PROJECT DOCUMENTATION

APPLICATION NAME: SHOPPING CART

Team Members:

- Shunmika. C
- Nandhini. S
- Gobika. T
- Jenifer Jose. R

TABLE OF CONTENTS:

Description:

The shopping cart is an application that enables the user to purchase products from online sellers. It is a one stop platform where the user can buy products from sellers across the world. This app has wide range of commodities so it would be an ease to shop more products at one place in lesser time.

App Info:

This shopping cart application is designed by using:

1. Ionic framework

2. Angular

3. Visual Studio code

4. Node.js

5. Mongo DB

Building Tools:

Ionic framework:

Ionic Framework is an open source UI toolkit for building performant, high-quality mobile and desktop apps using web technologies — HTML, CSS, and JavaScript — with integrations for popular frameworks like Angular and React. We have learned how to implement these frameworks to our mobile application through this internship. It helped us in focusing on the frontend UI interaction of our app.

We have used the following ionic components in our app:

1. Ion-item

2. Ion-list

3. Ion-label

4. Ion-card-header

5. Ion-card-content

6. Ion-button

7. Ion-fab-button

8. Ion-icon

9. Ion-input

10. Ion-card-title

11. Ion-row

12. Ion –col

Angular:

Angular is an application design framework and development platform for creating efficient and sophisticated single-page apps. Angular is written in TypeScript. In our application it helped us in navigating between various pages and storing and displaying product details from Mongo DB.

Routing is made possible by using the following code:

```
import { ActivatedRoute,Router } from '@angular/router';
```

The following command is used to perform HTTP requests:

```
import {HttpClientModule,HttpClient} from '@angular/common/http';
```
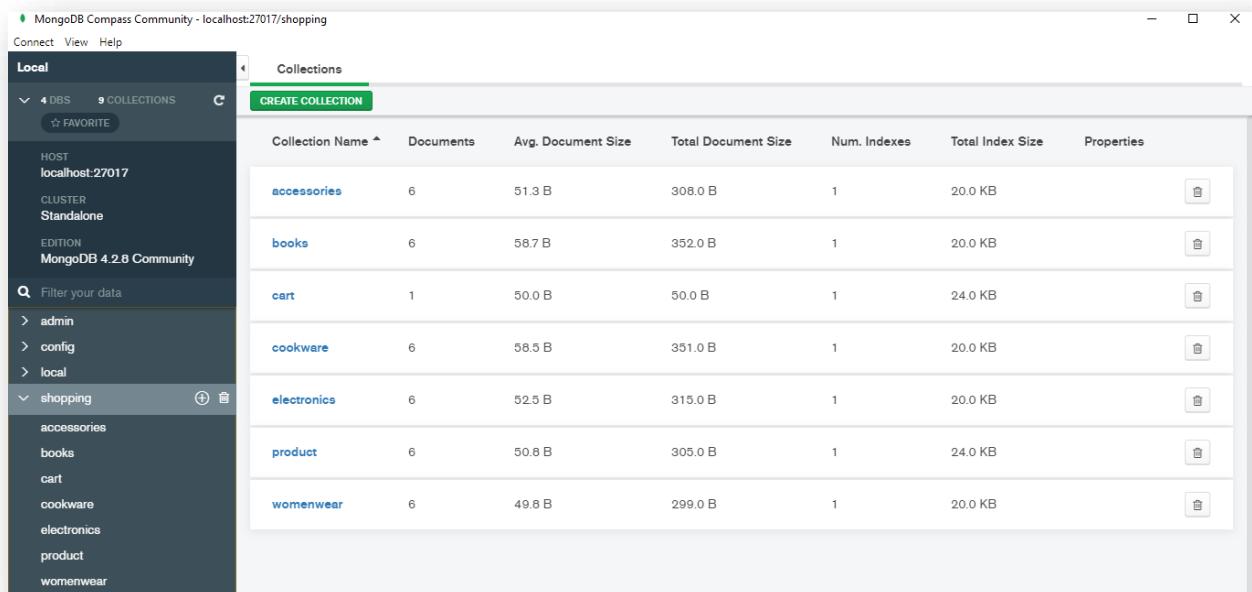
Node.js:

Node.js is an open source server environment. Node.js lets developers use JavaScript to write command line tools and for server-side scripting. In our application it helps to get and post product info in Mongo DB.

Visual Studio code:

Visual Studio Code is a source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js .

Mongo DB:

MongoDB is a distributed database. It stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time. The document model maps to the objects in your application code, making data easy to work with. In our application we have created different collections to store the entire product details which include its name, price and the user info who add it to the cart.

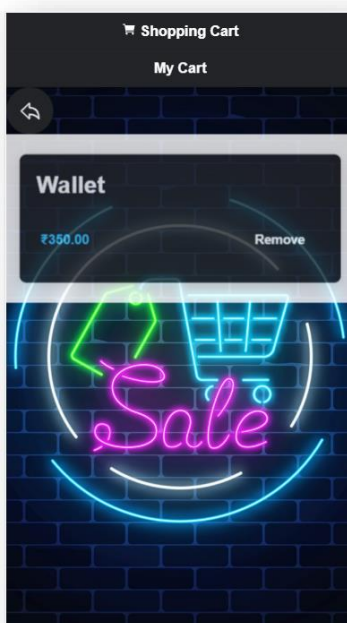| Collection Name ▲	Documents	Avg. Document Size	Total Document Size	Num. Indexes	Total Index Size	Properties
accessories	6	51.3 B	308.0 B	1	20.0 KB	
books	6	58.7 B	352.0 B	1	20.0 KB	
cart	1	50.0 B	50.0 B	1	24.0 KB	
cookware	6	58.5 B	351.0 B	1	20.0 KB	
electronics	6	52.5 B	315.0 B	1	20.0 KB	
product	6	50.8 B	305.0 B	1	24.0 KB	
womenwear	6	49.8 B	299.0 B	1	20.0 KB	

APP FLOW:

1. Login Page

2. Categories Page

4. Cart page

3. Respective Product Page
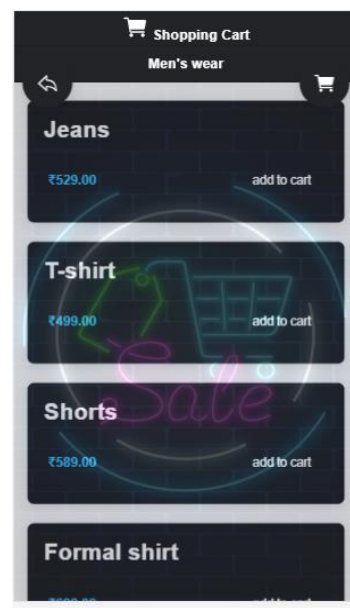
UI and server side:

1. Login Page:

Background	• Set using .scss page of login page
UI	• Username and password is received from user by ion-input component • Icons are designed using ion-icon
Navigation	• Routing from login page to categories page is done using angular command in login page.ts

2. Categories Page:

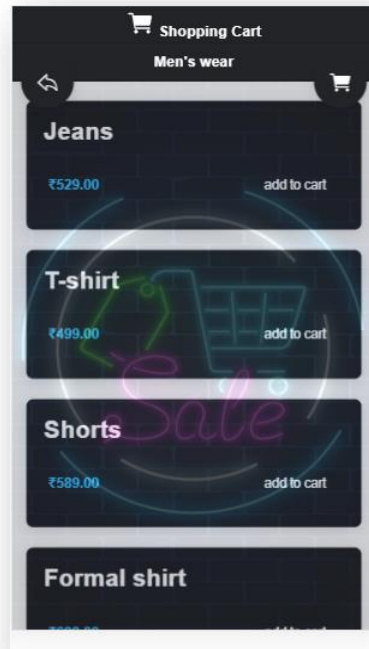| Background	• Set using .scss page of Categories page
UI	• Ion-fab-button are used to navigate for the respective product page and cart page • Icons are designed using ion-icon
Navigation	• Routing from categories page to product pages and cart page is done using angular command in categories page.ts

3. Product Page: (UI is similar for every page)

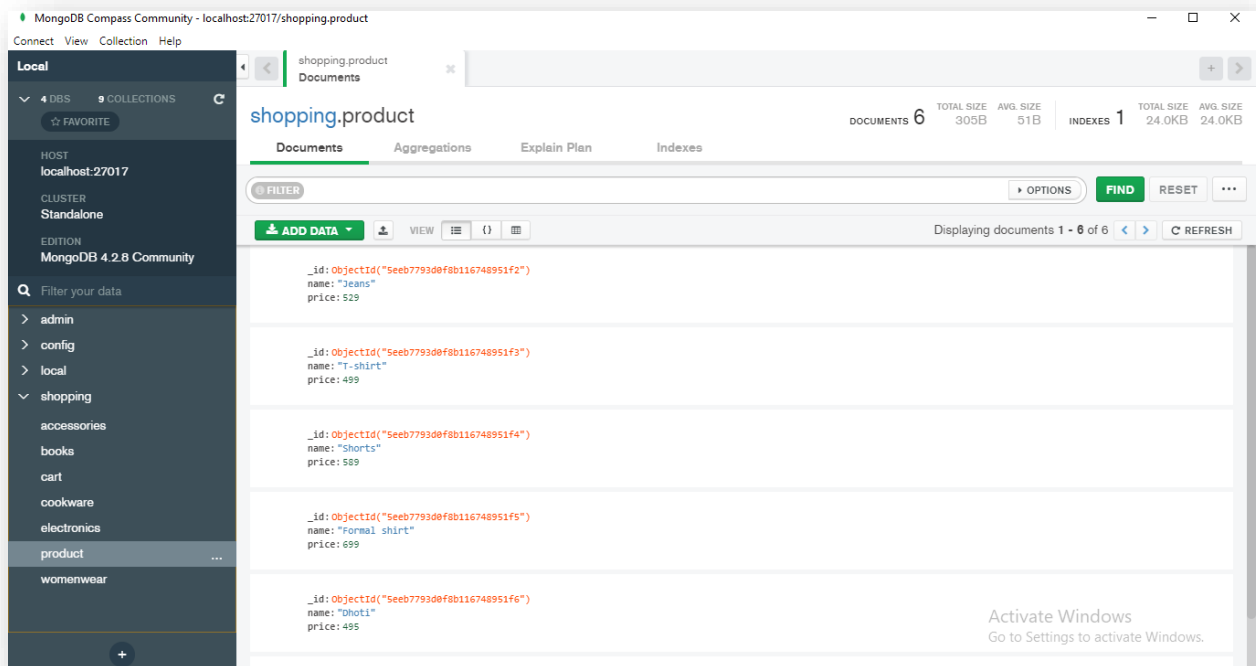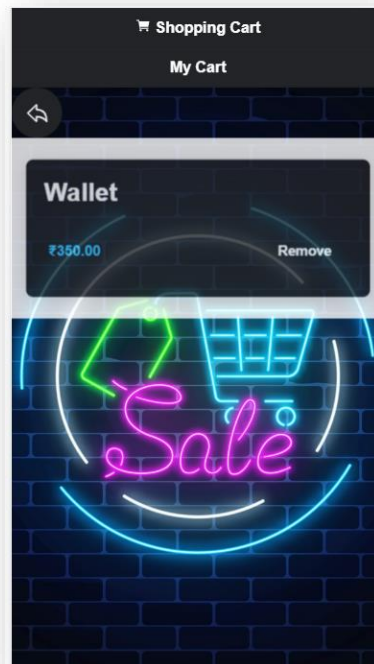| UI	• Product details are displayed using ion-card-header • Icons are designed using ion-icon • A back & cart button is kept to navigate back to the categories & cart page respectively.
Product info	• Product details (name and price) is stored using mongoDB community server.
Add to cart	• The 'add to cart' button pushes the corresponding product details to the cart page. This functionality is done with the help of node.js.

Product Info DB:

In our application we have created 6 product pages. Product details of each page are placed under different collections in MongoDB. Each product is given a name and price along with which an id is generated.

4. Cart Page:

| UI	• Product details are displayed using ion-card-header • Icons are designed using ion-icon • A back button is kept to navigate back to the categories page • Background is kept in cart.scss page
Product info	• Product details are taken from the respective products collection and is added to the cart collection in mongoDB
Remove	• Once the 'Remove ' button is clicked, corresponding product will be deleted from the Cart page as well as in cart collection

Cart Collection:

  In the cart collection the product added to the cart along with the user name is stored. By using this user name as a key the products that are displayed in the cart becomes user specific. In the similar way removing a product is also carried out.
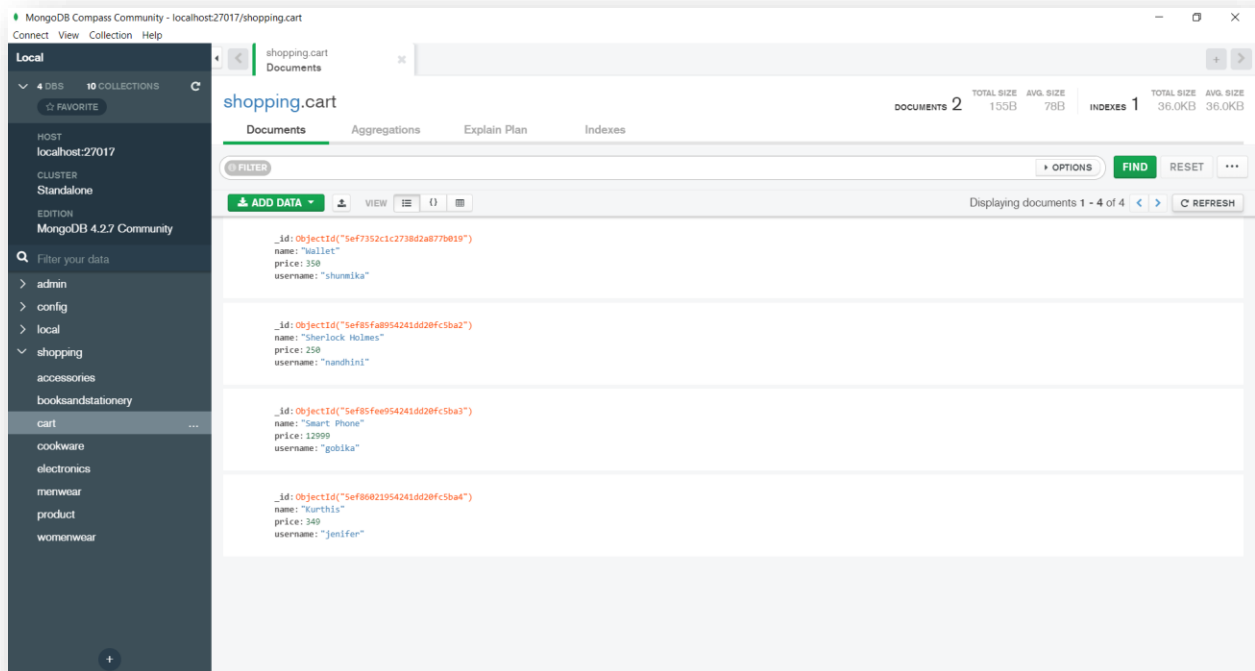
Conclusion:

  The Shopping Cart that we have developed helps the user to shop with ease. During the development of the Shopping Cart App, we were able to explore through different platforms like Ionic, GitHub, etc... We got real-time experience in application development.