



# BookVerse

Curated Reads Just For You

Rishabh Lokanathan Kaushick - [kaushick.r@northeastern.edu](mailto:kaushick.r@northeastern.edu)

Sai Ram Sudheendra Sabnavisu - [sabnavisu.s@northeastern.edu](mailto:sabnavisu.s@northeastern.edu)

Tejas Hemant Choudhari - [choudhari.t@northeastern.edu](mailto:choudhari.t@northeastern.edu)

Aneesh Arunjunai Saravanan - [saravanan.a@northeastern.edu](mailto:saravanan.a@northeastern.edu)

# Table of Contents

<b>Abstract.....</b>	<b>3</b>
<b>Project Background.....</b>	<b>3</b>
<b>Requirement Gathering.....</b>	<b>3</b>
<b>Agile Kanban Methodology.....</b>	<b>3</b>
<b>Domain Driven Design.....</b>	<b>4</b>
<b>How To Run The Project.....</b>	<b>5</b>
Backend (/service folder).....	5
Frontend (/app folder).....	5
<b>Technology Stack.....</b>	<b>5</b>
<b>Frontend Components.....</b>	<b>6</b>
Material UI Components:.....	6
React Components.....	7
Internationalization and Localization.....	9
REDUX AND STATE MANAGEMENT.....	10
<b>Backend Components.....</b>	<b>11</b>
MongoDB Database:.....	11
Stripe API.....	11
GEMINI API.....	12
FUGU CAPABILITIES.....	13
PWA and CACHING:.....	14
RESTful APIs.....	16
MongoDB Collections.....	17

# Abstract

...

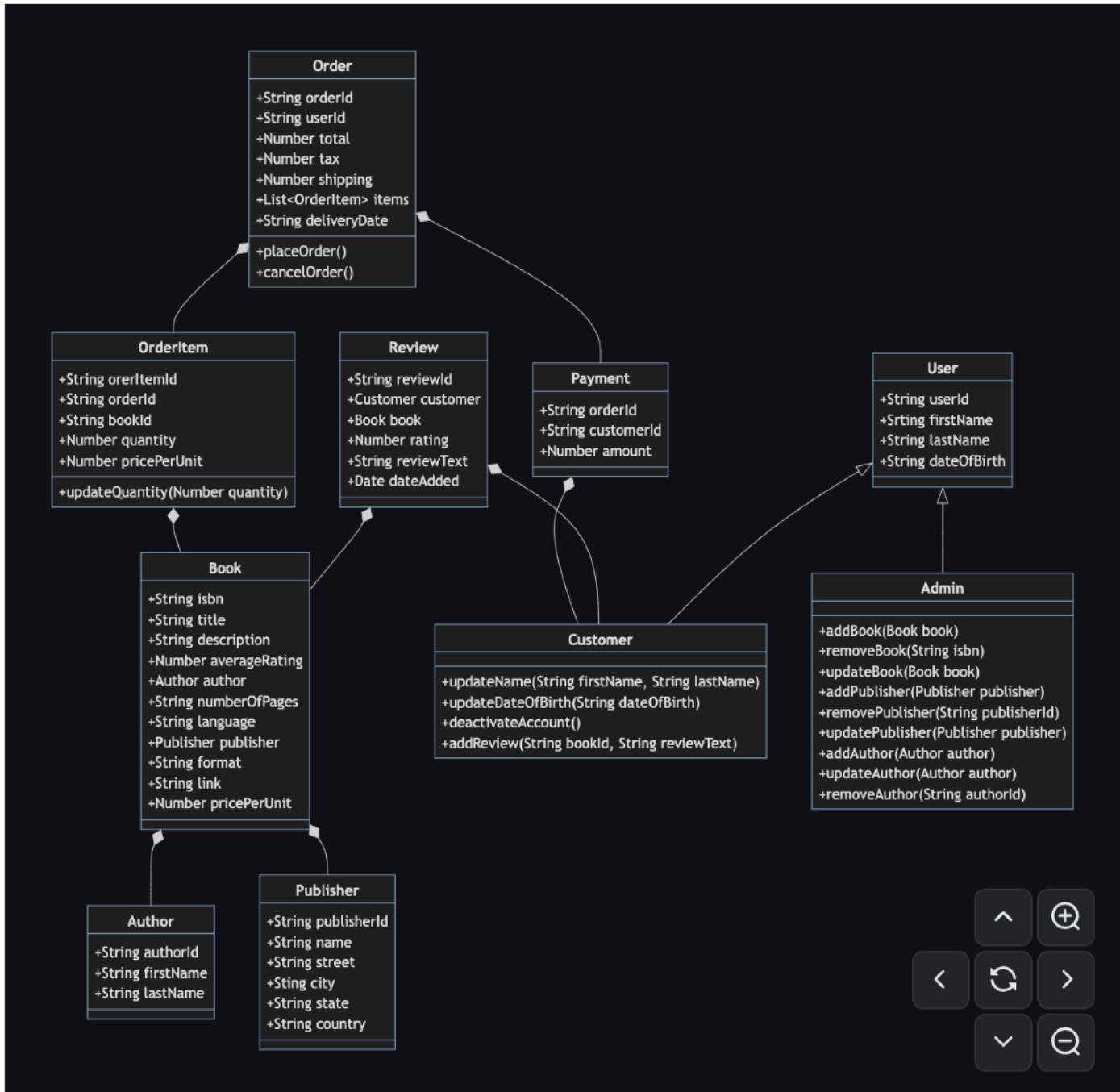
# Project Background

In this project, we have aimed to build an e-commerce book store which features a sophisticated recommendation system. Through this project we have worked on the following features & functionality:

1. Building an end-to-end application using Node.js, Express.js, MongoDB and React TypeScript.
2. Making it a PWA
3. Adding Internationalization and Localization
4. Integrating Various API's

...

# Domain Driven Design



# How To Run The Project

## Backend (/service folder)

1. Open Terminal or Command Prompt and navigate to the project directory.  
Change directory to the api folder using the command:  
`cd /service`
2. Initialize Node & installing dependencies:  
`npm install`
3. Initialize .env file with the following contents:  
`PORT = 3000`  
`MONGO_CONNECTION=<mongoDB_connection_url>`  
`JWT_SECRET=bookwave-secret-key`
4. Install python (if not present)  
`brew install python`
5. Install python in a virtual environment:  
`python3 -m venv bookVerseVenv`
6. Activate virtual environment:  
On UNIX/ MacOS: `source bookVerseVenv/bin/activate`  
On Windows: `bookVerse\Scripts\activate.bat`
7. Install python dependencies:  
`pip install -r app/src/requirements.txt`
8. Run the server:  
`npm run start`

## Frontend (/app folder)

1. Open Terminal or Command Prompt and navigate to the project directory.  
Change directory to the app folder using the command:  
`cd /app`
2. Initialize node  
`npm install`
3. Run the application  
`npm run dev`

# Technology Stack

## Frontend:

1. Vite
2. React TypeScript
3. Express

4. MongoDB
5. Material UI

## Backend:

1. Node.js
2. Gemini
3. Stripe
4. Bruno

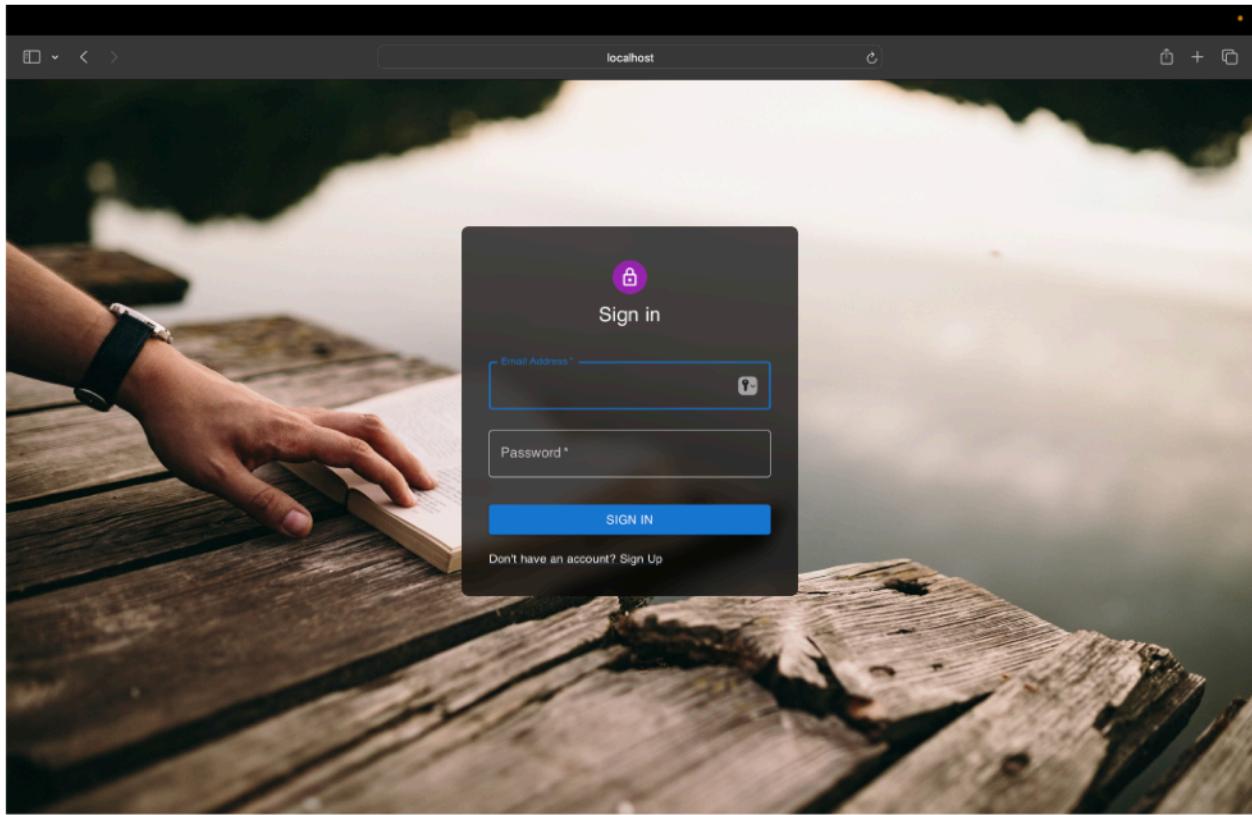
## Frontend Components

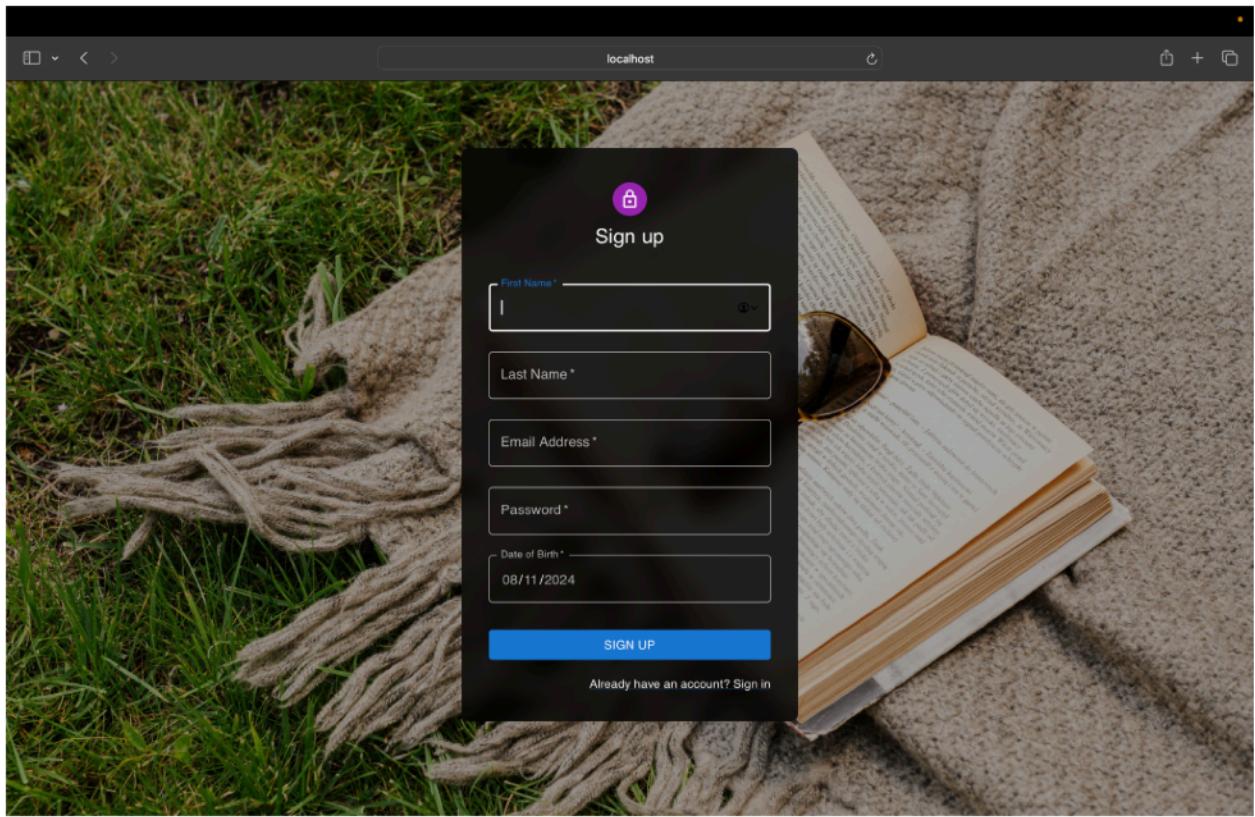
The frontend of the application is developed using the Vite framework, which offers a fast development environment and optimized build process. The UI is built with React.js and TypeScript, ensuring a component-based architecture and static type checking for improved code quality.

Internationalization is seamlessly handled with the i18n package, enabling dynamic translation of static text for multi-language support. The application also functions as a Progressive Web App (PWA), with offline capabilities, background sync, and push notifications powered by the built-in React service worker.

We have used Redux to do the state management

## Material UI Components:





## React Components

Developed UI using React

Microsoft Teams classic Edit View Window Help

Fugu Capabilities Stripe Improved Navbar v Final Project info-6150-summer Bookverse BookVerse - Google [42] Chuttamalle Sun Aug 11 9:10PM

localhost:3002/cart

BOOKS CART PROFILE ORDER HISTORY AI Chat Signout

**Under the Dome**  
Author: Stephen King  
Pricing: \$36.99  
Quantity: 1

Remove

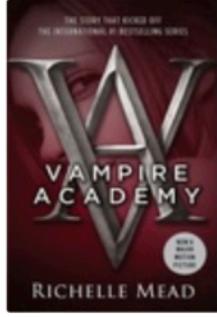
Total: \$36.99

Clear Cart Checkout

English Telugu Kannada Hindi Tamil Malayalam

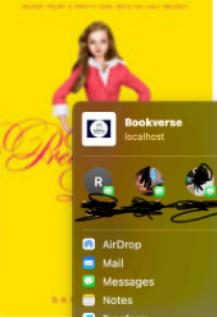
localhost

BOOKS CART PROFILE ORDER HISTORY AI Chat Signout

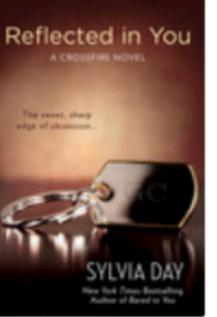
**Vampire Academy**  
(Vampire Academy, #1)  
  
THE STORY THAT INSPIRED THE INTERNATIONAL #1 BESTSELLING SERIES  
VAMPIRE ACADEMY  
RICHELLE MEAD

Author: Richelle Mead  
Pricing: 11.99

Add to Cart Details Share

**Pretty Little Liars (Pretty Little Liars, #1)**  
  
MEET THE LITTLE LIARS IN THIS NEW #1 BESTSELLER!  
Pretty Little Liars  
SARAH SHEPARD

Auth...  
P...  
Add to Cart Details Share

**Reflected in You**  
(Crossfire, #2)  
  
The newest, sharpest edge of obsession...  
REFLECTED IN YOU  
A CROSSFIRE NOVEL  
SYLVIA DAY  
New York Times bestselling author of Burned to You

Author: Sylvia Day  
Pricing: 18.99

Add to Cart Details Share

**This Girl (Slammed, #3)**  


**Born a Crime: Stories From a South African Childhood**  
TREVOR

Microsoft Teams classic Edit View Window Help

Fugu Capabilities Stripe Improved Navbar v Final Project info-6150-summer Bookverse BookVerse - Google [42] Chuttamalle Sun Aug 11 9:10PM

localhost:3002/cart

BOOKS CART PROFILE ORDER HISTORY AI Chat Sign Out

**Under the Dome**  
Author: Stephen King  
Pricing: \$36.99  
Quantity: 1

Total: \$36.99

Remove Clear Cart Checkout

English Telugu Kannada Hindi Tamil Malayalam

Chrome File Edit View History Bookmarks Profiles Tab Window Help

Voice search zoom Sun Aug 11 7:47PM

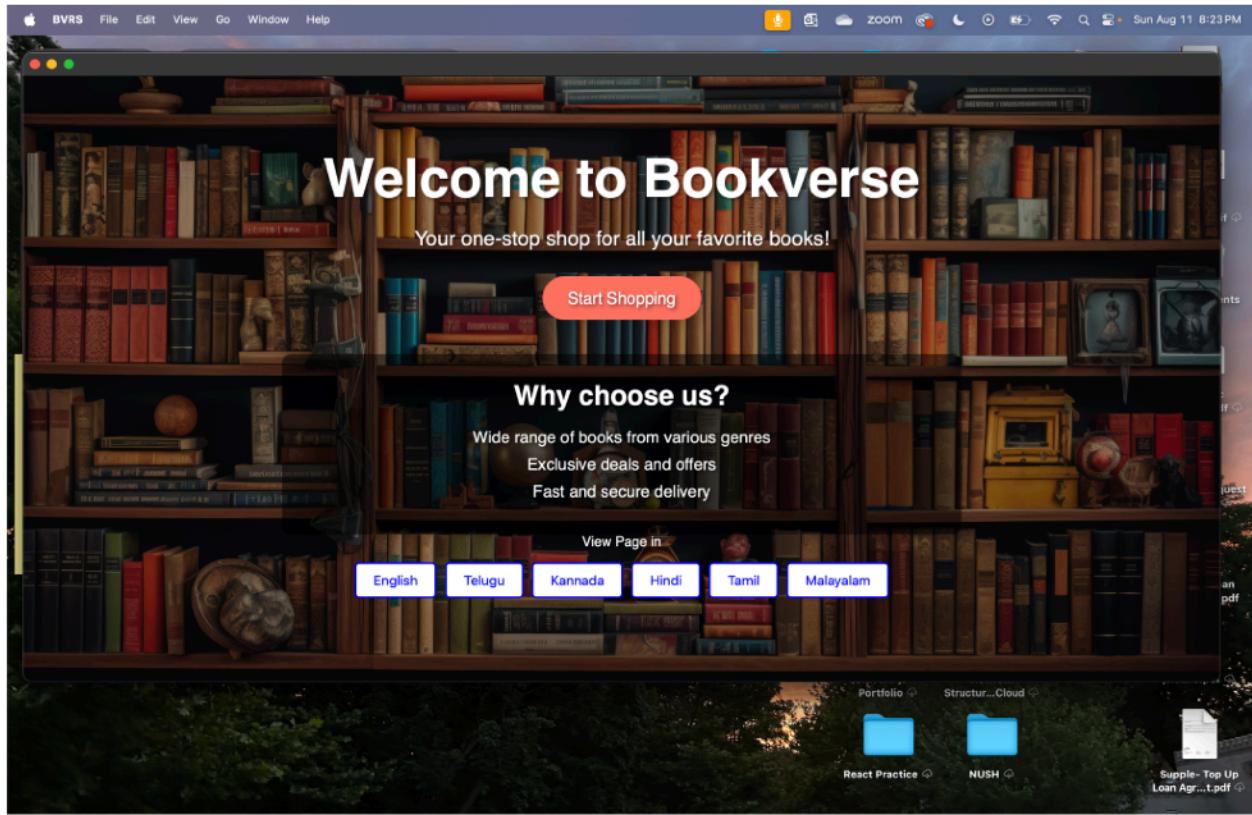
localhost:3002/admin/books

BOOKS ORDERS + ADD BOOKS Sign Out

SEARCH

Good Omens: The Nice and Accurate Prophecies of Agnes Nutter Author: Year: Price: \$46.99	Born a Crime: Stories From a South African Childhood Author: Year: Price: \$14.99	Anna and the French Kiss (Anna and the French Kiss, #1) Author: Year: Price: \$35.99	Sharp Objects Author: Year: Price: \$37.99
<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>
Betrayed (House of Night, #2) Author: Te4jas Year: Price: \$15.99	Unbroken: A World War II Story of Survival, Resilience, and Redemption Author: Year: Price: \$10.99	Just Kids Author: Year: Price: \$29.99	We Should All Be Feminists Author: Year: Price: \$48.99
<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>
I Am the Messenger Author: Year: Price: \$20.99	The Name of the Wind (The Kingkiller Chronicle, #1) Author:	The Girl Who Played with Fire (Millennium, #2) Author:	All the Missing Girls Author: Year: Price: \$20.99
<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>	<a href="#">EDIT</a> <a href="#">DELETE</a>

# Internationalization and Localization



# REDUX AND STATE MANAGEMENT

The screenshot shows a web browser window with multiple tabs open. The main content area displays a website for "BookVerse" with a banner stating "Your one-stop shop for all your favorite books!" and a "Start Shopping" button. Below the banner, there's a section titled "Why choose us?" with three bullet points: "Wide range of books from various genres", "Exclusive deals and offers", and "Fast and secure delivery". The background of this section features a photograph of a well-stocked bookshelf.

The browser's address bar shows the URL `localhost:3002`. The DevTools sidebar on the left lists various storage types: Manifest, Service workers, Storage, Local storage, Session storage, IndexedDB, Cookies, Private state tokens, Interest groups, Shared storage, and Cache storage. Under Cache storage, there are entries for "workbox-precache-v2-h" and "static-assets - http://loc...".

The Application tab in DevTools is active, showing network requests for the domain `http://localhost:3002`. The table below lists four requests:

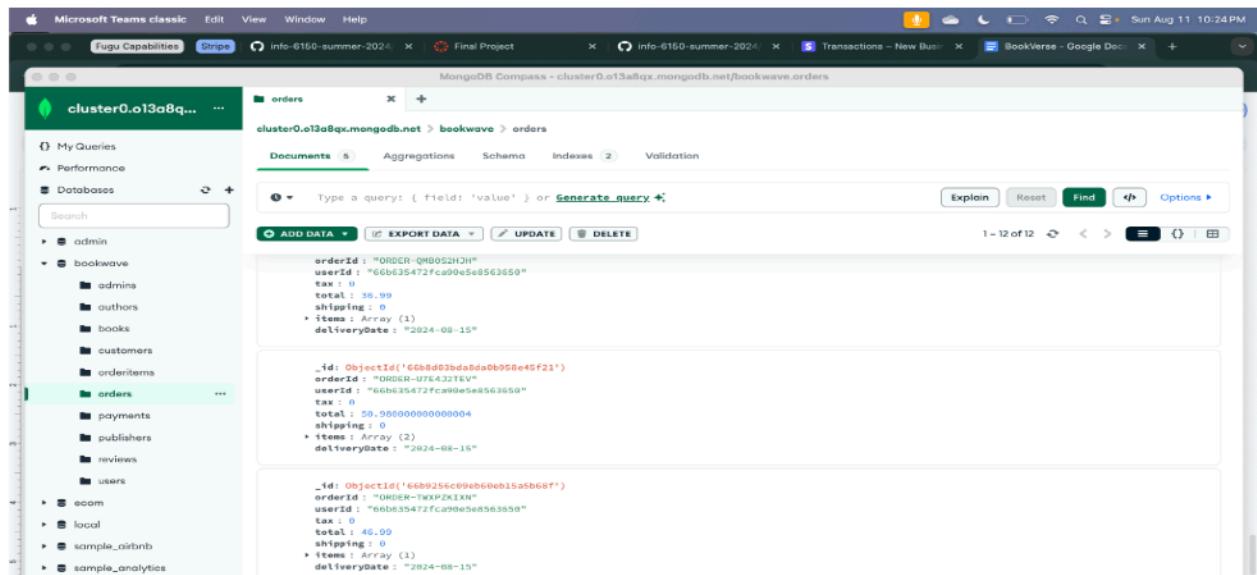
#	Name	Response-Type	Content-Type	Content-Length	Time Cached	Vary Header
0	/books	cors	application/json	816,675	8/11/2024, 7:21:58 PM	
1	/books/66b1979aceb447f6cb44cd8c	cors	application/json	2,474	8/11/2024, 7:22:10 PM	
2	/books/66b1979aceb447f6cb44cd92	cors	application/json	1,031	8/11/2024, 7:23:09 PM	
3	/books/66b1979aceb447f6cb44cd95	cors	application/json	1,490	8/11/2024, 9:55:30 ...	

# Backend Components

We have developed the backend server using JavaScript, with libraries such as Node.js, Express.js and Mongoose (for connectivity with MongoDB). Additionally, we have used a novel recommendation system implemented in Python which we invoke from the REST API. We have utilized Stripe API and Gemini API to integrate to our project

## MongoDB Database:

We have multiple Collections for multiple purposes



The screenshot shows the MongoDB Compass interface connected to a cluster. The left sidebar lists databases: admin, bookwave, admins, authors, books, customers, orderitems, orders, payments, publishers, reviews, users, ecom, local, sample\_airbnb, and sample\_analytics. The 'orders' database is selected. The main pane displays the 'orders' collection with three documents listed. Each document has fields: \_id, orderId, userId, tax, total, shipping, items, and deliveryDate. The first document's details are expanded:

```
_id: ObjectId('66b925cb9eb60b0b5b6ef')
orderId: "ORDER-QM05G2H0H"
userId: "66b635472fc99e5e8563650"
tax: 0
total: 36.99
shipping: 0
items: Array (1)
deliveryDate: "2024-08-15"
```

The second document's details are partially visible:

```
_id: ObjectId('66b925cb9eb60b0b5b6ef')
orderId: "ORDER-UT64Z2TEV"
userId: "66b635472fc99e5e8563650"
tax: 0
total: 50.98000000000004
shipping: 0
items: Array (2)
deliveryDate: "2024-08-15"
```

The third document's details are partially visible:

```
_id: ObjectId('66b925cb9eb60b0b5b6ef')
orderId: "ORDER-TWXPZKIXN"
userId: "66b635472fc99e5e8563650"
tax: 0
total: 46.09
shipping: 0
items: Array (1)
deliveryDate: "2024-08-15"
```

## Stripe API

Utilized Stripe API for Checkout

The screenshot shows the Stripe dashboard in 'Test mode'. The left sidebar includes links for New Business, Home, Balances, Transactions (which is selected), Customers, and Product catalog. Under Products, there are sections for Payments, Billing, Reporting, and More. The main area displays a table of transactions with the following columns: Amount, Payment method, Description, Customer, and Date. The table shows 22 succeeded transactions, all of which are Refunded. The transactions are listed with amounts ranging from \$37.99 to \$100.00 and dates from August 11 to August 10.

Amount	Payment method	Description	Customer	Date
\$37.99 USD	Succeeded ✓	pi_3Pml7DLKcdm260BZ1Wt2Iy4J	sabnavisu.s@northeastern.edu	Aug 11, 7:24 PM ...
\$15.99 USD	Succeeded ✓	pi_3PmjKcLKcdm260BZ0r7ujSqT	sabnavisu.s@northeastern.edu	Aug 11, 5:29 PM ...
\$36.99 USD	Succeeded ✓	pi_3PmVULKcdm260BZ1PbzPXiU	sabnavisu.s@northeastern.edu	Aug 11, 2:43 AM ...
\$46.99 USD	Succeeded ✓	pi_3PmVJ9LKcdm260BZ1kV5Xj	sabnavisu.s@northeastern.edu	Aug 11, 2:31 AM ...
\$62.98 USD	Succeeded ✓	pi_3PmVHSLKcdm260BZ0kElMmC6	sabnavisu.s@northeastern.edu	Aug 11, 2:29 AM ...
\$100.00 USD	Succeeded ✓	pi_3PmSt0LKcdm260BZ0oB0ZITI	sabnavisu.s@northeastern.edu	Aug 10, 11:56 PM ...
\$600.00 USD	Succeeded ✓	pi_3PmM2fLKcdm260BZ16NdojtY	sabnavisu.s@northeastern.edu	Aug 10, 4:37 PM ...
\$100.00 USD	Succeeded ✓	pi_3PmLznLKcdm260BZ0zXaJ8I7	sabnavisu.s@northeastern.edu	Aug 10, 4:34 PM ...
\$300.00 USD	Succeeded ✓	pi_3PmLt6LKcdm260BZ0epimCOK	sabnavisu.s@northeastern.edu	Aug 10, 4:27 PM ...
\$600.00 USD	Succeeded ✓	pi_3PmLnJLKcdm260BZ1Fvb82Dm	sabnavisu.s@northeastern.edu	Aug 10, 4:21 PM ...
\$100.00 USD	Succeeded ✓	pi_3PmLzXLKcdm260BZ19AvV3gL	sabnavisu.s@northeastern.edu	Aug 10, 4:07 PM ...
\$100.00 USD	Succeeded ✓	pi_3PmLI2LKcdm260BZ1z0TEHQ	sabnavisu.s@northeastern.edu	Aug 10, 3:49 PM ...
\$100.00 USD	Succeeded ✓	pi_3PmLFoLKcdm260BZ0Irk8qfC	sabnavisu.s@northeastern.edu	Aug 10, 3:47 PM ...

## GEMINI API

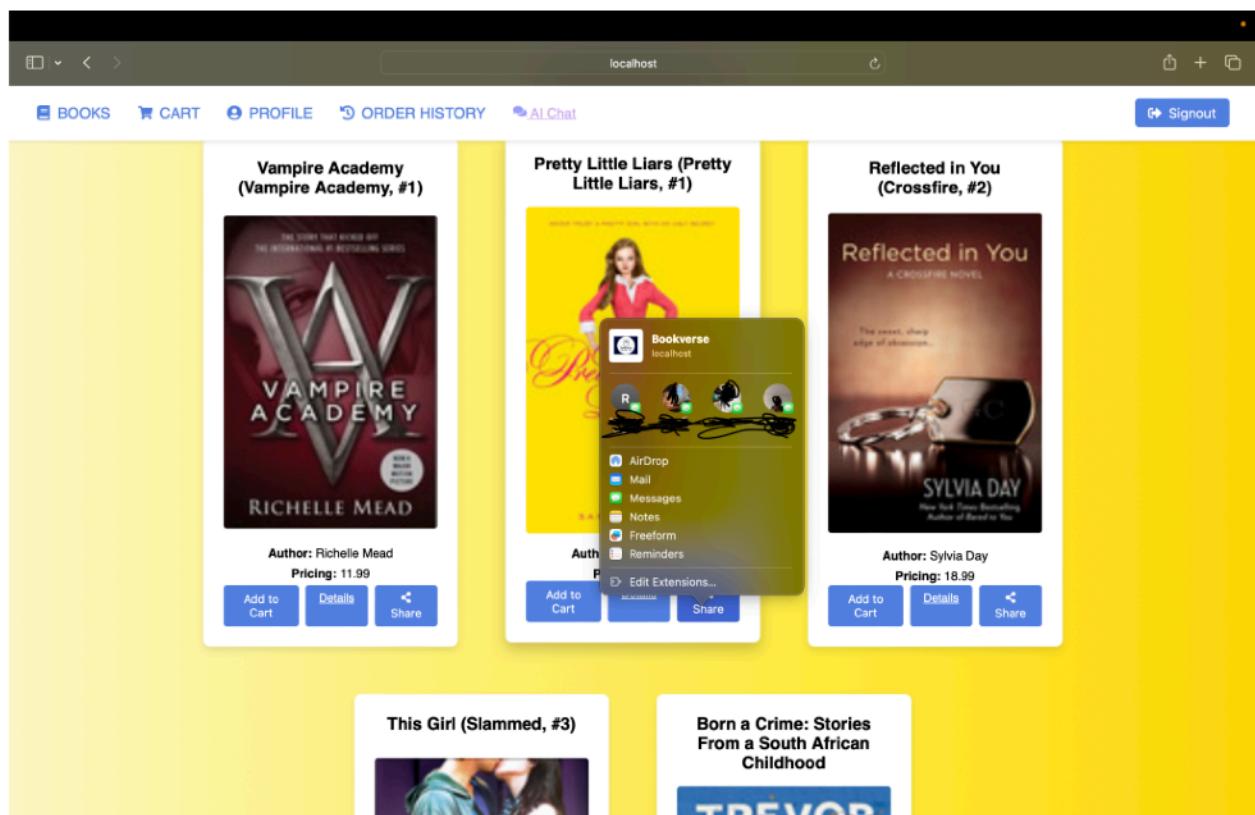
```

ts AIChatService.ts ×
app > src > services > TS AIChatService.ts > [e] getBookRecommendations
  2
  3 const API_KEY = 'AIzaSyDztI2-qCP2WB3MtpjSeavCa1Y7aB-CsD8';
  4
  5 const genAI = new GoogleGenerativeAI(API_KEY);
  6 const model: GenerativeModel = genAI.getGenerativeModel({ model: "gemini-pro" });
  7
  8 export const getBookRecommendations = async (prompt: string): Promise<string> => {
  9   try {
10     const result = await model.generateContent([{ text: prompt }]);
11     const response = await result.response;
12     const text = await response.text();
13     return text;
14   } catch (error) {
15     console.error("Error generating AI response:", error);
16     throw error;
17   }
18 };
19

```

## FUGU CAPABILITIES

Utilized web share API to share the books like a native application to Operating System specific shares



## PWA and CACHING:

Defined service workers to make application Installable and cacheable which helps the application to run even while server is down

#	Name	Response-Type	Content-Type	Content-Length	Time Cached	Vary Header
0	/books	cors	application/json	815,675	8/11/2024, 7:21:58 PM	
1	/books/66b1979aceb447f6cb44cd08c	cors	application/json	2,474	8/11/2024, 7:22:10 PM	
2	/books/66b1979aceb447f6cb44cd092	cors	application/json	1,031	8/11/2024, 7:23:09 PM	
3	/books/66b1979aceb447f6cb44cd095	cors	application/json	1,490	8/11/2024, 9:55:30 ...	



## RESTful APIs

To view the Open API specification for this project, please navigate to docs folder.