Book Buddy Deployment Manual

Table of Contents

1. Setup Work
   a. Clone github repo
   b. The backend server uses PostgreSQL, we recommend Postgres.app for easy set up if you are on Mac or any other Postgres installation. Keep track of the username and password and create a database for the server (named whatever you like)
      i. On first run of the application, it will automatically create tables and keys for you in the database
   c. Import project into IDE (optional)
   d. Create firebase account and set up Authentication feature, get API Key
   e. Install these packages for your local dev environment:
      i. Java: https://www.oracle.com/java/technologies/downloads/
      ii. Maven: https://maven.apache.org/install.html
      iii. Node for React Frontend: https://nodejs.org/en/download/

2. Backend Server
   a. Find application.properties file in codebase/bookbuddy/src/main/resources
   b. Input database name, user and password into properties.
   c. After inputting information, run "***mvn test***" from the command line in the bookbuddy folder. If you get all green then you are ready to run, if not use the error messages to find out what went wrong. If you use an IDE, it might be easier to find the cause
   d. From the command line run "***mvn spring-boot:run***" this will start the backend development server on port 8080.
   e. Go to "http://localhost:8080/swagger-ui/index.html" for interactive API documents and endpoint testing.
   f. Sample books have been included in the resource folder if you want to load books into it. Run the backend server first to create the books tables and then use a PSQL COPY command to load the information from the SampleBookDatabase.csv file into the database

- i. If you don't add these books to the database, there could be problems with the front end rendering and the ML API. It is highly recommended to use this sample database.
  - ii. This PSQL command will work:
       COPY books(title, author, price, description)
       FROM 'path/to/csv/here'
       DELIMITER E'\t'
       CSV HEADER;

3. Machine Learning API
   a. Navigate to
   b. In the terminal run: "**python3 -m venv mlapi**"
      - i. This will create a python virtual environment
   c. Then input: "**source mlapi/bin/activate**"
      - i. This will activate the virtual environment
   d. Then input: "**pip3 install -r requirements.txt**"
      - i. This will install all dependencies needed for the virtual environment
   e. Then input: "**python3 app.py**"
      - i. This will run the API and you can keep this running in the background.

4. Frontend
   a. With repo cloned navigate to codebase/frontend on the command line input "**sudo npm install**" and this should install all packages and dependencies needed to run the server
   b. Input Firebase API Key in firebase.js file
   c. Make sure to start Machine Learning API and Backend Server first
   d. In the same command line input "**npm run dev**" and the server will start on port 5173
   e. If you see a blank home page then check that the books were loaded correctly in the backend database.
   f. Signing up as a user will open all functionality of the site