**CSC 400 Final Report**

Chris Pantani

Feed Branford Kids Planner

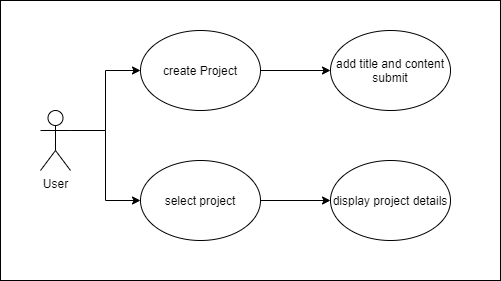
**Background and Project Conception**

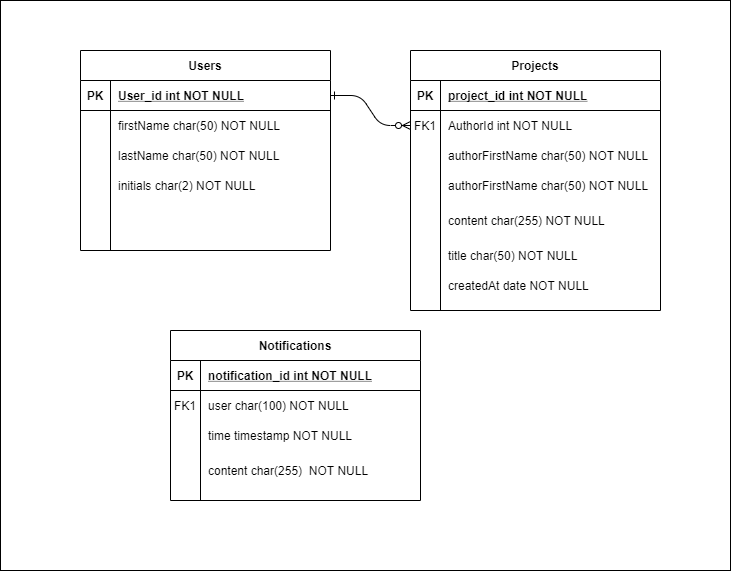
This Project was designed with the charity “Feed Branford Kids” in mind. It is an online project/task management system. The main objective is to allow users to create projects with a title and subsequent content further describing the project. The projects will be viewable on the main dashboard of the application as well as notifications depicting user creation and project creation in real time. The program was built with the group of volunteers for this organization in mind. Many of the volunteers are elderly and are not tech fluent so the design is very straightforward and easy to navigate.

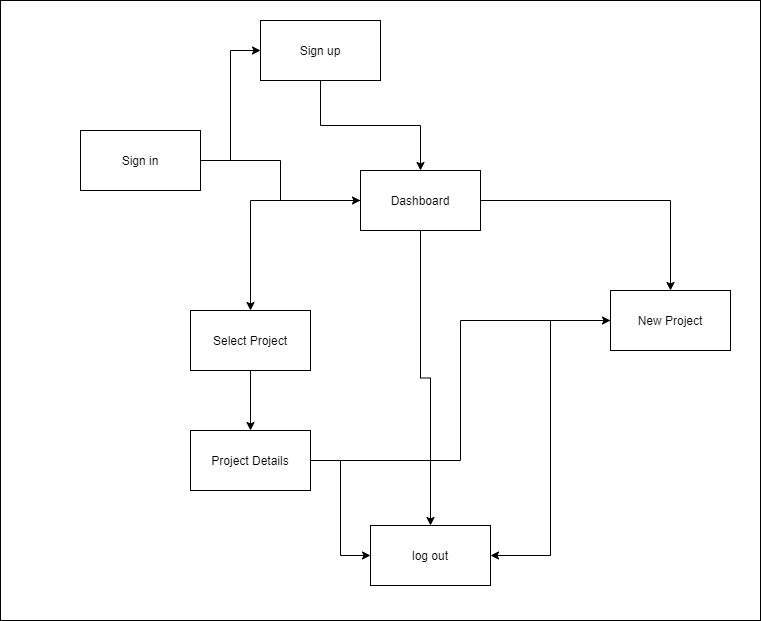
**User Stories:**

1. As an organization head it is helpful to be able to post projects and task that must be completed in an application that all members have easy access to
2. As someone who does not have much experience with computers it is nice to use an application that is easy to navigate and understand
3. As a member of a team it is helpful to be able to post project goals to stay on the same page with my teammates

**Use Case**



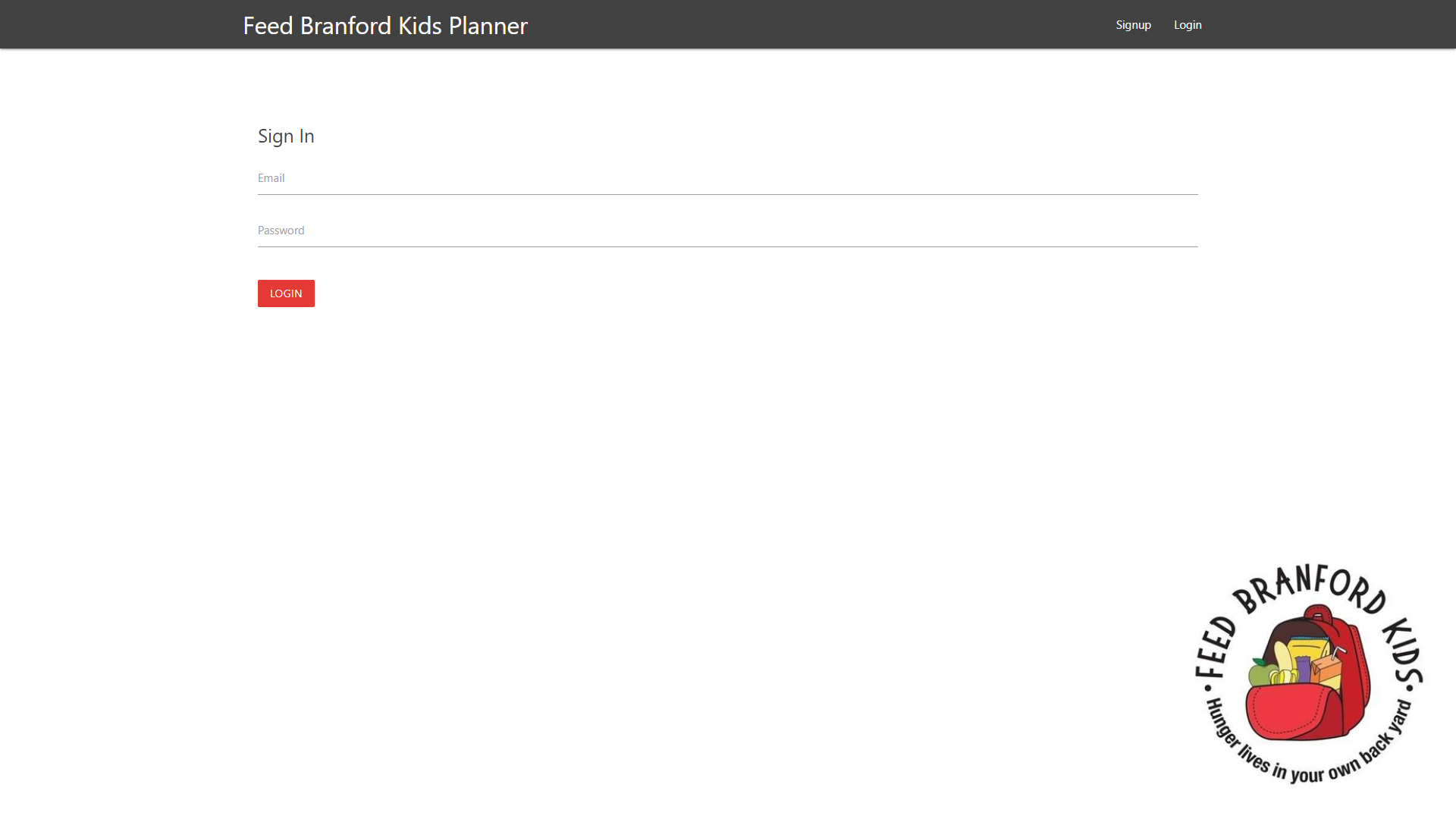
**Entity Relation**

**Site Map:**

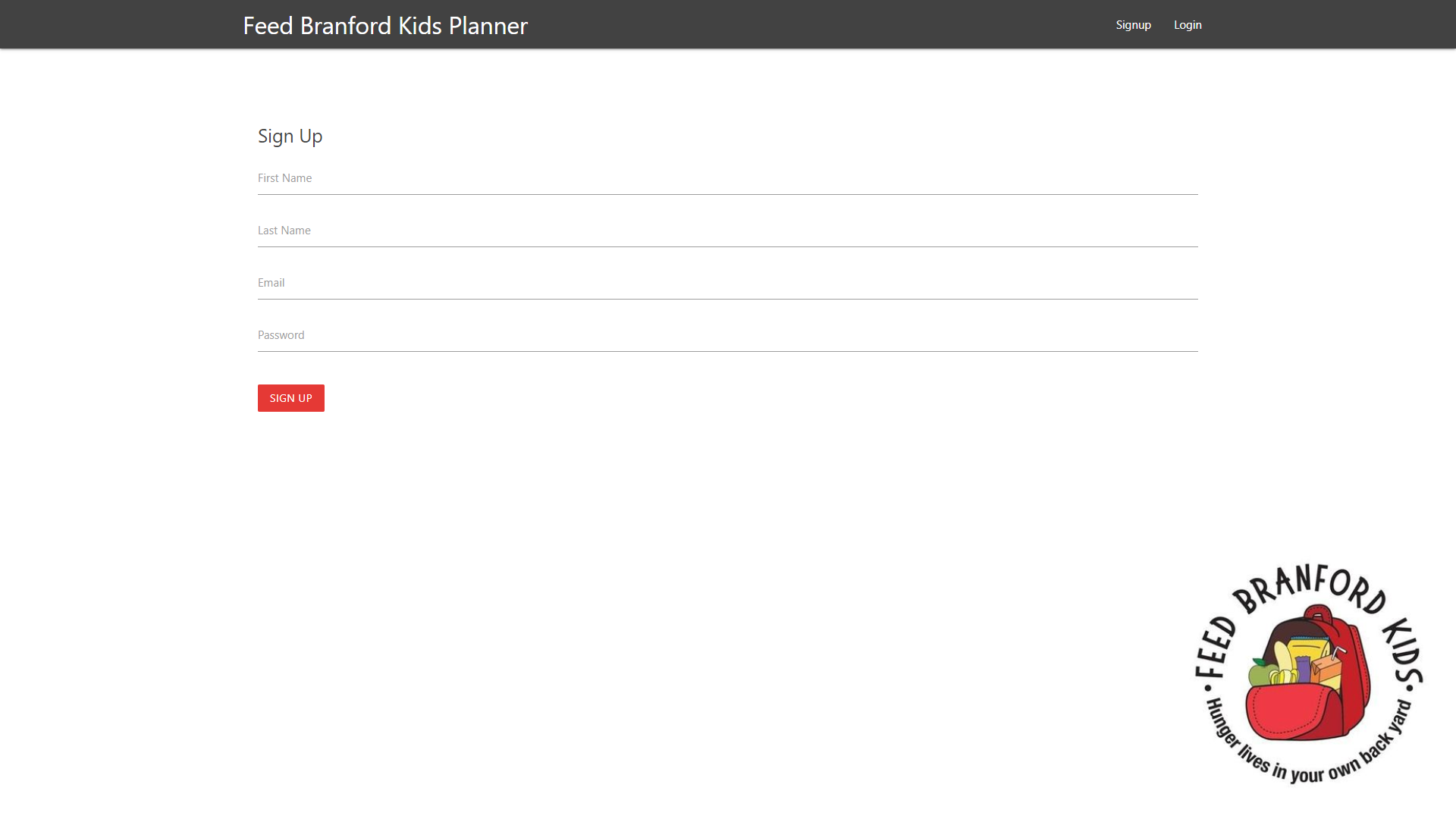
**deployment user guide:**

go to: <https://feed-branford-kids-plan.web.app/>

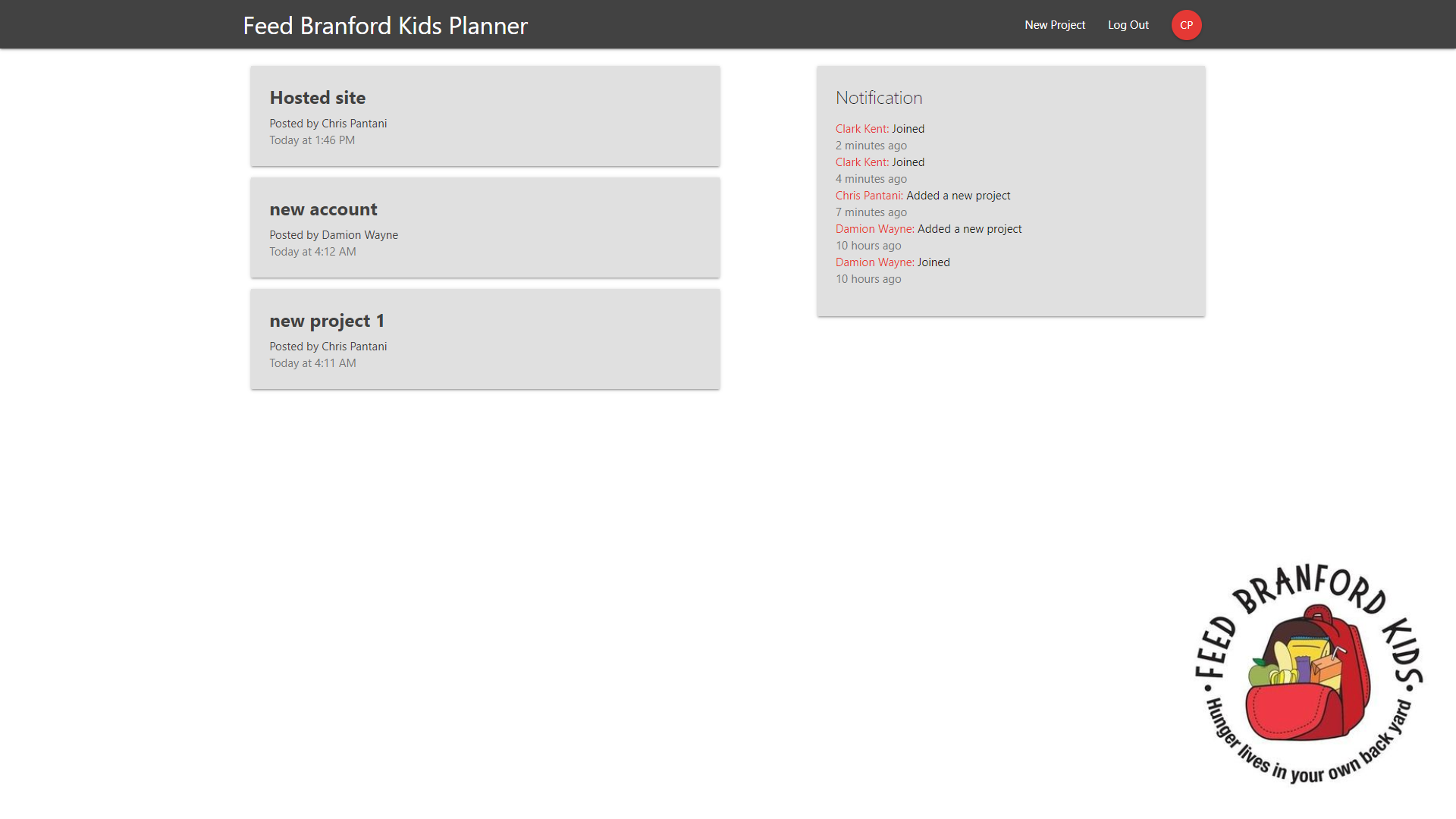
log in if account is created



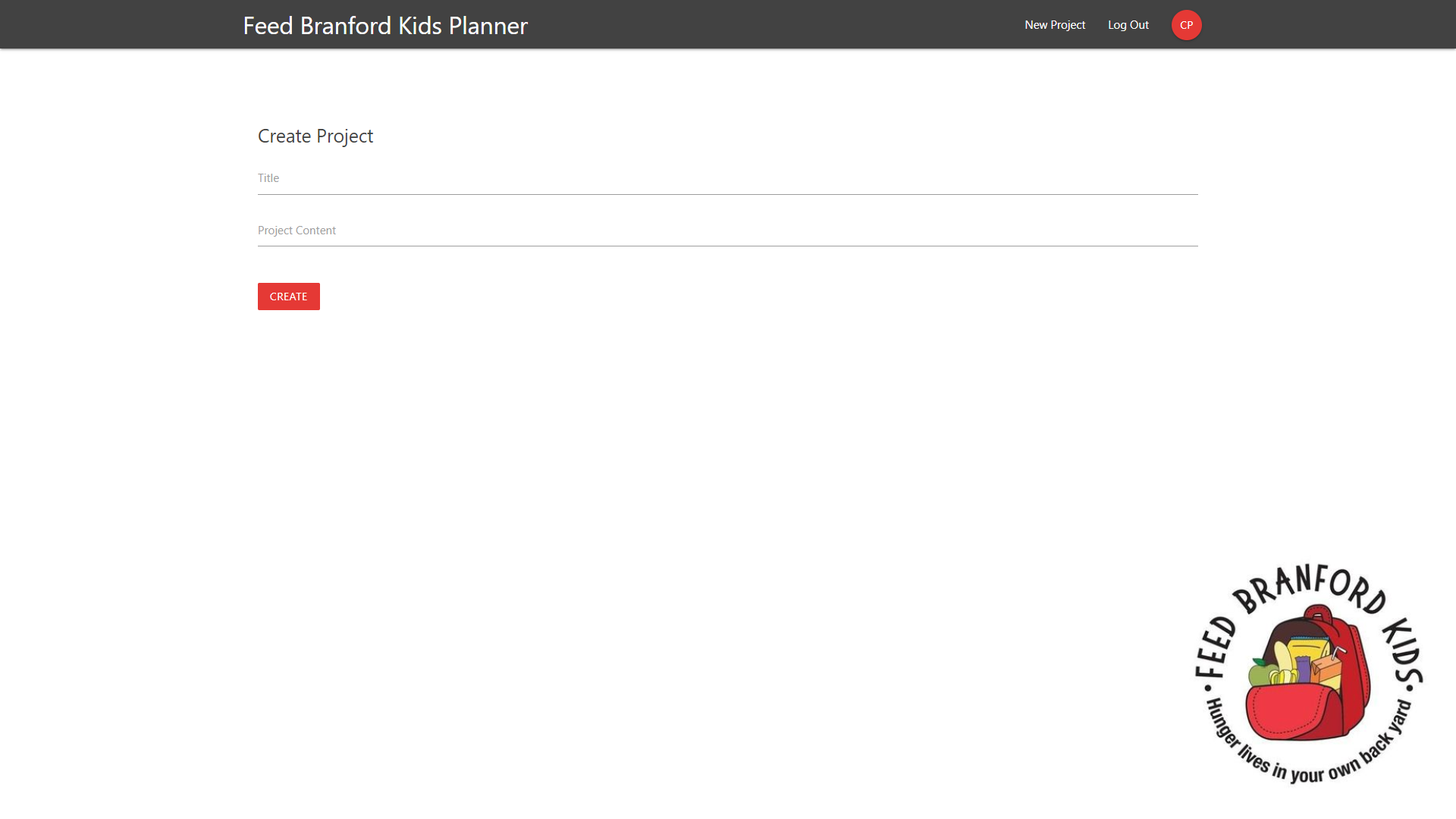
Or create account



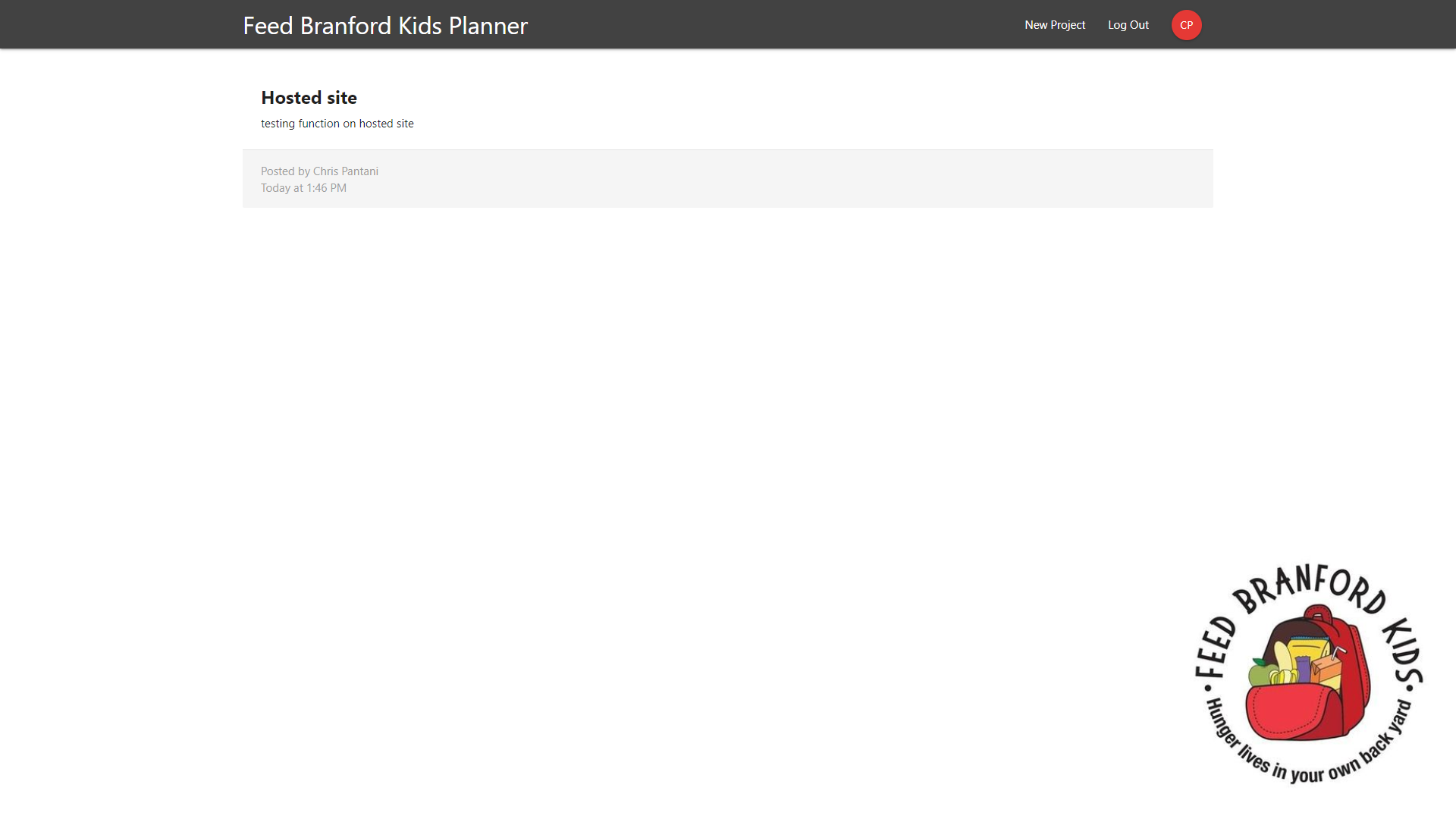
redirect to dashboard



Select new Project to create new project



Click on existing project to see project details



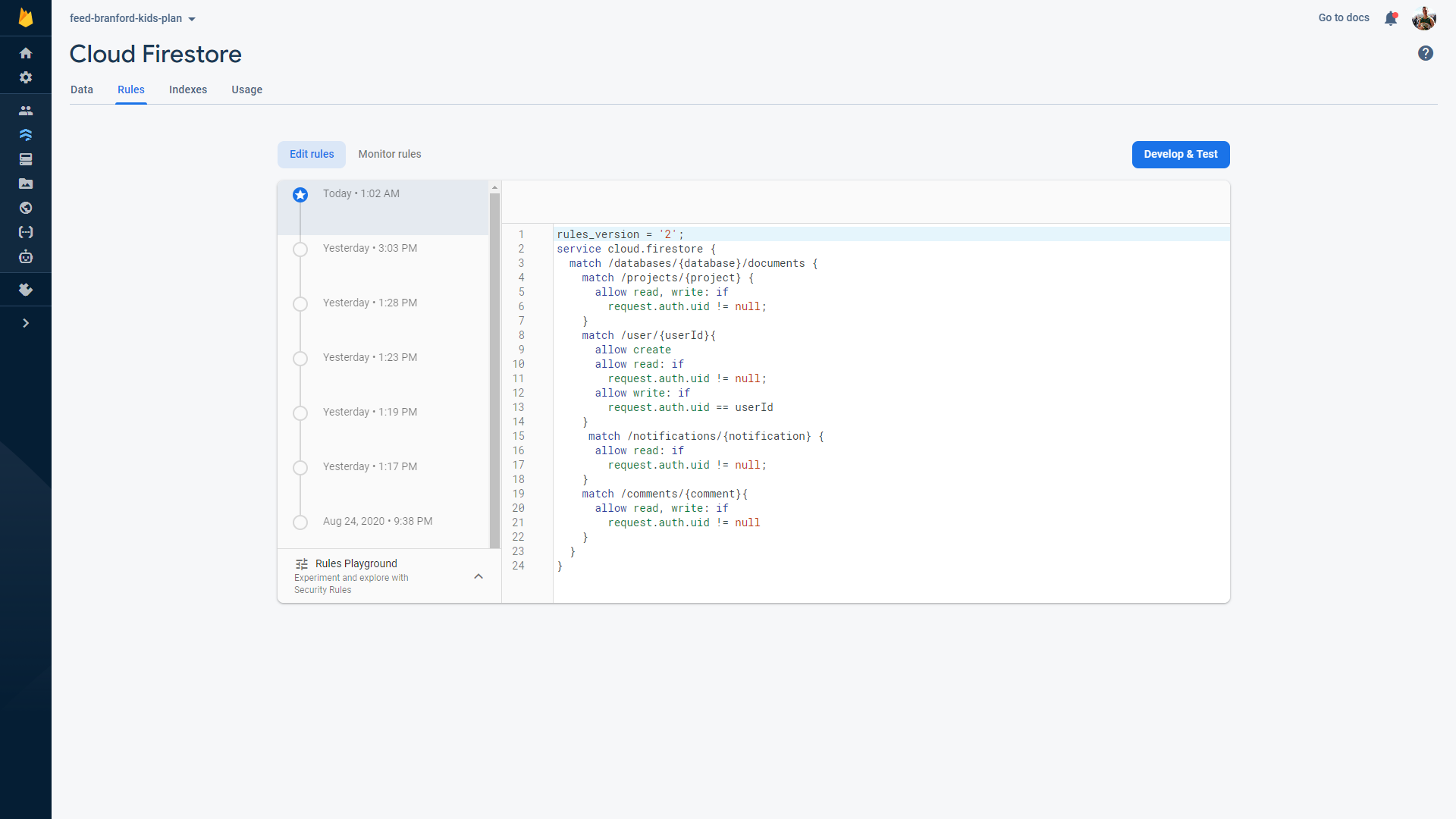
Clicking on application title or initial id bubble in navbar will redirect to dashboard

Click logout in dashboard to log out and end application.

**Core Algorithms:**

The project front end was built using React and using Redux to manage the state to connect with the firebase database. The action of creating users or projects is sent by taking data from the state and sending an object through firebase.firstore into the appropriate collection.

Security rules for access to firestore:



State of implementation:

**User creation (completed)**

**Project creation(completed)**

**Project summary list display (completed)**

Display a list of projects and minimal details on dashboard in descending order

**Project details (completed)**

Ability to select projects from project list to display full details

**Real Time Notifications (completed)**

Notifications sections that display created projects and new users in real time

**Testing and evaluation:**

I built and tested all features separately one at a time and integrated them together once the section worked independently. There were some unforeseen issues with the way that firebase is set up and how the collections in the firestore relate to one another that is very different from other db schemas I am familiar with. Otherwise firebase is a very good application to use as a server side for applications and very helpful if not a little hard to understand.

**Lessons learned:**

This project though smaller than what was expected of a class like this and completed in a few weeks provided me my first chance to build an application from scratch in an isolated environment. Much like how jobs will have to be completed in the work force. A major lesson was how difficult it can be to have no safety net as well as how rewarding it can be to complete something totally on your own. Even if that something is very small and inconsequential like the application I have built. Learning to use React along with Redux has been very helpful in furthering my understand and building my usable knowledge in the field. React has many libraries the made creating this application much easier than it would have been with just JavaScript and html. This was the first time I researched libraries that could be used instead of writing up a function or method and they were very helpful in designing and implementing my project.

**Version 2:**

I fully plan to continue to work on this project to add it to my portfolio and to sharpen my skills. Something I plan on adding is a comment section in each project details page for users to leave thoughts. I have already begun to work on it and have gotten to the point where comments can be created and added to the db along with all relevant information about the poster. I just need to do more research on firestore collections relations or Redux state manipulation to connect the correct comments to the correct project and display a list.

I also plan to add in functions that allow users to remove projects, or mark them as complete, inside the application so that there is no need to go through the server side to remove listing. That will take more research into firebase and firestore as well as a redesign of the project collection and the project creation functions.