

DocNetwork Full-Stack Challenge Design Document

API Considerations

1. Must use the express server framework
2. Must use a database of some type
3. Implement an API route that accepts a search query, passes that along to the Books API and responds with a collection of book results in a JSON format that includes the following details:
 - **Title**
 - **Subtitle**
 - **Authors**
 - **Description (First 140 characters)**
 - **Categories**
 - **Publisher**
 - **Published Date**
 - **Preview Link**
 - **Cover Image (URL)**
4. Each query should store the search query and associated responses. Entries should include all of the above information as well as a timestamp of when the entry was created.
5. When a search is run using a query that's already stored in your DB, return the results from your DB. If no existing match is found, continue with the Books API call and return those results as usual.

Browser App Considerations

1. The Application is built using React.
2. There should be a search bar to input a query.
3. The results should be rendered in the results area. You must include these fields (if available for the record):
 - **Cover Image**
 - **Title**
 - **Subtitle**
 - **Authors**
 - **Preview Link**
 - **Any other information as you see fit**
4. Each subsequent query should append or prepend its results to the results area.
 - Let's say the user searches "cars" then "dogs". Results for both search terms should appear on the page at the same time. If the user searches "cars" then "dogs" then "cars" again, the user should see more results (but not the same results) for "cars".
5. There must be no duplicates (by Google Books id).

6. The results should be visually appealing, styled logically, and fully responsive for mobile devices.

Application Architecture

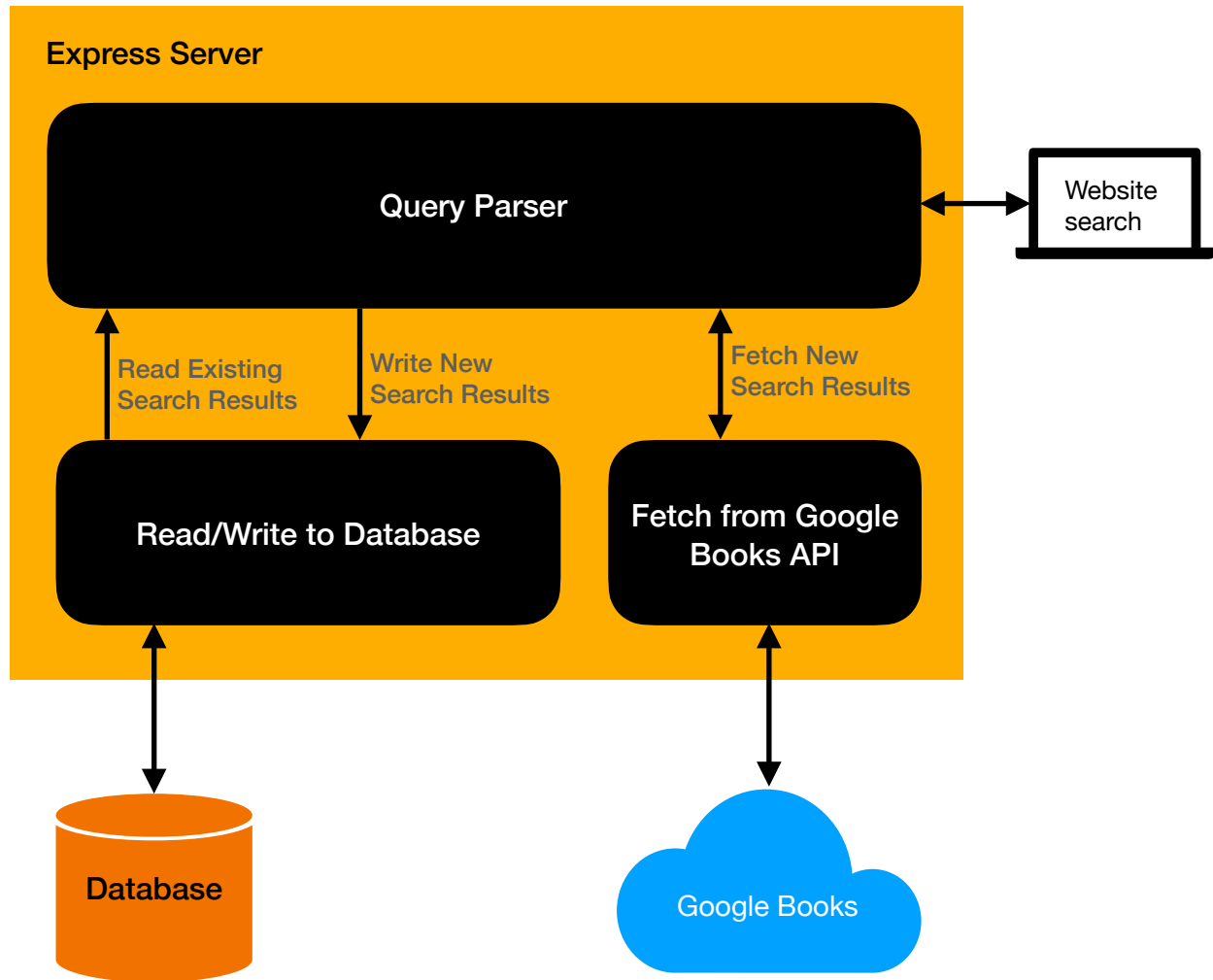


FIG 1: HIGH LEVEL SERVER ARCHITECTURE

The Server was written in **Express** with the database being implemented in **PostgreSQL**.

Web Application Layout Mockups

Below are some of the layout and user flow mockups for the web application that I prototyped before the final version that was implemented.

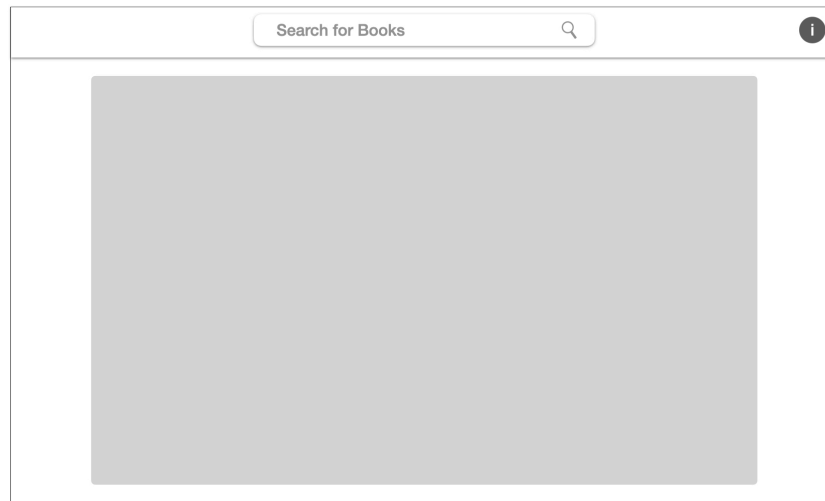


FIG 2: BASIC WEBSITE BASE LAYOUT (DESKTOP)

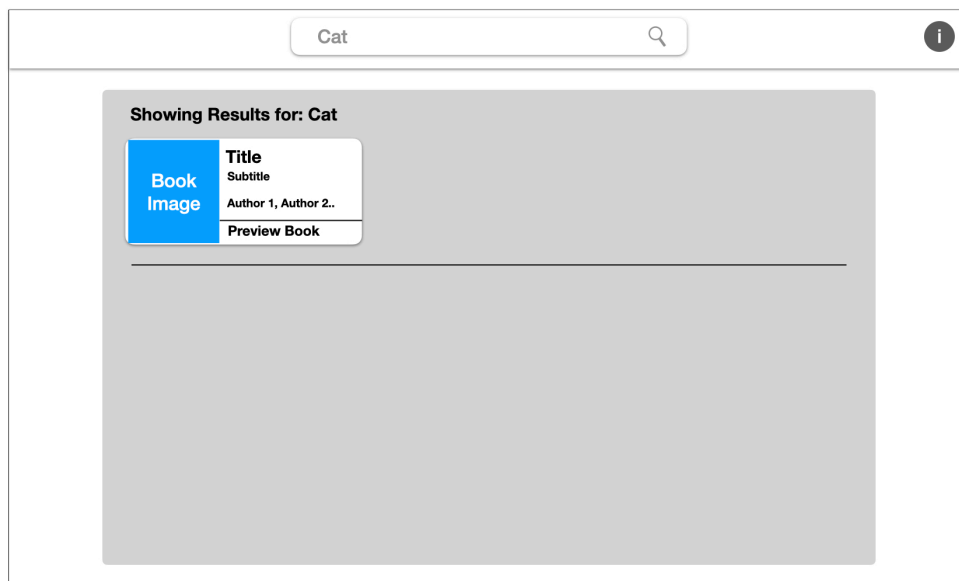


FIG 3A: WEBSITE: QUERY RESULTS DISPLAY (HORIZONTAL CARD VIEW)

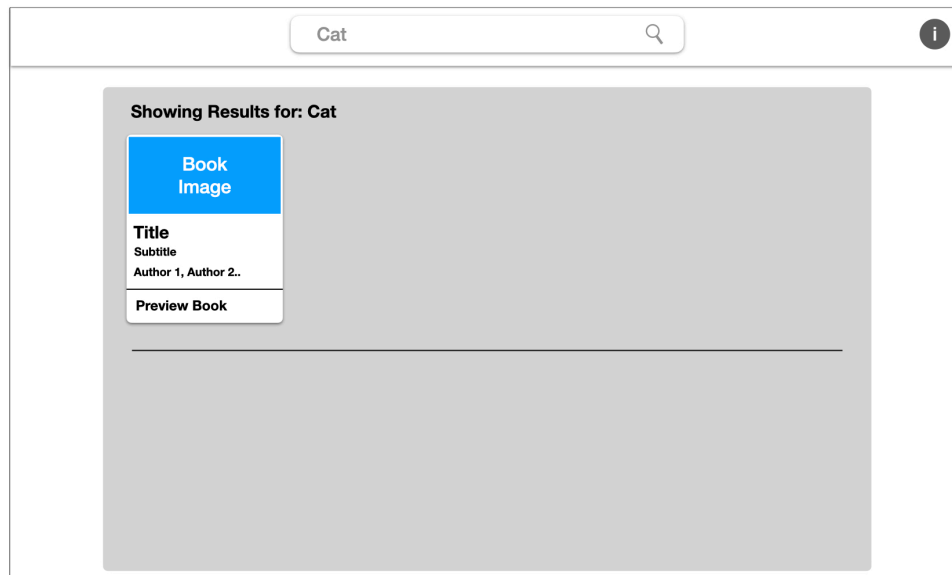


FIG 3B: WEBSITE: QUERY RESULTS DISPLAY (VERTICAL CARD VIEW)

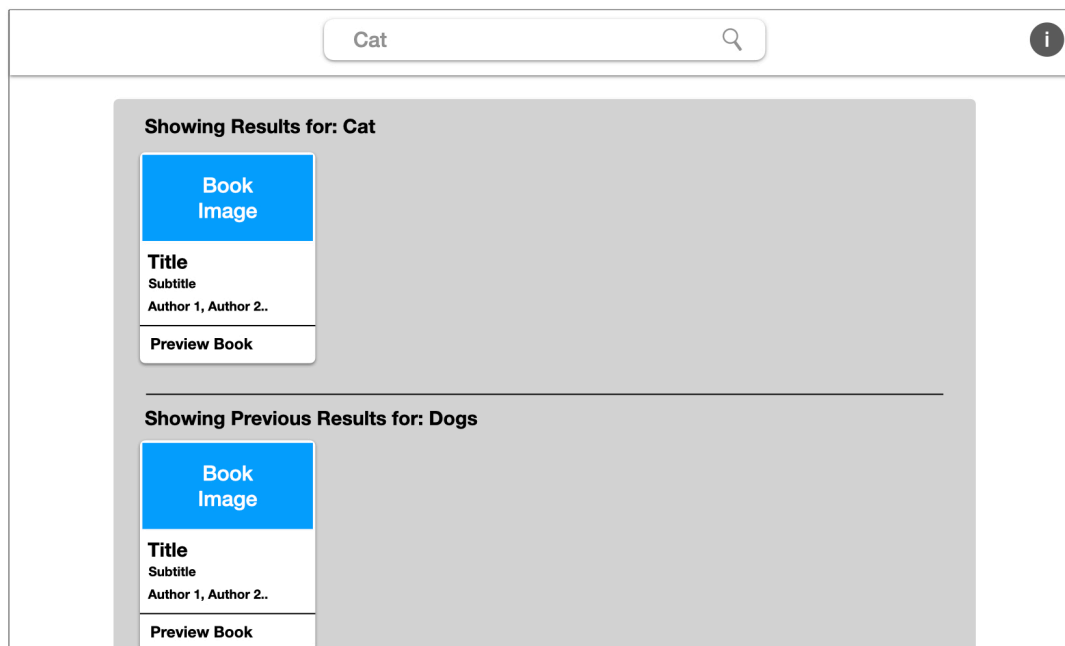


FIG 4: WEBSITE: QUERY RESULTS DISPLAY INCLUDING SEARCH HISTORY (VERTICAL CARD VIEW)

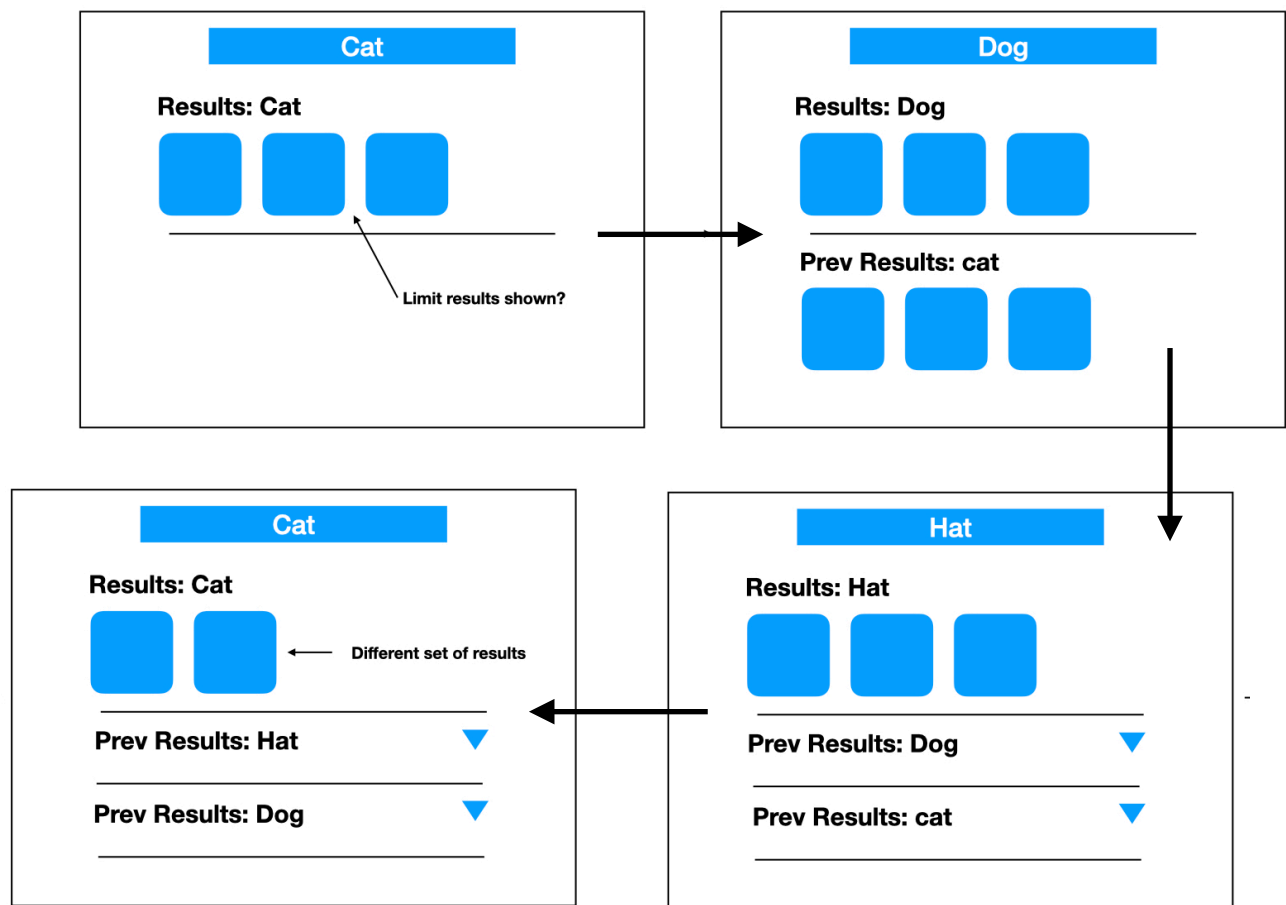


FIG 5: WEBSITE: USER SEARCH AND DISPLAY FLOW