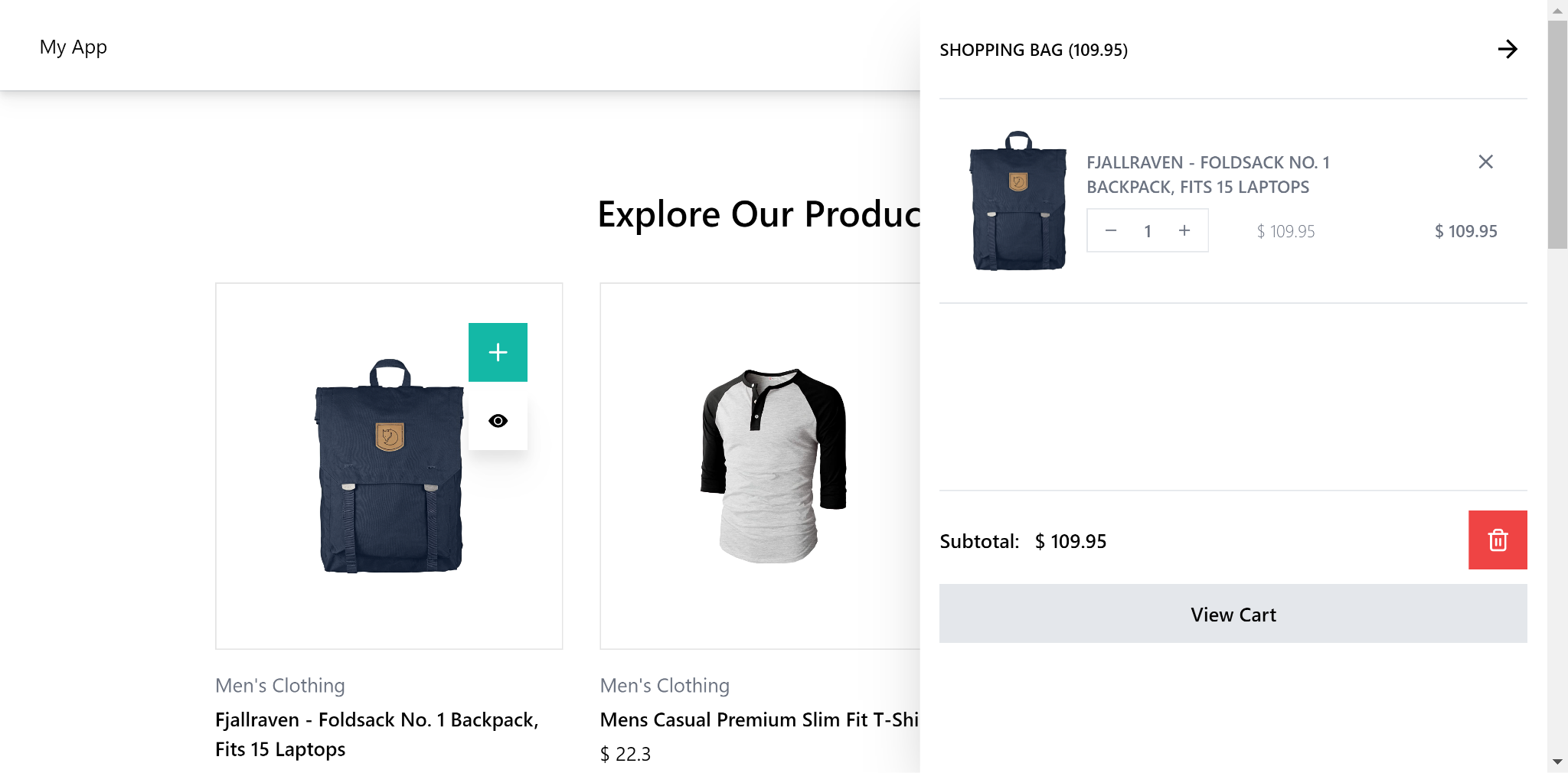
**Ecommerce Web App**

The product is an ecommerce web app for listing the products and users should be able to purchase and manage their orders.



Git repo : [repo](https://github.com/ManuSunny18/manu-ecommerce-app-test)

***Technologies***

* [Typescript](https://www.typescriptlang.org/)
* [React](https://react.dev/)
* [redux](https://redux.js.org/)
* [tailwindcss](https://tailwindcss.com/)
* [webpack](https://webpack.js.org/)
* [jest](https://jestjs.io/)
* [sass](https://sass-lang.com/)
* [RTK Query](https://redux-toolkit.js.org/rtk-query/overview)

***Integration***Node version above 18 is required to run the project. Once you install the dependencies npm start will start the web application in development mode.

***API***A dummy fakestore api is used to list and show the products. You will be able to find the service setup in api folder

***Folder Structure***

Root  
│ ───package.json # npm dependencies and run scripts  
│ ───Readme.md # Documentation  
│───public/index.html # Entry point of application  
│ ───src # Main application folder  
│ ├───main.scss # Application stylesheet  
│ ├───App.tsx # App component  
│ │───index.tsx #React starting point  
│ ├───components # UI Components used specific to the application

│ ├───common # UI Components for the design system (modal, loaders, accordion, dropdowns etc)  
│ ├───api #api configuration and hooks  
│ ├───store #redux store set up  
│ │ ├───slice # sub redux stores  
│ ├───assets # Static assets like images and fonts  
│ ───doc # Documentation data  
│───node\_modules # Dependencies

│───jest #Test suite setup  
│ ├───tailwind config # style variables and responsive criterias

│───webpack # build configurations

***Technical decisions and improvement***

The current setup is basic and it's a good starting point for starting the development. However it is designed in such a way that we can scale it further without much trouble. The production set up needs more work for caching, asset optimisations, better js/css splitting and static page generations.

Need to work on the env variables for ci/cd and application configs specific to the environments.

We have used tailwind as our base css set up, which will help us to develop a ui component system and better css system. As you can see for the demo we haven't used much extra css. All are from tailwind.

Redux is used for global state management and RTK query is used for api management. RTK query is set up is minimal and possible to enhance with caching.

***Current Setup***

* React for UI
* Redux store and RTK query for the state management and api services.
* Complete control over webpack for dev and production builds.
* Analyze set up for bundles
* typescript for consistent codebase
* Tailwind is used for styling and it is highly scalable due to its atomic design.
* React router with lazy loading features for better performance and route based chunking.
* Configurable Test cases setup using the latest react testing library and jest with complete HTML coverage report.

***What is next***

* Add more features to RTK Query. Including caching, mutation etc
* Redux store and action testing
* More types
* Common component developments with tailwind and framer motion
* Better route configuration for complex case
* Error boundary component for logging the errors
* PWA for assets caching and better user experience
* Responsive configurations for tailwind
* Eslint configurations
* Asset optimisations with webpack
* Gzip compression for webpack
* Virtualized scrolling for product listing

Read more here in [github](https://github.com/ManuSunny18/manu-ecommerce-app-test?tab=readme-ov-file#table-of-contents)