**User Account Log In**

**What does setting up user accounts involve?**

Setting up a basic user account for any type of web service and web application involves creating a username, password, user email address. Pantry is no exception, so to store and classify this information we need a database user table. In the case of our project, we want our users to be able to store ingredients they have in their pantries, recipes, meal plans, groceries they need to buy, as well as food restrictions. Each one of these categories is going to have a database table to store them, but to ensure that the users access their own information and not other users’, we need to designate the user an ID to relate the information of each category to its respective user.

**What sort of third party services will need to be utilized, if any?**

As we specified in our initial architecture document, the backend will be hosted in Google Firebase to store all the user account data as well as all the recipe and ingredients data the user would be able to customize and access.

**How does this interact with the cloud database**?

Google Firebase uses cloud-based NoSQL databases, Firestore, and a real-time database, to store information in the Google Cloud Storage bucket. Since it uses NoSQL databases it means that it saves information in collections and documents, which we can use to store user, recipes, ingredients, meal plans and grocery list information.

**How much work would it take on our end to set up a way for users to register an account? Describe the process.**

The Firebase Architecture is made up of many components, two of them are Cloud Firestore and Authentication which we can use to register user accounts. Cloud Firestore is a scalable database service for server, web, and mobile development, so we can use it for both our web and mobile versions of our app. For mobile use it supports Android and iOS so it can also help us widen our target audience. We can use this component to store user information as well as to sync it to our app. Authentication is included in the software development kit as UI libraries. It can be used as a backend service to register and login users through the use of passwords, phone numbers and governmental IDs. In our case, the user would only require an email address and password as a means of authentication.

The fastest way to add Authentication would be to use FirebaseUI Auth, which is a drop-in UI library. To use it we would need to create a Firebase project and register our app with that project, then we would need to install the software development kit (SDK) and initialize Firebase, finally we would have to import the Firebase Authentication libraries to it. Then we can start setting up a sign-in method we want our app to support, which in our case would be email address and password.

**How much work would it take on our end to set up a way for users to log in to an account?**

**Describe the process**

To enable users to login to their account we would also use the Authentication component, specifically FirebaseUI Auth. After a user account is created and linked to the credentials (email address and password), this new account is stored as an Auth object as part of our Firebase project. So, since the user’s registration information is already stored, we only need to pass the user’s input for email address and password to our Firebase project. In our Firebase Realtime Database and Cloud Storage Security Rules, we can obtain the authentication information from Auth.

**How much work would it take on our end to implement the program displaying a user's data in place of locally stored data? Describe the process**

In this case, we would have to retrieve data from our firebase project registered with our app. There are two ways to do this, the first one is by using asynchronous listeners. Data stored in our Firebase Database can be retrieved by attaching an asynchronous listener to a database reference. The second one is by using blocking reads. Data stored in a Firebase Database is retrieved by invoking a blocking method on a database reference, which returns the data stored at that reference, in our case, the user authentication information.

**How long would it take to implement those changes?**

Since we are new to using Google Firebase, we would need time to become familiar with this tool and all its functionalities. We also need to account for debugging delays. A reasonable amount of time would be from two to three weeks for an initial and fully working implementation of user registration and log-in functionalities.