

# Presenting - The Coffee Dex!

Available at: <https://coffee-dex-a7a8d.firebaseio.com/>

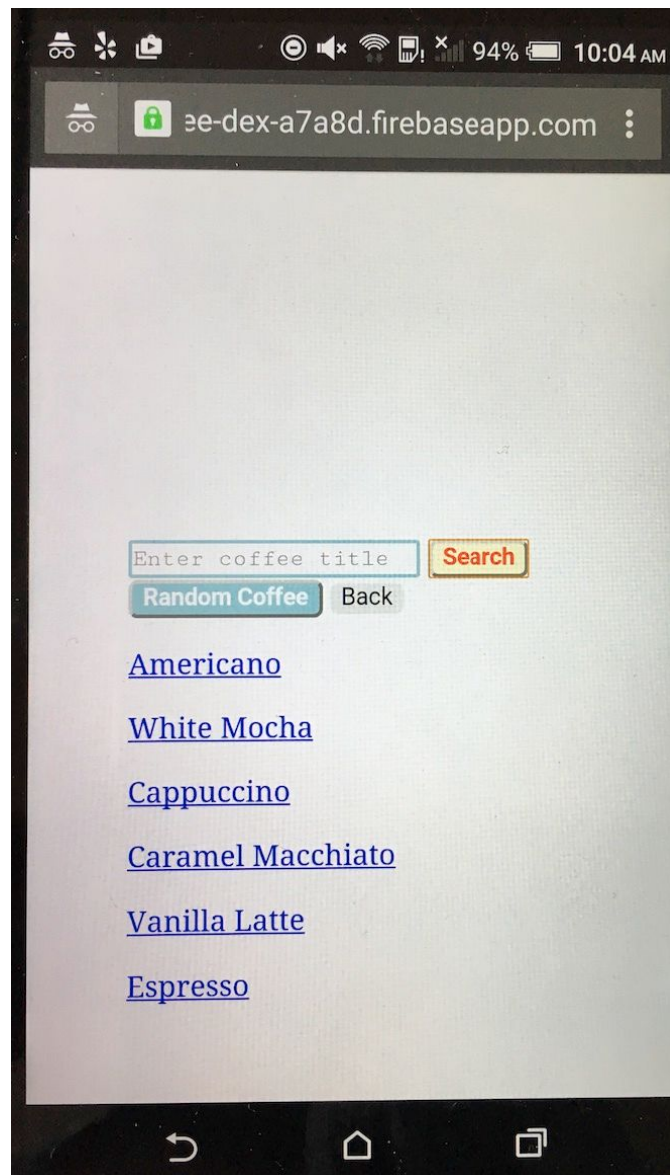
## What is it?

The Coffee Dex is an application designed to cater to the needs of coffee enthusiasts everywhere. There are so many different flavors and types of coffee in the world and it can be remembering exactly which ones tickled your taste buds the best. Users can search for various types of coffee, favorite coffees suited to their tastes, and add notes on why exactly they enjoy those types of coffee.



## Application Overview

- Frontend: HTML, CSS, JavaScript
- Backend: Firebase database
- Services Used:
- Tools Used: Sublime Text, Lighthouse, w3validator
- Main Target: mobile phones (Android) with web users being secondary



## Testing & Usability

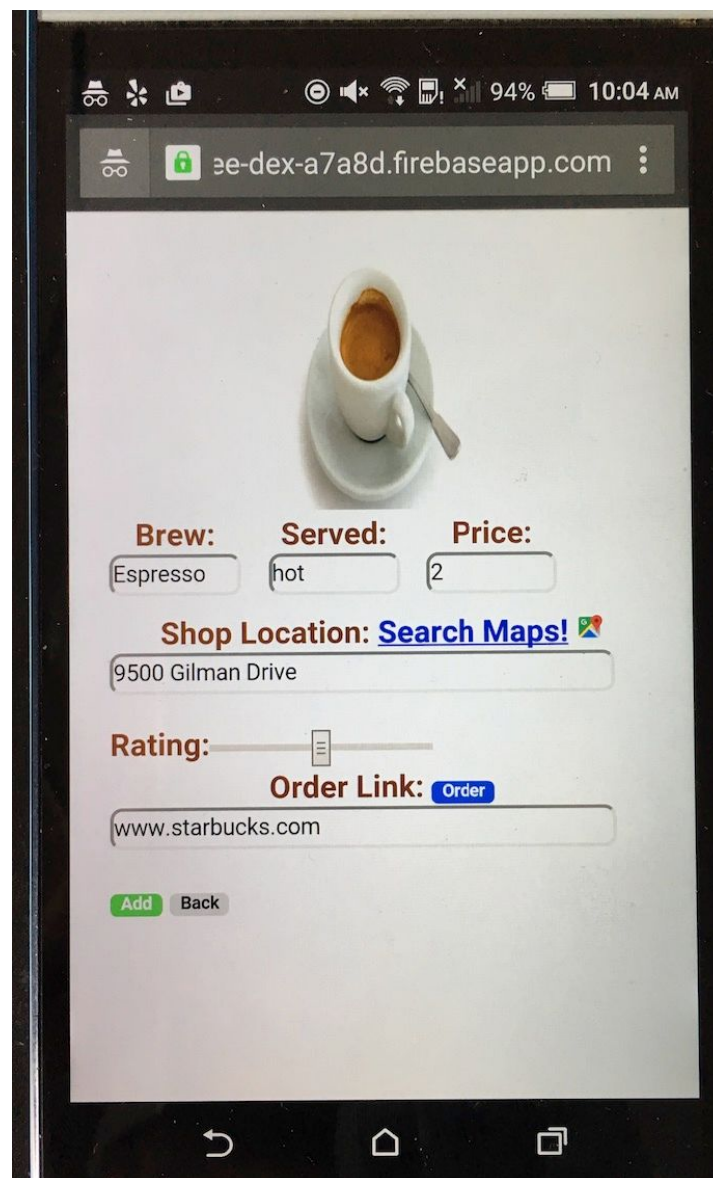
We used *Lighthouse* as a tool to test our web app's usability on Firebase. Following the report generated by Lighthouse we fixed bugs and potential issues it mentioned such as adding a manifest and a service worker, both of which are used to handle the offline services. Using the tips and suggestions generated by Lighthouse we managed to alter our application in such a way that potential problems were mitigated. In addition, we used older Android models to test the speed and loading times of our web app in order to improve user experience. Finally, we used W3's html validator in order to ensure most of our markup was proper. It should be noted, however, that this is in regards to *our* html. The 404.html page generated by Firebase doesn't validate because its img tag is missing an alt attribute. We didn't mess with this because it was generated by Firebase and because we didn't want to potentially screw anything up with Firebase's configurations.

## Performance

- We minimized our CSS/JS files and used vanilla CSS in order to reduce the amount of unused code we have sitting around.
- As previously mentioned, we used older Android models to test loading times and potential errors.
- Small images were intentionally sought after - in early performance tests we found that larger images would actually break our web app's pages on older phones, so smaller images were prioritized.

## Database Restrictions

Due to the way we designed our application, the coffee database is completely off-limits to users. That is to say, we are the ones who update the database containing all of the different types of entries. We did this because there is no guarantee that proper types of coffee will be updated if the users were given free reign which would hurt our app's credibility and reduce the user experience. At the same time, however, we acknowledge that it could be a potentially daunting task to keep track of the unreal amount of coffee shops around the world. Future application development would need to keep this in mind. Instead, we opted simply to give the users exactly what our app does - favorite coffees suited to their tastes.



ae-dex-a7a8d.firebaseio.com 10:04 AM

**Brew:** Espresso **Served:** hot **Price:** 2

**Shop Location:** [Search Maps!](#)  
9500 Gilman Drive

**Rating:**

**Order Link:** [Order](#)  
www.starbucks.com

[Add](#) [Back](#)

## Issues & Limitations

Due to time concerns, we did not have time to implement some features:

- **Share with friends**  
Send your favorite coffee including price, location and other information to your friends via email or text
- **Quick Compare**  
Select two or more types of coffee and it will generate a form comparing their prices, serving size, ratings and so on.
- **Proper Rating Button**  
When we first drew up the look of the Coffee Dex we originally had a rating system which accrued data from the userbase and displayed the current average. In addition, the rating bar was meant to use yellow stars to indicate a score.
- **Proper Favorites Loading**  
We encountered an error when attempting to load a user's list of favorite coffees - since we need to load data from Firebase's database our scripts run before the data is completely fetched, resulting in a blank favorites list unless there is a button to stagger the load
- **iOS Version**  
Currently, our app only works for desktop browsers and Android phones. Future development would need to take the iOS demographic into account.
- **Shoddy Browser Display**  
Our web app looks rather shoddy when displayed on a browser. Much of it was designed for a mobile platform and as a result, the display suffers when viewed on a desktop browser. However, the Bootstrap variant helps resolve this issue, albeit at a performance tradeoff.