

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Contents

1. Contents

Setting up REST API in your computer	2
BEST PRACTICES TO BE FOLLOWED:	2
REST API URLs	3
Create React Project	4
USER STORIES:.....	4
HOME PAGE IMPLEMENTATIONS	4
PRODUCT PAGE IMPLEMENTATIONS.....	5
PRODUCT DETAILS PAGE IMPLEMENTATIONS.....	7
ABOUT-US PAGE IMPLEMENTATIONS	9
CONTACT-US PAGE IMPLEMENTATIONS	11
Unit Test:.....	12
Expected output.....	13

Setting up REST API in your computer

Install JSON server in your computer

```
npm install -g json-server
```

Create a folder named **my-shop-rest-api** in any C:/

Create db.json file inside the above folder with the json data from this Github repository.

<https://github.com/arunprabu/genesys-react-course-2024/blob/master/db.json>

Then, from command prompt try the following command

```
json-server --watch db.json
```

(You could see similar messages like below)

Loading db.json

Done

Resources

http://localhost:3000/products

http://localhost:3000/categories

http://localhost:3000/contact

Home

http://localhost:3000

Do not disturb the above Terminal/command prompt window. Your REST API server is running now.

BEST PRACTICES TO BE FOLLOWED:

- Keep component creation as minimum. Reuse components wherever necessary
- Code should be testable. Hence always write clean code
- Define Proper file structure
- Prefer pure functions and immutable data wherever necessary
- Keep modules independent wherever applicable
- Follow appropriate naming of Props

REST API URLs

The following REST API urls will work on GET, POST, PUT, PATCH, DELETE

To list products

`http://localhost:3000/products/`

To limit only 3 products for Homepage

`http://localhost:3000/products?_limit=3`

To sort products based on maxRetailPrice in ascending order

`http://localhost:3000/products?_sort=maxRetailPrice&_order=asc`

To sort products based on maxRetailPrice in descending order

`http://localhost:3000/products?_sort=maxRetailPrice&_order=desc`

When sorting by price (desc) is performed in Products of specific category

`http://localhost:3000/products?category=Men&_sort=maxRetailPrice&_order=desc`

When sorting by price (asc) is performed in Products of specific category

`http://localhost:3000/products?category=Women&_sort=maxRetailPrice&_order=asc`

To display categories in sidebar

`http://localhost:3000/categories`

To get product details

`http://localhost:3000/products/1`

To submit contact form data

`http://localhost:3000/contact`

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Create React Project

Create React Project using vite in **C:/** and name it as **my-shop-react-app** and start running the server.

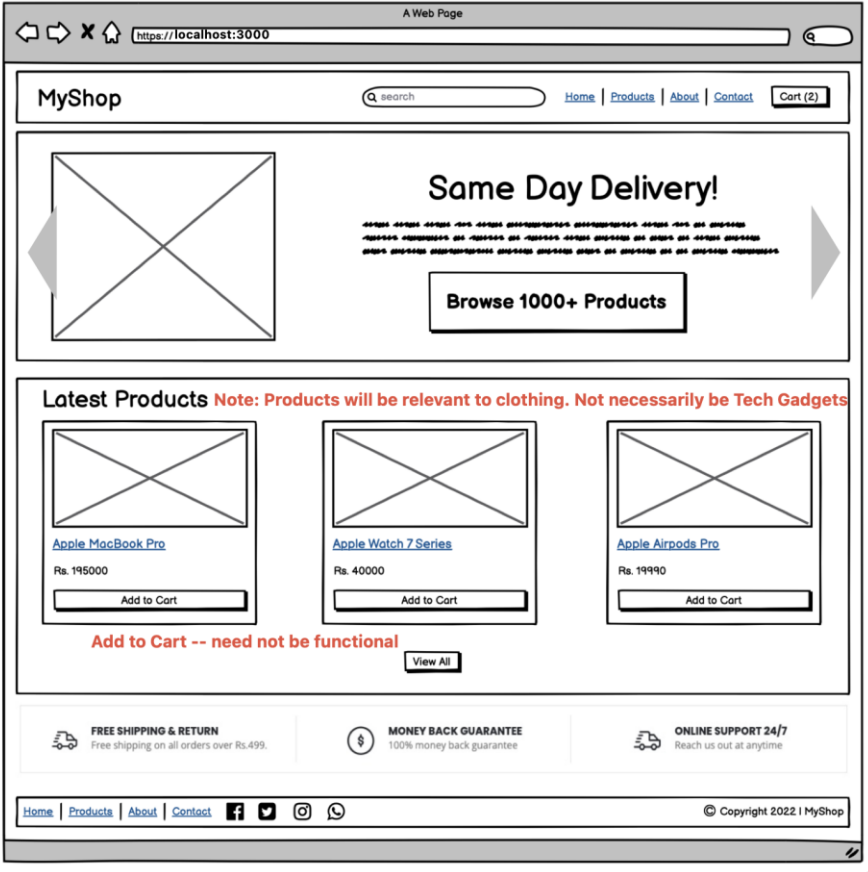
Server will be up on **http://localhost:5173**

USER STORIES:

HOME PAGE IMPLEMENTATIONS

Story no	U01
Title	Home Page Implementation
Description	Design a page as given in the expected output. Logo and designs are customisable
Requirement	<p>1. Split the Home page into following sections</p> <ul style="list-style-type: none">• Navbar• Menu• Image and Contents for Carousel <p>Implement Navbar & Carousel Functionality</p> <p>Menu with Icons should be available for below listed options</p> <ul style="list-style-type: none">• Logo with Name• HOME• PRODUCTS• ABOUT US• CONTACT <p>Routing URL for this page should be designed as below :</p> <p>(URL: localhost:5173/)</p>

CASE STUDY: MyShop CLOTHING SHOP- REACT18

<p>Acceptance Criteria</p>	<p>When we click Logo or Home, it should show Home Page. When we click ABOUT US/CONTACT US and PRODUCTS, it should show ABOUT US, PRODUCTS and CONTACT US respectively with its contents.</p> <p>Mandatory Criteria,</p> <ul style="list-style-type: none"> Write Unit test for all the code
<p>Assets Information</p> <ul style="list-style-type: none"> Screenshots for Expected Output Images Icons Stylesheet (You are supposed to add the text as per each user story. Also, you are free to add and modify the styles as per your customisation. This is just sample recommendation) Fonts 	<p>FOLLOW BELOW FOLDER STRUCTURE TO MODIFY/ADD NAVBAR Title And Image(logo):</p> <ul style="list-style-type: none"> Load footer social media icons from font-awesome. <p>EXPECTED OUTPUT :</p>  <p>Note: Ignore the Cart button</p>

PRODUCT PAGE IMPLEMENTATIONS

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Story no	U02
Title	Product Page Implementation
Description	Design a page by listing Products available under clothing. Expected output is given as sample
Requirement	<ol style="list-style-type: none"> 1. When navigating to the products page, products with clothing category should be listed along with images for the product and price details 2. Write grouping logic based on the categories and display under each category. 3. When clicking 'All' from 'category sidebar' the URL should be localhost:5173/products/ 4. When clicking 'Men' from 'category sidebar' the URL should be localhost:5173/products?category=men 5. When clicking 'Women' from 'category sidebar' the URL should be localhost:5173/products?category=women 6. When clicking 'Kids' from 'category sidebar' the URL should be localhost:5173/products?category=kids 7. Upon category clicked load only products that belong to the clicked category from the backend. Refer REST API urls in Page No 3. 8. Sort by Option should show only Price – Low to High and Price - High to Low. 9. When sorting is done in Women category, the React App's url should become like the following. <ul style="list-style-type: none"> • localhost:5173/products?category=women&_sort=maxRetailPrice&_order=asc • localhost:5173/products?category=women&_sort=maxRetailPrice&_order=desc <p>Routing URL for this page should be designed as below : (URL: localhost:5173/products/)</p>
Acceptance Criteria	<p>Once user click on the Products Title they will be redirected to Products details page.</p> <p>Mandatory Criteria,</p> <ul style="list-style-type: none"> • Write Unit test for all the code

CASE STUDY: MyShop CLOTHING SHOP- REACT18

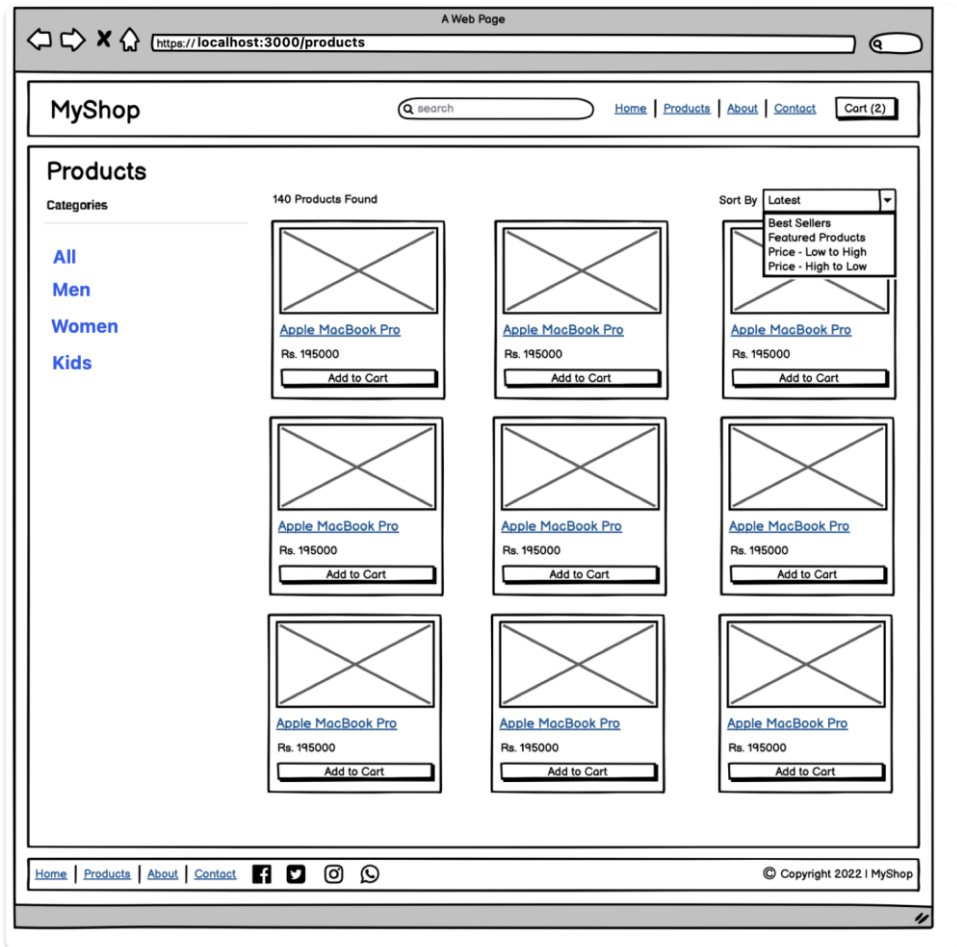
Assets Information

- Screenshots for Expected Output
- Images
- Icons
- Stylesheet (You are supposed to add the text as per each user story. Also, you are free to add and modify the styles as per your customisation. This is just sample recommendation)
- Fonts

FOLLOW BELOW FOLDER STRUCTURE TO ADD IMAGE AND TITLE:

- Also Create a component – to display categories such as All, Men, Women, Kids in sidebar in the appropriate project directory.
- Load products and categories also from REST API Backend.

EXPECTED OUTPUT :



PRODUCT DETAILS PAGE IMPLEMENTATIONS

Story no	U03
Title	Product Details Page Implementation

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Description	Design a page to display product details. Expected output is given as sample
Requirement	<ol style="list-style-type: none">1. When navigating to the product details page, the UI should show necessary product specific information2. Add the product title and description3. Work with id as URL parameter to make the URL dynamic for every product detail page. <p>Routing URL for this page should be designed as below:</p> <p>(URL: localhost:5173/products/1/</p> <p>localhost:5173/products/2/</p> <p>localhost:5173/products/3/ and so on)</p>
Acceptance Criteria	<p>Once user click on the Write a Review, bootstrap modal should appear with form to enter name, email, phone, review comments and rating.</p> <p>Mandatory Criteria,</p> <ul style="list-style-type: none">• Write Unit test for all the code

CASE STUDY: MyShop CLOTHING SHOP- REACT18

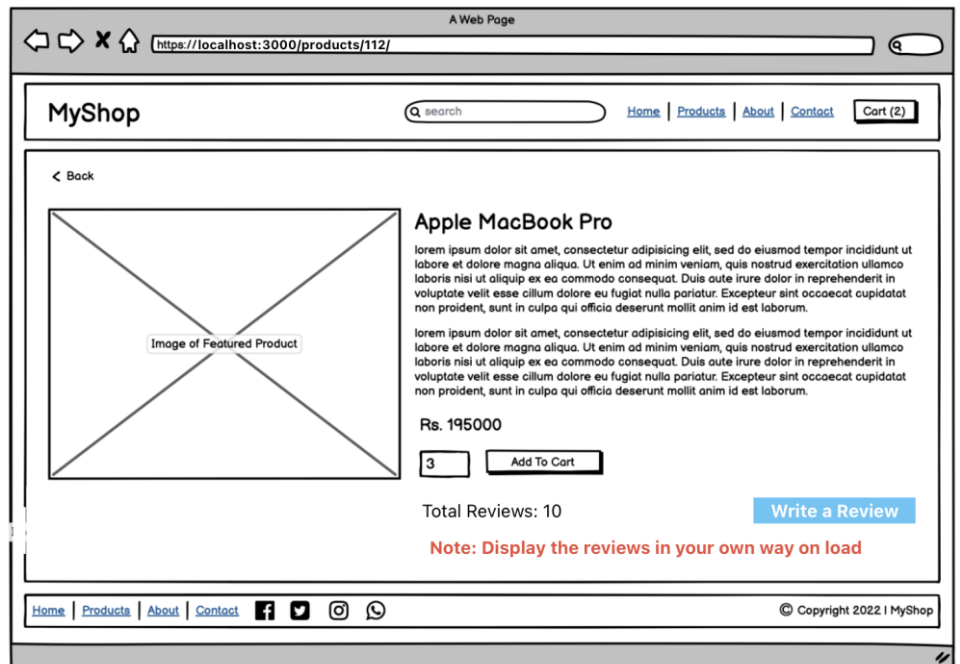
Assets Information

- Screenshots for Expected Output
- Images
- Icons
- Stylesheet (You are supposed to add the text as per each user story. Also, you are free to add and modify the styles as per your customisation. This is just sample recommendation)
- Fonts

FOLLOW BELOW FOLDER STRUCTURE TO ADD IMAGE AND TITLE:

- Load product details from REST API Backend.
- **IMPORTANT: Just display the reviews from the REST API data. 'Write a Review' logic need not be implemented**

EXPECTED OUTPUT:



ABOUT-US PAGE IMPLEMENTATIONS

Story no	U04
Title	About US Page Implementation
Description	Design a page to provide information about the Clothing Shop

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Requirement	<ol style="list-style-type: none"> 1. While clicking on About us page, the expected output provided should be appear. You can create your own them and icons. 2. About US – Some tagline with title and detailed description 3. Implement Nested Routing here. <p>Routing URL for this page should be designed as below:</p> <p>(URL: localhost:5173/about-us/ and localhost:5173/about-us/history)</p> <p>TEXT Contents for pages:</p> <p>==>About Us Tagline</p> <p>“We have the capabilities and experience to deliver the products you need to move forward.”</p> <p>==>The History</p> <p>The story of MyShop Clothing started in Bangalore in the year 2000 with wholesale of hosiery and woven garments, catering to the needs of various wholesalers and retailers across India. Bangalore textiles industry stands at USD 4.2 billion in 2012 and is globally famous for its hosiery products. The City has more than 5,000 garment manufacturing and job work units, and is one of the most organised processing and finishing garment clusters in India. The firm came into existence in the year 2005 as a Sole Proprietorship concern in Bangalore, Tamil Nadu. In the year 2007 it started as family enterprise with a shop selling Men’s apparel at much lesser price than market and expanded to full men’s wear during next six months. The journey continued and in 2011 launched its first exclusively brand store in Bangalore with bigger retailing space for Men’s wear, Women’s wear and Kids clothing.</p> <p>MyShop Clothings is engaged into multiple business activities which include manufacturing, supplying and trading in a wide array of clothing products which include Women’s Wear, Polo T-Shirt, Girls T-Shirt and many more. The range offered is well acknowledged for its availability in stylish pattern, perfect finish and comfort in wearing. Products are developed after careful attention and hard work of MyShop team. It aspires to render quality striped products which guarantee cheerfulness of end customer.</p>
Acceptance Criteria	<p>When clicking on About Us page, the expected output should appear.</p> <p>Mandatory Criteria,</p> <ul style="list-style-type: none"> • Write Unit test for all the code

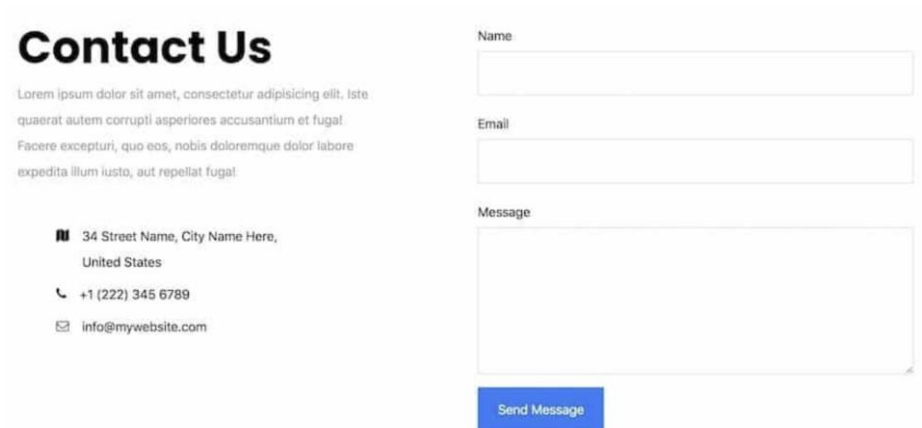
CASE STUDY: MyShop CLOTHING SHOP- REACT18

Assets Information <ul style="list-style-type: none">• Screenshots for Expected Output• Images• Icons• Stylesheet (You are supposed to add the text as per each user story. Also, you are free to add and modify the styles as per your customisation. This is just sample recommendation)• Fonts	EXPECTED OUTPUT : UI can be as per your wish. Have two URL's. (/about-us/ and /about-us/history) Implement Nested Routing for this feature.
--	---

CONTACT-US PAGE IMPLEMENTATIONS

Story no	U05
Title	Contact US Page Implementation
Description	Design a page to provide information about the Contact Details
Requirement	<p>While clicking on Contact us page, the below things should be appear. You can create your own them and icons.</p> <ul style="list-style-type: none">• Address• Phone• Email• Logo• Navigation to Home, Products, About Us and Contact US <p>Routing URL for this page should be designed as below : (URL: localhost:5173/contact-us/)</p>

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Acceptance Criteria	<p>When clicking on Contact us page, the expected output should appear.</p> <p>Mandatory Criteria,</p> <ul style="list-style-type: none"> Write Unit test for all the code
<p>Assets Information</p> <ul style="list-style-type: none"> Screenshots for Expected Output Images Icons Stylesheet (You are supposed to add the text as per each user story. Also, you are free to add and modify the styles as per your customisation. This is just sample recommendation) Fonts 	<p>Follow the below folder structure to add header in Contact Page:</p> <ul style="list-style-type: none"> Save the form data on the back-end side. Fetch the above address phone and email by sending REST API call over GET method. Submit the Form data to REST API over POST Method. <div>  </div> <p>Fetch the above address phone and email by sending REST API call over GET method. Submit the Form data to REST API over POST Method.</p>

Unit Test:

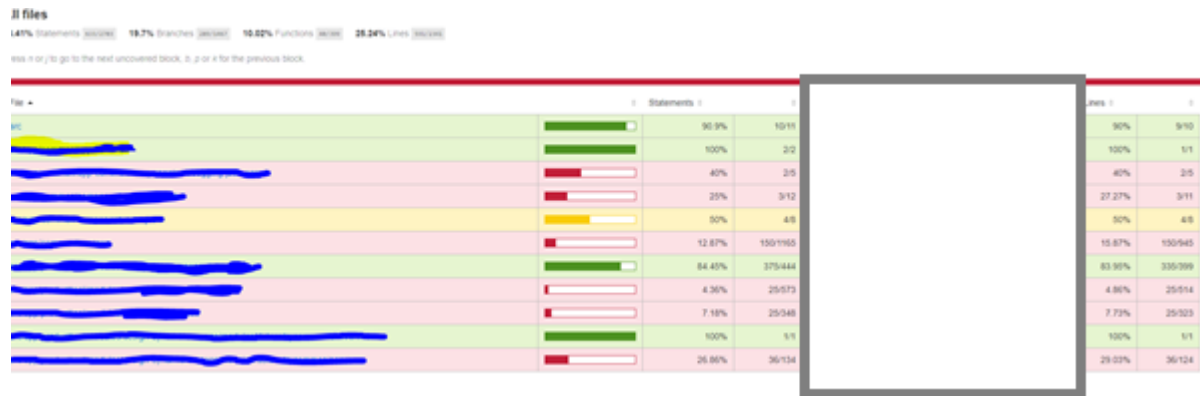
- All React components should have minimum **80%** code. Ensure that **Lines and Statements** code coverage individual should not be less than **80%**.
- All components must have individual spec file associated with component file.
- Run below command to generate coverage report:

ng test --code-coverage

CASE STUDY: MyShop CLOTHING SHOP- REACT18

Expected output

on application root folder inside **coverage** folder index.html as coverage report should generate which reflects required % as mentioned above as sample



Responsive Requirements:

- Application should be responsive which means the screen layout will adjust on mobile and Tab screen without element overlap
- Application should run on mobile browsers (ios and Android) (Chrome at least)
- On running on mobile browser or tab screen should not have horizontal scrollbar
- Application layout should follow **Mobile first approach**
- Application layout should adjust as free flow (Left to Right) additional element wrap downside.
- In mobile screen single column layout
- More than 600 2 columns and after 700px as desktop layout.