**Hacker sprint 4 assignment, Leaderboard.**

Mohammed Nasser Alshukaili

ICT, Fontys University of Applied Sciences Media Design Semester 3

Willems, Pim P.i.

­­  
Jun 08, 2022

I decided to use Vue.JS for this assignment because it offers many dynamic features that could be very confusing to make in vanilla JavaScript. For instance, v-binding and routing.

**Learning process**

This assignment is asking me to fetch a list of students that have different ranks, names, and points. After that, I will allow the user to sort the list the way they like. Finally, I will create a function that would add a new student to the backend database and immediately fetch the changes to display the new list.

For the database, I will use Firebase. I have many other options like mangoDB and SQLite. However, I will stick to Firebase database since this is just a small project that refreshes my Vue.JS skills and improve my API skills.

To practice components in Vue, I decided to divide the project into 2 components, one is for the navigation, and the other component is for the list of students + the other features it might need.

I will be using SCSS to style my app. This would strengthen my SCSS skills as I started learning it a month ago.

I also recently have got involved with external packages like vuex and axios.

I will be taking advantage of the features that vuex offer to manage the states throughout the whole app.

The first thing I did after setting up my vue-app, is creating a firebase database, then I copied the configuration keys and pasted it on the vue-app. I stored the keys in an external JS file called database.js. Now that my vue-app is connected to my firestore database, I should be ready to GET and POST information from firestore. There is still one step missing, I need to store the students’ names and points in firestore.

Graphical user interface, application

Description automatically generated

This feature allows the functions inside it to run immediately when loading the page. I used it call the getStudents function which GET all the data from the fire store by using the onSnapshot package from firebase.

onSnapshot is exactly like the getDocs, the only difference is that onSnapshot executes immediately after any change in the database.

Then I used another firebase package called query to order by the points to display ascending or descending. Fortunately, I am familiar with this concept because I studied SQL for a whole semester.

I saved the data array into a local array, then I simply created an unordered list to v-for through the loca array and display what is inside.

Now that I got all the students from the firestore, I need to have a function that adds people to that list. That should be easy with the help of addDoc package from firebase. I just called this function with the name of the database, and I can add students’ names and points and POST them to the firestore.

I used v-if to show the form only when clicking the + symbol.

I also added delete and update functions to the app. The idea behind them is to assign the doc id to the delete/update image as a dynamic ID, then after clicking the image, I get the ID of the specific doc saved locally. After that I use deleteDoc for deleting, and setDoc for updating.

**Personal Refraction**

Vue.JS and Firebase are super powerful together. I am now able to connect them easily with the usage of dynamic features like v-if v-for. After studying vanilla javascript for 8 months knowing that this is the power of dynamic websites, Vue came to show me even better ways to approach different things. I also liked SCSS language, it is more logical for me than the plain CSS. I can hardly imagine myself coding a dynamic website without using Vue/Vite, and SCSS.