Web Programming 1, Semester 2 - 2025

**Angular Project Report**

1. **Initial design proposal**

**Mobile**

Home Page Favourite Page

**A paper with lines and text

AI-generated content may be incorrect. A graph paper with lines and text

AI-generated content may be incorrect.**

**Desktop**

Home Page

**A paper with writing on it

AI-generated content may be incorrect.**

Favourites Page

**A paper with lines and symbols

AI-generated content may be incorrect.**

In the original plan, I wanted to display the book details on the same page as the search results. I thought I wouldn’t be able to have the Book Details on its page. In the final version, I added a separate page for book details. Now, after clicking on a book, the user is taken to a new page with more information. I also changed the search bar and filters. There are fewer filters now because I did not need all of them. I added a complex footer too. On mobile, books are shown one under another as cards. Apart from these changes, the application is mostly the same. I have also added an About page with some basic info about the app.

1. **Component diagram**

A diagram of a book card

AI-generated content may be incorrect.

I have used seven components: Footer, Navbar, Search, About, Favourites, BookCard, and Book Details. The Navbar and Footer appear on every page and are directly included in the AppModule. The Search and Favourites pages use the BookCard components to display books. When a user clicks on a BookCard, @Input receives data, and @Output navigates to the Book Details page.

1. **Architecture diagram**

**A diagram of a computer server

AI-generated content may be incorrect.**

The front end is built with Angular and Bootstrap and communicates with two services -**BooksApiService, which fetches** book data from the external **Google Books API**, and **FavoriteBookApiService, which manages** user favourites via CRUD operations through an **Express Server**. The server interacts with a **Mongodb** database to store the user's favourite books.

1. **Screenshots of the main elements**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a book finder

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a book

AI-generated content may be incorrect.**

1. **Links to my hosted application on AWS**

S3 - <http://s00255627atusligo.s3-website-eu-west-1.amazonaws.com/>

EC2 - <http://176.34.150.80:5050/books>

1. **Project reflection**

This project went quite well for me overall. I faced some struggles, especially when it came to working with interfaces. I had a hard time figuring out how to properly type both the data coming from the Google Books API and the favourites I was storing in Mongodb. At first, I wasn’t sure if I should put them into one interface or split them, which caused a lot of confusion. In the end, I created two interfaces for them. One for the API response and one for my Mongodb database.

Another challenge was keeping track of the app's overall structure as it grew. There were multiple pages, services, interfaces, and shared components. However, I feel I got much better at organising code, thinking through data flow, and consistently naming things.

I also tried to implement standalone components instead of traditional ones and added a toggle-favourite feature right on each book card. Clicking the heart icon stops navigation, emits an add or remove event, calls the backend, and updates the UI before the server responds.

Overall, I’m very happy with how much I learned about Angular during the development of this app.