CAPSTONE PROJECT

E-COMMERCE MOBILE APP WITH INVENTORY MANAGEMENT AND ORDER TRACKING

WIPRO IOS BATCH

Name: GOPICHAND AMPABATTUNI

E-mail: agopichand2001@gmail.com

Date: October 28, 2024

Problem Statement:

You are tasked with building a comprehensive ecommerce mobile app for a fictional retail company called "WiproTechWear". The app should allow users to browse and purchase tech-related products, track their orders, and manage their account information.

Requirements:

Frontend:

• **Web App:** Build using React, ensuring a user-friendly experience for browsing products, adding to the cart, and checking out.

• **Mobile App:** Develop for iOS using Swift and Swift UI, providing a native experience for iOS users.

• **User Interface Design:** Implement design principles for an intuitive user interface and clear navigation flow, similar to Figma. Ensure that each screen follows a consistent user journey.

Admin should be able to :

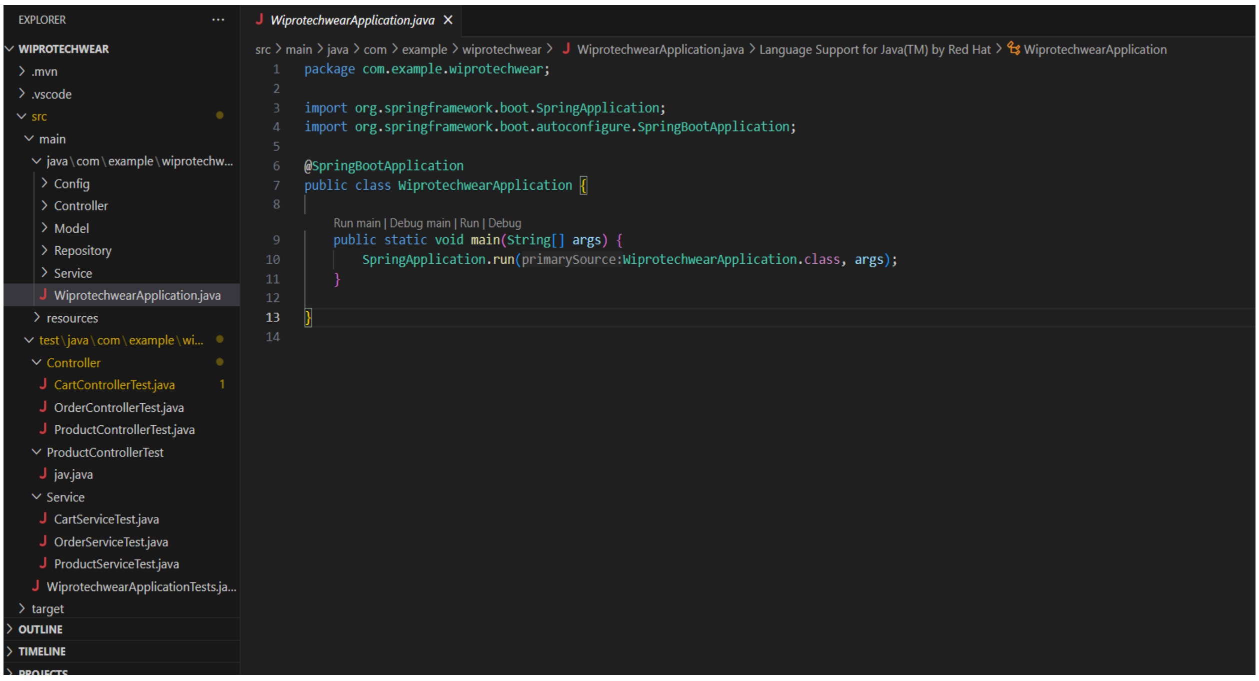
- Login to the web application with the admin credentials.
- Manage the products stock
- Manage the orders placed by the users

User should be able to :

- If the user is new, should be able to register.
- If already exists, he should be able to login
- Select the products and add them to cart
- Place order with the products added to the cart
- Track the order status

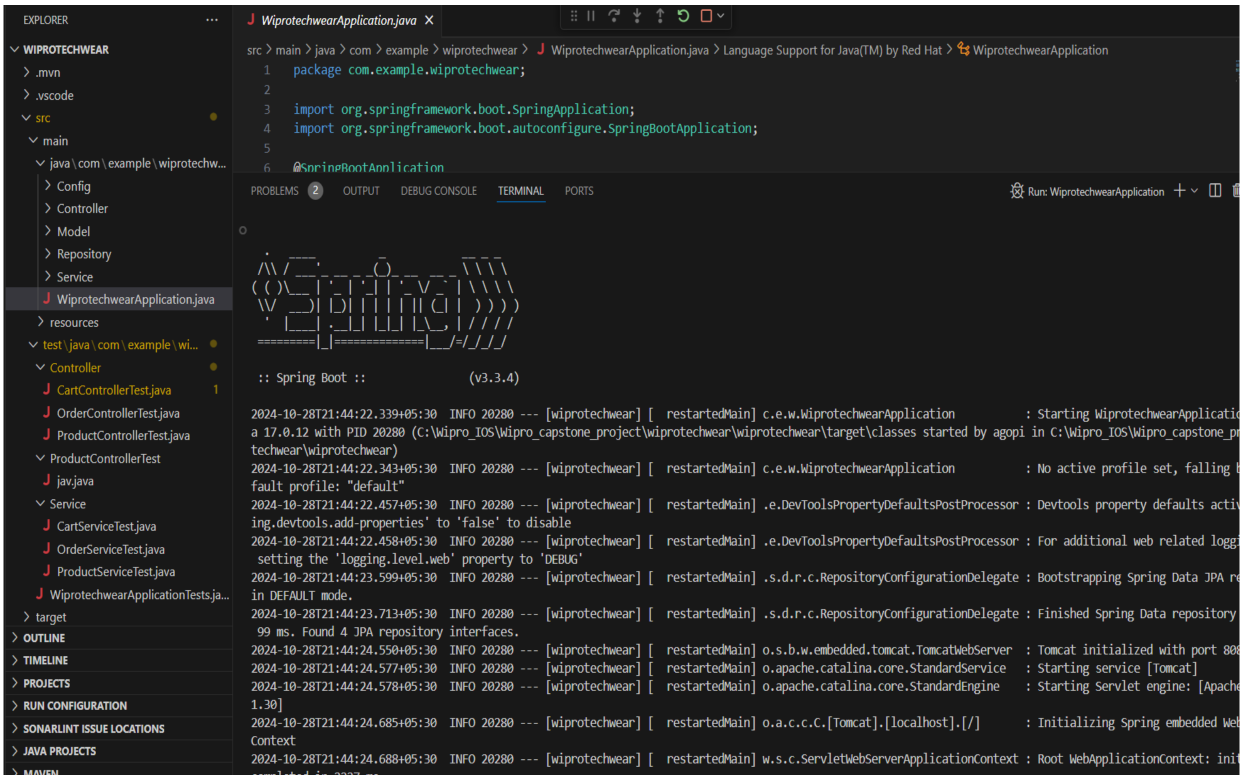
Web Application:

SpringbootApp - Backend:


EXPLORER
WIPROTECHWEAR
.mvn
.vscode
src
main
java \ com \ example \ wiprotechw...
Config
Controller
Model
Repository
Service
WiprotechwearApplication.java
resources
test \ java \ com \ example \ wi...
Controller
CartControllerTest.java
OrderControllerTest.java
ProductControllerTest.java
ProductControllerTest
jav.java
Service
CartServiceTest.java
OrderServiceTest.java
ProductServiceTest.java
WiprotechwearApplicationTests.ja...
target
OUTLINE
TIMELINE
WiprotechwearApplication.java
src > main > java > com > example > wiprotechwear > WiprotechwearApplication.java > Language Support for Java(TM) by Red Hat > WiprotechwearApplication
Run main | Debug main | Run | Debug

```java
package com.example.wiprotechwear;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class WiprotechwearApplication {

    public static void main(String[] args) {
        SpringApplication.run(primarySource:WiprotechwearApplication.class, args);
    }

}
```

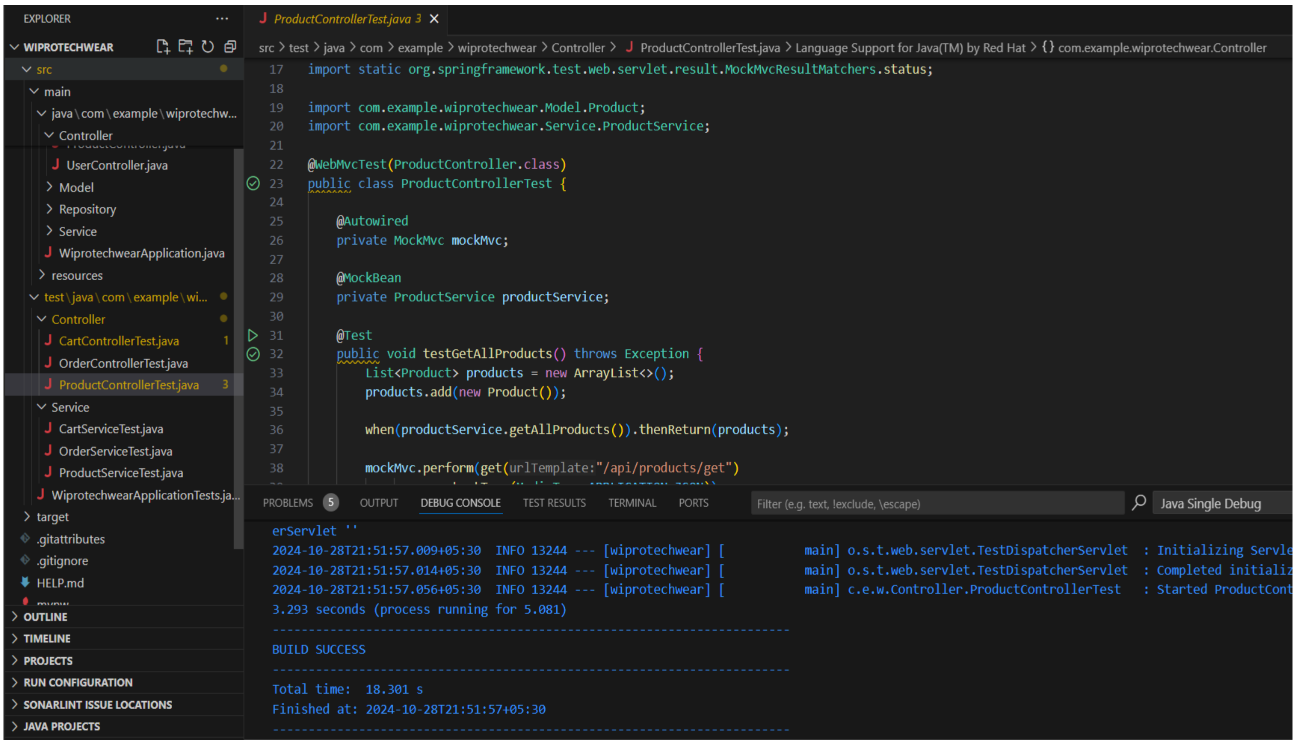
Running the Springboot application :


EXPLORER
WIPROTECHWEAR
.mvn
.vscode
src
main
java \ com \ example \ wiprotechw...
Config
Controller
Model
Repository
Service
WiprotechwearApplication.java
resources
test \ java \ com \ example \ wi...
Controller
CartControllerTest.java
OrderControllerTest.java
ProductControllerTest.java
ProductControllerTest
jav.java
Service
CartServiceTest.java
OrderServiceTest.java
ProductServiceTest.java
WiprotechwearApplicationTests.ja...
target
OUTLINE
TIMELINE
PROJECTS
RUN CONFIGURATION
SONARLINT ISSUE LOCATIONS
JAVA PROJECTS
WiprotechwearApplication.java
src > main > java > com > example > wiprotechwear > WiprotechwearApplication.java > Language Support for Java(TM) by Red Hat > WiprotechwearApplication
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Run: WiprotechwearApplication

```java
package com.example.wiprotechwear;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
```

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.3.4)

2024-10-28T21:44:22.339+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] c.e.w.WiprotechwearApplication           : Starting WiprotechwearApplicati
a 17.0.12 with PID 20280 (C:\Wipro_IOS\Wipro_capstone_project\wiprotechwear\wiprotechwear\target\classes started by agopi in C:\Wipro_IOS\Wipro_capstone_pr
techwear\wiprotechwear)
2024-10-28T21:44:22.343+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] c.e.w.WiprotechwearApplication           : No active profile set, falling b
fault profile: "default"
2024-10-28T21:44:22.457+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults activ
ing.devtools.add-properties' to 'false' to disable
2024-10-28T21:44:22.458+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related loggi
 setting the 'logging.level.web' property to 'DEBUG'
2024-10-28T21:44:23.599+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA re
in DEFAULT mode.
2024-10-28T21:44:23.713+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository
 99 ms. Found 4 JPA repository interfaces.
2024-10-28T21:44:24.550+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 808
2024-10-28T21:44:24.577+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2024-10-28T21:44:24.578+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache
1.30]
2024-10-28T21:44:24.685+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded Web
Context
2024-10-28T21:44:24.688+05:30  INFO 20280 --- [wiprotechwear] [  restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: init
```

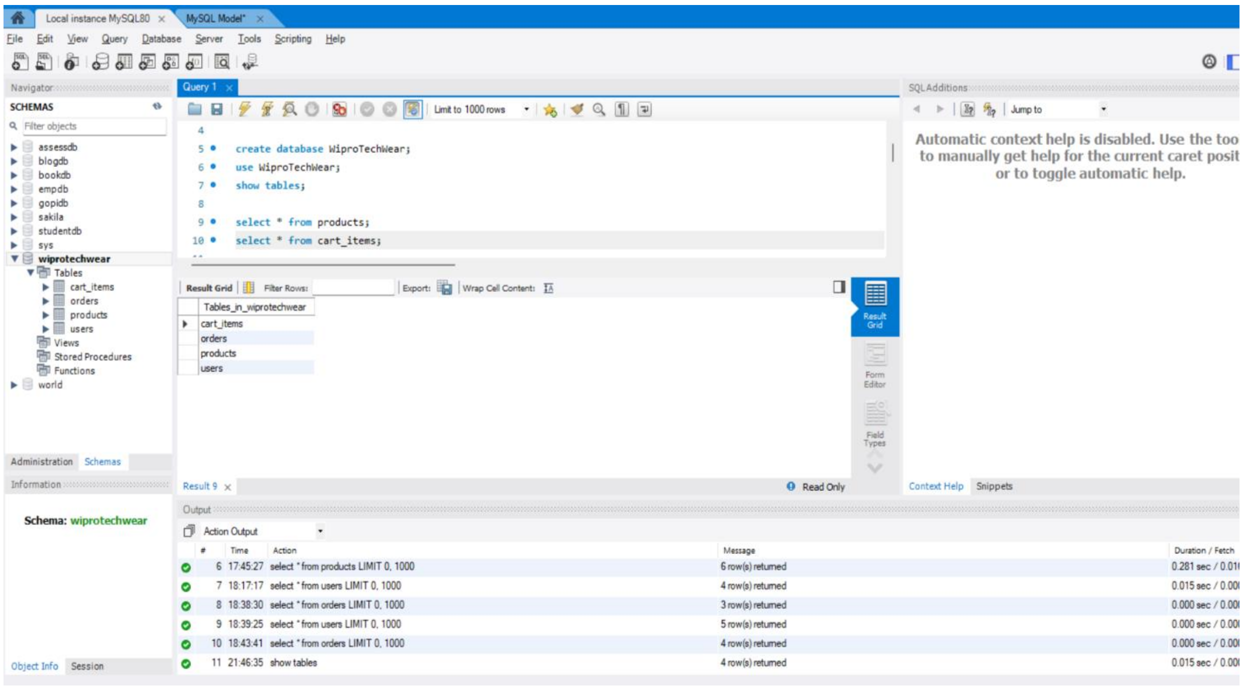
Test case for Springboot Application:


EXPLORER
WIPROTECHWEAR
src
main
java \ com \ example \ wiprotechw...
Controller
UserController.java
Model
Repository
Service
WiprotechwearApplication.java
resources
test \ java \ com \ example \ wi...
Controller
CartControllerTest.java
OrderControllerTest.java
ProductControllerTest.java
Service
CartServiceTest.java
OrderServiceTest.java
ProductServiceTest.java
WiprotechwearApplicationTests.ja...
target
.gitattributes
.gitignore
HELP.md
OUTLINE
TIMELINE
PROJECTS
RUN CONFIGURATION
SONARLINT ISSUE LOCATIONS
JAVA PROJECTS
ProductControllerTest.java 3
src > test > java > com > example > wiprotechwear > Controller > ProductControllerTest.java > Language Support for Java(TM) by Red Hat > {} com.example.wiprotechwear.Controller
PROBLEMS 5
OUTPUT
DEBUG CONSOLE
TEST RESULTS
TERMINAL
PORTS
Filter (e.g. text, !exclude, \escape)
Java Single Debug

```java
import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.status;

import com.example.wiprotechwear.Model.Product;
import com.example.wiprotechwear.Service.ProductService;

@WebMvcTest(ProductController.class)
public class ProductControllerTest {

    @Autowired
    private MockMvc mockMvc;

    @MockBean
    private ProductService productService;

    @Test
    public void testGetAllProducts() throws Exception {
        List<Product> products = new ArrayList<>();
        products.add(new Product());

        when(productService.getAllProducts()).thenReturn(products);

        mockMvc.perform(get(urlTemplate:"/api/products/get")
```

```
erServlet ''
2024-10-28T21:51:57.009+05:30  INFO 13244 --- [wiprotechwear] [           main] o.s.t.web.servlet.TestDispatcherServlet  : Initializing Servle
2024-10-28T21:51:57.014+05:30  INFO 13244 --- [wiprotechwear] [           main] o.s.t.web.servlet.TestDispatcherServlet  : Completed initializ
2024-10-28T21:51:57.056+05:30  INFO 13244 --- [wiprotechwear] [           main] c.e.w.Controller.ProductControllerTest   : Started ProductCont
3.293 seconds (process running for 5.081)
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  18.301 s
Finished at: 2024-10-28T21:51:57+05:30
------------------------------------------------------------------------
```

Setting up MySQL :'



```
create database WiproTechWear;
use WiproTechWear;
show tables;

select * from products;
select * from cart_items;
```

Schema: wiprotechwear

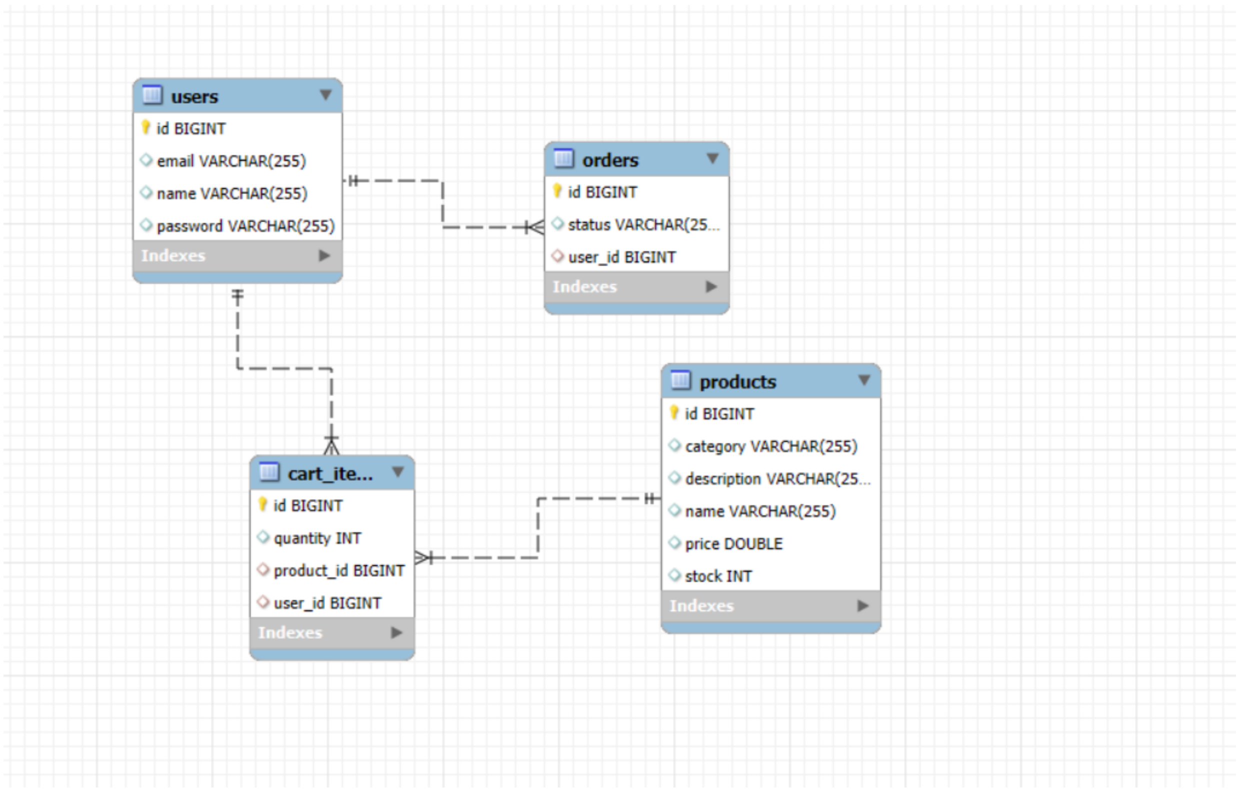
Tables_in_wiprotechwear
cart_items
orders
products
users

Automatic context help is disabled. Use the too to manually get help for the current caret posit or to toggle automatic help.

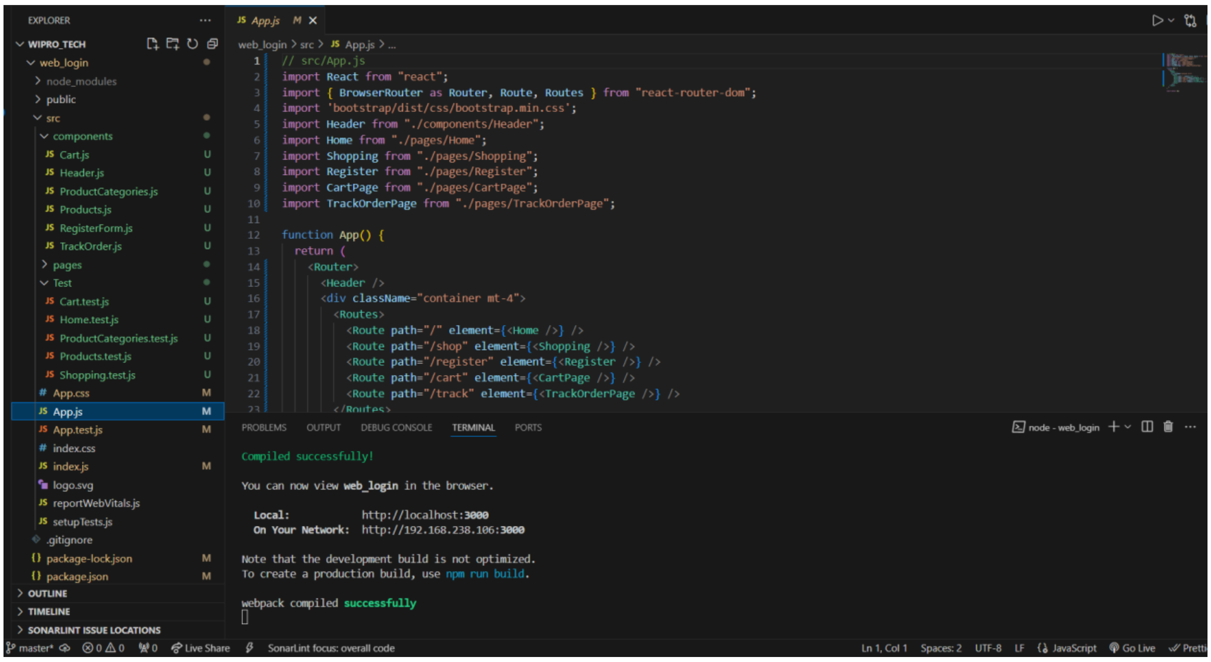
Action Output

```
#   Time      Action                                  Message              Duration / Fetch
6   17:45:27  select * from products LIMIT 0, 1000    6 row(s) returned    0.281 sec / 0.01
7   18:17:17  select * from users LIMIT 0, 1000       4 row(s) returned    0.015 sec / 0.00
8   18:38:30  select * from orders LIMIT 0, 1000      3 row(s) returned    0.000 sec / 0.00
9   18:39:25  select * from users LIMIT 0, 1000       5 row(s) returned    0.000 sec / 0.00
10  18:43:41  select * from orders LIMIT 0, 1000      4 row(s) returned    0.000 sec / 0.00
11  21:46:35  show tables                             4 row(s) returned    0.015 sec / 0.00
```

ER diagram :



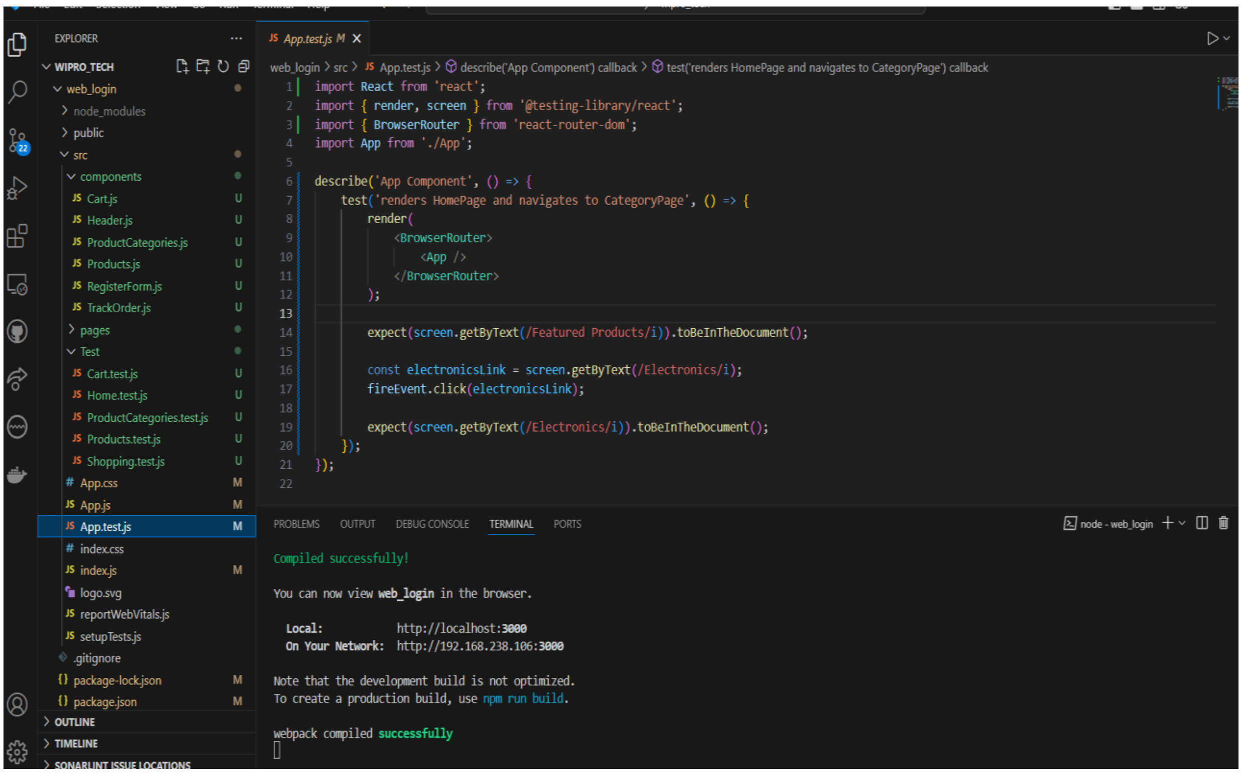
Frontend Application – ReactJS :



```javascript
// src/App.js
import React from "react";
import { BrowserRouter as Router, Route, Routes } from "react-router-dom";
import 'bootstrap/dist/css/bootstrap.min.css';
import Header from "./components/Header";
import Home from "./pages/Home";
import Shopping from "./pages/Shopping";
import Register from "./pages/Register";
import CartPage from "./pages/CartPage";
import TrackOrderPage from "./pages/TrackOrderPage";

function App() {
  return (
    <Router>
      <Header />
      <div className="container mt-4">
        <Routes>
          <Route path="/" element={<Home />} />
          <Route path="/shop" element={<Shopping />} />
          <Route path="/register" element={<Register />} />
          <Route path="/cart" element={<CartPage />} />
          <Route path="/track" element={<TrackOrderPage />} />
        </Routes>
```

```
Compiled successfully!

You can now view web_login in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.238.106:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

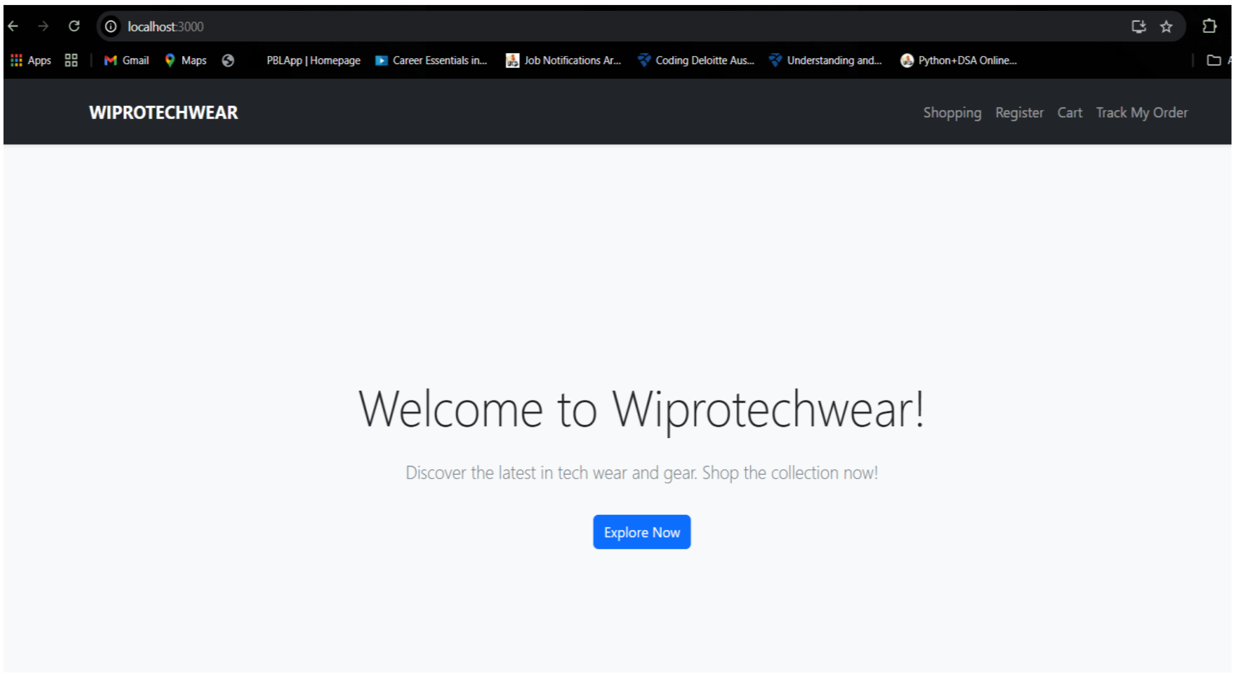
Test cases – ReactJS :



```javascript
import React from 'react';
import { render, screen } from '@testing-library/react';
import { BrowserRouter } from 'react-router-dom';
import App from './App';

describe('App Component', () => {
    test('renders HomePage and navigates to CategoryPage', () => {
        render(
            <BrowserRouter>
                <App />
            </BrowserRouter>
        );

        expect(screen.getByText(/Featured Products/i)).toBeInTheDocument();

        const electronicsLink = screen.getByText(/Electronics/i);
        fireEvent.click(electronicsLink);

        expect(screen.getByText(/Electronics/i)).toBeInTheDocument();
    });
});
```

```
Compiled successfully!

You can now view web_login in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.238.106:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

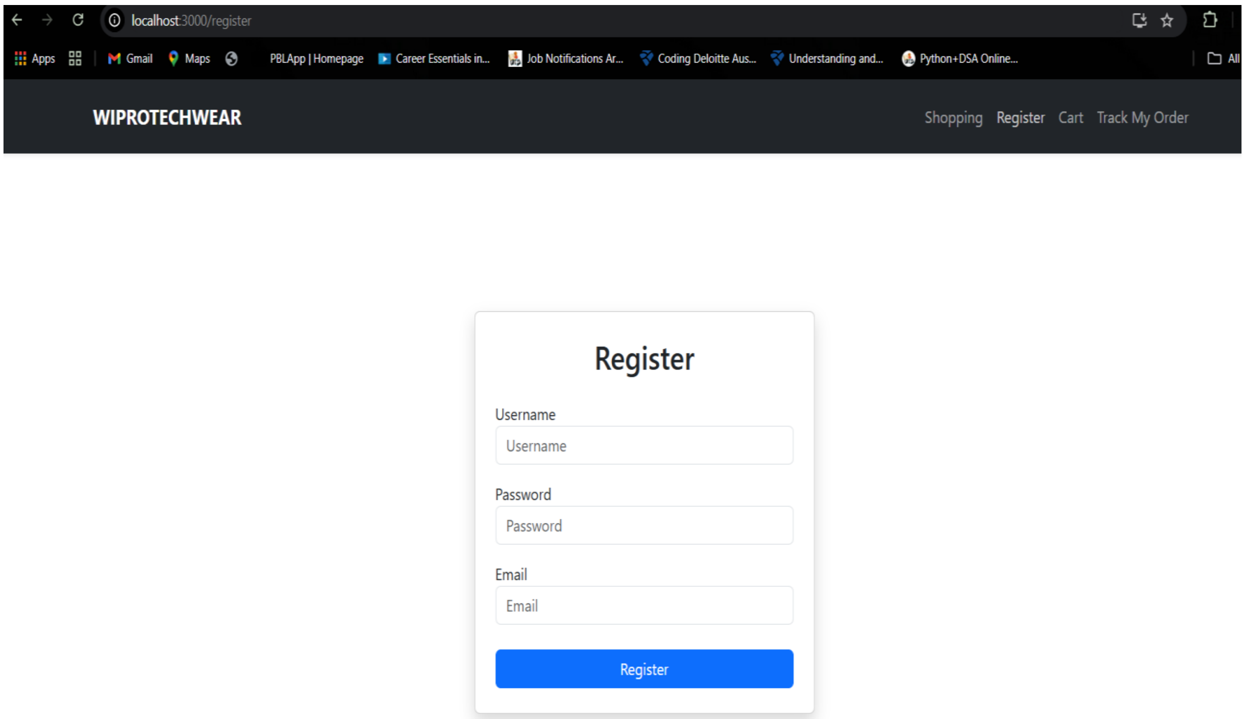
Home page :



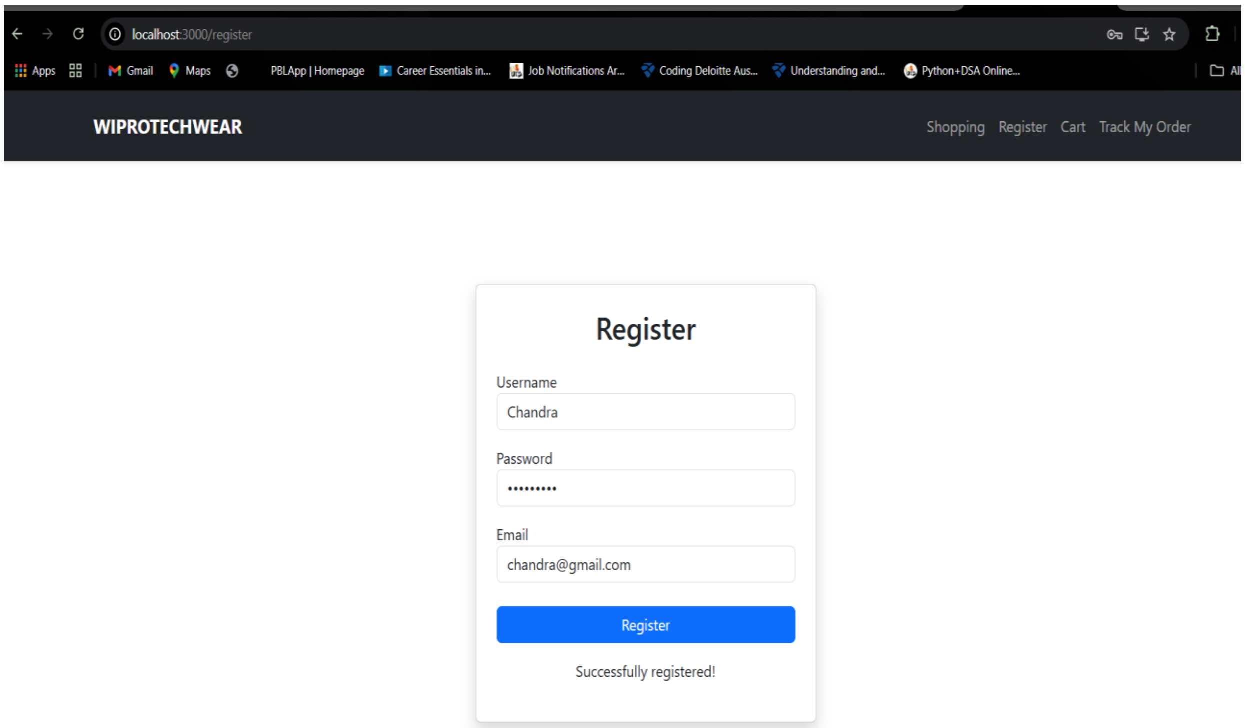
Welcome to Wiprotechwear!

Discover the latest in tech wear and gear. Shop the collection now!

Register page :



Successful registration of user :



localhost:3000/register

Register

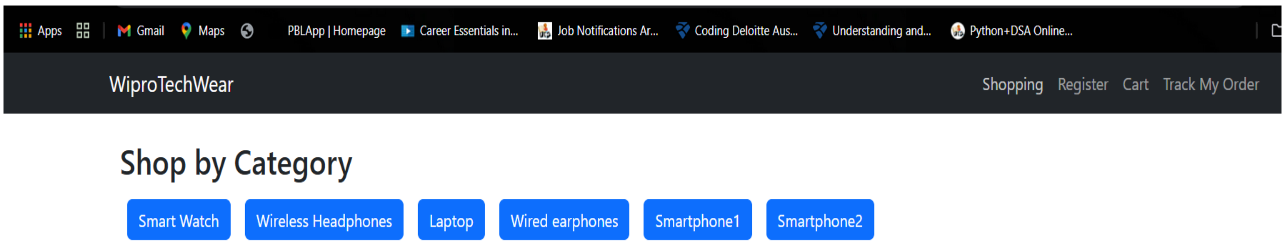
Username: Chandra

Password: •••••••••

Email: chandra@gmail.com

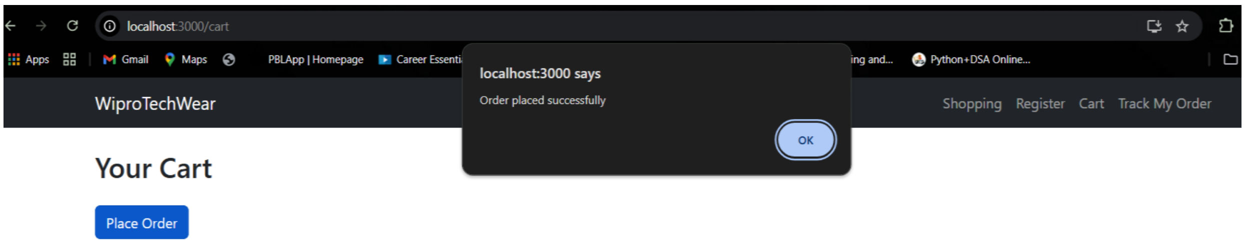
Successfully registered!

Shopping Page :



Shop by Category

Placing order from cart :



localhost:3000/cart

WiproTechWear

Shopping Register Cart Track My Order

localhost:3000 says

Order placed successfully

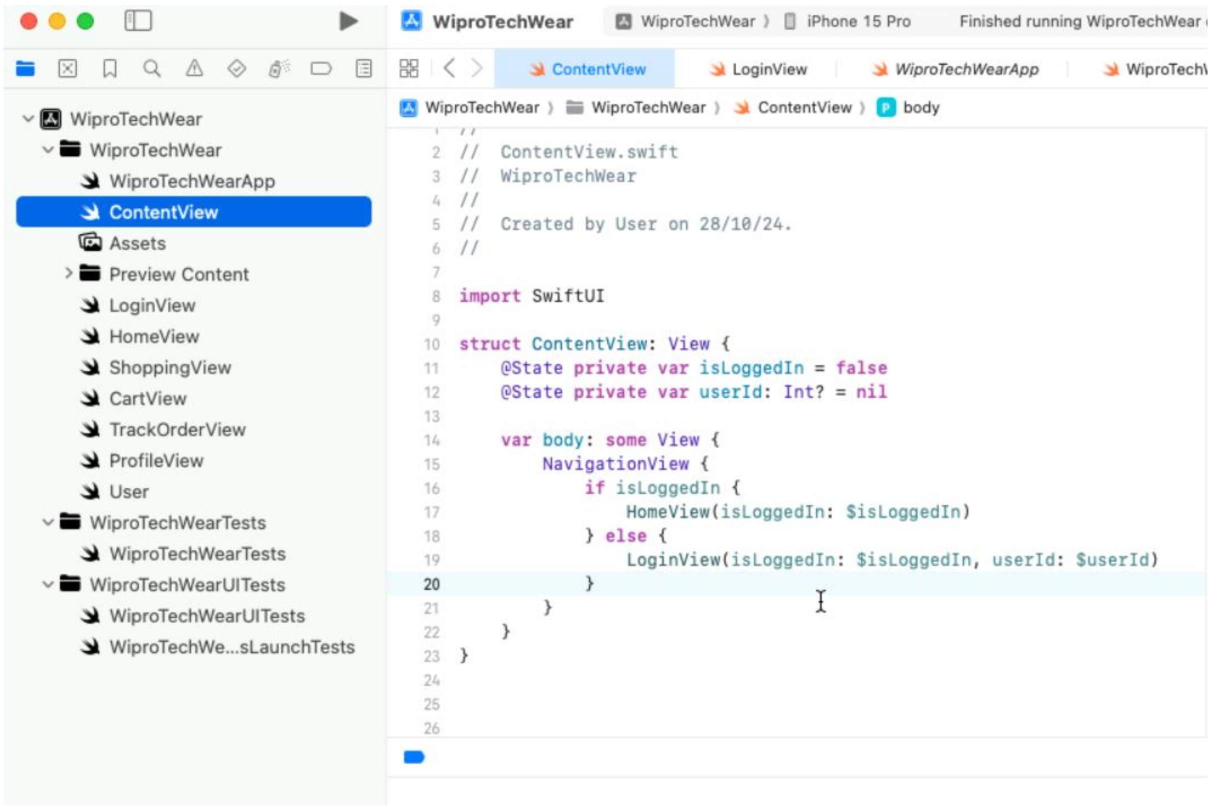
OK

Your Cart

Place Order

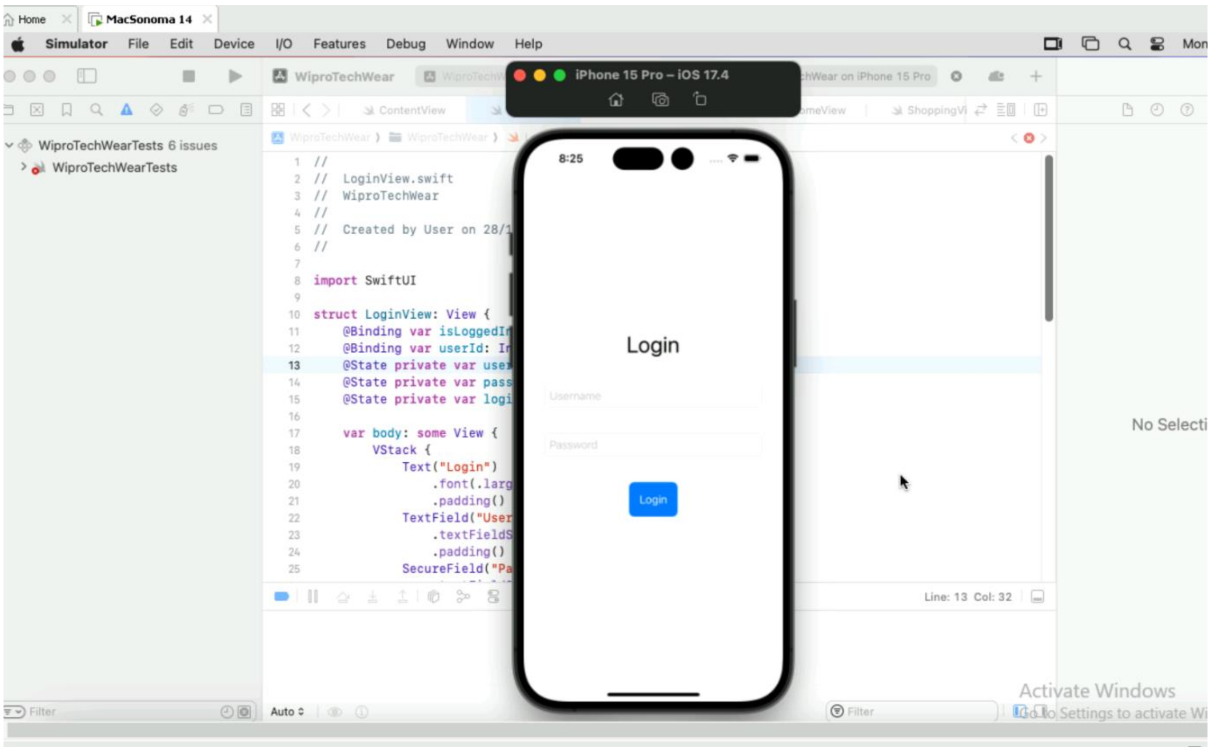
Mobile Applocation :

SwiftUI view – LoginView :



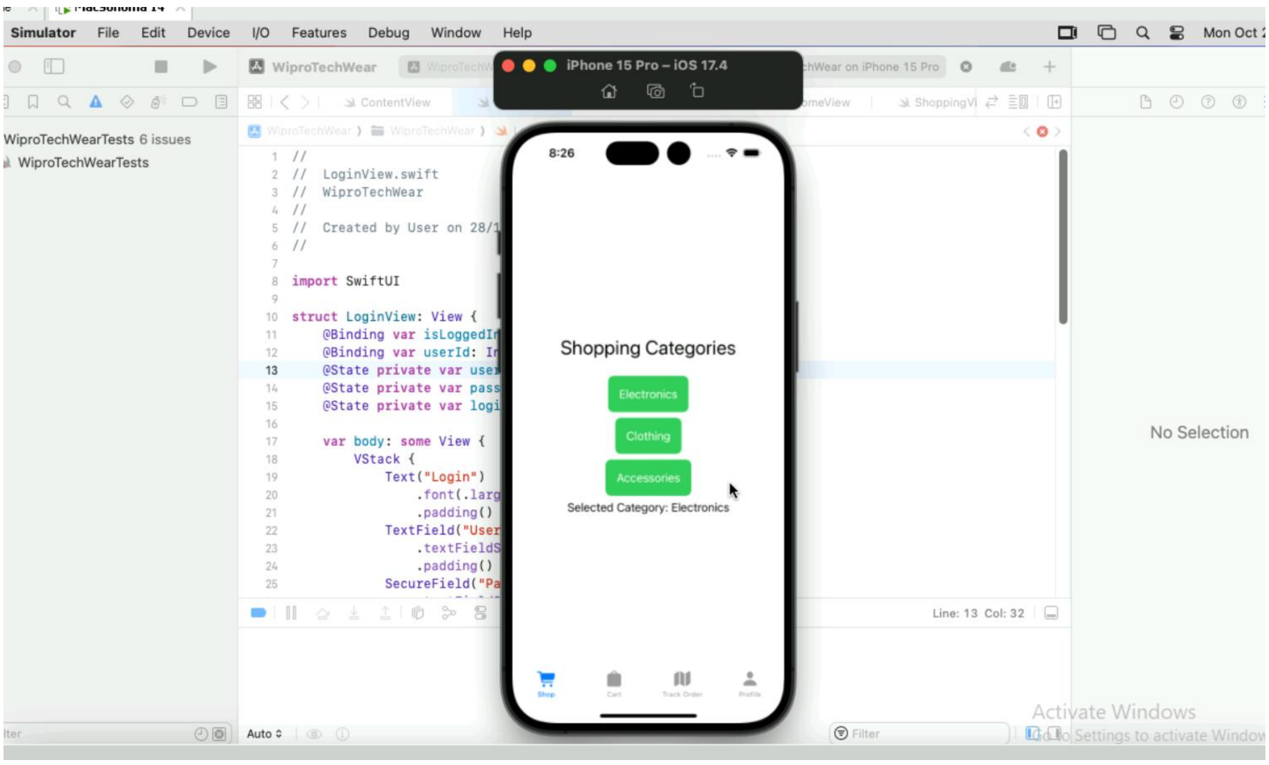
```swift
//
//  ContentView.swift
//  WiproTechWear
//
//  Created by User on 28/10/24.
//

import SwiftUI

struct ContentView: View {
    @State private var isLoggedIn = false
    @State private var userId: Int? = nil

    var body: some View {
        NavigationView {
            if isLoggedIn {
                HomeView(isLoggedIn: $isLoggedIn)
            } else {
                LoginView(isLoggedIn: $isLoggedIn, userId: $userId)
            }
        }
    }
}
```

Login screen :



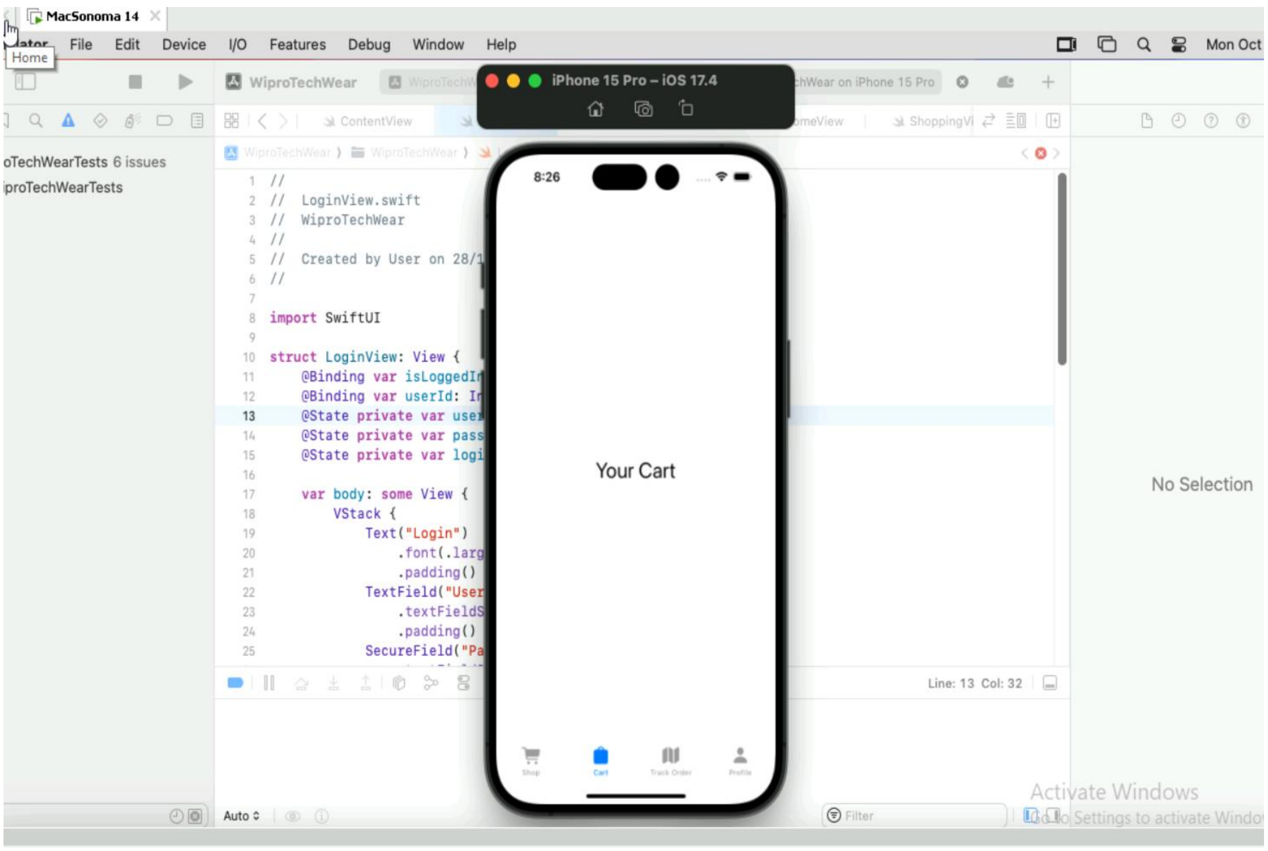
```
//
//  LoginView.swift
//  WiproTechWear
//
//  Created by User on 28/1
//

import SwiftUI

struct LoginView: View {
    @Binding var isLoggedI
    @Binding var userId: I
    @State private var use
    @State private var pass
    @State private var logi

    var body: some View {
        VStack {
            Text("Login")
                .font(.larg
                .padding()
            TextField("User
                .textFieldS
                .padding()
            SecureField("Pa
```

Shopping screen :



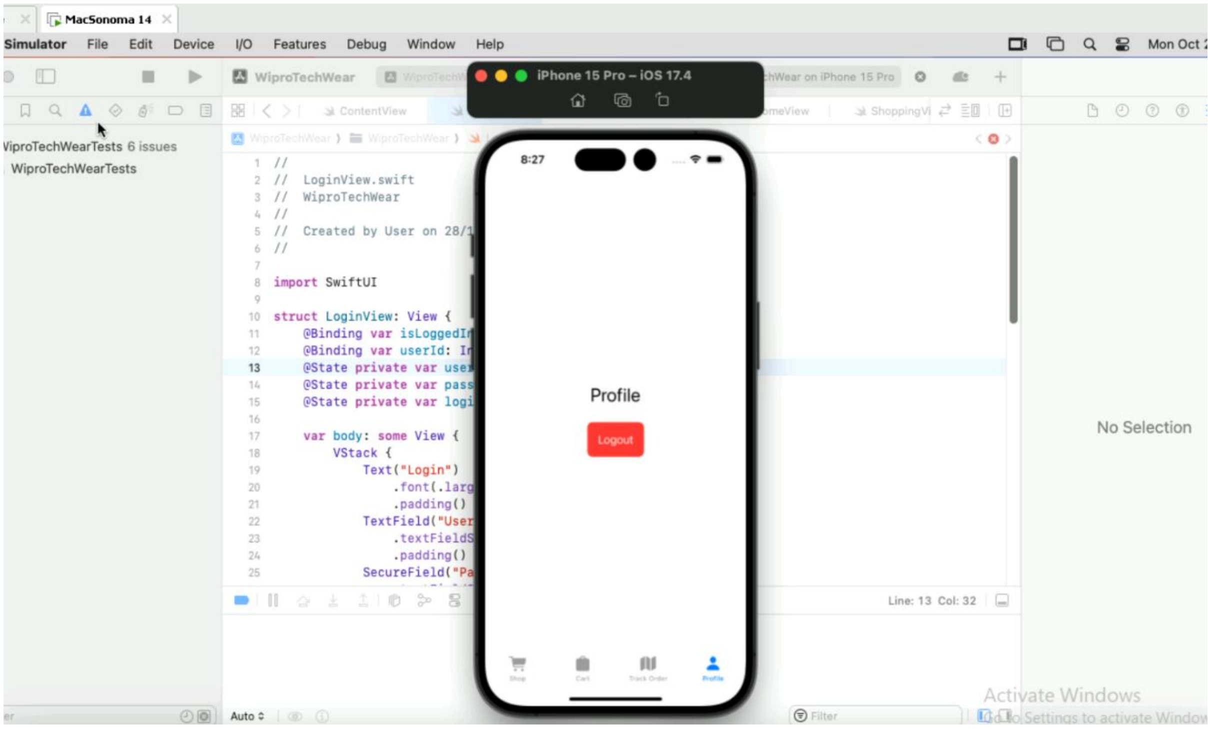
```
//
//  LoginView.swift
//  WiproTechWear
//
//  Created by User on 28/1
//

import SwiftUI

struct LoginView: View {
    @Binding var isLoggedI
    @Binding var userId: I
    @State private var use
    @State private var pass
    @State private var logi

    var body: some View {
        VStack {
            Text("Login")
                .font(.larg
                .padding()
            TextField("User
                .textFieldS
                .padding()
            SecureField("Pa
```

Cart screen :



```
//
//  LoginView.swift
//  WiproTechWear
//
//  Created by User on 28/1
//

import SwiftUI

struct LoginView: View {
    @Binding var isLoggedIn
    @Binding var userId: In
    @State private var user
    @State private var pass
    @State private var logi

    var body: some View {
        VStack {
            Text("Login")
                .font(.larg
                .padding()
            TextField("User
                .textFieldS
                .padding()
            SecureField("Pa
```

Logout screen :



```
//
//  LoginView.swift
//  WiproTechWear
//
//  Created by User on 28/1
//

import SwiftUI

struct LoginView: View {
    @Binding var isLoggedIn
    @Binding var userId: In
    @State private var user
    @State private var pass
    @State private var logi

    var body: some View {
        VStack {
            Text("Login")
                .font(.larg
                .padding()
            TextField("User
                .textFieldS
                .padding()
            SecureField("Pa
```

- Register application is developed in the web application
- Login application is developed in the mobiles application
- In mobile application admin pages will not be there.
- All others features like selecting the products, adding to the cart and placing order etc are common for both mobile and web applications

I implemented most of the use cases and along with the test cases………

THANK YOU