UI and API Tests Documentation

Sdet task

Table of Contents

[**UI Tests** 2](#_Toc180919269)

[Automated Tests 2](#_Toc180919270)

[**Smoke Tests** 2](#_Toc180919271)

[Contact Us Smoke Test 2](#_Toc180919272)

[Home Page Smoke Test 2](#_Toc180919273)

[Search Results Page Smoke Test 3](#_Toc180919274)

[**Checklist Tests** 3](#_Toc180919275)

[Contact Us Checklist Test 3](#_Toc180919276)

[Dress Search Checklist Test 6](#_Toc180919277)

[Manual Tests 8](#_Toc180919278)

[**Overview** 8](#_Toc180919279)

[1. Testing Product Navigation 8](#_Toc180919280)

[2. Testing Changing Product Size 8](#_Toc180919281)

[3. Testing Display All Pictures 9](#_Toc180919282)

[4. Changing Model Color 9](#_Toc180919283)

[5. Displaying All Pictures and Changing Color 9](#_Toc180919284)

[6. Find Instock Product 9](#_Toc180919285)

[7. Add to Cart 9](#_Toc180919286)

[8. Product Naming 10](#_Toc180919287)

[9. Product Description 10](#_Toc180919288)

[10. Product Reviews 10](#_Toc180919289)

[11. Page Elements: Visa/MasterCard Icons 10](#_Toc180919290)

[**Bug Reports** 11](#_Toc180919291)

[UI Tests Bugs 11](#_Toc180919292)

[**Contact Us Form** 11](#_Toc180919293)

[1. Submit form without order reference 11](#_Toc180919294)

[2. Submit form without email and message/Submit empty form 11](#_Toc180919295)

[**Dress Search Results** 11](#_Toc180919296)

[1. Search Result Verification: Dresses 11](#_Toc180919297)

[2. Sort Price from Low to High 12](#_Toc180919298)

[3. Sort Price from High to Low 12](#_Toc180919299)

[5. Buying Instock Item 13](#_Toc180919300)

# **UI Tests**

## Automated Tests

### **Smoke Tests**

#### Contact Us Smoke Test

1. Contact Us Visibility Test

|  |  |
| --- | --- |
| Test Case ID | TC #01 |
| Objective | Checks contact us form’s elements’ presence on page navigation |
| Preconditions | Navigation to the contact us page |
| Expected Result | Elements are visible |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Contact Us Basic Functionality Test

|  |  |
| --- | --- |
| Test Case ID | TC #02 |
| Objective | Checks contact us form’s elements’ basic functionalities |
| Preconditions | Navigation to the contact us page |
| Expected Result | Buttons are clickable and input fields accept and retain input |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

#### Home Page Smoke Test

1. Home Page Visibility Test

|  |  |
| --- | --- |
| Test Case ID | TC #03 |
| Objective | Check search bar elements’ visibility on home page |
| Preconditions | Navigation to home page |
| Expected Result | Search field is visible and submit search button is visible |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Home Basic Functionality Test

|  |  |
| --- | --- |
| Test Case ID | TC #04 |
| Objective | Check search bar elements’ visibility on home page |
| Preconditions | Navigation to home page |
| Expected Result | Search input field accepts and retains input, search submit button is clickable |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

#### Search Results Page Smoke Test

1. Search Result Page Visibility Test

|  |  |
| --- | --- |
| Test Case ID | TC #05 |
| Objective | Check search results page elements’ visibility on search submittal |
| Preconditions | Navigation to a hardcoded search result page with results displayed |
| Expected Result | Search title is visible, grid/list view options are visible, sort by drop down list is visible, product list area is visible |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Search Result Page Basic Functionality Test

|  |  |
| --- | --- |
| Test Case ID | TC #06 |
| Objective | Check search results page elements’ basic functionality on search submittal |
| Preconditions | Navigation to a hardcoded search result page with results displayed |
| Expected Result | Grid/list view is selectable, sort by options from the drop-down list are selectable |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

### **Checklist Tests**

#### Contact Us Checklist Test

1. Submit Message without Heading Test

|  |  |
| --- | --- |
| Test Case ID | TC #07 |
| Objective | Check if the form allows submission without selecting a header from the drop down |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission without user selecting a subject header and notifies user of error |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message without Email Test

|  |  |
| --- | --- |
| Test Case ID | TC #08 |
| Objective | Check if the form allows submission without an email address |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission without an email address and notifies user of error |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message without Order Reference – Customer Service

|  |  |
| --- | --- |
| Test Case ID | TC #09 |
| Objective | Check if the form allows submission without order reference |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission without order reference and notifies user of error |
| Actual Result | Form submits |
| Pass/Fail | Fail |
| Remarks | N/A |

1. Submit Message without Message Body

|  |  |
| --- | --- |
| Test Case ID | TC #10 |
| Objective | Check if the form allows submission without a message |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission without a message body and notifies user of error |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message with Invalid Email Format

|  |  |
| --- | --- |
| Test Case ID | TC #11 |
| Objective | Check if the form allows submission with an invalid email address |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission with an invalid email address and notifies user of error |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message with PDF File

|  |  |
| --- | --- |
| Test Case ID | TC #12 |
| Objective | Check if the form allows submission with a PDF file format |
| Preconditions | Navigation to form page |
| Expected Result | Form allows submission with PDF files |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message with JPG File

|  |  |
| --- | --- |
| Test Case ID | TC #13 |
| Objective | Check if the form allows submission with a JPG file format |
| Preconditions | Navigation to form page |
| Expected Result | Form allows submission with JPG files |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message with JSON File

|  |  |
| --- | --- |
| Test Case ID | TC #14 |
| Objective | Check if the form allows submission with a JSON file format |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission with JSON files and notifies user of the error |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Submit Message without Email and without Message Test

|  |  |
| --- | --- |
| Test Case ID | TC #15 |
| Objective | Check if the form behavior when two necessary fields are missing |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission without email and without message body, and notifies user of the errors |
| Actual Result | Form does not allow submission, however, it only notifies user of one error |
| Pass/Fail | Fail |
| Remarks | N/A |

1. Submit Empty Form

|  |  |
| --- | --- |
| Test Case ID | TC #16 |
| Objective | Check if the form behavior when it is empty |
| Preconditions | Navigation to form page |
| Expected Result | Form does not allow submission with empty fields, and notifies user of the errors |
| Actual Result | Form does not allow submission; however, it notifies user of only one error about one missing field |
| Pass/Fail | Fail |
| Remarks | N/A |

#### Dress Search Checklist Test

1. Search Result Verification: Matches

|  |  |
| --- | --- |
| Test Case ID | TC #17 |
| Objective | Checks if the search yields any results |
| Preconditions | Navigation to home page, searching for dress and submitting |
| Expected Result | Search should yield matches |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | used .assert() instead of .verify() in case no test matches are found, then the test suite fails and stops execution |

1. Search Result Verification: All Dresses

|  |  |
| --- | --- |
| Test Case ID | TC #18 |
| Objective | Checks if the search yields anything other than dresses |
| Preconditions | Navigation to home page, searching for dress and submitting |
| Expected Result | Search should only yield result for dresses |
| Actual Result | Search yields results for other items that are not dresses |
| Pass/Fail | Fail |
| Remarks | N/A |

1. Sort Price Low to High

|  |  |
| --- | --- |
| Test Case ID | TC #19 |
| Objective | Checks result items prices after sorting them from low to high to verify sort |
| Preconditions | Navigation to home page, searching for dress and submitting, sorting prices from low to high |
| Expected Result | Results should be sorted according to price from low to high |
| Actual Result | Not all products were in their correct sorted place |
| Pass/Fail | Fail |
| Remarks | N/A |

1. Sort Price High to Low

|  |  |
| --- | --- |
| Test Case ID | TC #20 |
| Objective | Checks result items prices after sorting them from high to low to verify sort |
| Preconditions | Navigation to home page, searching for dress and submitting, sorting prices from high to low |
| Expected Result | Results should be sorted according to price from high to low |
| Actual Result | Not all products were in their correct sorted place |
| Pass/Fail | Fail |
| Remarks | N/A |

1. Sort Products by Name in Ascending Order

|  |  |
| --- | --- |
| Test Case ID | TC #21 |
| Objective | Checks result items names after sorting them in ascending order |
| Preconditions | Navigation to home page, searching for dress and submitting, sorting names in ascending order |
| Expected Result | Results should be sorted according to name in ascending order |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Sort Products by Name in Descending Order

|  |  |
| --- | --- |
| Test Case ID | TC #22 |
| Objective | Checks result items names after sorting them in descending order |
| Preconditions | Navigation to home page, searching for dress and submitting, sorting names in descending order |
| Expected Result | Results should be sorted according to name in descending order |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

1. Buy In Stock/Out of Stock Items

|  |  |
| --- | --- |
| Test Case ID | TC #23 |
| Objective | Checks if in stock items are buyable, and out of stock items are not |
| Preconditions | Navigation to home page, searching for dress and submitting, sorting items by stock availability |
| Expected Result | In stock items’ add to cart button is enabled, while out of stock items’ add to cart button is disabled |
| Actual Result | Could not add in stock items to cart |
| Pass/Fail | Fail |
| Remarks | N/A |

1. List View

|  |  |
| --- | --- |
| Test Case ID | TC #24 |
| Objective | Checks if view changes on selecting list view option |
| Preconditions | Navigation to home page, searching for dress and submitting, clicking on list view |
| Expected Result | Items view changes from grid view to list view |
| Actual Result | As expected |
| Pass/Fail | Pass |
| Remarks | N/A |

## Manual Tests

### **Overview**

* Webpage: My Store <http://automationpractice.multiformis.com/index.php> -- Printed Summer Dress
* Purpose: Test and verify site’s design and user functionalities are intuitive, easy-to-follow and accessible.

### Testing Product Navigation

* Purpose: Test and verify ease of navigation to products
* Steps:
  + Start on the home page of My Store
  + Hover over Women 🡪 Dresses or Click on Dresses
  + Click on the required dress model
* Results: Failed
* Comments: Navigation to dresses is redundantly implemented under the women category and under dress category. Website is only selling women clothes, no need to have it listed.

### Testing Changing Product Size

* Purpose: Test and verify ease of selection of sizes and having the necessary size information
* Steps:
  + Choose a size from the drop down list of sizes
  + Verify size selection
* Results: Failed
* Comments: Sizes selected are reflected on the website, however, there are no size charts on the website to inform the user what the sizes correspond to.

### Testing Display All Pictures

* Purpose: Verify that display all pictures functionality works as intended and displays all model images
* Steps:
  + - Click on display all pictures under product photo
    - Verify that all color model pictures are now shown
  + Results: Passed
  + Comments: N/A

### Changing Model Color

* Purpose: Verify that the chosen color is highlighted and reflected in the model photo
* Steps:
  + Choose a color from the available colors
  + Check that color is highlighted among the options
  + Check model photo to see the color reflected on product
* Results: Passed
* Comments: N/A

### Displaying All Pictures and Changing Color

* Purpose: Verify that choosing a product option does not reset other options
* Steps:
  + Click on display all pictures
  + Check that all pictures are displayed
  + Change product color
  + Check that product color is changed
  + Check that all pictures are still displayed
* Results: Failed
* Comments: Changing product color resets the display all pictures function

### Find Instock Product

* Purpose: Verify ease of navigation to instock products
* Steps:
  + Go to a product page that is not out of stock
  + Look for options to view in stock products
* Results: Failed
* Comments: Product page does not clearly display which products are in stock, instead the user is required to go through colors and sizes to find which are in stock.

### Add to Cart

* Purpose: Verify that adding a product to cart reflects in the cart with the chosen size, model and quantity
* Steps:
  + Find an in stock product
  + Choose color, size and quantity
  + Click add to cart
  + Check user’s cart
  + Verify model, color, size and quantity
* Results: Passed
* Comments: N/A

### Product Naming

* Purpose: Ensure quality in product titles
* Steps:
  + Check if product names are unique under Dress category
* Results: Fail
* Comments: Product names are repeated and nondescriptive

### Product Description

* Purpose: Ensure quality and verbosity in product descriptions for keeping users informed
* Steps:
  + Navigate to a product’s page
  + Check product details
* Results: Fail
* Comments: Products do not have lengthy descriptions. Their descriptions omit much of the necessary details for potential customers including fabric composition ratios and wash instructions. Product pictures are only one per model and from one view only.

### Product Reviews

* Purpose: Test that users can write and view reviews for a product ensuring transparency.
* Steps:
  + Navigate to a product’s page
  + Check for reviews
* Results: Fail
* Comments: Products do not have a customer reviews section.

### Page Elements: Visa/MasterCard Icons

* Purpose: Explore the functionality of the added visa/mastercard icons
* Steps:
  + Navigate to product page
  + Check if the icons are clickable
* Results: Fail
* Comments: Items are not clickable. They serve no purpose on the product page and could make more sense being included in the checkout page.

# **Bug Reports**

## UI Tests Bugs

### Contact Us Form

#### Submit form without order reference

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #01 // TC #09 |
| Summary | Error message missing when order reference field is blank |
| Description | When submitting the contact form without an order reference, the form submission goes through without displaying an error message. |
| Steps to Reproduce | 1. Navigate to contact us page 2. Fill out name, email, message 3. Choose subject header to customer service 4. Leave order reference empty 5. Submit |
| Expected Result | Error – Cannot submit form without order reference |
| Actual Result | Submission Success |
| Environment | Chrome 130  Windows 10 |
| Priority | High |
| Severity | Medium |

#### Submit form without email and message/Submit empty form

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #02 // TC #15 & TC #16 |
| Summary | Error message incomplete errors when more than one parameter is missing |
| Description | When submitting the contact form with more than one required empty field, the error message displayed only counts one error and only lists one error. |
| Steps to Reproduce | 1. Navigate to contact us page 2. Leave form empty 3. Submit |
| Expected Result | 4 Errors – Cannot submit form with invalid email, without message, without header, without order reference |
| Actual Result | 1 Error – Invalid Email |
| Environment | Chrome 130  Windows 10 |
| Priority | Medium |
| Severity | High |

### **Dress Search Results**

#### Search Result Verification: Dresses

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #03 // TC #18 |
| Summary | Dress search results are not all dresses |
| Description | When searching for dresses, the products listed are not all dresses but instead a t-shirt and a blouse are shown. |
| Steps to Reproduce | 1. Navigate to store’s homepage 2. Search for Dress 3. Go through results to check if all contain the word ‘Dress’ |
| Expected Result | Only results with the word ‘Dress’ should be shown |
| Actual Result | A t-shirt and a blouse are also included in the search results |
| Environment | Chrome 130  Windows 10 |
| Priority | Medium |
| Severity | Medium |

#### Sort Price from Low to High

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #04 // TC #19 |
| Summary | Sorting search results by price from low to high does not apply sort to all products |
| Description | When choosing sort to be by price from low to high, two items are not in their right place due to them being sorted based on the original price not the discounted price. |
| Steps to Reproduce | 1. Navigate to store’s homepage 2. Search for Dress 3. Sort by price from low to high 4. Check item prices to be increasing in ascending order |
| Expected Result | Items are sorted correctly according to their prices from low to high |
| Actual Result | Two items are not in their correct sorted places |
| Environment | Chrome 130  Windows 10 |
| Priority | High |
| Severity | Medium |

#### Sort Price from High to Low

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #05 // TC #20 |
| Summary | Sorting search results by price from high to low does not apply sort to all products |
| Description | When choosing sort to be by price from high to low, two items are not in their right place due to them being sorted based on the original price not the discounted price. |
| Steps to Reproduce | 1. Navigate to store’s homepage 2. Search for Dress 3. Sort by price from high to low 4. Check item prices to be increasing in ascending order |
| Expected Result | Items are sorted correctly according to their prices from high to low |
| Actual Result | Two items are not in their correct sorted places |
| Environment | Chrome 130  Windows 10 |
| Priority | High |
| Severity | Medium |

#### Buying Instock Item

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #06 // TC #23 |
| Summary | In stock items shown in search results page do not have add to cart option enabled |
| Description | When choosing sort to be by in stock, the product item that is not out of stock does not have add to cart option enabled due to the sort not sorting through the model to show the in stock color. |
| Steps to Reproduce | 1. Navigate to store’s homepage 2. Search for Dress 3. Sort by in stock 4. Check item that is not out of stock 5. Click add to cart |
| Expected Result | Instock items can be added to cart after sorting by instock |
| Actual Result | Instock item cannot be added to cart on results page |
| Environment | Chrome 130  Windows 10 |
| Priority | High |
| Severity | High |

## API Tests Bugs

### Create User Tests

#### Creates User with Required Fields

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #01 // TC #01 |
| Summary | Creating user does not return user token |
| Description | When creating a user with valid email, username and password, the sign up page succeeds in creating the user. However, it does not return a user token. |
| Steps to Reproduce | 1. Send a POST request to /api/v1/users with all required fields in the body. 2. Check the response body. |
| Expected Result | User is created successfully and user token is received |
| Actual Result | User is created successfully with no user token |
| Environment | Platform: Localhost (Port 8000)  Testing Framework: Jest With Supertest |
| Priority | High |
| Severity | Medium |

#### Creates User with Empty Username

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #02 // TC #03 |
| Summary | Sign up accepts user creation with an empty username field |
| Description | When creating user with an empty username, it creates it and outputs a success message. |
| Steps to Reproduce | 1. Send a POST request to /api/v1/users with username empty, but with a valid email and password. 2. Check the response body. |
| Expected Result | Error on user creation—Cannot be created with no username. |
| Actual Result | User is created successfully. |
| Environment | Platform: Localhost (Port 8000)  Testing Framework: Jest With Supertest |
| Priority | High |
| Severity | High |

#### Creates User with Invalid Email

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #03 // TC #04 |
| Summary | Sign up accepts user creation with an invalid email format |
| Description | When creating a user with a valid username but an invalid email format (i.e. no @domain, no .com), the user is still created. |
| Steps to Reproduce | 1. Send a POST request to /api/v1/users with a username, an invalid email format and a password. 2. Check the response body. |
| Expected Result | Error on user creation—Invalid Email |
| Actual Result | User is created successfully. |
| Environment | Platform: Localhost (Port 8000)  Testing Framework: Jest With Supertest |
| Priority | High |
| Severity | High |

### Get User by Token Tests

#### Does not Get User with Valid Token

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #04 // TC #09 |
| Summary | Get user by valid token returns error instead of user details |
| Description | When attempting to get user by a valid token, it is expected to receive user details: username, email, password and image. Instead, an error message is returned. This is likely due to the server not generating tokens on user creation in the first place. |
| Steps to Reproduce | 1. Send a GET request to /api/v1/users with a valid authorization token in the headers. 2. Check the response body. |
| Expected Result | Returns user with user info |
| Actual Result | 403 Error |
| Environment | Platform: Localhost (Port 8000)  Testing Framework: Jest With Supertest |
| Priority | High |
| Severity | High |

### Patch User by Token Tests

#### Does not Patch User with Valid Token

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #05 // TC #011 |
| Summary | Patch user by valid token returns error instead of success message |
| Description | When sending a PATCH request to update user details with a valid token, an error response is returned instead of a success message. This is likely due to the server not generating tokens on user creation in the first place. |
| Steps to Reproduce | 1. Send a PATCH request to /api/v1/users with a valid authorization token in the headers. 2. Include valid data in the request body for the user update. 3. Check the response body. |
| Expected Result | A success message should be returned indicating that the user details were successfully updated |
| Actual Result | 403 Error |
| Environment | Platform: Localhost (Port 8000)  Testing Framework: Jest With Supertest |
| Priority | High |
| Severity | High |

### Delete User by Token Tests

#### Does not Delete User by Valid Token

|  |  |
| --- | --- |
| Bug Ticket ID/Testcase ID | Bug #06 // TC #014 |
| Summary | Delete user by valid token returns error instead of success message |
| Description | When sending a DELETE request to update user details with a valid token, an error response is returned instead of a success message. This is likely due to the server not generating tokens on user creation in the first place. |
| Steps to Reproduce | 1. Send a DELETE request to /api/v1/users with a valid authorization token in the headers. 2. Check the response body. |
| Expected Result | A success message should be returned indicating that the user has been deleted successfully |
| Actual Result | 403 Error |
| Environment | Platform: Localhost (Port 8000)  Testing Framework: Jest With Supertest |
| Priority | High |
| Severity | High |