Low Level Design Document (LLD)

BOOK FINDER APPLICATION

By Pabitra Banerjee

Document Version Control

Date Issued	Version	Description	Author
10-01-2023	1.0.0	Final Version	Pabitra Banerjee

Contents

Document Version Control 2
Abstract 4
1. Introduction 5
1.1 Why this Low-Level Design Document? 5
1.2 Scope 5
1.3 Constraints 5
1.4 Out of Scope 5
2. Technical Specifications 5
2.1 JavaScript Libraries 5
2.2 Linters 6
2.3 API 6
2.4 Deployment 6
3. Technology Stack 6
3.1 Frontend 6
3.2 Backend 6
4. Proposed Solution 7
5. Work flow as a user 7

Abstract

The book finder application us a website where users can find the trending books right now and also, they can search for the books and view detailed information about the books. This application uses the Google Books API to fetch the meta data of the books. Using this application users can the books by the title, author and keywords in descriptions if the books.

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 main objective of The Book Finder application is to provide a distraction free environment to the user to search for the books they can read and find meta data of the books they want to read.

1.3 Constraints

We can only find the books which are present in the database of Google books API. It is huge catalogue but it can be a limitation.

1.4 Out of Scope

Features are like login 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 are building can me made into a single page application for the user so it is better to use JavaScript libraries ReactJS for better developer experience. Here is the list of libraries we are using along with ReactJS.

Library	Version
react	18.2
react-router-dom	6.4
react-slick	0.29
slick-carousel	1.8.1
tailwindcss	3.2.1
postcss	8.4.18

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	2.7.1
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

Name	Source
Google Books API	*https://developers.google.com/books/*

2.4 Deployment

To host our application, we have chosen Netlify as a hosting provider because it is free and fast for hosting small projects.

Figure 1. Netlify 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

3.2 Backend

Google Books API	To fetch meta data of books

4. Proposed Solution

For this Book Finder Application we have decided to use Google Books API to fetch meta data of books and build a distraction free UI for the users to find the books they want to search using our application.

5. Work flow as a user

As a user one should be able to input the search query of the book, they want to find meta data and able to read meta data of the books in intuitive way with simple user interface.

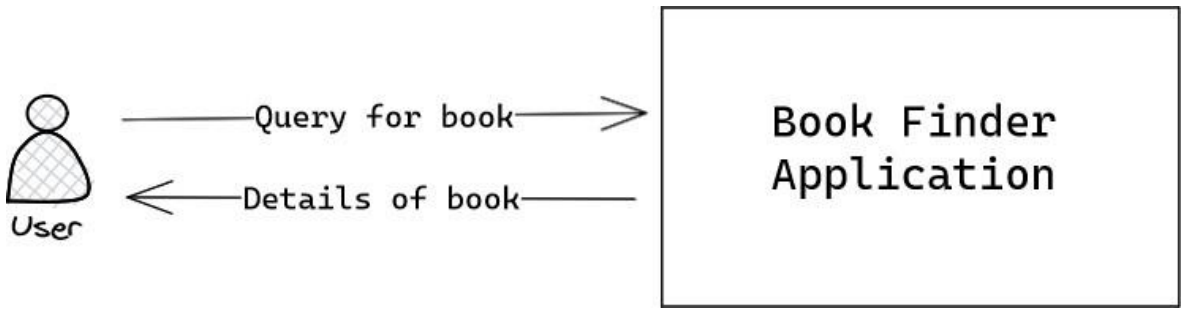

Figure 2. Workflow as a user