

METCS673 Group 6 - Project Status Report 4

TerrierMap

Team Members:

- Ryan Christopher
- Misael Gared
- Jasmine Hughes
- Siddhraj Parmar

TerrierMap GitHub Repo:

<https://github.com/ryan-christopher/CS673-TerrierMap/tree/dev-landingpage>

Live Link:

<https://terriermap-git-dev-landingpage-ryanchristophers-projects.vercel.app/>

Meeting Decisions:

- Remove the backend folder as Next.js supports [API routes](#) which was the main reason for our backend
- Use the 'pages/api' folder that Next.js has built-in support for to handle all of our API calls for the routing machine and Firestore database search
- Will use the [unittest python library](#)

Completed Tasks:

- Registered Firebase project for web app use
- Created environment variables for Firebase access on both our local machines and on the Vercel deployment
- Updated readme files and added SDD to the project artifacts folder
- Created a component to list all of the classrooms from our Firestore database as this will later be used to implement in the API folder

Assigned Tasks:

- Add API functions in the API Routes folder to take a building room as input and return the result including the address and coordinates | Ryan Christopher
- Implement a destination pin to the Leaflet map | Siddhraj Parmar

Individual Contributions:

Ryan Christopher: This week I edited the format of our project to use the [API routes](#) that Next.js supports as it more accurately aligns with what our 'backend' functions require. Then, I added the environment variables to my machine and to our Vercel project in order for our Firebase information to be securely stored as we continue to develop. After the variables were set up I connected our Next.js project to the Firestore database and created a component that pulls all the entries from the 'classrooms' collection that we have. This will later be used to search for existing classrooms once the user enters in a building code and room number. Finally, I began writing a test file in our 'api' folder to begin passing an entered classroom code to return a matching result from our database.

Misael Gared:

Jasmine Hughes: This week I wrote the Software Design Document. After creating the backend folder, we realized that the folder was not necessary to connect our Firestore database to our app. I removed the folder and instead added the needed Firebase files (like firebase.js, firestore.rules, etc.). I also registered the Firebase project for web use. After setting up the Firebase files, I configured the database to use environment variable names, so our application's data isn't accessible publicly. I have a copy of the environment variable on my machine for testing purposes. With Ryan's help, I was able to troubleshoot and get the database information displayed on the web application from the local host.

Siddhraj Parmar: This week, I implemented dynamic map shifting to auto-center based on the user's location, eliminating the need for manual adjustments and enhancing navigation. I also improved input handling for "Building Code" and "Room Number" fields by adding validation constraints to ensure accurate data entry. Additionally, I refined the UI by updating the styling of input fields and buttons.