Low Level Design Document (LLD)

BOOK FINDER APPLICATION

~By Aman Kumar Sinha

Document Version Control

Date Issued	Version	Description	Author
11-07-2024	1.0	Initial LLD	Aman Kumar Sinha
19-07-2024	1.5	Final Draft	Aman Kumar Sinha
08-08-2024	2.0	Final Version	Aman Kumar Sinha

Contents

Abstract

The Book Finder application is a user-friendly website designed to help readers discover trending books and explore a vast collection of literary works. By leveraging the Google Books API, the platform provides detailed metadata, allowing users to search for books by title or its author. Whether you're looking for the latest bestsellers or hidden gems, this application offers an intuitive way to find and learn about books that match your interests. Users can easily browse trending titles, read in-depth details, and make informed decisions on what to read next.

1. Introduction

1.1 Why this Low-Level Design Document?

The purpose of this document is to provide a detailed description of The Book Finder application. We will explain the features and purpose of the application and explain each and every component used in our project.

1.2 Scope

The primary goal of The Book Finder application is to offer a focused, distraction-free environment where users can search for and access metadata on the books they wish to read.

1.3 Constraints

We can only locate books available in the Google Books API database. Although it's an extensive catalog, this can be a limitation.

1.4 Out of Scope

Features are like login/signup and storing user data in a database for further use is out of scope for this project.

2. Technical Specifications

2.1 JavaScript Libraries

The application we're developing can be converted into a single-page application for the user, so it's advantageous to use JavaScript libraries like ReactJS for an improved developer experience. Here is the list of libraries we're using alongside ReactJS.

Library	Version
react	19.0
react-router-dom	6.25.1
react-slick	0.30.2
slick-carousel	1.8.1
tailwindcss	3.4.7
react-icons	5.2.1
framer-motion	11.3.21
postcss	8.4.40

2.2 Linters

To maintain good standards across the projects we have used JavaScript linters to weed out the potential bugs and errors in JavaScript code.

Library	Version
ES Lint	VS Code Built-in
prettier	10.4.0
prettier-plugin-tailwindcss	0.1.3

2.3 API

To fetch the meta data of the books we have used google books API provided by Google. It has a big catalogue of books available in its database. And for Trending Books we have used Draftbit API.

Name	Source
Google Books API	*https://developers.google.com/books/*
Draftbit API	*https://example-data.draftbit.com/books?_limit=10*

2.4 Deployment

We chose Vercel to host this application due to its ease of use and cost-effectiveness compared to other hosting alternatives.

Figure 1. Vercel Hosting Provider

3. Technology Stack

The Book Finder app is frontend only application we use Google books API service to fetch meta data of the books. Detailed breakdown of frontend technologies used as follows.

3.1 Frontend

ReactJS	Render application
Tailwind CSS	Styling the application

React Router DOM	Client-side routing
Framer Motion	Typing Animation
React Icons	Social Media Icons

3.2 Backend

Google Books API	To fetch meta data of books
Draftbit API	To fetch data for Trending Books

4. Proposed Solution

For the Book Finder Application, we have chosen to utilize the Google Books API to retrieve book metadata and create a distraction-free user interface that allows users to easily search for the books they are interested in.

5. Work flow as a user

As a user, you should be able to enter a search query for a book to retrieve its metadata and view the information in an intuitive and user-friendly interface.

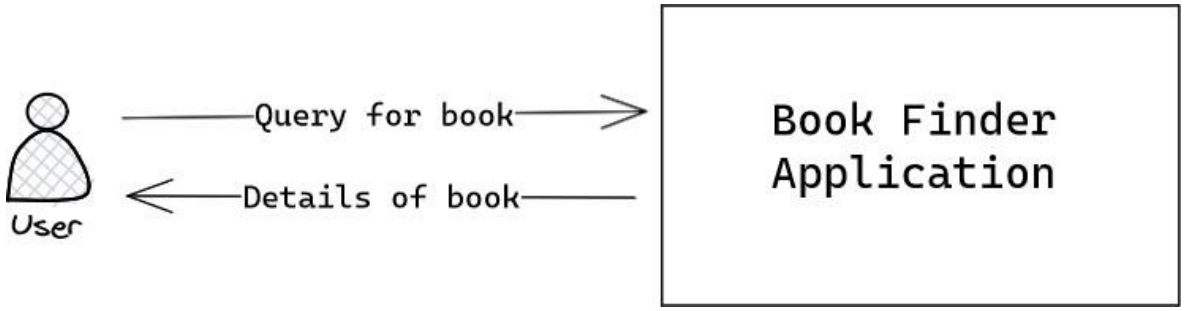

Figure 2. Workflow as a user