**Full Stack Flashcard Project Overview**

This is an application that will allow a user to be able to study, create, update, and delete flashcards. It will have a welcome page, login page, home page, and logout page. Once you are on the home page you will have 2 options. The options are: 1 study flashcards, 2 manage flashcards which you will be able to create new flashcards, update an existing flashcard, and delete an existing flashcard. Lastly, this application needs to have a navigation bar with some features like logout.

**Frontend: React**

* Adding Frameworks to React Application
  + npm add axios
  + npm add react-router-dom
  + npm add formic “Optional”
* Welcome Page
  + This application needs to have an area for user login.
    - User login by username and password
    - Button for creating new user
      * New user button needs go to the create new user page
      * The newly created user needs to be saved in the database
* Home Page
  + Option 1: Study Flashcards
    - This will send you to a page where you will be studying your flashcards.
  + Option 2: Manage Flashcards
    - This will send you to the Manage Flashcards Page that will have a list of all the flashcards with buttons for either create, update, or delete:
      * Create Flashcard Button
        + Once the user selects the create button it will direct you to a new page for creating a new flashcard and then return you back to the Manage Flashcards Page.
      * Update Existing Flashcard Button
        + Once the user selects the update button it will direct you to a new page for updating the current flashcard and then return you back to the Manage Flashcards Page.
      * Delete Selected Flashcard
        + Once the user selects the delete button it will delete the current flashcard. It will also need to let the user know in some way that the flashcard was deleted.
* Logout Page
  + This page will say thank your using my website or something. It will also need give you some kind option to log back in and study some more.

**Backend: Spring Boot RestAPI with Hibernate and MySQL**

* Project setup dependencies
  + Spring Web
  + Spring Data JPA
  + MySQL Driver
* Flashcard API
  + Flashcard Entity
  + Flashcard DAO
  + Flashcard Controller
    - Retrieve all existing flashcards
    - Retrieve Flashcard by id
    - Retrieve random flashcards
    - Create new flashcards
    - Update existing flashcards
    - Delete flashcard by id
* User Login API
  + User login Entity
  + User login DAO
  + User Login Controller
    - Create new user login credentials
    - Retrieve user login by id to log into the system

**Backend: MySQL**

* Create database for flashcard and user login
  + Script for creating local host
  + Script for creating the Flashcard Database
    - Flashcard table
    - User login table
* Flashcard Database
  + Flashcard table column
    - ID: Primary Key
    - Question
    - Answer
  + User login table column
    - ID: Primary Key
    - Username
    - Password