current state (e.g. unsatisfied users, lost revenue)?

The desired state is to create a platform that simplifies exercise planning and tracking while minimising distractions for clients. There are many existing projects in the fitness industry, such as running platforms like Nike, which incorporate leaderboards for tracking progress but there is none for badminton. This could potentially fill that gap but it would have to be approached through a specific niche.

Industry/ domain

The project operates in the competitive exercise and sports industry, driven by demand for personalised fitness programs.

Stakeholders

Who are the stakeholders?
The primary stakeholders are **users** and **coaches**. **Users** benefit from a secure and accessible exercise tracking system, ensuring their assigned exercises can be easily accessible. **Coaches** rely on the software for exercise management, progress tracking, program creation and user management.

Stakeholders' Interests:
Users expect a secure platform that safeguards their assigned exercises and ensures the privacy of their programs. They value accessibility and a user-friendly interface. Coaches, on the other hand, seek efficient exercise management tools, progress monitoring, and administrative capabilities

Product Description

Architecture Diagram

See Appendix A

User Stories

#	User Story	Available action to satisfy	Priority
1	"I have multiple students in the same group, I need to assign the same exercises to them"	**Coach**: Coaches can assign multiple exercises to multiple users simultaneously through the data grid. Users can be filtered according to their group.	High
2	"I need to search the specific exercises to assign"	**Coach**: There is a search button in the data grid, that helps filter the exercises according to the category wanted or search the exercise specifically.	High
3	"I want to be proactive about my client's program and set up a program for them for the next four weeks, but I don't want them to see it."	Coaches can set multiple exercises with specific date ranges as to when the client needs to execute them by specifying a start date and an end date. The clients will only be able to see exercises that are assigned to them for those assigned dates. If they are not within the assigned dates, the client will not see the exercises.	High
4	"I want to see if my other group members are doing their exercises"	The leaderboard shows users who are in the same group as the user who is logged in. They will be able to see the progress of other group members, depending on what they have logged during the week. They are not able to see other users' progress if they are not in their group.	High
5	"I have come up with a new exercise and I need to see if this exercise already exists for the footwork category."	Coaches can filter their search in the exercise category and add a new exercise if it cannot be found in the grid.	High
6	"A student no longer wants to be part of a group and wants to train individually instead."	The coach can edit the user in the User-Admin page that only coaches can access. Here the coach can edit the user and leave the group section as null.	High

7	“I want to assign weekly exercises to a group but each member will have different sets of exercises”	In the Assign Exercise page, coaches can select exercises to a specific user. This will not be displayed to other members in the group, just the specific user.	High
8	“I see huge improvements in this student and I believe he can now take his own clients.”	Coaches can edit the user’s role in the User Grid.	
9	“I want to talk to my group members on how hard these exercises are”	Not implemented into the project. The idea is to add a comment section or discussion forum that can only be seen by the group.	Low

User Flow

See Appendix C

Entity	**Action**	
User	➔ Create an account ➔ Update points on the leaderboard ➔ View leaderboard ➔ View exercises	➔ Users can view the exercises assigned by the coach. ➔ Assigned weekly exercises are displayed to the user on a leaderboard based on the provided date. ➔ Users can add points to the leaderboard.
	Limitations:	➔ Users can create an account. ➔ Users can edit their account details. ➔ Users can update their points on the leaderboard.
Coach	CRUD - Create, Read, Update, Delete ● Users : CRUD ● Exercises: CRUD ● Assigned Exercises: CRUD	➔ Coaches can assign multiple exercises to multiple users in bulk by selecting specific exercises and users. ➔ Coaches have CRUD (Create, Read, Update, Delete) capabilities for exercises, allowing them to manage exercise data.

		➔ Coaches also have CRUD capabilities for users, enabling them to manage user accounts.

Wireframe Design

Key pages were designed with wireframe designs on Figma.
Changes have been made between the initial design and the final product. This is due to better functionality from resources found during the project. Further explanation is provided in project planning

See Appendix B

Open Questions/Out of Scope

- Optimisation for mobile: The data grid that I am currently using displays too much valuable information to be removed.
- Adding a live comment stage for the group leaderboards.
- Having a “student” view from the coach’s account
- Redux implementation (need more time)
- Displaying exercises in group / history (need more time)

Non-functional Requirements

- **What are the key security requirements? (e.g. login, storage of personal details, inactivity timeout, data encryption)**

  Users sign up with an email, username, and password. All data created during the sign up process is stored in the SQL database server as plain text except for the password. The password is being encrypted and cannot be viewed by anyone in the backend.

- **How easy to use does the software need to be?**
  Very easy for users as the purpose is to minimise distractions for the users to perform their exercises.
  Coaches should also need to easily navigate themselves within pages to create, users, exercises and assign exercises.

Ease of Use

Consistent formatting of data throughout the application.

Using the same format of data grid, with the same buttons to help users simply understand use.

- **How quickly should the application respond to user requests?**
  Relatively quick as the scores in the leaderboard are supposed to update on every click.

  Users need to see assigned exercises made by coaches as soon as possible.

- **How reliable must the application be? (e.g. mean time between failures)**
  Coach-Space needs to be reliable almost at all times possible. Users should be able to access their programs at any point in time. Coaches should also be able to create programs at any point in time.

- Does the software conform to any technical standards to ease maintainability?

  Backend
  - MVC Structure.

  Front End
  - Folder structure made to easily track components that belong to a certain functionality (eg. User folder, Exercise folder)

Project Planning

Each component that was required to achieve the functionality of the project was divided into tasks to help stay on track for completion. Each task was divided into front end and front end.

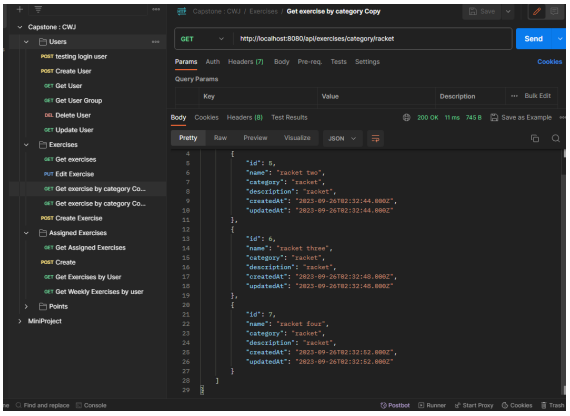
Trello is used to track these tasks. (*See appendix D)*

The design process for the app started off by defining the main features and components for Coach-Space. This was then mapped through to the pages to determine which component will appear on a specific page.

The UI/UX planning process was done with a LOFI diagram on Figma. Having a rough idea on which main colors will be used for the application. There have been a few changes made as the project progressed since there were resources that showed better functionality than my original plan. After some UI research, keeping it simple keeps the user's attention. Therefore the colour scheme also differs from the original planning (See Appendix B)

Testing Strategy

Backend
All requests for each feature were being tested through postman. Using HTTP requests from Postman to help see if routes were performing the functionality that was needed when called.



Frontend
The client must be able to easily navigate and interact with each feature seamlessly. When UI is interacted with, it must perform the function that is being enforced in the backend. (eg. clicking the exercise that has been ‘completed’ and successfully adding the points to the correct user for the leaderboard)

How did you handle edge cases?
Displaying exercises at the wrong time
These were tested in the backend through postman to ensure that the correct data was being displayed before being integrated to the front end.

Adding points to the wrong user for the leaderboard
Points are dependent on the current user that is active. Points can only be added to the current user that is found within the User Context. If there is no user signed in, the page with the leaderboard is inaccessible. If a user does manage to access the page, points can only be deployed if user context is not null.

Leaderboards are also dependent on the current user. As it is only populated based on the current user’s information. This was tested by deploying multiple users in different groups and roles, to ensure that the leaderboard was displaying correctly.

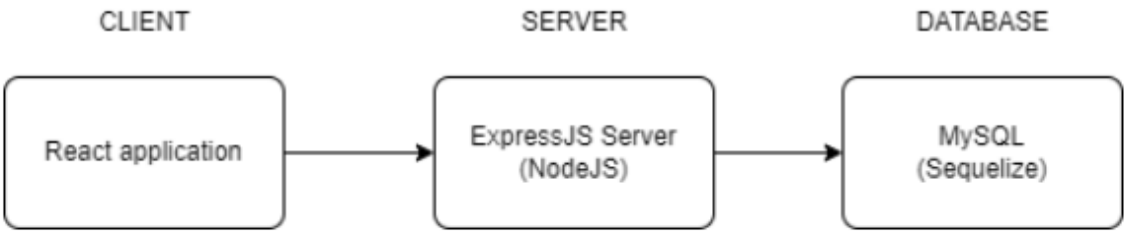
Assigning exercises in dates that could potentially collide.
Users will not be able to differentiate the overlap of the exercises. Currently no solution due to time constraints but exercises can successfully be displayed.

Implementation

The Coach-Space application uses a React application for the client,

ExpressJS server running in NodeJS for backend

MySQL for the database.



End-to-end solution

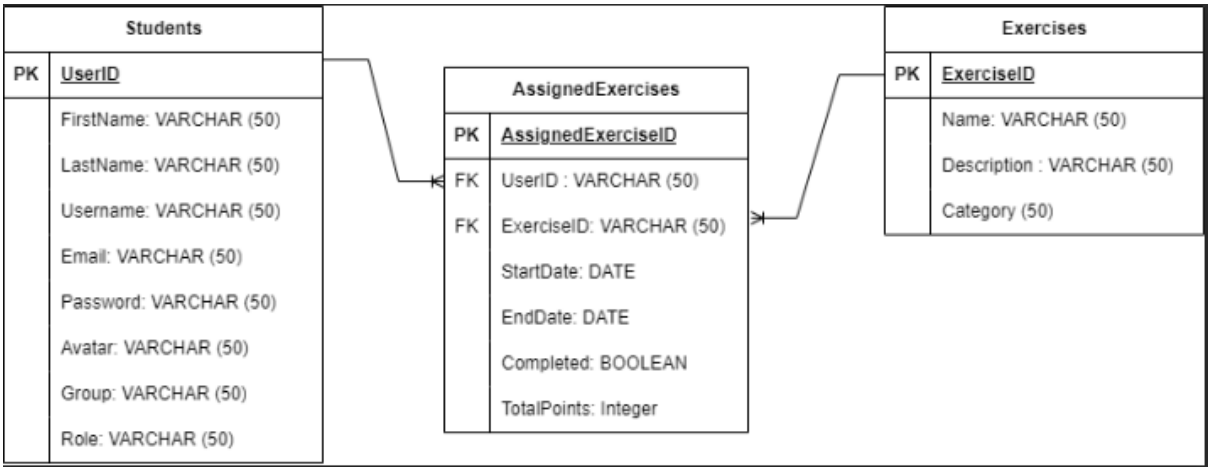
The final product and deployment met all the main requirements for application. A few out-of-scope features have not been implemented but did not disrupt the usability of the application.

References

- React
- MUI
- Axios
- Express JS
- Jsonwebtoken
- Bcrypt
- MySQLServer

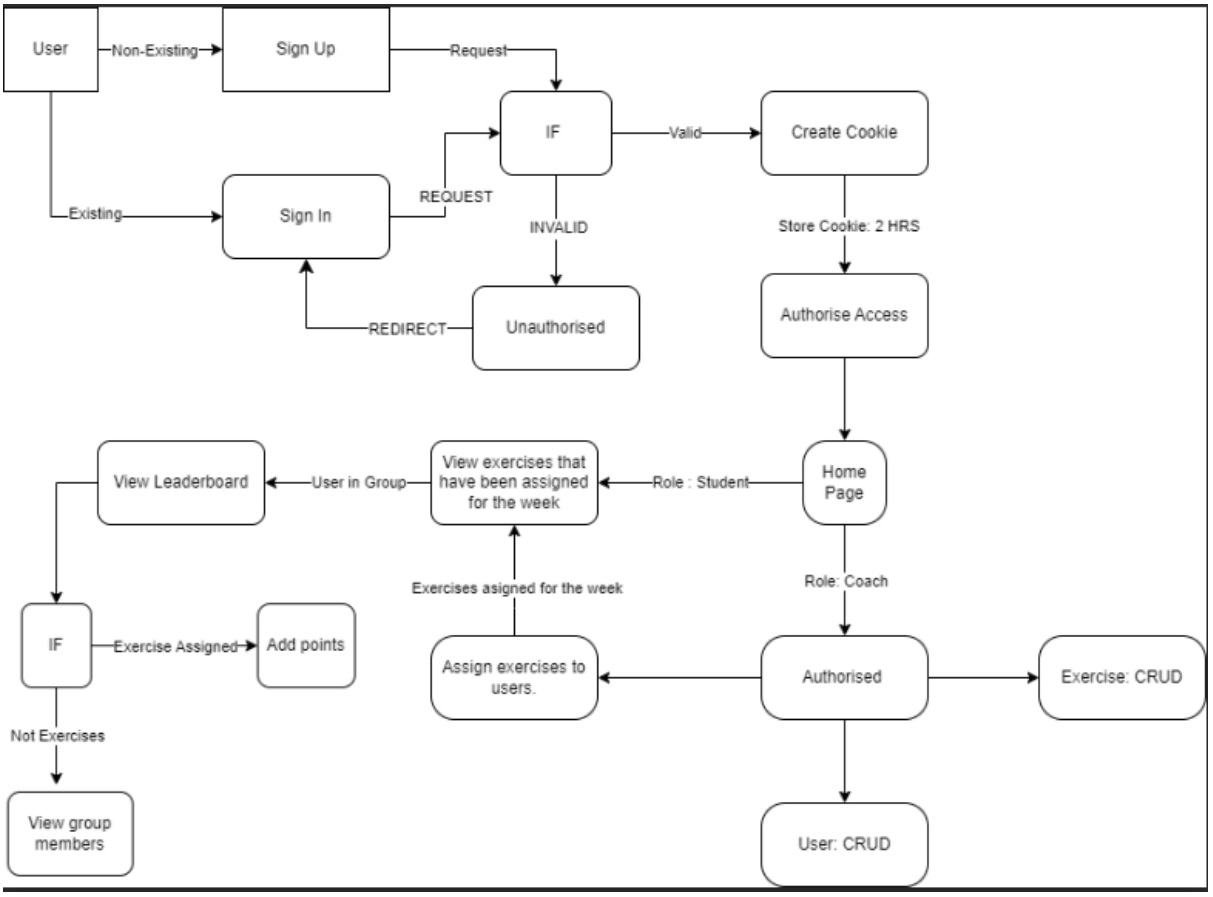
Github: https://github.com/villegasjustine/capstone.git

Appendix A



Appendix B

Appendix C



Appendix D