**Exp no : 1** **Develop the tests plan for testing an e-commerce**
**Date:** **web/mobile-application**

## Aim

To develop the test plan for testing an e-commerce web/mobile application

## Procedure

### 1.Scope:

- User Interface (UI) Testing: Evaluate layout consistency, responsive design, and navigation functionality across web and mobile platforms.
- Functional Testing: Verify critical features like user registration/login, product search/filtering, shopping cart, checkout process, order confirmation, and user account management.
- Performance Testing: Assess application load handling under high traffic and measure response times during various scenarios.
- Security Testing: Check data protection measures, authentication vulnerabilities, and ensure compliance with payment security standards
- Usability and Compatibility Testing: Gather user experience feedback and ensure compatibility across different browsers, devices, and operating systems.
- Integration and Regression Testing: Test interactions with third-party services and verify that new features do not disrupt existing functionalities.

### 2. Test Objectives:

- Verify that users can browse and search for products.
- Confirm that products are displayed accurately with correct details.
- Test the user registration and login process.
- Validate the shopping cart and checkout process.
- Test payment processing and order placement.
- Check for responsiveness and usability on mobile devices.
- Verify security aspects such as data encryption and session management.

### 3. Test Environment:

- Web Browsers: Chrome, Firefox, Safari, Edge.
- Mobile Devices: iOS and Android devices.
- Testing Framework: Selenium WebDriver with Java.
- Load Testing: Apache JMeter for performance testing.
- Security Testing: OWASP ZAP or similar tool.

**4. Test Cases:**

- Test cases should cover registration, login, product browsing, cart management, payment processing, etc.
- Include positive and negative scenarios, edge cases, and stress tests.

**5. Test Execution:**

- Automate test cases using Selenium WebDriver.
- Execute tests across different browsers and mobile devices.
- Use load testing tools to simulate high user traffic and monitor performance.
- Perform security testing to identify vulnerabilities.

**6. Test Data:**

- Use real and dummy data for testing.
- Test accounts, products, payment methods, addresses, etc.

**7. Risks and Assumptions:**

- Risk: Changes in the website's structure might break automation scripts.
- Risk: Mobile app updates might affect test scripts and compatibility.
- Assumption: Test data in the environment matches real-world scenarios.

**8. Deliverables:**

- Test reports summarizing test results, defects found, and performance analysis.
- Documentation of test cases, procedures, and configurations.
- Recommendations for improvements and fixes.

### Test Plan

Inorder to test the E-Commerce application we test its individual features and ensure their proper functioning. The following are the features that we will have to test to ensure proper functioning of the website.

1. Login
2. Search Feature
3. Product Details Page (PDP)
4. Shopping Cart
5. Order Confirmation Page

### 1 – Testing the Login flow of an eCommerce Site

- Verify if the customer can purchase items as a guest user, assuming the site permits it.

- Verify if a guest user can easily create an account from the order confirmation page after placing an order.
- Verify if a guest user can register as a member from the registration page.
- Verify if the products in the cart are correctly associated with the logged-in member's account.
- Verify if the user is still being prompted to sign in once logged in; this should not happen.
- Verify if user sessions are being maintained for the intended time period.
- Verify if the user's session times out and expires after the stipulated time.

**2 – Testing the Search feature of an eCommerce Site**

- Verify if the search feature is robust enough to allow searching by product names, brand names, or product categories.
- Verify if the search results are relevant for all variations of search keywords.
- Verify if the search results page has sorting options. If sorting options are available, check if customers can sort the results based on price (high to low, low to high), date ranges, brand, review/ratings, etc.
- Verify how many results are displayed on each page.
- Verify if the next page loads automatically when the user scrolls. If not, verify if the user can easily navigate to the next search result page using pagination.

**3 – Testing the Product Details Page (PDP) of an eCommerce Site**

- Verify if the page displays all the product-related information (e.g., product name, SKU, price, discount if any, product images, quantity changer, etc.) as expected.
- Verify if the page displays other similar products that were bought by customers who purchased this item.
- Verify if the customer can customize the product if it is customizable and available with variations (e.g., size, colors, etc.).
- Verify if the page shows the stock availability accurately.
- Verify if the user can select the required number of items using the quantity selector and then add the product to the cart.
- Verify if the product gets added to the cart when added.
- Verify if the quantity in the cart gets updated when the product is added to the cart.

**4 – Testing the Shopping Cart of an Online Shopping Platform**

- Verify that the cart updates to show the updated quantity when an item is added to it.
- Verify if you can increase or decrease the quantity of an item from both the cart and mini-cart.
- Verify if you can remove an item from both the cart and mini-cart.
- Verify if you can add the same item several times to the cart.
- Verify if you can add several variations (e.g., color, size, etc.) of the same item to the cart.
- Verify if you can add several items of different types to the cart.
- Verify if you can visit the product detail page by clicking on an item from the cart.
- Verify if the cart remembers the items, their quantity, and price correctly even if you accidentally close and then reopen the browser.
- Verify if the discount is correctly applied when valid coupon codes and discount vouchers are used in the cart.

**5 – Testing the Order Confirmation Page of an Online Shopping Portal**

- Verify if the page displays the Order ID correctly.
- Verify if the page displays the product details correctly.
- Verify if the page offers an easy way for guest users to create an account.
- Verify if the customer receives an order confirmation email or SMS text message along with the order.

| **Evaluation Parameter** | **Max. Marks** | **Marks Awarded** |
|---|---|---|
| Observation | 20 | |
| Implementation | 40 | |
| Output | 10 | |
| Viva | 10 | |
| Record | 20 | |
| **Total Marks** | **100** | |

**RESULT**

Thus the test plan for testing an e-commerce web/mobile application was successfully created and output verified.

**Exp No : 2** **Design the test cases for testing the e-commerce application**
**Date :**

## Aim

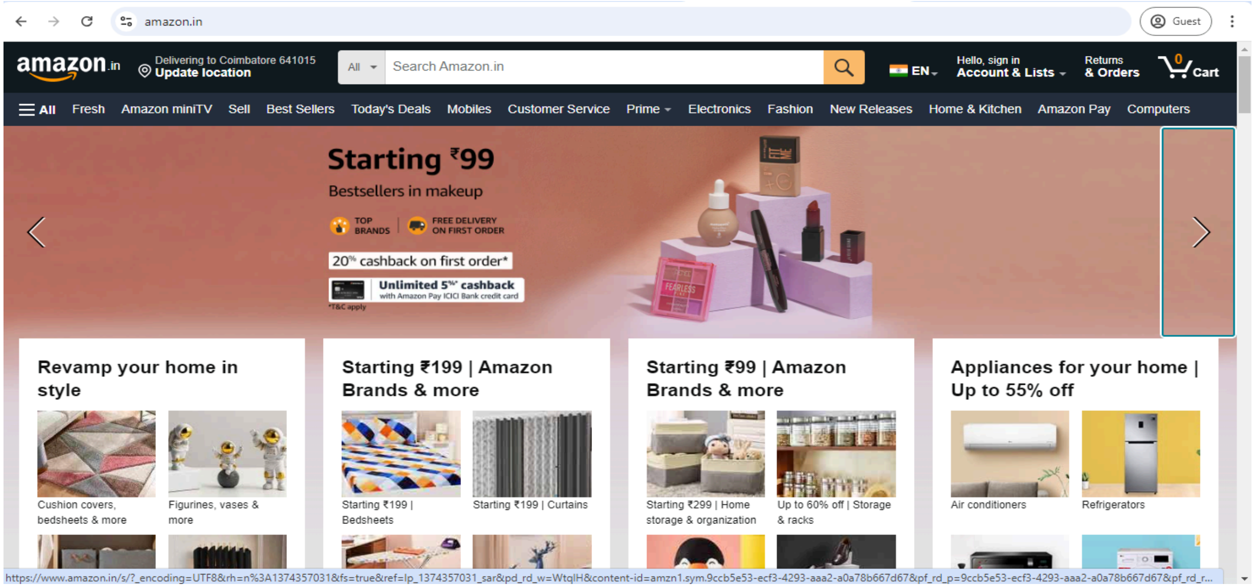
To design the test cases for testing the e-commerce application(amazon.in)

## Procedure

The following are the scenarios that are designed for our test cases to ensure a smooth user experience while using the ecommerce website .

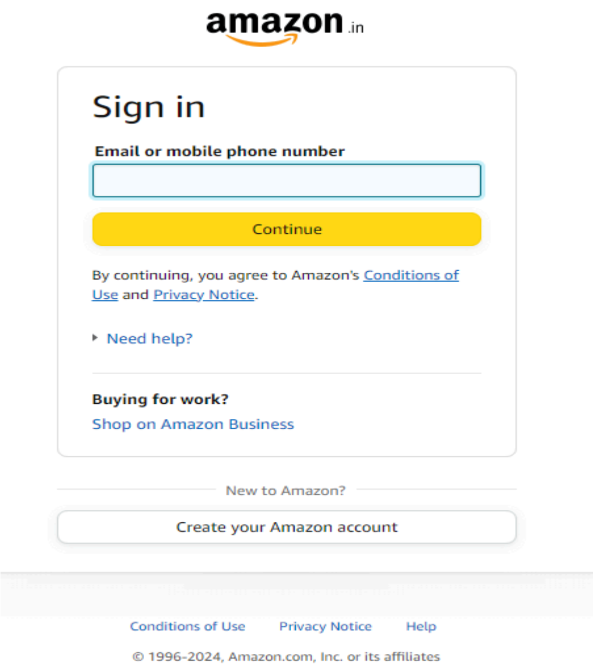
## Home Page Test cases

- To verify the home page should be displayed after a successful login or Not.
- To verify that the user name is displayed on the homepage or not.
- To verify that the home page is working in different browsers.
- To verify that the products are displayed on the home page or not.
- To verify that the search functionality is available on the home page or not.


amazon.in
Guest
amazon.in
Delivering to Coimbatore 641015
Update location
All
Search Amazon.in
EN
Hello, sign in
Account & Lists
Returns
& Orders
0
Cart
All
Fresh
Amazon miniTV
Sell
Best Sellers
Today's Deals
Mobiles
Customer Service
Prime
Electronics
Fashion
New Releases
Home & Kitchen
Amazon Pay
Computers
Starting ₹99
Bestsellers in makeup
TOP BRANDS
FREE DELIVERY ON FIRST ORDER
20% cashback on first order*
Unlimited 5% cashback
with Amazon Pay ICICI Bank credit card
*T&C apply
Revamp your home in style
Cushion covers, bedsheets & more
Figurines, vases & more
Starting ₹199 | Amazon Brands & more
Starting ₹199 | Bedsheets
Starting ₹199 | Curtains
Starting ₹99 | Amazon Brands & more
Starting ₹299 | Home storage & organization
Up to 60% off | Storage & racks
Appliances for your home | Up to 55% off
Air conditioners
Refrigerators
https://www.amazon.in/s/?_encoding=UTF8&rh=n%3A1374357031&fs=true&ref=lp_1374357031_sar&pd_rd_w=WtqlH&content-id=amzn1.sym.9ccb5e53-ecf3-4293-aaa2-a0a78b667d67&pf_rd_p=9ccb5e53-ecf3-4293-aaa2-a0a78b667d67&pf_rd_r...

## Login test cases

- To verify Valid Username and Valid Password
- To verify Invalid Username and Valid Password
- To verify Valid Username and Invalid Password
- To verify Invalid Username and Invalid Password



amazon.in

Sign in

Email or mobile phone number

Continue

By continuing, you agree to Amazon's Conditions of Use and Privacy Notice.

Need help?

Buying for work?
Shop on Amazon Business

New to Amazon?

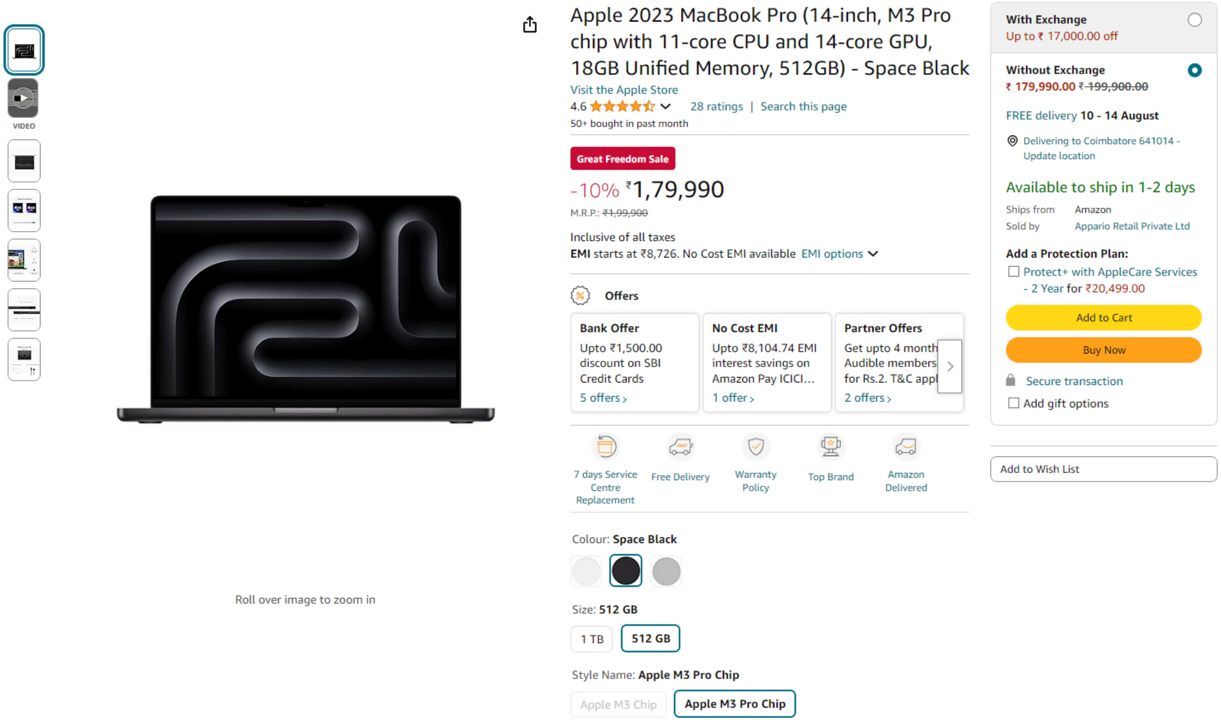
Create your Amazon account

Conditions of Use   Privacy Notice   Help

© 1996-2024, Amazon.com, Inc. or its affiliates

## Product Detail Test Cases

- To verify that the images of products are displayed on the page correctly or not.
- To verify that the price of the product should be displayed.
- To verify that the product specifications should display.
- To verify that the product reviews are displayed on the page correctly or not.



Roll over image to zoom in

Apple 2023 MacBook Pro (14-inch, M3 Pro chip with 11-core CPU and 14-core GPU, 18GB Unified Memory, 512GB) - Space Black

Visit the Apple Store

4.6 ★★★★☆ 28 ratings | Search this page

50+ bought in past month

Great Freedom Sale

-10% ₹1,79,990

M.R.P.: ~~₹1,99,900~~

Inclusive of all taxes

**EMI** starts at ₹8,726. No Cost EMI available EMI options

Offers

**Bank Offer**
Upto ₹1,500.00 discount on SBI Credit Cards
5 offers

**No Cost EMI**
Upto ₹8,104.74 EMI interest savings on Amazon Pay ICICI…
1 offer

**Partner Offers**
Get upto 4 month Audible members for Rs.2. T&C appl
2 offers

Colour: **Space Black**

Size: **512 GB**

1 TB   512 GB

Style Name: **Apple M3 Pro Chip**

Apple M3 Chip   Apple M3 Pro Chip

**With Exchange** ○
Up to ₹ 17,000.00 off

**Without Exchange** ●
₹ 179,990.00 ~~₹ 199,900.00~~

FREE delivery **10 - 14 August**

Delivering to Coimbatore 641014 - Update location

Available to ship in 1-2 days

Ships from Amazon
Sold by Appario Retail Private Ltd

**Add a Protection Plan:**
□ Protect+ with AppleCare Services - 2 Year for ₹20,499.00

Add to Cart

Buy Now

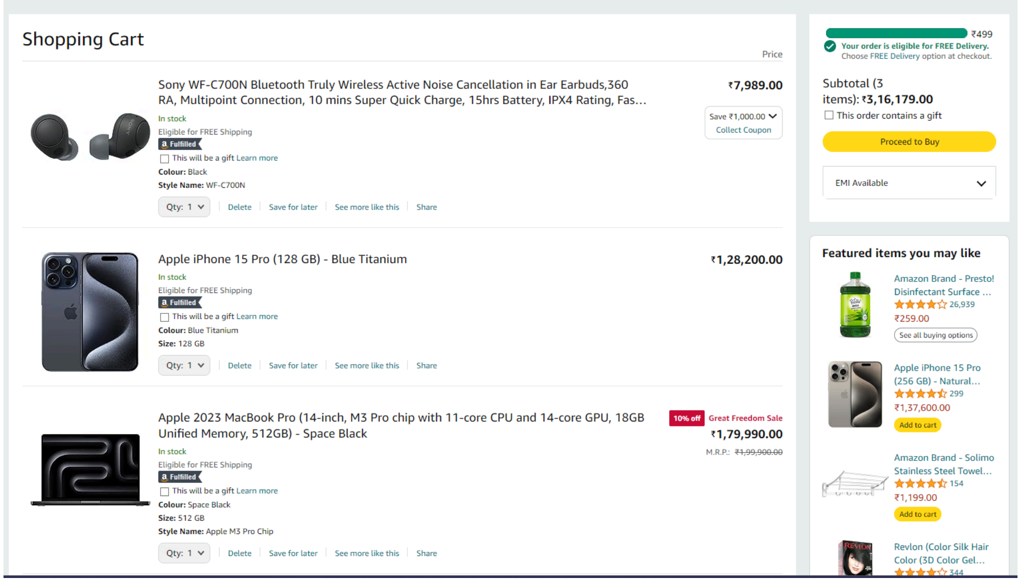
Secure transaction

□ Add gift options

Add to Wish List

## Shopping Cart

- To verify that the Add to cart button is clickable or not.
- To verify after clicking Add to cart button, Products are moved to Cart or not.
- To verify that the total amount of all the items is displayed or not.
- To verify that the tax information is displayed or not.
- To verify that the user is able to see shipping charges as per location



Shopping Cart

Price

Sony WF-C700N Bluetooth Truly Wireless Active Noise Cancellation in Ear Earbuds,360 RA, Multipoint Connection, 10 mins Super Quick Charge, 15hrs Battery, IPX4 Rating, Fas...

₹7,989.00

Save ₹1,000.00 Collect Coupon

In stock

Eligible for FREE Shipping

a Fulfilled

□ This will be a gift Learn more

**Colour:** Black

**Style Name:** WF-C700N

Qty: 1 | Delete | Save for later | See more like this | Share

Apple iPhone 15 Pro (128 GB) - Blue Titanium

₹1,28,200.00

In stock

Eligible for FREE Shipping

a Fulfilled

□ This will be a gift Learn more

**Colour:** Blue Titanium

**Size:** 128 GB

Qty: 1 | Delete | Save for later | See more like this | Share

Apple 2023 MacBook Pro (14-inch, M3 Pro chip with 11-core CPU and 14-core GPU, 18GB Unified Memory, 512GB) - Space Black

10% off Great Freedom Sale

₹1,79,990.00

M.R.P.: ~~₹1,99,900.00~~

In stock

Eligible for FREE Shipping

a Fulfilled

□ This will be a gift Learn more

**Colour:** Space Black

**Size:** 512 GB

**Style Name:** Apple M3 Pro Chip

Qty: 1 | Delete | Save for later | See more like this | Share

₹499

Your order is eligible for FREE Delivery.
Choose FREE Delivery option at checkout.

Subtotal (3 items): **₹3,16,179.00**

□ This order contains a gift

Proceed to Buy

EMI Available

**Featured items you may like**

Amazon Brand - Presto! Disinfectant Surface ...
26,939
₹259.00
See all buying options

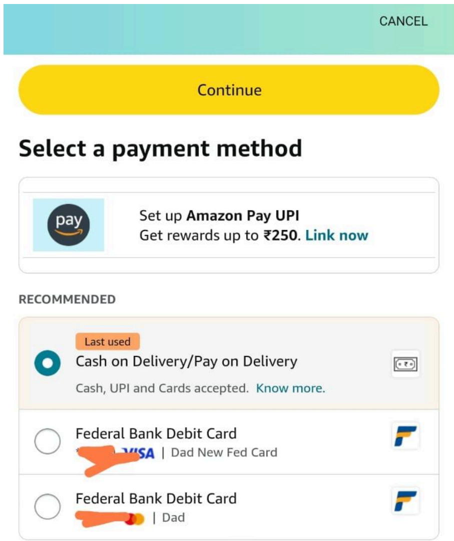
Apple iPhone 15 Pro (256 GB) - Natural...
299
₹1,37,600.00
Add to cart

Amazon Brand - Solimo Stainless Steel Towel...
154
₹1,199.00
Add to cart

Revlon (Color Silk Hair Color (3D Color Gel...
344

## Payments

- To verify that the payment options are displayed on the checkout page or not.
- To verify the Name on Card, Card Number, Expiry Date, and CVV number are mandatory fields on the Payment Page or not.
- To verify that the error message is displayed when the user enters incorrect data in all the mandatory fields on the payment page.
- To verify that the user is able to place an order successfully.



CANCEL

Continue

**Select a payment method**

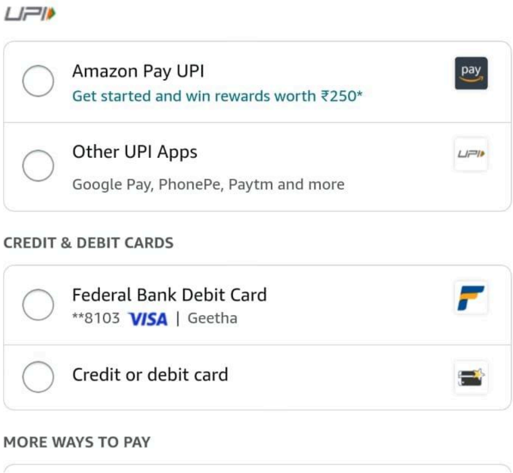
Set up **Amazon Pay UPI**
Get rewards up to **₹250**. **Link now**

RECOMMENDED

Last used
● Cash on Delivery/Pay on Delivery
Cash, UPI and Cards accepted. Know more.

○ Federal Bank Debit Card
[redacted] | Dad New Fed Card

○ Federal Bank Debit Card
[redacted] | Dad



UPI

○ Amazon Pay UPI
Get started and win rewards worth ₹250*

○ Other UPI Apps
Google Pay, PhonePe, Paytm and more

CREDIT & DEBIT CARDS

○ Federal Bank Debit Card
**8103 VISA | Geetha

○ Credit or debit card

MORE WAYS TO PAY

**Test Cases Table**

| Test Scenario Id | Test Scenario Description | Test Case Id | Test Case Description | Test Steps | Precondition | Test Data | Expected Result |
|---|---|---|---|---|---|---|---|
| TS_HomePage | To test the functionality of the Home Page of the application | TC_HP_001 | To verify the home page should be displayed after a successful login or not. | 1.Login with correct credentials | 1.Valid URL<br>2.Test Data | **Username:** test<br>**Password:** correct | The user is redirected to the Home Page of the application |
| | | TC_HP_002 | To verify that the user name is displayed on the homepage or not. | 1.Login with correct credentials | 1.Valid URL<br>2.Test Data | **Username:** test<br>**Password:** correct | User name is displayed on the homepage. |
| | | TC_HP_003 | To verify that the home page is in different browsers. | 1.Open URL in different Browsers | 1.Valid URL<br>2.Test Data | **URL** : www.amazon.in | The user is directed to the Home Page of the application |
| | | TC_HP_004 | To verify that the products are displayed on the home page or not. | 1.Open URL in different Browsers | 1.Valid URL<br>2.Test Data | URL : www.amazon.in | The Home Page displays the Products to the user. |
| | | TC_HP_005 | To verify that the search functionality is available on the home page or not. | 1.Open URL in different Browsers | 1.Valid URL<br>2.Test Data | URL : www.amazon.in | The Seach bar is available to the user on the top of the page. |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TS_Login | To test the login Functionality of the applicaion | TC_L_001 | Enter Valid Username and Valid Password | 1.Enter Valid Username<br>2.Enter Valid Password<br>3.Click Login | 1.Valid URL<br>2.Test Data | Username: test@gmail.com<br>Password: correct | User should be able to succesfully login |
| | | TC_L_002 | Enter Invalid Username and Valid Password | 1.Enter Invalid Username<br>2.Enter Valid Password<br>3.Click Login | 1.Valid URL<br>2.Test Data | Username: wrongtest@gmail.com<br>Password: correct | User should get a prompt for incorrect credentials |
| | | TC_L_003 | Enter Valid Username and Invalid Password | 1.Enter Valid Username<br>2.Enter Valid Password<br>3.Click Login | 1.Valid URL<br>2.Test Data | Username: test@gmail.com<br>Password: wrong | User should get a prompt for incorrect credentials |
| | | TC_L_004 | Enter Invalid Username and Invalid Password | 1.Enter Invalid Username<br>2.Enter Invalid Password<br>3.Click Login | 1.Valid URL<br>2.Test Data | Username: wrongtest@gmail.com<br>Password: wrong | User should get a prompt for incorrect credentials |
| TS_Product Details | To test the Product Details displayed for a selected Product | TC_PD_001 | To verify that the images of products are displayed on the page correctly or not. | 1. Select a Product<br>2.Click on it | 1.Valid URL<br>2. Test Data | Click The Displayed Product | The user should be displayed the images of the selected product |
| | | TC_PD_002 | To verify that the price of the product should be displayed. | 1. Select a Product<br>2.Click on it | 1.Valid URL<br>2. Test Data | Click The Displayed Product | The user should be displayed the Price of the selected |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | product |
| | | TC_PD_003 | To verify that the product specifications should display. | 1. Select a Product<br>2.Click on it | 1.Valid URL<br>2. Test Data | Click The Displayed Product | The user should be displayed the specification of the selected product |
| | | TC_PD_004 | To verify that the product reviews are displayed on the page correctly or not. | 1. Select a Product<br>2.Click on it | 1.Valid URL<br>2. Test Data | Click The Displayed Product | The user was displayed the product reviews of the selected product |
| TS_ShoppingCart | To test the Shopping Cart functionality of the selected Product | TC_SC_001 | To verify that the Add to cart button is clickable or not. | 1.Select a Product<br>2.Click Add to Cart Button | 1.Valid Product<br>2.Test Data | Click The Add to Cart Button | The Button is Clickable |
| | | TC_SC_002 | To verify after clicking Add to cart button, Products are moved to Cart or not. | 1.Select a Product<br>2.Click Add to Cart Button | 1.Valid Product<br>2.Test Data | Click The Add to Cart Button | The Product Gets Added to the Cart |
| | | TC_SC_003 | To verify that the total amount of all the items is displayed or not. | 1.Go to Cart<br>2.Adjust the item Quantity | 1.Valid Product<br>2.Test Data | Adjust the quantity of the item in cart | The total count of product is displayed correctly |
| | | TC_SC_004 | To verify that the tax information is displayed or not. | 1.Go to Cart<br>2.Navigate to the bottom of Page | 1.Valid Product<br>2.Test Data | Click the cart total | The Tax info of the items is displayed |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | TC_SC_005 | To verify that the user is able to see shipping charges as per location | 1.Go to Cart<br>2.Navigate to the bottom of Page | 1.Valid Product<br>2.Test Data | Click the cart total | The Shipping info of the items is displayed |
| TS_Payment | To test the payments functionality of the checked out items | TC_P_001 | To verify that the payment options are displayed on the checkout page or not. | 1.Go to Cart<br>2.Checkout | 1.Valid Product<br>2.Test Data | Click Checkout | All the payment Options are Displayed |
| | | TC_P_002 | To verify the Name on Card, Card Number, Expiry Date, and CVV number are mandatory fields on the Payment Page or not. | 1.Go to Cart<br>2.Checkout<br>3.Click Checkout | 1.Valid Product<br>2.Test Data | Name on Card: **TEST**<br>Card No : **1234 4678 9123 1568**<br>Expiry Date: **01/2029**<br>CVV: **123** | The user is redirected to the Bank Payment Page |
| | | TC_P_003 | To verify that the error message is displayed when the user enters incorrect data in all the mandatory fields on the payment page. | 1.Go to Cart<br>2.Checkout<br>3.Click Checkout | 1.Valid Product<br>2.Test Data | Name on Card: **WrongTEST**<br>Card No : **1234 4678 9123 1568**<br>Expiry Date: **01/2029**<br>CVV: **123** | The user is prompted for incorrect Credentials |
| | | TC_P_004 | To verify that the user is able to place an order successfully. | 1.Go to Cart<br>2.Checkout<br>3.Make Payment | 1.Valid Product<br>2.Test Data | Enter OTP and wait for succesful Payment | The order is placed succesfully |

| **Evaluation Parameter** | **Max. Marks** | **Marks Awarded** |
|---|---|---|
| Observation | 20 | |
| Implementation | 40 | |
| Output | 10 | |
| Viva | 10 | |
| Record | 20 | |
| **Total Marks** | **100** | |

**RESULT**

Thus the design of  the test cases for testing the e-commerce application(amazon.in) was created successfully.

**Exp No:** **Test the e-commerce application and report the defects in it.**
**Date:**

**AIM:**

To test an e-commerce application (amazon.in) and report the defects in it.

**PROCEDURE:**

1. Run the command pip install selenium to install the Selenium library in your Python environment.
2. Download the appropriate WebDriver for your browser (e.g., ChromeDriver for Chrome) and place it in a known location.
3. Set up a function that initializes the WebDriver (e.g., ChromeDriver) and navigates to the desired website (in this case, Amazon).
4. Implement a mechanism to pause the execution in case a Captcha appears, allowing the user to manually resolve it before continuing.
5. Create a login page using flask with an input field for username and password
6. Write a function in selenium that testes the username and password field with various inputs for both the field

**PROGRAM:**

**For the Login Page:**

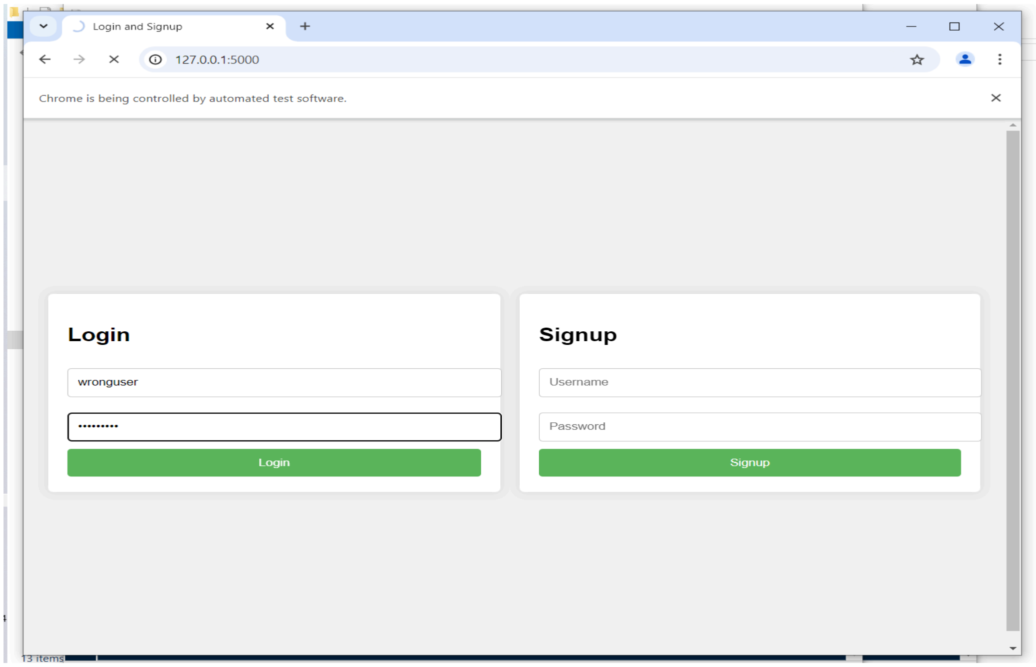
```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time


# Set up the WebDriver (this example uses ChromeDriver)
# If you have ChromeDriver installed in a different location, specify the full path
driver = webdriver.Chrome('F:/Sem VII Labs/chromedriver-win64/chromedriver.exe')


# Open the login page
```

```
driver.get("http://127.0.0.1:5000")

# Test case: Attempt to log in with invalid credentials
def test_invalid_login():
    driver.get("http://127.0.0.1:5000")  # Load the Flask login page

    # Find the username and password fields
    username_field = driver.find_element("name", "username")
    password_field = driver.find_element("name", "password")

    # Input invalid credentials
    username_field.send_keys("wronguser")
    password_field.send_keys("wrongpass")

    # Submit the login form
    password_field.send_keys(Keys.RETURN)

    # Wait for the alert to show up
    time.sleep(2)

    # Check for alert by switching to alert and getting the text
    alert_text = driver.switch_to.alert.text
    if "Invalid username or password" in alert_text:
        print("Test Case Passed: Invalid login warning displayed.")
    else:
        print("Test Case Failed: No invalid login warning displayed.")

    # Close the alert
    driver.switch_to.alert.accept()
```

```
# Test case: Attempt to log in with valid credentials
def test_valid_login():
    driver.get("http://127.0.0.1:5000")  # Load the Flask login page

    # Find the username and password fields
    username_field = driver.find_element("name", "username")
    password_field = driver.find_element("name", "password")

    # Input valid credentials
    username_field.send_keys("admin")
    password_field.send_keys("password123")

    # Submit the login form
    password_field.send_keys(Keys.RETURN)

    # Wait for the alert to show up
    time.sleep(2)
    # Check for alert by switching to alert and getting the text
    alert_text = driver.switch_to.alert.text
    if "Login successful" in alert_text:
        print("Test Case Passed: Valid login successful.")
    else:
        print("Test Case Failed: No success message displayed.")
     # Close the alert
    driver.switch_to.alert.accept()
# Running the test cases
test_invalid_login()  # Test with invalid credentials
```

```
time.sleep(1)        # Short pause between tests
test_valid_login()    # Test with valid credentials
# Close the browser
driver.quit()
```

**OUTPUT:**

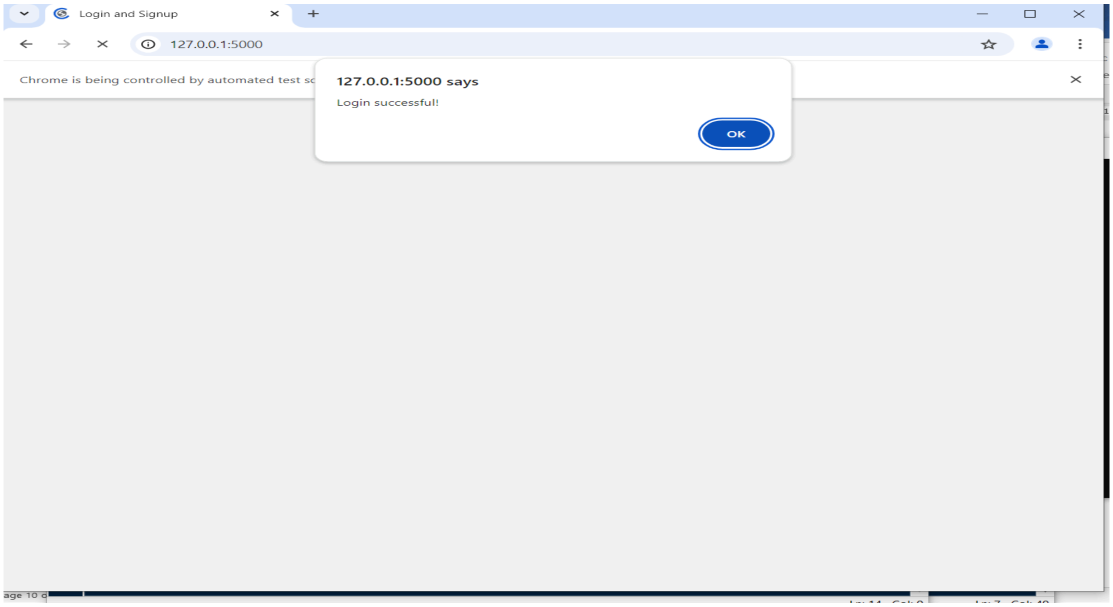


Chrome is being controlled by automated test software.

Login

wronguser

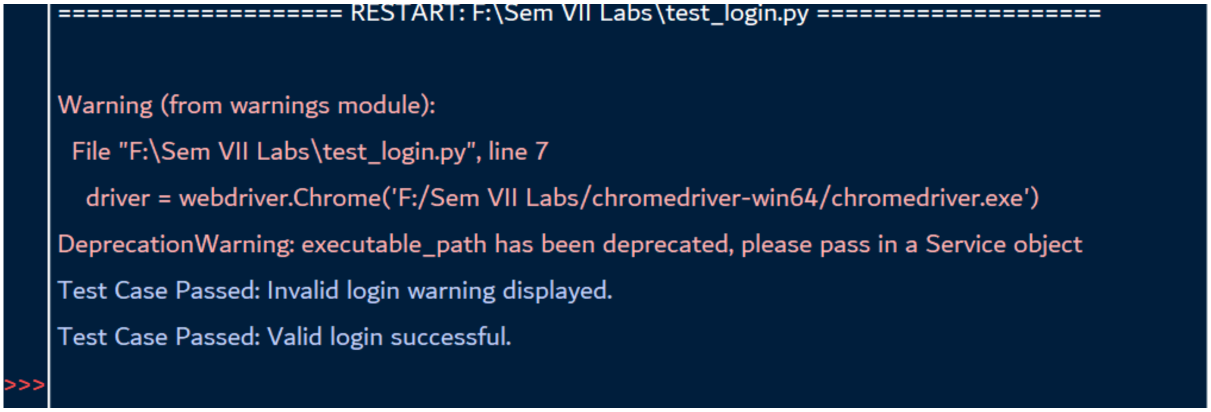
•••••••••



Chrome is being controlled by automated test so

127.0.0.1:5000 says

Login successful!



```
=================== RESTART: F:\Sem VII Labs\test_login.py ===================


Warning (from warnings module):
  File "F:\Sem VII Labs\test_login.py", line 7
    driver = webdriver.Chrome('F:/Sem VII Labs/chromedriver-win64/chromedriver.exe')
DeprecationWarning: executable_path has been deprecated, please pass in a Service object
Test Case Passed: Invalid login warning displayed.
Test Case Passed: Valid login successful.
>>>
```

| **Evaluation Parameter** | **Max. Marks** | **Marks Awarded** |
|---|---|---|
| Observation | 20 | |
| Implementation | 40 | |
| Output | 10 | |
| Viva | 10 | |
| Record | 20 | |
| **Total Marks** | **100** | |

**Result**

Testing an e-commerce application (amazon.in) using Selenium and the defects present in it are reported.

**EX NO:4 DEVELOP TEST PLAN AND DESIGN TEST CASE FOR INVENTORY**

**DATE: CONTROL SYSTEM**

**Aim:**

To develop a test plan and to design test cases for testing an inventory control system to ensure its proper functionality in tracking and managing inventory levels, orders, and stock movements

**Procedure:**

**1. Scope:**

- **User Interface (UI) Testing:** Evaluate the layout, navigation, and accessibility across web and mobile platforms.
- **Functional Testing:** Test core features like adding/removing items, updating inventory levels, managing suppliers, tracking stock, and generating reports.
- **Performance Testing:** Assess the system's response time under high inventory operations and large datasets.
- **Security Testing:** Check for vulnerabilities in data protection, user access controls, and compliance with data security standards.
- **Usability and Compatibility Testing:** Ensure the application is compatible with multiple browsers, devices, and operating systems, and gather feedback for user experience improvements.
- **Integration and Regression Testing:** Test interactions with third-party services (like suppliers, financial systems) and ensure new updates do not disrupt existing functionalities.

**2. Test Objectives:**

- Verify users can add, update, and delete inventory items.
- Confirm real-time inventory tracking and accurate stock levels.
- Test user roles and permissions (admin, manager, staff).
- Validate supplier management and order history.
- Ensure generation of inventory reports (daily, weekly, monthly).
- Test the application's responsiveness on mobile devices.
- Verify data encryption and secure handling of sensitive information.

**3. Test Environment:**

- **Web Browsers:** Chrome, Firefox, Safari, Edge.
- **Mobile Devices:** iOS and Android devices.

- **Testing Framework:** Selenium WebDriver for UI testing.
- **Load Testing:** Apache JMeter for performance assessment.
- **Security Testing:** OWASP ZAP or similar tools for vulnerability checks.

**4. Test Cases:**

- Test cases should cover item creation, stock updates, supplier management, user permissions, and reporting.
- Include both positive and negative scenarios, edge cases, and stress tests.

**5. Test Execution:**

- Automate test cases using Selenium WebDriver for web testing.
- Execute tests across different browsers and mobile devices.
- Perform load testing using tools like JMeter to simulate high data transactions.
- Conduct security testing to identify potential vulnerabilities.

**6. Test Data:**

- Use real and dummy data for items, stock levels, suppliers, and transactions.
- Test accounts with different user roles (admin, staff).

**7. Risks and Assumptions:**

- **Risk:** Changes in the inventory structure may affect automation scripts.
- **Risk:** Mobile app updates could introduce compatibility issues.
- **Assumption:** Test data closely matches real-world usage scenarios.

**8. Deliverables:**

- Detailed test reports summarizing test results, defects found, and performance metrics.
- Documentation of test cases, procedures, and configurations.
- Recommendations for improvements and fixes.

**Test Plan**

To ensure the proper functionality of the inventory control system, we will test its individual features as outlined below:

1. **Inventory Management**
2. **Supplier Management**
3. **User Permissions**
4. **Reporting**
5. **Notifications and Alerts**

**1 – Testing the Inventory Management Module**

- Verify users can add a new item to the inventory.
- Verify users can update stock levels accurately.
- Verify the system alerts when stock reaches a reorder point.
- Verify users can delete items and adjust inventory for returned goods.
- Verify items are categorized properly and searchable.

**2 – Testing the Supplier Management Module**

- Verify users can add new suppliers with correct contact details.
- Verify users can update supplier details.
- Verify the system tracks supplier order history.
- Verify users can generate purchase orders for suppliers.

**3 – Testing User Permissions**

- Verify different access levels for users (admin, manager, staff).
- Verify that only authorized users can add, modify, or delete inventory.
- Verify that restricted users cannot access sensitive data (like supplier financials).
- Verify that users can change their password and manage their profile.

**4 – Testing Reporting Functionality**

- Verify users can generate inventory reports (e.g., current stock levels, out-of-stock items).
- Verify the system generates reports for a specified date range.
- Verify users can download and print reports.
- Verify the accuracy of reports by comparing data with actual inventory.

**5 – Testing Notifications and Alerts**

- Verify the system sends notifications when stock reaches a low level.
- Verify the system alerts the admin when an order is delayed.
- Verify users can configure alert settings (e.g., email, SMS).
- Verify that all notifications are logged for auditing purposes.

**Test Cases Table:**

| Test Category | Test Case ID | Test Case Description | Precondition | Test Steps | Expected Result | Test Type | Priority |
|---|---|---|---|---|---|---|---|
| **Home Page** | TC_HP_001 | Verify that the home page is displayed after a successful login. | Valid login credentials. | 1. Login with valid credentials.<br>2. Check if redirected to the home page. | User is successfully redirected to the home page. | Functional | High |
| | TC_HP_002 | Verify that the user's name is displayed on the home page. | Valid login credentials. | 1. Login with valid credentials.<br>2. Check for the username at the top-right corner. | Username is displayed correctly. | Functional | Medium |
| | TC_HP_003 | Verify that the home page works across multiple browsers (Chrome, Firefox, Safari, Edge). | Valid login credentials. | 1. Open URL in various browsers.<br>2. Verify that the home page loads correctly. | The home page is displayed correctly on all browsers. | Non-functional (Cross-browser) | Medium |
| | TC_HP_004 | Verify that inventory summary (total stock, low stock warnings, etc.) is displayed on the home page. | Valid user login. | 1. Login to the system.<br>2. Check for inventory summary details on the home page. | Inventory summary matches the actual stock data. | Functional | High |
| | TC_HP_005 | Verify search functionality for inventory items | Valid user login. | 1. Use search bar to search for | Search results are | Usability | High |

| Test Category | Test Case ID | Test Case Description | Precondition | Test Steps | Expected Result | Test Type | Priority |
|---|---|---|---|---|---|---|---|
| | | on the home page. | | inventory by name, SKU, or category.<br>2. Verify the displayed results. | relevant to the query. | | |
| **Login & Authentication** | TC_L_001 | Verify login with valid username and password. | Valid username and password. | 1. Enter valid credentials.<br>2. Click on "Login." | User is logged in and redirected to the dashboard. | Functional | High |
| | TC_L_002 | Verify login with an invalid username and valid password. | Invalid username and valid password. | 1. Enter invalid username.<br>2. Click "Login." | Error message is displayed: “Invalid credentials.” | Functional | Medium |
| | TC_L_003 | Verify login with a valid username and invalid password. | Valid username and invalid password. | 1. Enter valid username and invalid password.<br>2. Click "Login." | Error message is displayed: “Incorrect password.” | Functional | Medium |
| | TC_L_004 | Verify password reset functionality. | User with a valid registered email. | 1. Click "Forgot Password."<br>2. Enter registered email.<br>3. Check for reset link email. | Reset link is sent to the user's email address. | Functional | Medium |
| **Inventory Management** | TC_IM_001 | Verify the addition of new items to the inventory. | Admin user rights. | 1. Navigate to "Add New Item."<br>2. Fill in item details. | New item is added to the inventory. | Functional | High |

| **Test Category** | **Test Case ID** | **Test Case Description** | **Precondition** | **Test Steps** | **Expected Result** | **Test Type** | **Priority** |
|---|---|---|---|---|---|---|---|
| | | | | 3. Click "Save." | | | |
| | TC_IM_002 | Verify that existing inventory items can be updated. | Admin user rights. | 1. Select an existing item.<br>2. Modify details.<br>3. Click "Save." | Item details are updated. | Functional | High |
| | TC_IM_003 | Verify that inventory quantities are reduced after a sales order is placed. | Item in stock. | 1. Place a sales order.<br>2. Verify the updated stock levels. | Inventory quantity is reduced accordingly. | Functional | High |
| | TC_IM_004 | Verify that low stock alerts are generated when inventory falls below a threshold. | Item stock below threshold. | 1. Set inventory below threshold.<br>2. Check for low stock alert. | Low stock alert is displayed. | Functional, Regression | High |
| | TC_IM_005 | Verify that items can be removed from the inventory. | Admin user rights. | 1. Select an item.<br>2. Click "Remove" and confirm. | Item is successfully removed. | Functional | Medium |
| **Order Management** | TC_OM_001 | Verify creation of a new sales order. | Valid inventory items. | 1. Navigate to "Create Sales Order."<br>2. Add items.<br>3. Confirm the order. | Sales order is successfully created. | Functional | High |
| | TC_OM_002 | Verify that a purchase order | Valid supplier and items. | 1. Navigate to "Create Purchase | Purchase order is | Functional | High |

| Test Category | Test Case ID | Test Case Description | Precondition | Test Steps | Expected Result | Test Type | Priority |
|---|---|---|---|---|---|---|---|
|  |  | can be generated. |  | Order."<br>2. Select items.<br>3. Confirm the order. | successfully created. |  |  |
|  | TC_OM_003 | Verify that order statuses update correctly (e.g., "Pending," "Shipped"). | Valid order. | 1. Place an order.<br>2. Change order status.<br>3. Verify the updated status. | Status is updated accordingly. | Functional | Medium |
|  | TC_OM_004 | Verify that users can view order history. | Valid user login. | 1. Navigate to "Order History."<br>2. Filter past orders by date or customer. | Order history is displayed correctly. | Usability | Medium |
| **Reports & Analytics** | TC_RA_001 | Verify that users can generate stock level reports. | Valid inventory data. | 1. Navigate to "Reports" and select "Stock Levels."<br>2. Generate report. | Stock level report is generated accurately. | Functional | Medium |
|  | TC_RA_002 | Verify that users can generate sales reports. | Valid sales data. | 1. Navigate to "Reports" and select "Sales Report."<br>2. Choose time frame.<br>3. Generate report. | Sales report is generated accurately. | Functional | Medium |
|  | TC_RA_003 | Verify that users can | Generated report. | 1. Select report. | Report is exported | Functional | Medium |

| Test Category | Test Case ID | Test Case Description | Precondition | Test Steps | Expected Result | Test Type | Priority |
|---|---|---|---|---|---|---|---|
| | | export reports in CSV or PDF formats. | | 2. Export in desired format (CSV or PDF). | successfully in chosen format. | | |

| **Evaluation Parameter** | **Max. Marks** | **Marks Awarded** |
|---|---|---|
| Observation | 20 | |
| Implementation | 40 | |
| Output | 10 | |
| Viva | 10 | |
| Record | 20 | |
| **Total Marks** | **100** | |

**Result:**

Thus, the test plan and the test cases for testing an inventory control system were successfully designed and validated for the Inventory Control System.