**Trending Electric Products Application**

Document Properties

|  |  |
| --- | --- |
| **Item** | **Details** |
| Document | Architecture Design Document |
| Application | Trending Electric Products Application |
| Author | Pooja Paigude |
| Version | V1.0 |
| Creation Date | 16/09/2020 |
| Last Updated |  |
| Status | Completed |

Contents

[Document Properties 1](#_Toc50408770)

[**1.** **Introduction** 3](#_Toc50408771)

[**2.** **Use-Case View** 3](#_Toc50408772)

[**3.** **Flow Chart** 4](#_Toc50408773)

[**4.** **Components** 5](#_Toc50408774)

[**5.** **Services** 6](#_Toc50408775)

[**6.** **Store** 6](#_Toc50408776)

[**7.** **Validations** 6](#_Toc50408777)

[**8.** **Tools and Libraries** 6](#_Toc50408778)

[**9.** **Unit Testing** 6](#_Toc50408779)

[**10.** **Responsive** 7](#_Toc50408780)

[**11.** **Accessibility** 7](#_Toc50408781)

[**12.** **Open Items** 7](#_Toc50408782)

[**13.** **Setup** 7](#_Toc50408783)

# **Introduction**

In this document, we can see the overview for the application which covers its basic flow and scenarios of the Trending Electric Products in market.

It shows trending products with category of that product and login user will be able to perform CRUD operations on that table of product.

Also, there is a login and sign up functionality provided for the user.

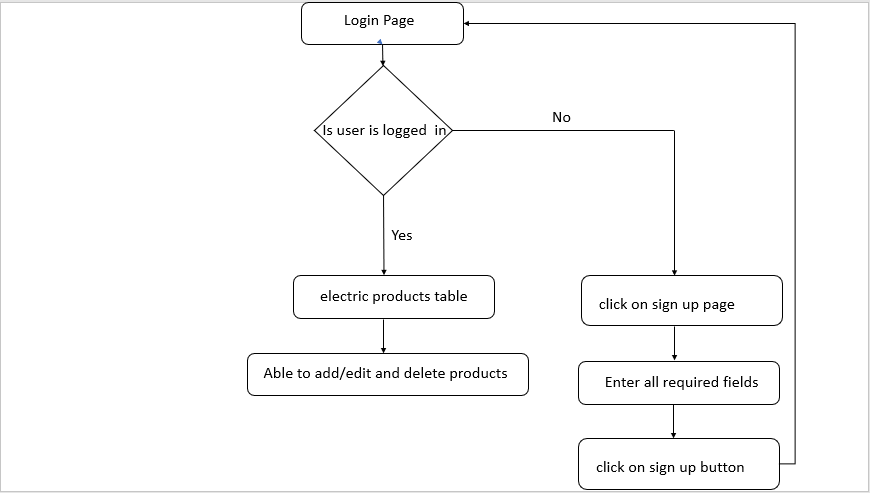
# **Use-Case View**

The application use cases for which user authentication is mandatory are -

* **Add a product** – The application provides the functionality for adding a new product after user login.
* **Update a product** – This functionality provides the facility to update the existing product for the user.
* **Delete a product –** The user has an option to delete the product.

Use cases for which the user authentication is not mandatory.

* **List all product** -The user has the option to view all the products and their description.
* **Login**– The user can login into the application with his/her credentials.
* **Sign up** – The user can sign up to login the application.

1. **FlowChart**

# **Components**

* AppComponent – This component is used for Application Bootstrapping
* ElectronicsListingComponent – This is the initial page of the application where user will be able to see table of trending electronic products.
* LoginComponent – User will be able to login through this component.
* SignUpComponent – This component manages the signing up and login flow of the user
* AddProductComponent – This component is used to manage the edit/create flow of the product data.

# **Services**

1. UserAccessService – Handles all API calls for login/register operations.
2. ProductsService – This service takes care of the API requests for CRUD operations of the products.

# **Store**

The application uses ngRx for state management for user and products data.

# **Validations**

1)When an unregistered user tries to login, the application does not allow it to do so, first user needs to register/sign up then he can login.

2) Before deleting any product the application asks for user confirmation.

# **Tools and Libraries**

- Angular version: 10.0

- TypesScript: 3.9.5

- Rxjs: 6.5.5

- Jest: 26.4.2 (This is a unit testing framework )

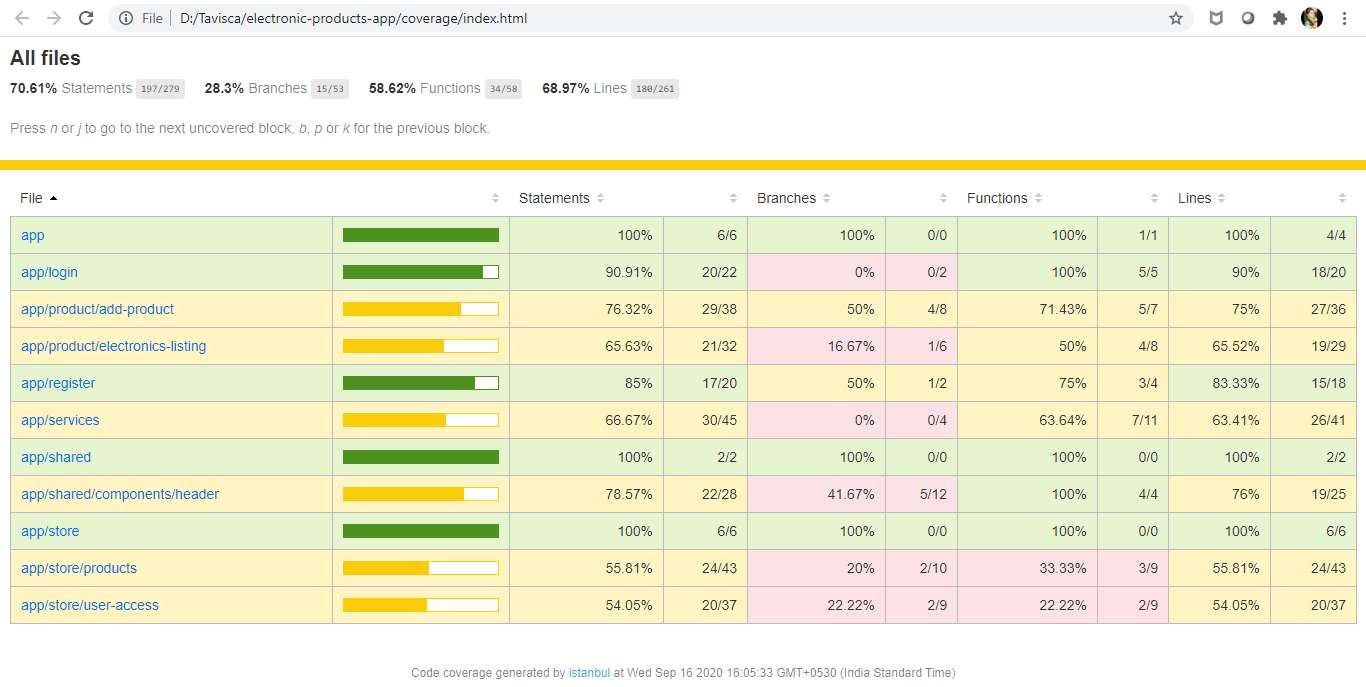
# **Unit Testing**

Added Unit Test cases for services.

It consists of following services -

* + product.service.spec.ts – It contain test cases for the all the API calls for CRUD operations for products.
  + User-access.service.spec.ts - It contain test cases for API calls for the authentication.

**Screenshot for unit test cases -**



# **Responsive**

* Created responsive table which grows/shrinks in width according to the device.
* No Horizontal scrollbar in application.

# **Accessibility**

For Accessibility, the HTML markup contains aria-labels for buttons and tab indexes for input fields and table data.

# **Open Items**

1. Storybook

# **Setup**

1. Download and install node.js (version: 12.16)
2. Install npm and install @angular (version: 10.1.0)
3. Git clone <https://github.com/poojapaigude/electronic-products-app> origin master
4. Navigate to cloned folder & run npm install
5. Run command npm run json:server
6. For testing use: npm run test
7. Run ng serve