# 

**CN5006 2324 (T1) Web and Mobile Application Development**

|  |  |
| --- | --- |
| Module code: | CN5006 |
| Module Title: | Web and Mobile Application Development |
| Group |  |
| Names | Resham Adhikari, U2428505  Student 2 (Name, ID)  Student 3 Name ,ID |
| Case Study Title | Report Writing for Back End and Front End |
| Module leader | Dr Nadeem Qazi |
| Assignment Title | Group Report Assignment |

* **Contents**
  + **Introduction……………………………………………….** 
    - **Aims/Objective…………………………………………**
    - **Project Plan…………………………………………….**
* **Methodology…………………………………………………**
* **Development tools……………………………………**
* **Web browser………………………………………….**
* **Sub Task and Sample codes………………………….**
* **Design choices…………………………………………………..**
* **Reflection.…………………………………………………………**
* **Analysis of the work…………………………………….**
* **Survey/Feedback………………………………………..**
* **Conclusion………………………………………………**
* **Link of the website………………………………………………**
* **References……………………………………………………….**
* **Introduction**

The Module **CN5006** has an assignment of website development (Backend and Frontend) using the React for user-interface. Following the criteria, we have developed a site for a Football data containing various data (Teams, Games Played, Win, Draw, Loss, Goals For, Goals against, Points, Year)

**Aims/Objective**

As we studied the React in this term module CN5006, a group project is assigned to design a web server application consisting of backend and front end ,using node js express , mongo dB and mongoose to design the RESTAPI by using the React on the basis of knowledge that we gained while studying this module. As we assigned, we are provided the Football CSV file to develop a web server application. We believed that it will help us to horn our skills on developing and designing the webserver.

**Project Plan:**

We (our group) had different virtual and physical group meetings to discuss about the coursework and we finalize the role of every member’s. As we have to develop and design the Back end and Front end, we are divided into two groups i.e. Back End designing will be led by Resham Adhikari and Aashutosh will help in Back end meanwhile, Saugat Tamang will lead the Front End while other remaining two members will work in Front End. The Report writing and Video presentation will be done by discussing and every member will contribute on what they have done in project.

-Dividing the structure for project

We break down the project into small logical components/modules that helped for better understanding of the project. It is like dividing the code into layers. We started to work on every element and the queries that helps to complete the task in a better way. Example: Firstly, we work on the connecting the data in MongoDB and secondly, we started working on VS code to create a web server. In each section, we focused on only one element at a time.

* **Methodology**
* **Development tools**

As we studied React to develop a web server and MongoDB for the data collection, We use React coding in the Visual Studio (VS) code for the Back End and MongoDB for the data collection (FootballCSV). We use the **Netlify** to create and host the link for the website.

* **Web browser**

While developing the webserver, we checked and used in different web browser. During designing, we used the Microsoft Browser and Google chrome. After the completion of web designing, We checked the webserver in the Brave too to find out its output and it works properly in all the web browsers that We have used.

* **Sub Task and Sample codes**

Sub Task 1.1

To complete this subtask, we have created a MongoDB collection called FootballData to hold the data from the given CSV i.e. FootballCSV file which is running in local host: 27017 of MongoDB as shown in the Snapshot below.

A screenshot of a computer

Description automatically generated

SubTask 1.2

I have created a mongoose Schema in a separate folder Modles called Football.js for this dataset and from this Schema I have created a model to be used in add, delete, update to find the queries given in other sub task. Here, the line require(‘mongoose’) help to imports the Mongoose library, which is used to interact with MongoDB in a Node.js. Each field is defined with its type and, if applicable, additional constraints (e.g., required: true) like, Team:{ type: String}, requuired: true. This data will exports a Mongoose model named ‘Footballmodel’ based on the ‘FootballScheme’ schema. The model is associated with the MongoDB collection named ‘FootballData’. The coding of given query is given below:

A screen shot of a computer

Description automatically generated

Sub Task 1.3

To connect the file that uses mongoose Schema into MongoDB, I have created a js file called Connection which is connected to the URI of Mongoose by the require function as shown in the given picture below.

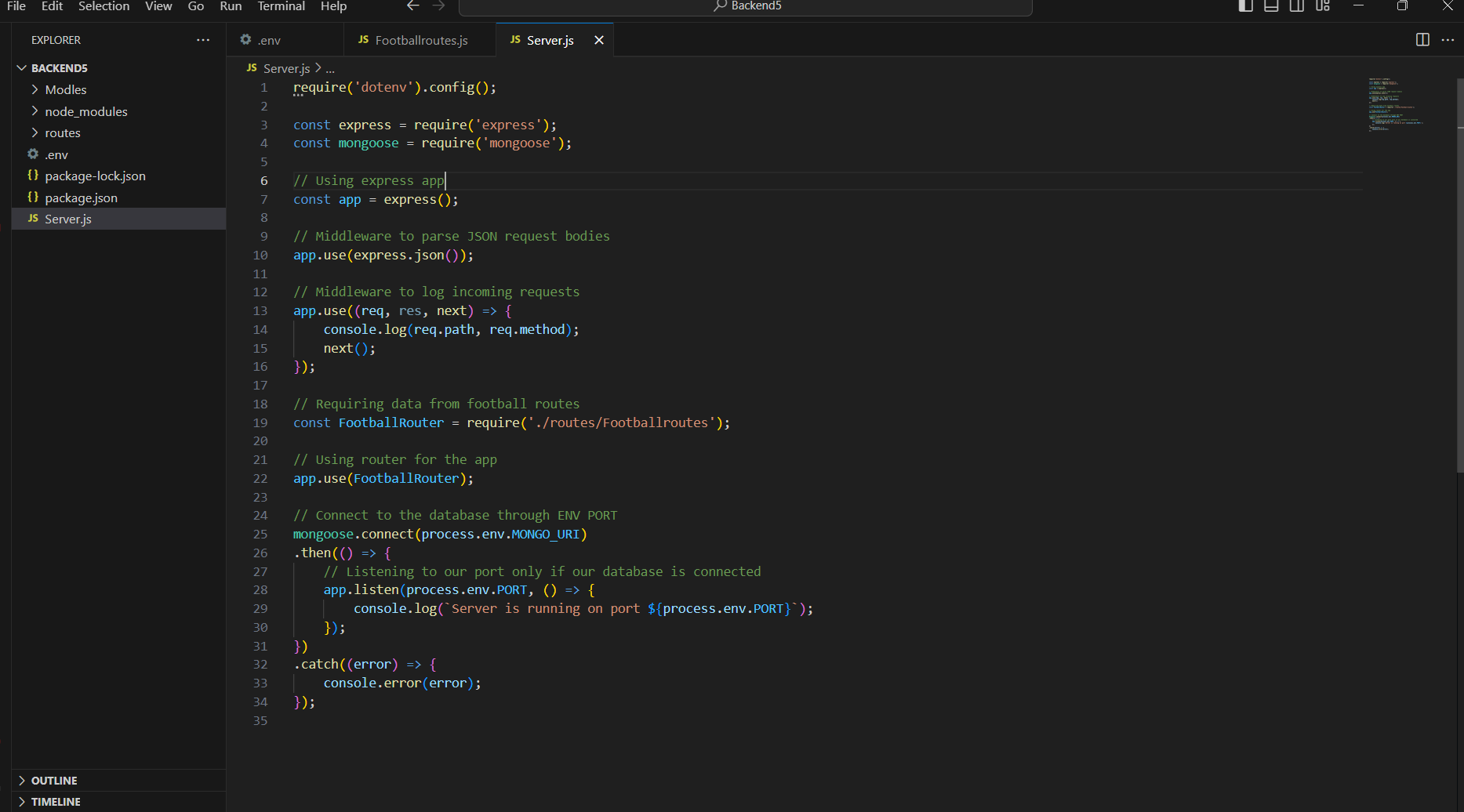
A screenshot of a computer program

Description automatically generated

Sub Task 1.4

We created a separate JS file (server.js ) that runs a web server using express and import

the files created in 1.2 and 1.3. This code sets up an Express web server using the express framework. It utilizes middleware for JSON parsing and request logging. The server connects to MongoDB by using the URI and starts working on the specified port.



This Server will run in port 5000 and the code for this is given below: A screenshot of a computer

Description automatically generated

Sub Task 1.5

This POST method of the REST API is able to Add the data to the Football collection in mongoDB like { Team, Games Played, Win, Draw, Loss, Goals Against, Points} as shows in given code.

A screenshot of a computer program

Description automatically generated

Sub task 1.6

A screenshot of a computer

Description automatically generatedThis POST method of the REST API is able to update the data to the Football collection in mongoDB like { Team, Games Played, Win, Draw, Loss, Goals Against, Points} based on the provided team identifier and the new data. If there is an error during the update operation, the server responds with a status code of 500 and a JSON object containing an error message.The source code for updating a data is given below.

Sub Task 1.7

Using POST method for deleting record for a given Team:

This delete route, defined at the endpoint /deleteData/:id, is designed to handle requests for deleting data from the Football collection in MongoDB. It uses findbyidand delete function to delete the item by searching from their ID. If there is an error during the deletion operation, the server logs the error to the console, responds with a status code of 500, and sends the message 'Failed to delete team'.

A screenshot of a computer

Description automatically generated

Sub Task 1.8

A screen shot of a computer

Description automatically generatedThis get route, defined at the endpoint ‘/stats/:Year’, retrieves aggregated statistics for a specific year from the Football collection in MongoDB. Here, year is mandatory to find the dara while Team Name is optional where user can find the stats as they want.

Sub Task 1.9

A screenshot of a computer

Description automatically generatedThe following code will display first 10 record from the Football database (display all the Teams including all nine columns) for match “won” greater than a value entered by the user .

Sub Task 2.0

This GET route, defined at the endpoint /AverageGoal/:Year, retrieves aggregated statistics for each team in the Football collection based on the average "Goal For" for a given year. The code for following Query is attached below.

A screen shot of a computer

Description automatically generated

Frontend:

Step 1: To initiate a new React project that comprises components (which can be either functional or class-based) for creating the user interface and accessing suitable REST API methods using Axios, a commonly used tool is npx create-react-app. This command, as shown in figures F.1 and F.2, sets up a project named frontend5. The project includes a functional component that utilizes Axios for interfacing with the necessary API, effectively handling user interactions and data retrieval.

fig F.1**A screenshot of a computer

Description automatically generated**

fig F.2A screenshot of a computer program

Description automatically generated

Sub Task 2.1

To Add Data

To set up a component that facilitates user interaction through a form interface for entering data such as Team, Games Played, Wins, Losses, Goals For, Goals Against, Points, and Year into a MongoDB database, I utilize figures F.3 and F.4 as references. This component, integral to the frontend, is designed to activate backend functions and communicate with the server using Axios for API calls. The figures mentioned (F.3 and F.4) include detailed comments to aid in understanding the component's functionality and its role in the data flow from the front-end to the back-end, ultimately leading to data storage in MongoDB.

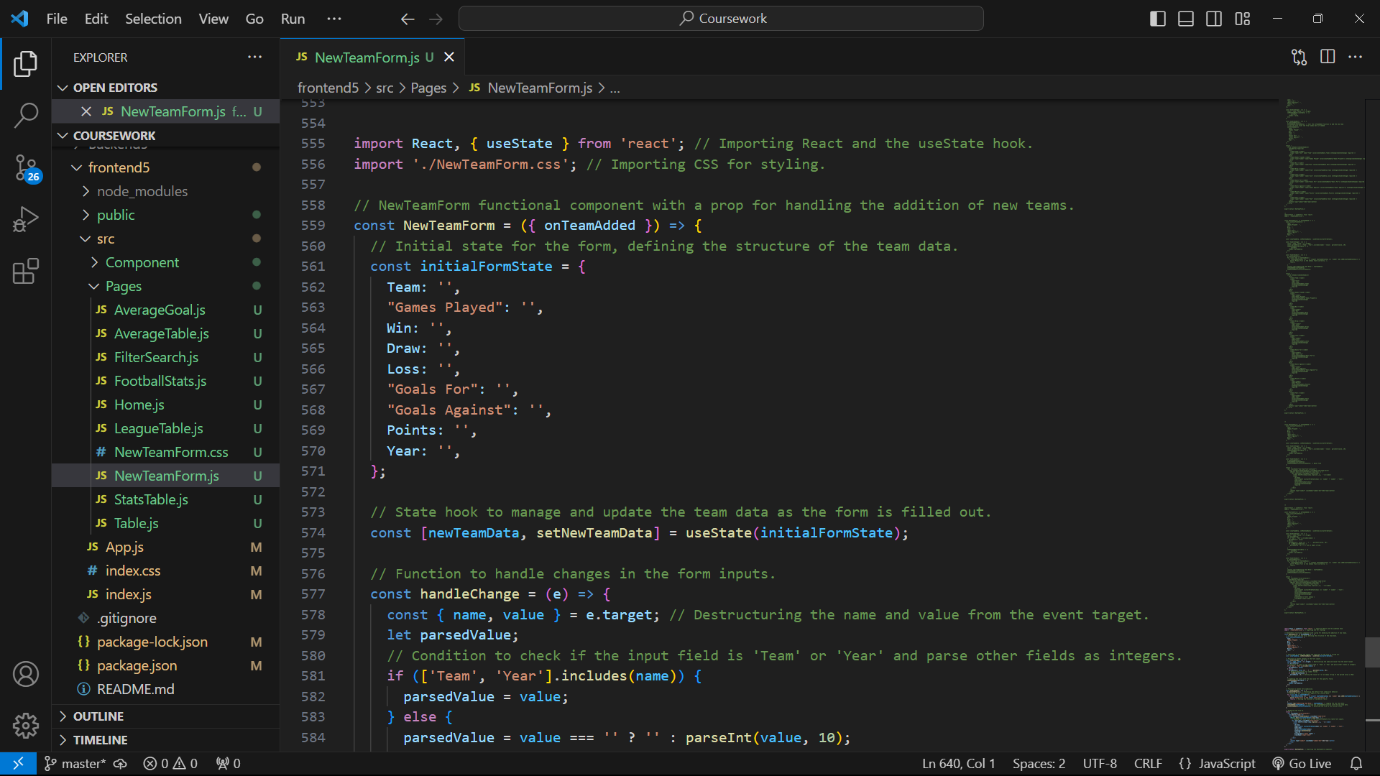
fig F.3****

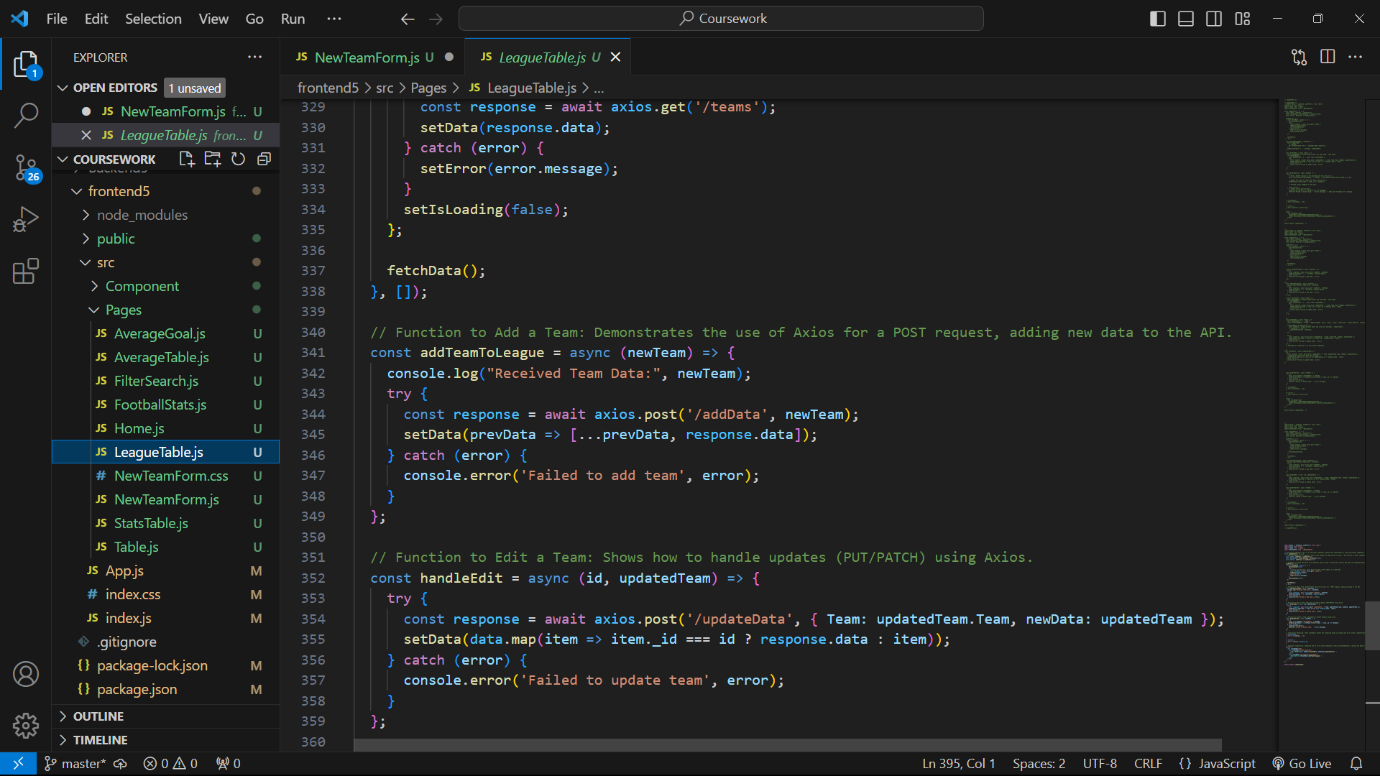
fig F.4

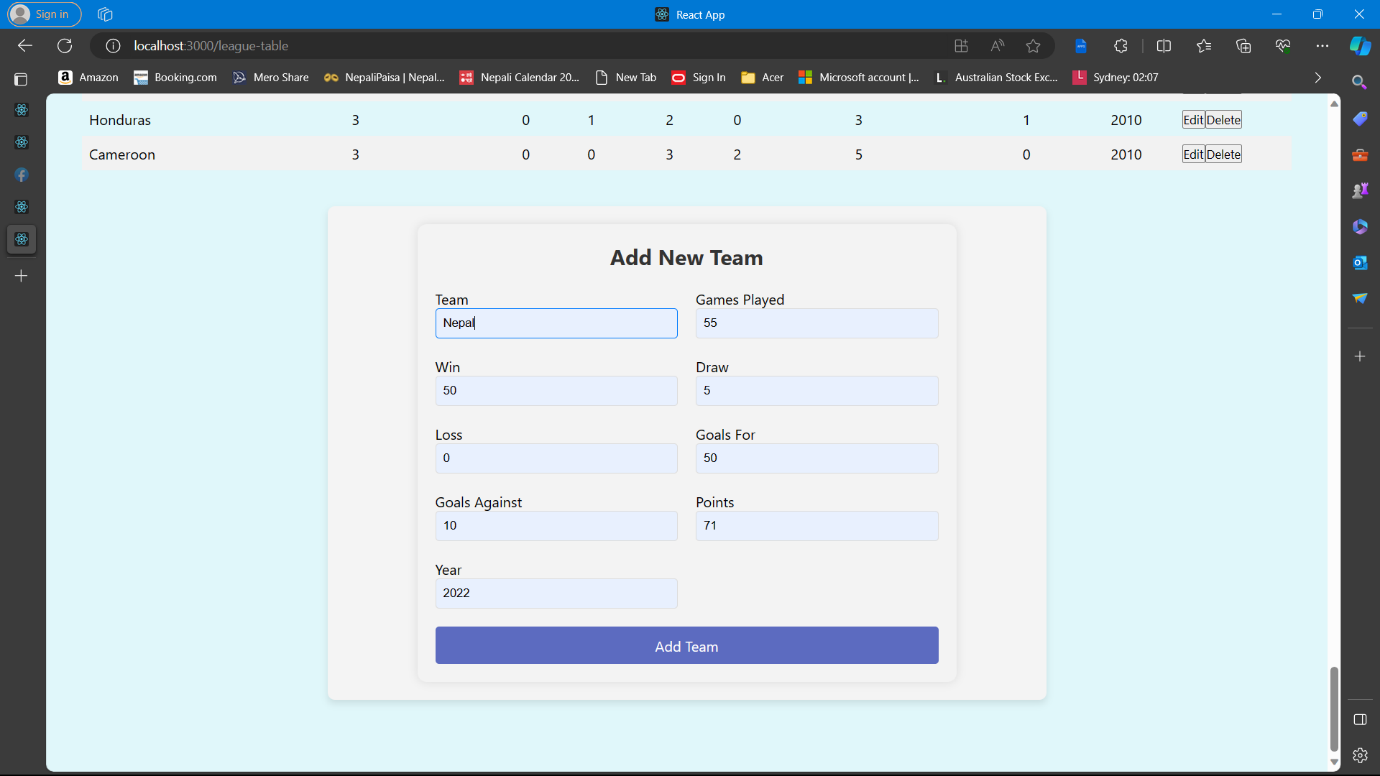
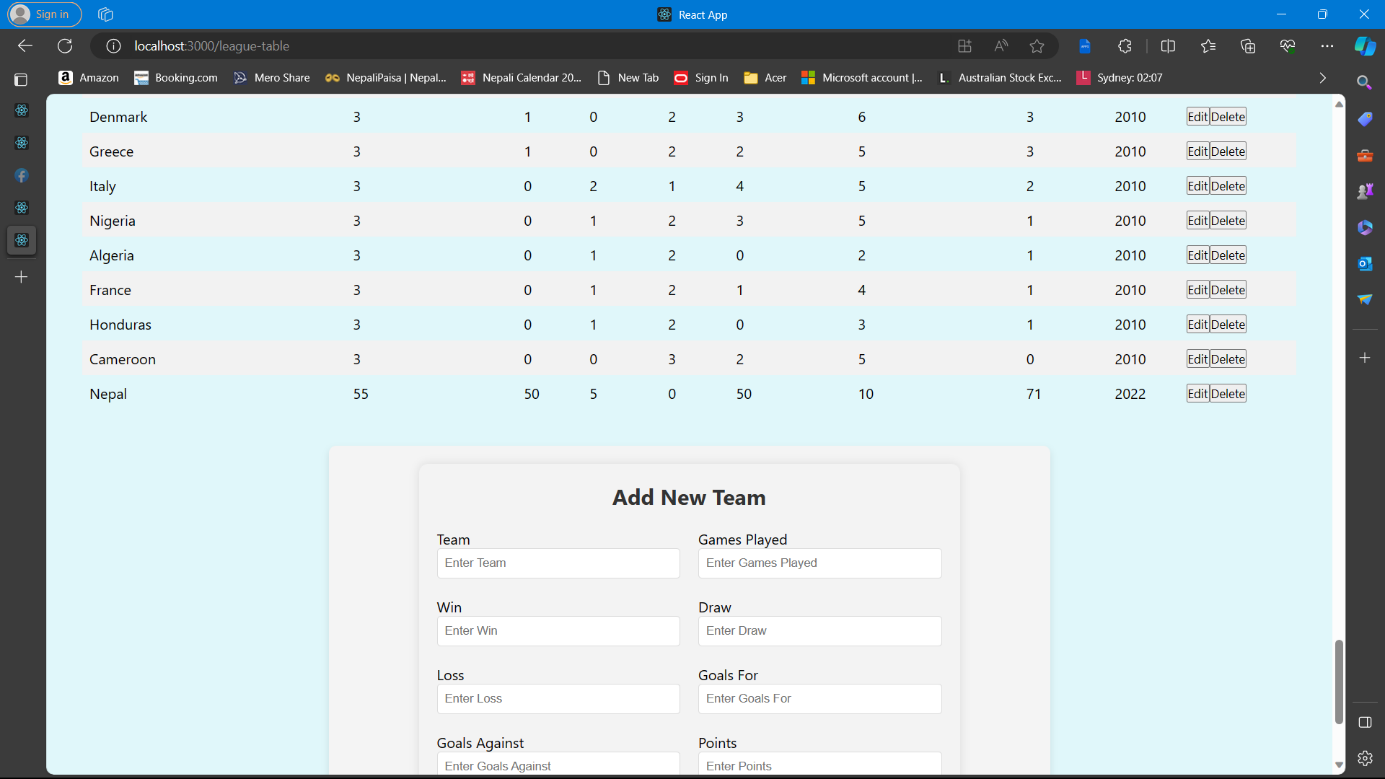
fig F.5****

fig F.6

Sub Task 2.2 and 2.4

To Update the Data and Delete the Data

To establish a component that enables user interaction via a form interface for inputting data like Team, Games Played, Wins, Losses, Goals For, Goals Against, Points, and Year and removing data related to a specific team, I refer to figures F.7, F.8 F.9, F.10 F.11, F.12 for guidance. This component is crucial for editing purposes and deleting and communicating with the server. The detailed comments in figures F.7, F.8 F.9, F.10 F.11, and F.12 help clarify the component's functions and demonstrate how it facilitates data flow from the front-end to the back-end, resulting in the updated data being stored and remove in MongoDB

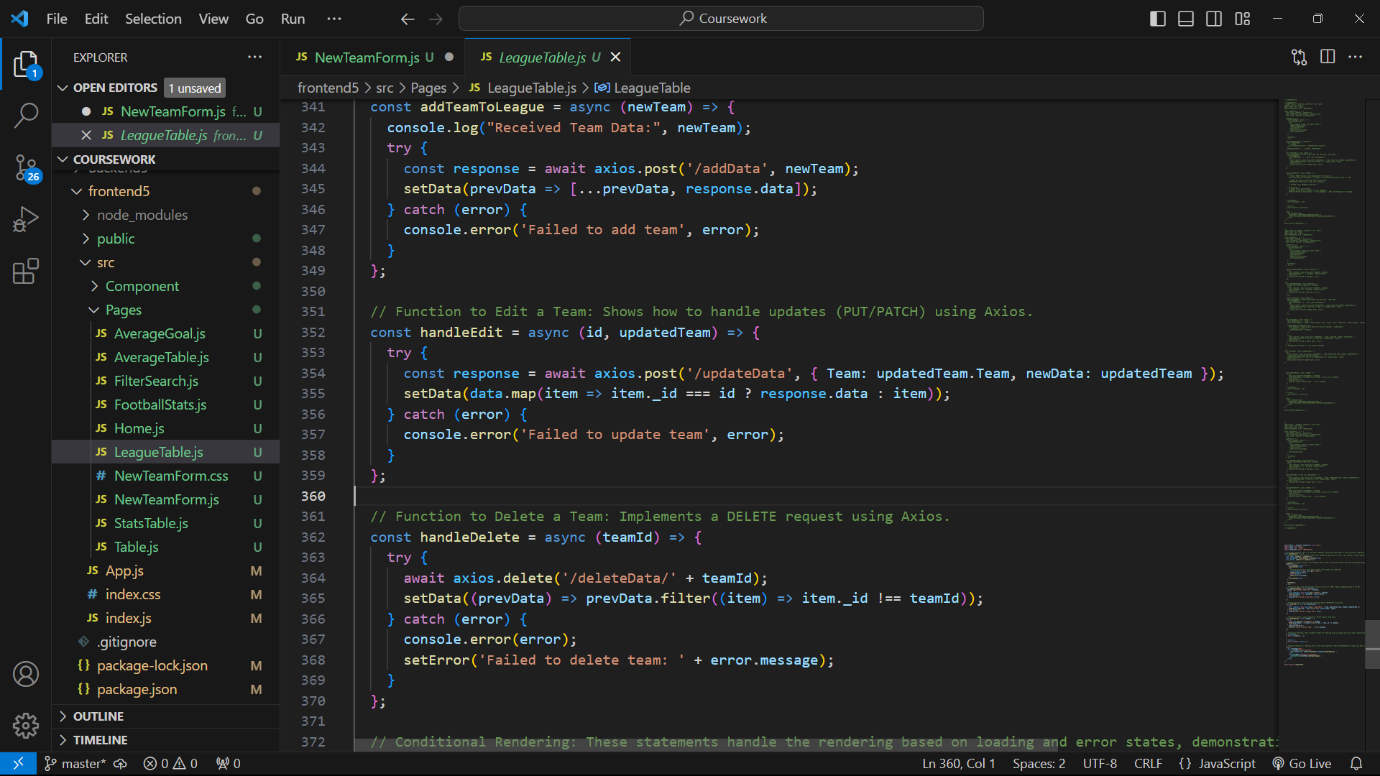
fig F.7****

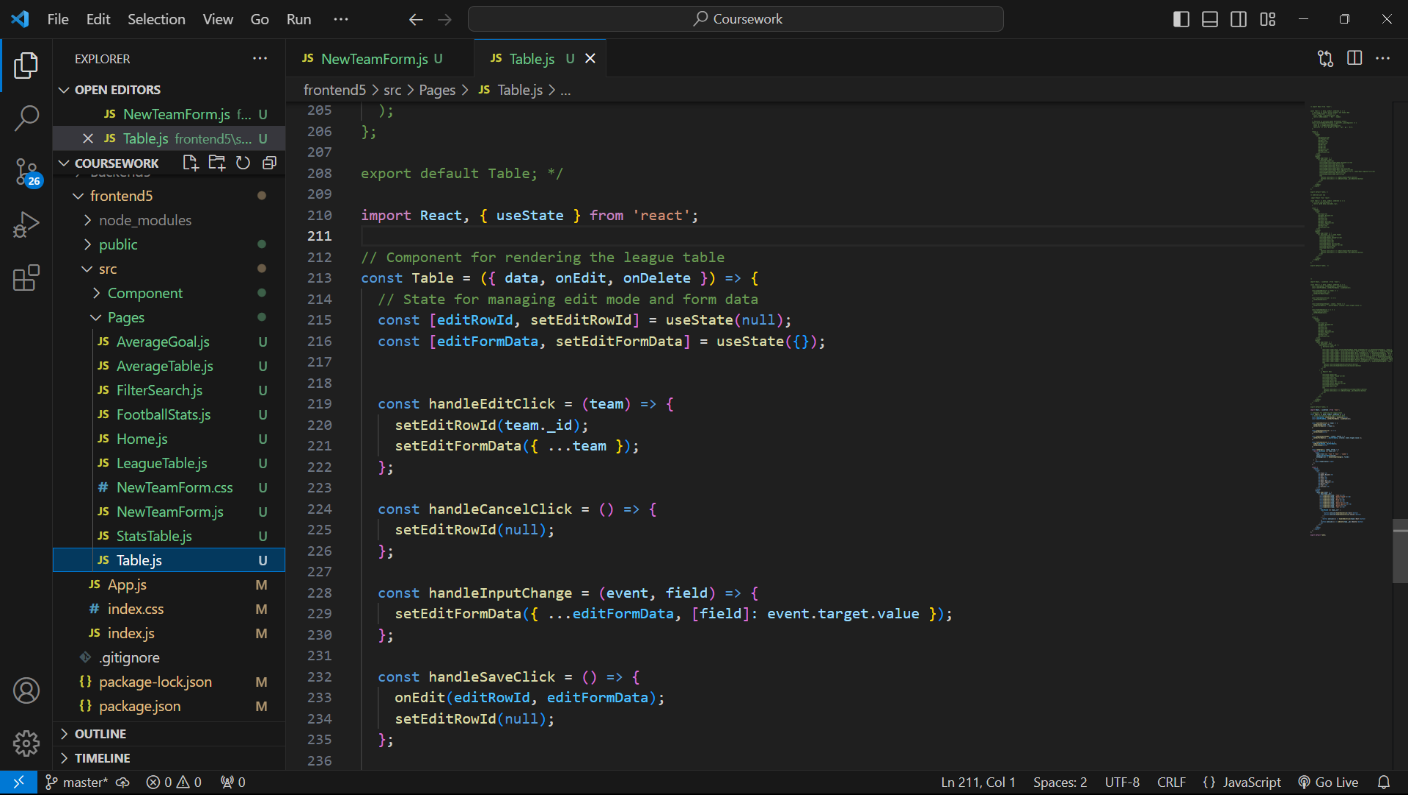
fig F.8

fig F.9

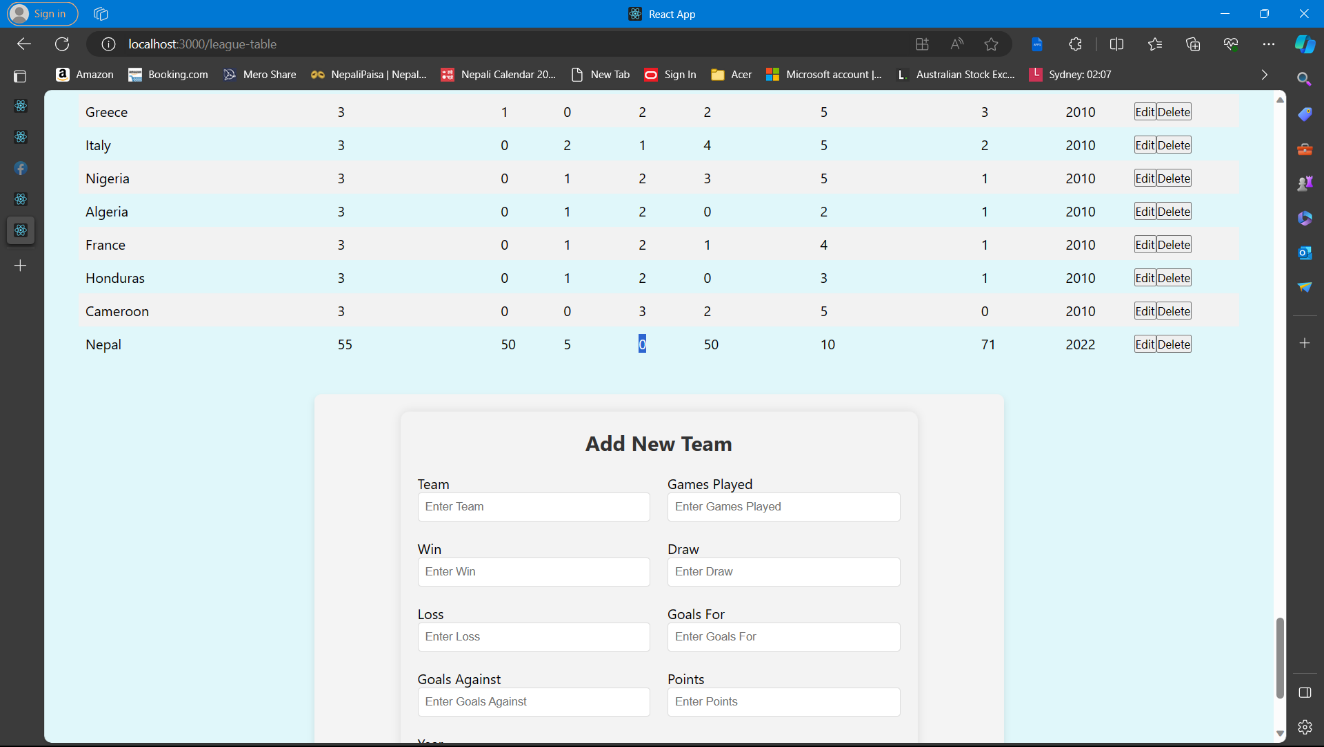
****

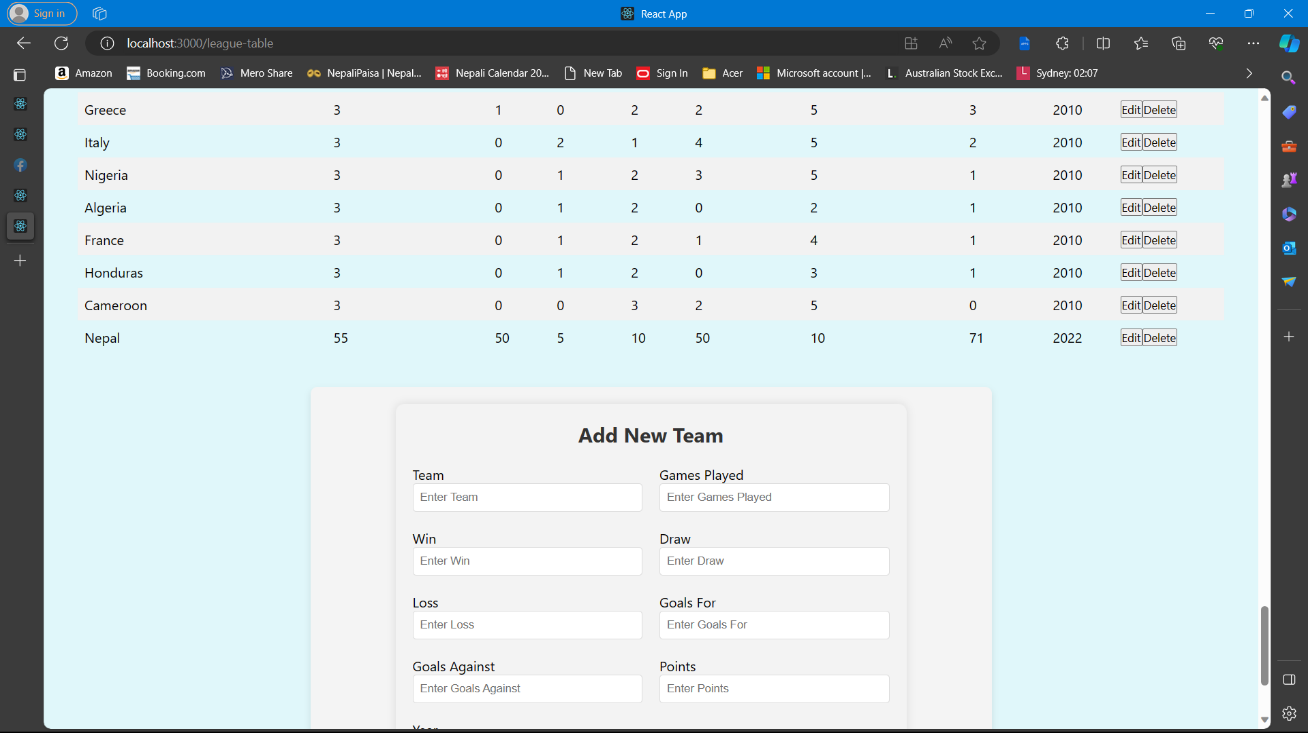
fig F.10

fig F.11A screenshot of a computer

Description automatically generated

fig F.12A screenshot of a computer

Description automatically generated

Sub Task 2.3

**Season Overview: [Team Name] [Year] - Games Played, Draws, Wins, and More**

According to the query of 2.3, I have implemented the user interface and function so the user can search specific team name data or specific year of that time which is total game played win draw and many more Additionally I also added the function so the user can search the total game played of a team or of that year or with both with team name and the year of that give a team name as you can see from the fig F.10, F.11 F.12 F.13

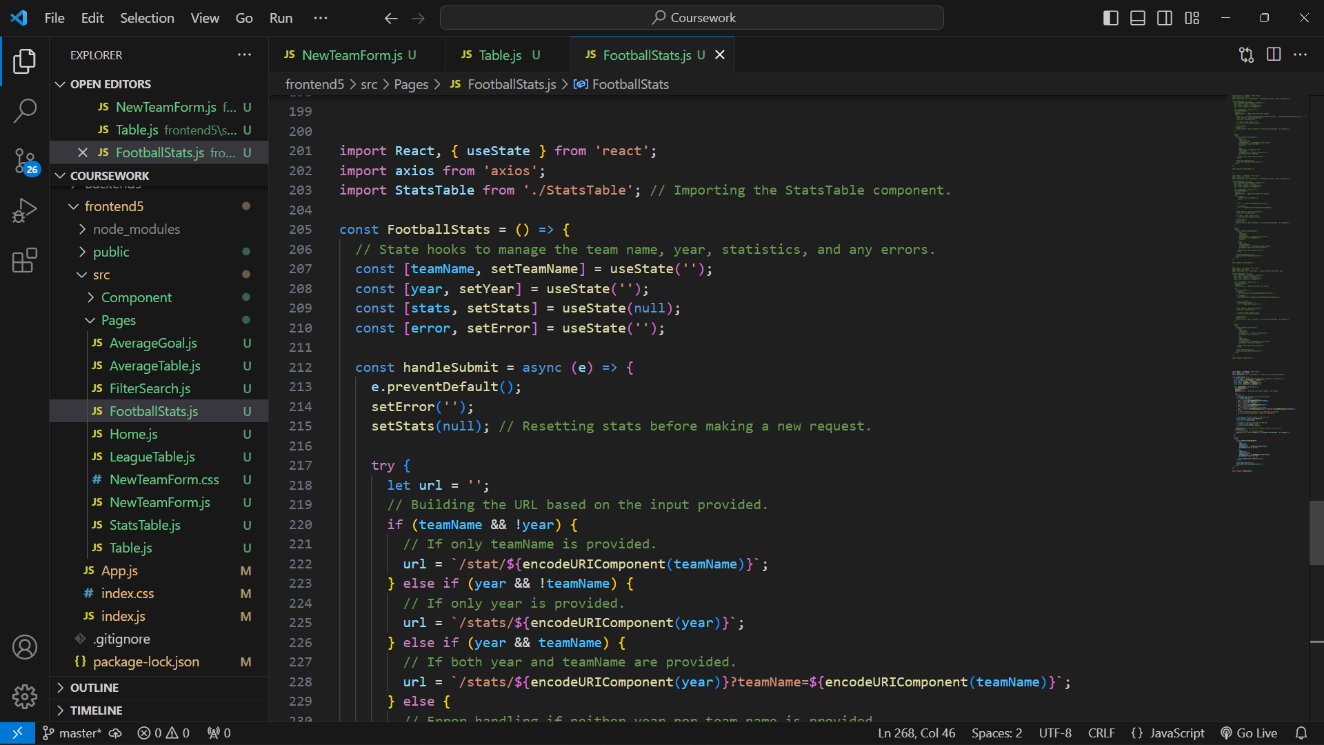
fig F.11****

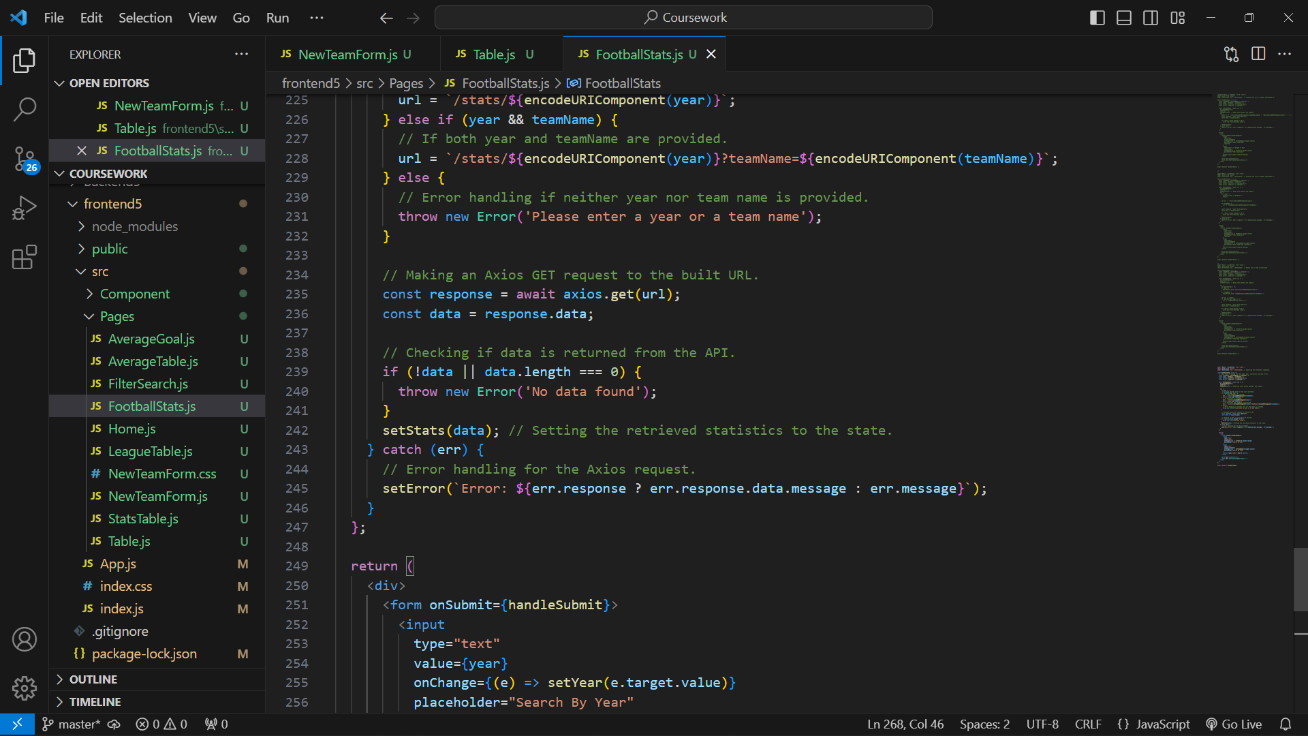
fig F.12

fig F.13

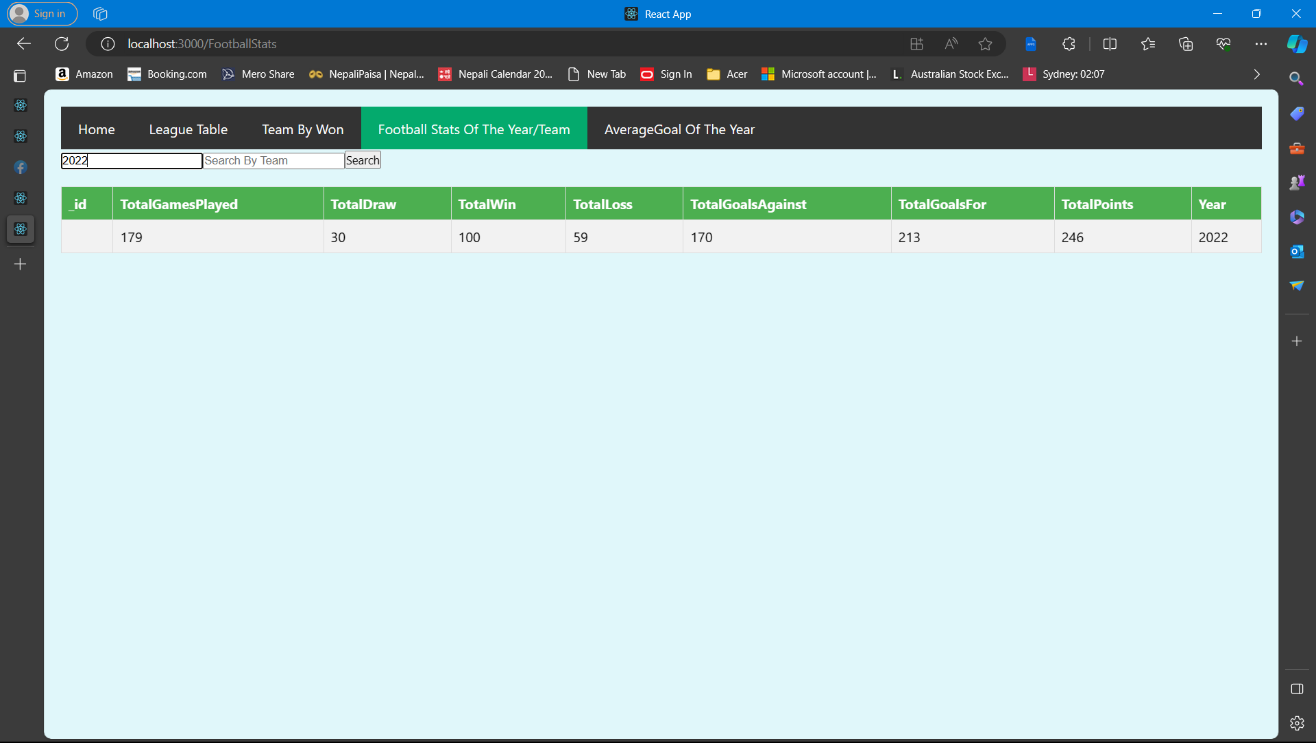
****

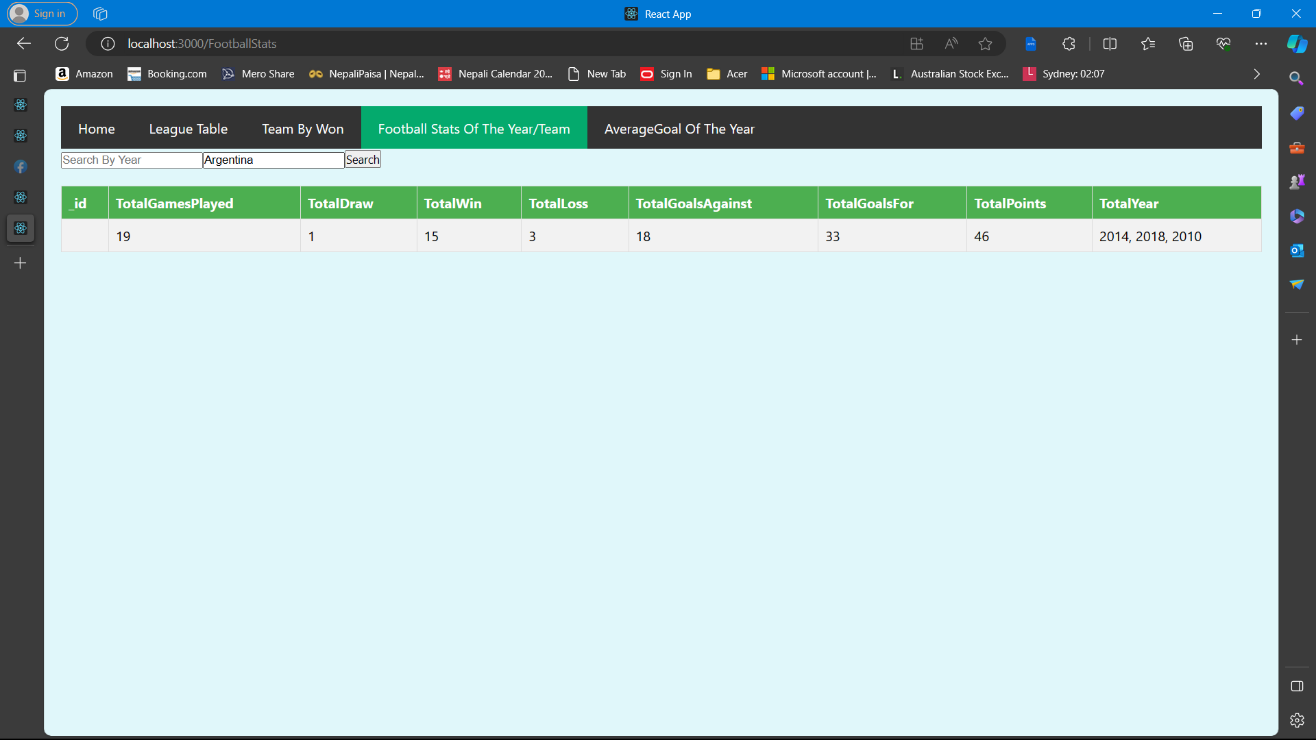
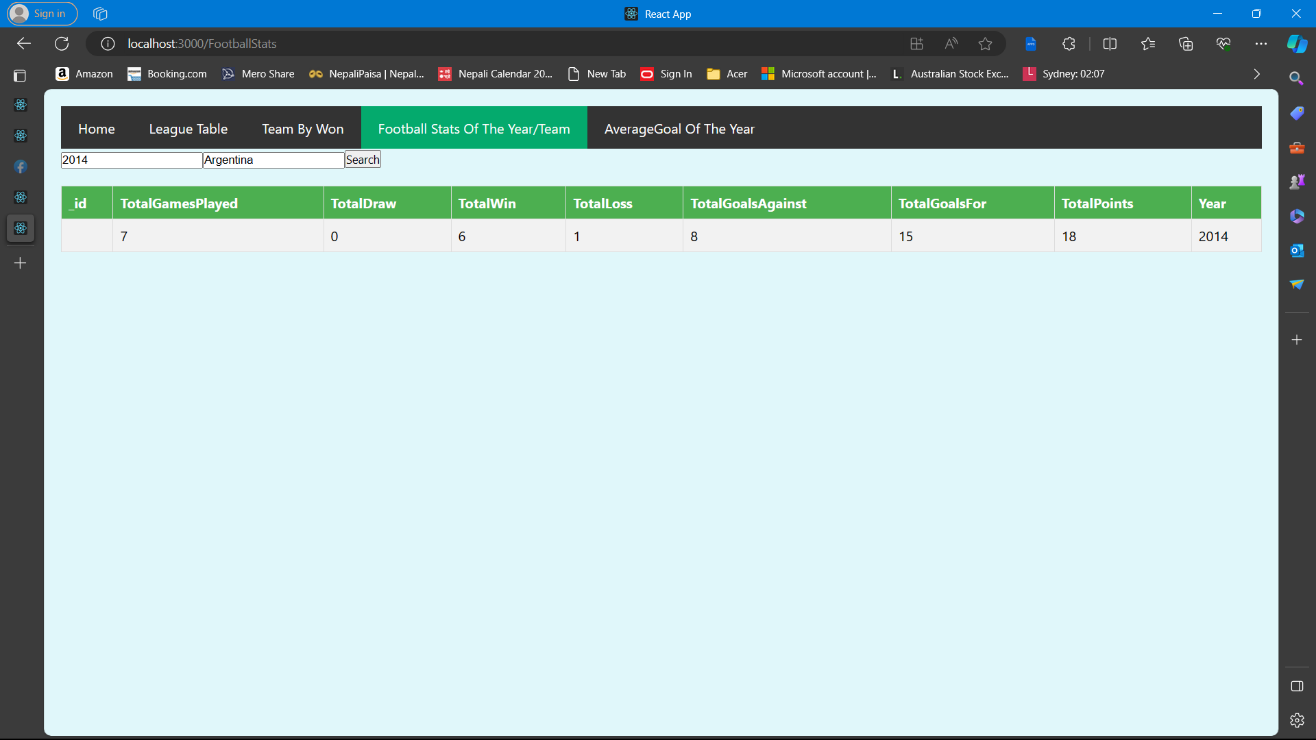
fig F.14

fig F.15

Sub Task 2.4

* **Design choices**

The webserver is simple and easy to use. Webserver contains 5 web pages with same header and footer. The webpage starts with name with logo and there is date below the name. The links are placed in a display flex with black colour in background and colour of the text is white. Due to the colour combination, the links part really looks good with provide great experience while using the website. There is hover effect when the link is touched by the mouse i.e. change in colour. All the links can be directly connected in their webpages except daily news. The daily news has dropdown effect as it contains three webpages i.e. National, International and Sports. This design help user to visit and read the news of different field as per their interest. Due to this effect, they don’t have to look and search from each and every links. CSS is used for designing my website.

In the home page, user can see the news of different sector attached with links in their headings which help user to read the whole news in their respective web page. All these news are attached with the pictures for great experience in reading. There is slide animation which shows the breaking news which helps user to get information if any things happen or there is any emergency. The footer has black colour in background with white colour text for good visibility and great experience.

* **Reflection**
* **Analysis of the work**

As a beginner, developing the website is really a challenging work for me. Before developing the website, I was little nervous about implementation of codes and designs as it is my first time to develop any site. At first, I prepared the layout in my copy and then started to design the website. Before starting the website, I had so many things in my mind about the designing of website but later on, I really found that it is impossible to apply all the designs in the single website. I tried to apply all the animations, design that I learned in my module. During the designing the website, I included the images only along with news articles, and in future I also planning to add videos related to the news which will undoubtedly help the user to get information more quickly and effectively. While designing the website, I went through different other websites and YouTube videos which really helps me to learn more about designing. The website can become more beneficial and can be more eye-catching if I can include all the designs that I planned before designing. However, in the future, I intend to update my portal regularly and add more interesting and beneficial content, which will enhance the user experience and engagement with the site.

* **Survey/Feedback**

After the completion of designing and developing, we were curious and excited to use the newly designed webserver. After completion of designing we thought, every user should be happy, for that, user flow should be easy and simple. Therefore, we asked some of our friends to use it and they really liked it. According to them, it is easy and simple to handle. Due to the design, it really looks good and every effect has given them a great experience while using the webserver. They told that its easy to find the details and output is also satisfying. Some of our friends also gave some elements need to add which will make it more effective. According to their feedbacks, we will definitely try our best to make it more attractive and will apply those things that they suggested in near future.

* **Conclusion**

Overall, we had really a good experience while developing and designing the Webserver for Football data. Developing and designing the web server as we assigned helps to get real life experience which help to hone our skills on developing the server, get to know many new elements, different techniques, various ideas from different web sites which we believe that will help us to our future in this field. We really had a great time and enjoyed a lot working in a team. Developing a web server by using React was really a good challenge for us too and we glad that we managed to crack this challenge and we are focused on to make this webserver more effective for the user in the near future.

* **Link of the website**

User can go through this link to enter the Web server :

<https://ubiquitous-scone-c644d3.netlify.app>

* **References :**

W3schools (n.a)., HTML and CSS [Online]., Available at : <https://www.w3schools.com/> **.,** Accessed at [23th December 2023].