Additional Packages used

1. Material UI core and Material UI Icons – for Icons
2. react-router-dom – used for navigating to other pages
3. react-currency-format – used to format curreny

Snippets:

1. To print a component multiple time,

{Array(count).fill().map(() => ( <p>⭐</p> ))}

1. Click on image and redirect to a page (using {Link} from react-router-dom)

<Link to='/checkout'>

<img className='header\_\_logo' src={logo} />

</Link>

1. Navigation in the Header Component using react-router-dom

import { BrowserRouter as Router, Route, Routes } from "react-router-dom";

<Router>

        <Routes>

          <Route path="/login" element={<Login />} />

          <Route path="/checkout" element={<Checkout />} />

          <Route path="/" element={<Home />} />

        </Routes>

</Router>

Using Context API / Redux:

* When the “Add to basket” is click the product data will pushed to “data layer”
* This data layer is React context API / Redux
* From this data layer we can get the info to other components (i.e., Cart)

Refer StateProvider.js and reducer.js for Context API Code

**StateProvider.js** is a template which created the data layer and provides access to data layer

**reducer.js** is used to manipulate the data layer

**reducer.js** will contain the contain for adding the item to cart and delete the item from cart

**Firebase Setup:**

1. Create new Project in firebase and create a new web app in the project
2. Navigate to Project Settings and scroll down to **SDK setup and configuration**
3. Click on **Config** and copy the configuration object shown
4. Create a new file **firebase.js** in project folder and paste it
5. Install firebase in the project folder using the commands

For the first time use this to install firebase globally:

**(npm install -g firebase-tools)**

Next, In the project directory,

**npm i firebase**

**Check this link for user register and sign in, sign out process using firebase**

<https://firebase.google.com/docs/auth/web/start>

Deploy app in firebase:

firebase login

firebase init

* Select **Hosting**
* use existing directory
* public directory? **build**
* single page app? **y**

npm run build

(if any changes are made afterwards, then **npm run build** should run again)

firebase deploy

**PART-3**

Stripe functionality

Build Checkout/Payment Page with stripe frontend

Deploy Cloud functions to process stripe payment (serverless architecture)

Build “My Orders” Page

Store orders for logged in users

Firebase for real-time database (Firestore)