CSE 134B Homework 4

David Lee A09548716
Nikhilesh Aiyer A11247204
Eddy Kim A11690057

Authentication

Using Firebase tools, we were able to set up both email and Google account authentication for our application. We used the TA provided sample along with the Firebase API and guides for this portion of the assignment. We were able to receive email and Google account information into our Firebase, but we initially ran into problems with checking authentication on each page. We solved this problem by merging our Javascript files and adding a check for authentication state in the beginning of our file. Our application is now able to check the authentication state for the specified user, and hide and display the proper web pages depending on the user's authentication state.

CRUD

In the beginning, we had some problems getting started on how we should approach applying CRUD functionality to our application. We initially decided to stay away from Vue JS, as none of us were familiar with the framework, and were able to retrieve form data on the application and store that data inside our Firebase database. Our group member, David, was able to figure out how to implement Vue JS for our database, and after talking and learning from him, we decided to switch over to primarily use Vue JS to handle our database requests.

One thing our group could've done better for this assignment was to decide earlier on our structure of our database. As we were working through different functions, we were all saving our data differently with no apparent structure. This presented a road block in our development, as time was wasted fixing code and rearranging data items in order to fit a more structured database. For the "delete" portion of our assignment, we added a temporary delete button that performs our function, but will change the styling for the final turn in.

A topic of question that arose during our application development for this assignment was how we are going to handle adding/deleting classes. We decided for this assignment we should focus on implementing CRUD functionality to a class page, as we are able to cover all four functions on our class, professor, ta, and feedback pages.

Since we were all familiar with using Vue JS in conjunction with our Firebase database, implementing asset management was a lot smoother than authentication and initial CRUD

functionality. We did manage to get our CRUD asset management to function properly, but after we implemented this functionality for our images, it messed with our page styling. Due to the way Vue displays objects, we had to change our initial layout of the professor and ta pages. Certain elements of our page had to be tweaked to fit close to the original design. We are going to continue to work with our page styling in order for things to look more consistent and even for the final turn in.

Repo Organization and Coding Consistency

Although our group did strive to be consistent in our file naming and organization, we made sure we were very critical in our consistency and organization for this assignment. Even though we are three individuals with our coding habits, we were constantly working collaboratively on our files, so coding consistency was improved for this assignment. As for repo organization, we were keeping careful, as our repos grow larger and more complex, and making sure that our repos were easy to navigate and logical.

Testing

After implementing our CRUD features and including any other optimizations and improvements, we tested remotely on an iPhone 6. We noticed that logging in took a considerable amount of time compared to all the other pages, and even fetching data from firebase database was fairly quick. We believe this is the case because authenticating and connecting a user to a database takes longer and more data is being passed such as session state, etc. Below is a summary of the times it took to load up each document, including any .css and .js files that had to be loaded as well.

File (including any .js and .css)	Time (average in milliseconds)
index.html	467ms
login_signup.html	548ms
class_finder.html (w/ log in)	3.28s

class_finder.html (w/o log in)	574ms
overview.html	451ms
class_edit.html	452ms
class_rate.html	459ms
professor.html	471ms
prof_edit.html	407ms
prof_rate.html	584ms
ta.html	427ms
ta_edit.html	579ms
ta_rate.html	624ms
feedback.html	417ms

Conclusion

This assignment has definitely proven to be one of more challenging projects of the quarter. We had to learn many new technologies in order to implement our site in the way we wanted to. This has given us a far greater understanding of the relationship between our frontend and backend, as well as the different methods we can utilize to implement our functions. As a result of our trials through this project, we feel that we can greatly improve on our product for the next assignment.