**Manual Testing**

**Test Case 1: Register User**

**Field Description**

Test Case ID TC01

Title Register a New User

Objective To verify that a user can successfully register, log in, and delete the account

Preconditions User is on the homepage of https://automationexercise.com

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Verify home page is visible

4. Click on Signup / Login button

5. Verify New User Signup! is visible

6. Enter name and email address

7. Click Signup button

8. Verify ENTER ACCOUNT INFORMATION is visible

9. Fill in: Title, Name, Email, Password, Date of birth

10. Select checkbox Sign up for our newsletter!

11. Select checkbox Receive special offers from our partners!

12. Fill in: First Name, Last Name, Company, Address, Address2, Country, State, City, Zipcode, Mobile Number

13. Click Create Account button

14. Verify ACCOUNT CREATED! is visible

15. Click Continue button

16. Verify Logged in as [username] is visible

17. Click Delete Account button

18. Verify ACCOUNT DELETED! is visible and click Continue

Expected Result User should be able to register successfully and see confirmation messages. Account should be deleted successfully.

Postconditions User account should be removed from the system.

Test Data Name: Nipuni Navodya

Email: navo.jayakodi@gmail.com

Password: 1111

Date of Birth: 26 June 1999

Country: India

Status Pass

**Test Case 2: Login User with Correct Email and Password**

**Field Description**

Test Case ID TC02

Title Login with Valid Credentials

Objective To verify that a registered user can successfully log in using correct email and password

Preconditions User must have a valid account on https://automationexercise.com

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Verify home page is visible

4. Click on Signup / Login button

5. Verify Login to your account is visible

6. Enter valid email and password

7. Click Login button

8. Verify Logged in as [username] is visible

9. Click Delete Account button

10. Verify ACCOUNT DELETED! is visible

Expected Result User should log in successfully and be able to delete the account

Postconditions Account should be deleted

Test Data Email: navo.jayakodi@gmail.com

Password: 1111

Status Pass

**Test Case 3: Login User with Incorrect Email and Password**

**Field Description**

Test Case ID TC03

Title Login with Invalid Credentials

Objective To verify that login fails when incorrect email or password is provided

Preconditions User is on the login page of https://automationexercise.com

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Verify home page is visible

4. Click on Signup / Login button

5. Verify Login to your account is visible

6. Enter incorrect email and/or password

7. Click Login button

Expected Result Login should fail, and error message "Your email or password is incorrect!" should be displayed

Postconditions User remains on the login page

Test Data Email: navo1.jayakodi@gmail.com

Password: 112

Status Pass

**Test Case 4: Logout User**

**Field Description**

Test Case ID TC04

Title Logout from User Account

Objective To verify that a logged-in user can successfully log out of their account

Preconditions User must be logged in to their account on https://automationexercise.com

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Click on Signup / Login button

4. Enter valid email and password

5. Click Login button

6. Verify Logged in as [username] is visible

7. Click on Logout button

Expected Result User should be logged out and redirected to the login page

Postconditions User session should be terminated

Test Data Email: navo.jayakodi@gmail.com

Password: 1111

Status Pass

**Test Case 5: Register User with Existing Email**

**Field Description**

Test Case ID TC05

Title Attempt to Register with an Existing Email

Objective To verify that the system prevents user registration using an already registered email address

Preconditions User must already have a registered account on https://automationexercise.com

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Click on Signup / Login button

4. Enter an already registered name and email address in the New User Signup! section

5. Click Signup button

Expected Result Error message "Email Address already exist!" should be displayed

Postconditions User remains on the signup/login page

Test Data Name: Nipuni Navodya

Email: navo.jayakodi@gmail.com

Status Pass

**Test Case 6: Contact Us Form**

**Field Description**

Test Case ID TC06

Title Submit Contact Us Form

Objective To verify that users can submit inquiries using the Contact Us form

Preconditions User is on the Contact Us page

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Click on Contact Us button

4. Fill in name, email, subject, and message

5. Upload a file

6. Click Submit button

7. Click OK on the alert

8. Verify success message is displayed

Expected Result Form should be submitted successfully with confirmation message

Postconditions User is redirected back to home page

Test Data Name: Nipuni Navodya

Email: navo.jayakodi@gmail.com

Subject: hello

Message: hi

Attachment: testfile.pdf

Status Pass

**Test Case 7: Verify Test Cases Page**

**Field Description**

Test Case ID TC07

Title Access and Verify Test Cases Page

Objective To confirm that the test cases page is accessible and displayed correctly

Preconditions User is on the homepage

Test Steps 1. Launch the browser

2. Navigate to https://automationexercise.com

3. Click on Test Cases button

Expected Result User should be navigated to the Test Cases page and view test cases

Postconditions User remains on Test Cases page

Test Data N/A

Status Pass

**Test Case 8: Verify All Products and Product Detail Page**

**Field Description**

Test Case ID TC08

Title View All Products and Product Details

Objective To verify that products list and individual product detail pages work correctly

Preconditions User is on the Products page

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click on Products

4. Verify list of all products is visible

5. Click on any product

6. Verify product detail is visible

Expected Result Products list and product detail page should load correctly

Postconditions User can view details and interact with products

Test Data N/A

Status Pass

**Test Case 9: Search Product**

**Field Description**

Test Case ID TC09

Title Search for a Product

Objective To verify that users can search for products using keywords

Preconditions User is on the Products page

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click on Products

4. Enter product name in search bar (e.g., "dress")

5. Click Search button

6. Verify relevant products are displayed

Expected Result Matching products should be shown based on search term

Postconditions User can view filtered product list

Test Data Search Term: dress

Status Pass

**Test Case 10: Verify Subscription in Home Page**

**Field Description**

Test Case ID TC10

Title Subscription Field in Home Page

Objective To verify that users can subscribe to the newsletter from the homepage

Preconditions User is on the homepage

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Scroll to bottom of the homepage

4. Enter email in Subscription input field

5. Click Arrow button

6. Verify success message is shown

Expected Result Subscription should be successful and confirmation message shown

Postconditions Email is recorded for newsletter subscription

Test Data Email: [navo.jayakodi@gmail.com](mailto:navo.jayakodi@gmail.com)

Status Pass

**Test Case 11: Verify Subscription in Cart Page**

**Field Description**

Test Case ID TC11

Title Subscription Field in Cart Page

Objective To verify the newsletter subscription from the cart page

Preconditions User is on the cart page

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click on Cart

4. Scroll to bottom of the cart page

5. Enter email in Subscription field

6. Click Arrow button

7. Verify success message is displayed

Expected Result Subscription should be successful from the cart page

Postconditions Email is added to the subscription list

Test Data Email: navo.jayakodi@gmail.com

Status Pass

**Test Case 12: Add Products in Cart**

**Field Description**

Test Case ID TC12

Title Add Products to Cart

Objective To verify that users can add products to the cart

Preconditions User is on the products page

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click on Products

4. Click Add to cart for any product

5. Click Continue Shopping or View Cart

6. Verify the product appears in the cart

Expected Result Product should be successfully added and visible in the cart

Postconditions Product appears in cart with correct details

Test Data Product: blue top

Status Pass

**Test Case 13: Verify Product Quantity in Cart**

**Field Description**

Test Case ID TC13

Title Update and Verify Product Quantity in Cart

Objective To verify that users can change product quantity in the cart

Preconditions Product is already added to the cart

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Add product to cart

4. Click on Cart

5. Update the quantity field (e.g., set to 4)

6. Click Update (if needed)

7. Verify updated quantity and price

Expected Result Quantity should update and reflect correct total

Postconditions Updated cart reflects quantity change

Test Data Quantity: 4

Status Pass

**Test Case 14: Place Order - Register While Checkout**

**Field Description**

Test Case ID TC14

Title Place Order with New Registration During Checkout

Objective To verify that users can register and complete order during checkout

Preconditions No account registered beforehand

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Add product to cart

4. Click Cart > Proceed To Checkout

5. Click Register / Login

6. Complete registration

7. Verify logged in and redirected to checkout

8. Fill in delivery/payment info

9. Click Place Order

10. Verify success message

Expected Result Order should be placed successfully with account registered

Postconditions Order appears in user history

Test Data User details + valid order info

Status Pass

**Test Case 15: Place Order - Register Before Checkout**

**Field Description**

Test Case ID TC15

Title Place Order After Registering Before Checkout

Objective To verify that users who register before checkout can place orders successfully

Preconditions User must be registered and logged in

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click Signup/Login, register a new account

4. Login and return to homepage

5. Add product to cart

6. Click Cart > Proceed To Checkout

7. Fill delivery/payment info

8. Click Place Order

9. Verify success message

Expected Result Order is placed successfully

Postconditions Order should be visible in order history

Test Data Registered user credentials, order info

Status Pass

**Test Case 16: Place Order - Login Before Checkout**

**Field Description**

Test Case ID TC16

Title Place Order After Logging In Before Checkout

Objective To verify that a logged-in user can place an order

Preconditions User is already registered

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click Signup/Login and log in

4. Add product to cart

5. Click Cart > Proceed To Checkout

6. Verify address and order details

7. Fill in payment and delivery info

8. Click Place Order

9. Verify order success message

Expected Result Order is placed successfully

Postconditions Order saved in user's order history

Test Data Existing user credentials, payment info

Status Pass

**Test Case 17: Remove Products From Cart**

**Field Description**

Test Case ID TC17

Title Remove Item from Cart

Objective To verify that a user can remove product(s) from the cart

Preconditions At least one product added to cart

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Add product to cart

4. Click Cart

5. Click X or Remove button on product

6. Verify product is removed

Expected Result Product is removed and no longer appears in cart

Postconditions Cart is updated to reflect changes

Test Data blue top

Status Pass

**Test Case 18: View Category Products**

**Field Description**

Test Case ID TC18

Title View Products by Category

Objective To verify that users can browse products by category

Preconditions None

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Scroll to left panel

4. Click on any category (e.g., Women > Dress)

5. Verify category page and products list appears

Expected Result Selected category's products are shown

Postconditions Category page loaded successfully

Test Data Category: Women > Dress

Status Pass

**Test Case 19: View & Cart Brand Products**

**Field Description**

Test Case ID TC19

Title View and Add Brand Products to Cart

Objective To verify brand-wise product browsing and cart addition

Preconditions None

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Scroll to Brands section

4. Click on any brand (e.g., Polo)

5. Verify brand product list

6. Click Add to cart on any brand item

7. View cart and verify item

Expected Result Brand items are listed and can be added to cart

Postconditions Cart updated with brand product

Test Data Brand: Polo

Status Pass

**Test Case 20: Search Products and Verify Cart After Login**

**Field Description**

Test Case ID TC20

Title Search Products, Add to Cart, and Verify After Login

Objective To verify product search, cart addition, and cart persistence post-login

Preconditions User not logged in

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Search for a product (e.g., Tshirt)

4. Click Add to cart

5. Click Signup/Login, log in

6. Navigate to Cart

7. Verify product is still in the cart

Expected Result Cart should persist product even after login

Postconditions Product remains in cart

Test Data Search term: Tshirt

Status Pass

**Test Case 21: Add Review on Product**

**Field Description**

Test Case ID TC21

Title Add a Review to a Product

Objective To verify that users can add product reviews

Preconditions User is on a product detail page

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Click Products and select any product

4. Scroll to Write Your Review section

5. Enter name, email, and review

6. Click Submit button

7. Verify success message is displayed

Expected Result Review is submitted and success message appears

Postconditions Review is saved

Test Data Nipuni Navodya, [navo.jayakodi@gmail.com,](about:blank) good quality

Status Pass

**Test Case 22: Add to Cart from Recommended Items**

**Field Description**

Test Case ID TC22

Title Add Products to Cart from Recommended Items

Objective To verify that recommended items can be added to cart

Preconditions None

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Scroll to bottom of homepage

4. Verify Recommended Items section

5. Click Add to cart on any item

6. View cart and verify product is added

Expected Result Item is added to cart successfully

Postconditions Cart contains the recommended product

Test Data blue top

Status Pass

**Test Case 23: Verify Address Details in Checkout Page**

**Field Description**

Test Case ID TC23

Title Verify Address Details During Checkout

Objective To ensure billing and delivery addresses are correct

Preconditions User is logged in with profile details filled

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Log in

4. Add product to cart and click Cart > Proceed To Checkout

5. Verify address details shown under Billing and Delivery sections

Expected Result Address info matches user profile

Postconditions Checkout process continues normally

Test Data User’s saved address

Status Pass

**Test Case 24: Download Invoice After Purchase Order**

**Field Description**

Test Case ID TC24

Title Download Invoice After Successful Purchase

Objective To ensure users can download invoices after placing an order

Preconditions Order must be successfully placed

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Log in

4. Place a complete order

5. On confirmation page, click Download Invoice

6. Verify invoice download

Expected Result Invoice is downloaded in PDF format

Postconditions File available in user’s system

Test Data Order ID and purchase details

Status Pass

**Test Case 25: Verify Scroll Up using 'Arrow' Button and Scroll Down Functionality**

**Field Description**

Test Case ID TC25

Title Test Scroll Up via Arrow Button and Scroll Down Function

Objective To verify page scrolls correctly using the arrow button

Preconditions Page must be long enough to scroll

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Scroll down to footer

4. Click scroll-up arrow button

5. Verify that user is returned to top of the page

Expected Result Scroll functionality works correctly

Postconditions Page lands at top

Test Data None

Status Pass

**Test Case 26: Verify Scroll Up without 'Arrow' Button and Scroll Down Functionality**

**Field Description**

Test Case ID TC26

Title Test Scroll Up without Arrow Button

Objective To ensure users can scroll manually using the mouse or keyboard

Preconditions Page must be scrollable

Test Steps 1. Launch browser

2. Navigate to https://automationexercise.com

3. Scroll down to bottom manually

4. Scroll back up using mouse/keyboard

5. Verify content at the top of the page is visible

Expected Result Page scrolls smoothly without the arrow button

Postconditions Page is fully accessible by scrolling manually

Test Data None

Status Pass

**Test Coverage Analysis**

**Feature Covered By Covered? Any Gaps?**

User Registration TC1, TC5, TC14, TC15 Yes None

User Login TC2, TC3, TC16 Yes None

Password Recovery No No Missing test (Add "ForgotPassword")

Contact Form TC6 Yes None

Product Search TC9, TC20 Yes None

Product Filter No No Missing test (Filter and sorting functionality)

Wishlist No No Missing test (Add test for wishlist/favorites feature)

Product Details TC8 Yes Could add test for product image zoom

Cart Functionality TC12, TC13, TC17 Yes None

Checkout & Orders TC14, TC15, TC16, TC23 Yes Could add tests for payment failures and order cancellation

Invoice Download TC24 Yes None

Subscription TC10, TC11 Yes None

Reviews TC21 Yes None

Scroll Functionality TC25, TC26 Yes None

User Profile Update No No Missing test (Update name, email, password, etc.)

Email Verification No No Missing test (Email verification after registration)

Negative Tests TC3, TC5, TC16 No Could add more negative tests like invalid email formats, empty fields, broken links

**Suggestions for Improving Coverage:**

* Add tests for the "Forgot Password" functionality - Ensure the reset process works correctly.
* Test for product filtering and sorting - Verify that users can filter products by price, color, and other criteria.
* Add Wishlist/Favorites tests - Users should be able to save products to a wishlist and view them later.
* Test editing user profile - Ensure users can update their name, email, password, and address.
* Test email verification - After registration, an email verification step could be added.
* Negative Tests - Add tests for:

Invalid email formats

Empty required fields in forms

Broken links or images on the site

**API Testing**

### API 1: Get All Products List

**Test Case ID**: API-001  
**Test Description**: Verify the response of the GET method for retrieving all products.  
**Test Steps**:

Send a GET request to /api/productsList.

Verify that the response status code is 200.

Verify the structure of the response to include product details such as id, name, price, etc.

**Expected Result:**

Response code should be 200 OK.

The response body should contain a list of products with correct fields like id, name, price, brand, and category.

**Actual Result**:

{"responseCode": 200, "products": [{"id": 1, "name": "Blue Top", "price": "Rs. 500", "brand": "Polo", "category": {"usertype": {"usertype": "Women"}, "category": "Tops"}}, {"id": 2, "name": "Men Tshirt", "price": "Rs. 400", "brand": "H&M", "category": {"usertype": {"usertype": "Men"}, " ……………..etc.

**Test Status**: Pass

### API 2: POST To All Products List

**Test Case ID**: API-002  
Test Description: Verify the response when sending a POST request to /api/productsList.

**Test Steps**:

Send a POST request to /api/productsList.

Observe the response code.

Verify that the request is rejected with a 405 Method Not Allowed status.

**Expected Result**:

Response code should be 405 Method Not Allowed.

**Actual Result**:

{"responseCode": 405, "message": "This request method is not supported."

**Test Status**: Pass

### API 3: Get All Brands List

**Test Case ID**: API-003  
**Test Description**: Verify the response of the GET method for retrieving all brands.  
**Test Steps**:

Send a GET request to /api/brandsList.

Verify that the response status code is 200.

Verify the structure of the response to include all brand names.

**Expected Result**:

Response code should be 200 OK.

The response body should contain a list of brands like Polo, H&M, etc.

**Actual Result**:

{"responseCode": 200, "brands": [{"id": 1, "brand": "Polo"}, {"id": 2, "brand": "H&M"}, {"id": 3, "brand": "Madame"},

**Test Status**: Pass

### API 4: PUT To All Brands List

**Test Case ID**: API-004  
**Test Description**: Verify the response when sending a PUT request to /api/brandsList.  
**Test Steps**:

Send a PUT request to /api/brandsList.

Observe the response code.

Verify that the request is rejected with a 405 Method Not Allowed status.

**Expected Result**:

Response code should be 405 Method Not Allowed.

**Actual Result**:

{"responseCode": 405, "message": "This request method is not supported."}

**Test Status**: Pass

### API 5: POST To Search Product

**Test Case ID**: API-005  
**Test Description**: Verify the response when sending a POST request to search for a product.  
**Test Steps**:

Send a POST request to /api/searchProduct with the payload:

{

"search\_product": "dress"

}

Observe the response code.

Verify that the response contains a list of products matching the search term "dress".

**Expected Result**:

Response code should be 200 OK.

The response body should contain a list of products related to "dress".

**Actual Result**:

{"responseCode": 200, "products": [{"id": 3, "name": "Sleeveless Dress", "price": "Rs. 1000", "brand": "Madame",

"category": {"usertype": {"usertype": "Women"}, "category": "Dress"}}, {"id": 4, "name": "Stylish Dress", "price": "Rs.

1500", "brand": "Madame", "category": {"usertype": {"usertype": "Women"}, "category": "Dress"}}, {"id": 16, "name":

"Sleeves Top and Short - Blue & Pink", "price": "Rs. 478", "brand": "Babyhug", "category": {"usertype": {"usertype":

"Kids"}, "category": "Dress"}}, {"id": 19, "name": "Sleeveless Unicorn Patch Gown - Pink", "price": "Rs. 1050", "brand":

"Allen Solly Junior", "category": {"usertype": {"usertype": "Kids"}, "category": "Dress"}}, {"id": 20, "name": "Cotton

**Test Status** : Pass

### API 6: POST To Search Product without search\_product parameter

**Test Case ID**: API-006  
**Test Description**: Verify the response when sending a POST request to search for a product without providing the search\_product parameter.  
**Test Steps**:

Send a POST request to /api/searchProduct with the empty payload:

Observe the response code.

Verify that the response indicates a bad request (400).

**Expected Result**:

Response code should be 400 Bad Request.

The response body should indicate that the search\_product parameter is required.

**Actual Result**:

{"responseCode": 400, "message": "Bad request, search\_product parameter is missing in POST request."}

**Test Status**: Pass

### API 7: POST To Verify Login with valid details

**Test Case ID**: API-007  
**Test Description**: Verify the response when posting valid login credentials.  
**Test Steps**:

Send a POST request to /api/verifyLogin with the payload:

{

"email": "valid@example.com",

"password": "correctpassword"

}

Observe the response code.

Verify that the response contains a message confirming the user exists.

**Expected Result**:

Response code should be 200 OK.

The response body should confirm that the user exists.

**Actual Result**:

{"responseCode": 200, "message": "User exists!"}

**Test Status**: Pass

### API 8: POST To Verify Login without email parameter

**Test Case ID**: API-008  
**Test Description**: Verify the response when the email parameter is missing from the login request.  
**Test Steps**:

Send a POST request to /api/verifyLogin with the payload:

{

"password": "password123"

}

Observe the response code.

Verify that the response indicates a missing email parameter.

**Expected Result**:

Response code should be 400 Bad Request.

The response body should mention that the email parameter is required.

**Actual Result**:

{"responseCode": 400, "message": "Bad request, email or password parameter is missing in POST request."}

**Test Status**: Pass

### API 9: DELETE To Verify Login

**Test Case ID**: API-009  
**Test Description**: Verify the response when trying to DELETE using the /verifyLogin endpoint.  
**Test Steps**:

Send a DELETE request to /api/verifyLogin.

Observe the response code.

Verify that the request is rejected with a 405 Method Not Allowed status.

**Expected Result**:

Response code should be 405 Method Not Allowed.

**Actual Result**:

{"responseCode": 405, "message": "This request method is not supported."}

**Test Status**: Pass

### API 10: POST To Verify Login with invalid details

**Test Case ID**: API-010  
**Test Description**: Verify the response when posting invalid login credentials.  
**Test Steps**:

Send a POST request to /api/verifyLogin with the payload:

{

"email": "navo1.jayakodi@gmail.com",

"password": "112"

}

Observe the response code.

Verify that the response indicates the user is not found or invalid credentials.

**Expected Result**:

Response code should be 404 Not Found or similar.

The response body should indicate "User not found."

**Actual Result**:

{"responseCode": 404, "message": "User not found!"}

**Test Status**: Pass

### **API 11: PUT Method for Verify Login**

**Test Case ID**: API-011

**Test Description**: Verify the response when using PUT method on the /api/verifyLogin endpoint.

#### **Test Steps:**

Send a **PUT** request to /api/verifyLogin with dummy login credentials:

{

"email": "navotest1.jayakodi@gmail.com",

"password": "1111"}

Observe the response code.

Verify that the method is not allowed.

#### **Expected Result:**

**Response code**: 201

**Message**: "User created!"

#### **Actual Result:**

{"responseCode": 201, "message": "User created!"}

#### **Test Status:** Pass

### **API 12: DELETE Method for Verify Login**

**Test Case ID**: API-012

**Test Description**: Verify the response when using DELETE method on the /api/verifyLogin endpoint.

#### **Test Steps:**

Send a **DELETE** request to /api/verifyLogin.

Observe the response code and message.

#### **Expected Result:**

**Response code**: 200

**Message**: "Account deleted!"

#### **Actual Result:**

{"responseCode": 200, "message": "Account deleted!"}

#### **Test Status:** Pass

### **API 13: GET Request to Search Product**

**Test Case ID**: API-013

**Test Description**: Verify the response when using a GET request instead of POST on /api/searchProduct.

#### **Test Steps:**

Send a **GET** request to /api/searchProduct.

Observe the response code and message.

#### **Expected Result:**

**Response code**: 200

**Message**: "User updated!"

#### **Actual Result:**

{"responseCode": 200, "message": "User updated!"}

#### **Test Status:** Pass

### ✅ **API 14: DELETE Method for Search Product**

**Test Case ID**: API-014

**Test Description**: Verify the response when using DELETE method on /api/searchProduct.

#### **Test Steps:**

Send a **DELETE** request to /api/searchProduct.

Observe the response.

#### **Expected Result:**

**Response code**: 200

**Message**: User Details

#### **Actual Result:**

{"responseCode": 200, "user": {"id": 662441, "name": "Nipuni Navodya", "email": "navo.jayakodi@gmail.com", "title":

"Mrs", "birth\_day": "26", "birth\_month": "June", "birth\_year": "1999", "first\_name": "Nipuni", "last\_name": "Navodya",

"company": "AgroWorld", "address1": "Saranankara Mawatha", "address2": "Gaiyala", "country": "India", "state": "north

western province", "city": "kuliyapitiya", "zipcode": "60200"}}

#### **Test Status:** Pass