COS30043 – Interface Design and Development

10.2HD – High Distinction Project

Huy Vu Tran - 104177995

1. **Introduction**

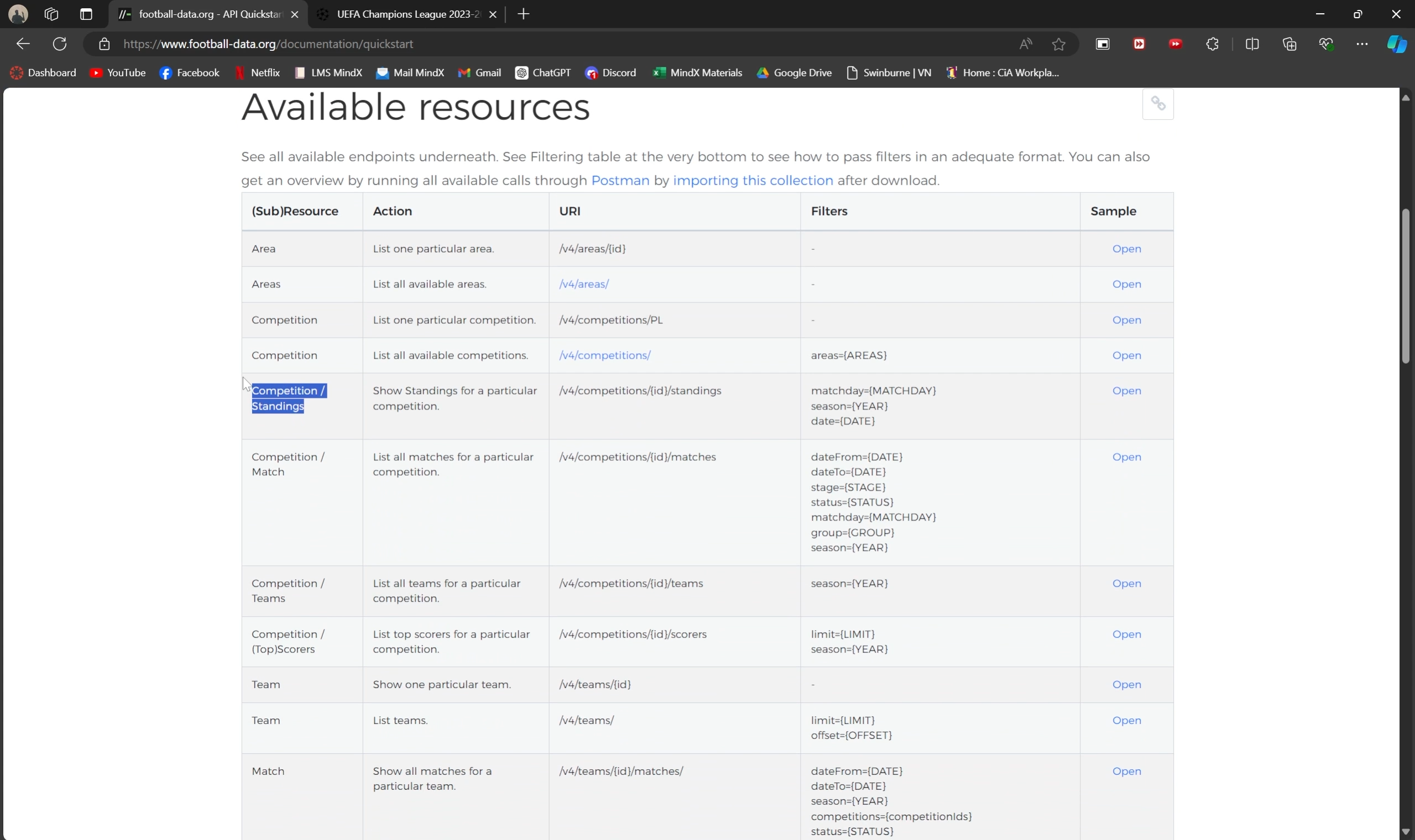
In this report, I will discuss the process of creating a single-page application (SPA) with a focus on the Standings Page of my custom-app project. Specifically, I will demonstrate the coding involved in implementing the Standings Page, which displays the group standings from A to H, along with additional relevant information.

A screenshot of a computer

Description automatically generated

1. **API service**

Before delving into the coding details, I will provide some background information on the API I'm using for this project - the footballAPI. This API allows me to retrieve football data from some of the most well-known leagues in the world. I will explain the steps required to register for the API and obtain the necessary API key to access the data.



To begin, I will show how the retrieved data from the footballAPI looks like in Postman. The API response contains an array of standings, as well as other pieces of information. I will focus on filtering out the unnecessary parts and concentrating solely on the standings data.

A screenshot of a computer

Description automatically generated

1. **Project configuration**

Next, I will walk through the steps required to configure the project before coding. This includes creating a Vue project using the vue create c1 command, installing Axios to make HTTP requests, and installing Bootstrap to style the website.

A screenshot of a computer

Description automatically generated

1. **Coding process**

One of the challenges I encountered during the development process was the CORS (Cross-Origin Resource Sharing) error. This error occurs when the web application tries to make a request to a different domain, and the browser enforces a security feature known as the Same-Origin Policy. To bypass these restrictions, I will explain how to set up a proxy server in the vite.config.js file, which will act as an intermediary, making the API requests on behalf of the application.

Moving on, I will dive into the code in the api.js file, where I can fetch the desired data from the API. I will demonstrate the use of Axios to create an instance with the base URL pointing to the proxy endpoint, and the configuration of the necessary headers.

Finally, I will provide a detailed explanation of the code in the HelloWorld.vue file, where the Standings Page is implemented. I will cover the process of fetching the data, handling any potential errors, and the structure of the template, including the use of v-for directives to display the group standings and team information.

A screenshot of a computer

Description automatically generated

1. **Conclusion**

In conclusion, I have covered the process of creating a single-page application with a focus on the Standings Page in my custom-app project. I have provided an overview of the footballAPI, explained how to configure the project, and demonstrated the coding involved in fetching and displaying the data. By addressing the CORS issue and setting up a proxy server, I was able to seamlessly integrate with the external API and deliver a functional Standings Page. This report should serve as a comprehensive guide for anyone interested in building a similar type of SPA with a Standings Page feature. Here’s the video demonstration for this project: <https://www.youtube.com/watch?v=BV4uUZRxi-A>