

## React4 Guided Project

### Introduction

This week we will be building a product detail page by fetching product details from an external API. We will be using Redux and Axios to store and manage the state of products and users related information.

The Base URL we will be using for fetching products and users data is -

<https://store-api-o6m1.onrender.com/v1/>

You can refer to the **Store API Guideline.pdf** file provided to you to get the detailed information about the APIs for this Base URL.

To run the React App in the localhost kindly follow the below mentioned steps -

1. Download and extract the **02\_React4 Guided Project\_Source Code** folder from the **02\_React4 Guided Project\_Source Code.zip** folder
2. The 02\_React4 Guided Project\_Source Code folder contains a sub folder named **ecommerce\_app\_client** folder.
3. Open a terminal and navigate to the **ecommerce\_app\_client** folder in it.
4. Run the *"npm install"* command to install the required packages. This will create a new folder named **node\_modules** in the **ecommerce\_app\_client** folder.
5. Run the *"npm start"* command to run the application with the localhost.
6. To login to the application, login using the username as "john35@d" and password as "johnd" or any other credentials provided in the **Store API**. (Refer <https://store-api-o6m1.onrender.com/v1/users> for more details.)

**Login functionality is already implemented using Redux and Axios in the application.**

### Housekeeping points

- This is a minimal example and may not follow some standard practices.
- We focus on the main flow, and not much error handling.

### Problem Statement

#### 1. Display Products

- Make use of **Fetch Products API** to display name and image of all products available on the Home Page using the following steps-
  - Define constants in **productActionConstants.js** file
  - Complete the code of **listProducts()** and **fetchProductDetails()** functions in **productActions.js** file to define actions to display the list of products and product details on the user side of the application. Make use of the **GET\_ALL\_PRODUCTS\_API** variable from **backend.js** file for the backend URLs.

- Complete the code of `listProductsReducer()` and `fetchProductDetailsReducer()` functions in `productReducers.js` file to define the reducers.
- Complete the code of `storage.js` file to define the initial values of the global state values.
- Complete the code in `productsListPage.js` and `productPage.js` files to dispatch actions to display products list and product details.