**PART 3 QA CHALLENGE: Andrés Vergara**

CHICKS GOLD

# TEST EXECUTION AND REPORTING

**1. User Authentication**

**Test cases:**

**1.1. Verify user login with valid credentials**

- **Objective:** Ensure users can log in successfully with correct credentials.
- **Steps:**
    1. Submit a valid **username and password**.
    2. Verify the response contains a valid authentication token.
    3. Check that the user is redirected to the dashboard or homepage.

**Result: The testing was carried out successfully, and the expected result was achieved**
https://order-products-images.s3.amazonaws.com/1d6f800f-cec9-46df-9491-4ae3f36b72db

**1.2. Verify error message for invalid credentials**

- **Objective: Ensure proper error message is returned when login credentials are invalid.**
- **Steps:**
    1. **Submit an invalid username and/or password.**
    2. **Verify the response includes an appropriate error message (e.g., "Invalid username or password").**

**Result: The testing was carried out successfully, and the expected result was achieved**
https://order-products-images.s3.amazonaws.com/dd2e445e-5e51-4547-b696-d64c42b0614d

**1.3. Test token generation and expiration**

- **Objective: Verify token generation and expiration mechanisms work as expected.**
- **Steps:**
    1. Log in with valid credentials.
    2. Verify token generation.
    3. Wait for the specified token expiration time.

**Result: The result obtained showed the presence of a fixed token, such as the sample token, which means that upon refreshing the session, it will not persist. This is due to the backend lacking a dynamic token. As QAs, we can use this information to propose an improvement by implementing a dynamic token in the project.**

**PART 3 QA CHALLENGE: Andrés Vergara**
https://order-products-images.s3.amazonaws.com/7b9c96d2-7ac1-4173-ba8a-83d34a6bea8f

**1.4. Verify login with multiple concurrent sessions**

- **Objective: Ensure that a user can be logged in from multiple devices or sessions simultaneously without issues.**
- **Steps:**
    1. Log in with valid credentials on one device.
    2. Log in with the same credentials on a different device or browser.
    3. Verify that both sessions remain active and independent.

**Result: The testing was carried out successfully, and the expected result was achieved**
https://order-products-images.s3.amazonaws.com/a012dc60-3b96-481f-a2ac-a5e018951e37

**1.5. Verify account lockout after multiple failed login attempts**

- **Objective: Ensure that the system locks the account after a set number of failed login attempts.**
- **Steps:**
    1. Attempt to log in with invalid credentials multiple times (e.g., 5 failed attempts).
    2. Verify that the account is temporarily locked and a proper message is displayed

**Result: The result showed that there is no account lockout after multiple attempts, which opens the opportunity to propose the creation of this feature in the future for the project's continuous improvement.**
https://order-products-images.s3.amazonaws.com/2eff74c2-1bff-4d89-b61f-470db2d6fa3f

**1.6. Verify login with empty username**

- **Objective:** Ensure that the system rejects a login attempt when the username field is empty.
- **Steps:**
    1. Leave the username field empty.
    2. Enter a valid password.
    3. Attempt to log in.

**Result:** The result is correct, with a 401 error indicating invalid credentials.
https://order-products-images.s3.amazonaws.com/e7d151ff-0e90-410d-922f-b88dd4091ed2

**PART 3 QA CHALLENGE: Andrés Vergara**

CHICKS GOLD

### 1.7. Verify login with incorrect password

**Objective: Ensure that the system rejects a login attempt when the password is incorrect.**

**Steps:**

1. Enter a valid username.
2. Enter an incorrect password.
3. Attempt to log in.

**Result:** The result is correct, with a 401 error indicating invalid credentials. The obtained result was correct as the login did not proceed, but the displayed message remains "Logging failed." As an improvement, it could be updated to "Username not found."
https://order-products-images.s3.amazonaws.com/6f5cc793-7201-4b38-9417-10a33e3c1961

### 1.8. Verify login with incorrect username

- **Objective:** Ensure that the system rejects a login attempt when the username is incorrect.
- **Steps:**
    1. Enter an incorrect username.
    2. Enter the correct password.
    3. Attempt to log in.

**Result:** The obtained result was correct as the login did not proceed, but the displayed message remains "Logging failed." As an improvement, it could be updated to "Incorrect password."
https://order-products-images.s3.amazonaws.com/845298d5-f7c6-4f75-9ab7-435655ba798a

### 1.9. Verify login with both username and password empty

- **Objective:** Ensure that the system rejects a login attempt when both the username and password fields are empty.
- **Steps:**
    1. Leave both the username and password fields empty.
    2. Attempt to log in.

**Result:** The obtained result was correct as the login did not proceed but the displayed message remains "Logging failed." As an improvement, it could be updated to "Username and password are required"
https://order-products-images.s3.amazonaws.com/180e940a-9b34-4541-8a90-9b346bf94029

**PART 3 QA CHALLENGE: Andrés Vergara**

CHICKS GOLD

**1.10. Verify login with leading or trailing spaces in username**

- **Objective:** Ensure that the system handles leading or trailing spaces in the username correctly.
- **Steps:**
    1. Enter a valid username with leading or trailing spaces (e.g., " user " or " user").
    2. Enter a valid password.
    3. Attempt to log in.

**Result:** As an improvement, it could trim the spaces and log in successfully if the credentials are correct.
https://order-products-images.s3.amazonaws.com/d21542df-66bd-48f2-bbf3-8a78992a0f84

**1.11. Verify login with leading or trailing spaces in password**

- **Objective:** Ensure that the system handles leading or trailing spaces in the password correctly.
- **Steps:**
    1. Enter a valid username.
    2. Enter a valid password with leading or trailing spaces (" p assword " or " password").
    3. Attempt to log in.

**Result:** As an improvement, it could trim the spaces and log in successfully if the credentials are correct.
https://order-products-images.s3.amazonaws.com/bd35dbd6-e164-4366-b6a3-7b89ba9beedb

**1.12. Verify login with special characters in username and password**

- **Objective:** Ensure that the system supports special characters in both usernames and passwords.
- **Steps:**
    1. Enter a username and password containing special characters (!@#$%^&*())
    2. Log in using these credentials.

**Result:** The testing was carried out successfully
https://order-products-images.s3.amazonaws.com/71ea515c-df30-4a5d-8561-2d494ad746d5

**2. Product Management**

**2.1.** Create, read, update, and delete (CRUD) operations for products

- **Objective: Validate CRUD operations for managing product data.**
- **Steps:**

**PART 3 QA CHALLENGE: Andrés Vergara**

1. Create a new product with valid data.
2. Retrieve the product details.
3. Update the product information.
4. Delete the product.

**Result:**

**Creation**
https://order-products-images.s3.amazonaws.com/72834d4e-19c0-4705-9df1-bc6f50bdceb7

**Update of the product**
https://order-products-images.s3.amazonaws.com/5c0c8228-5786-4ec7-aaa9-6f42d2c6be57

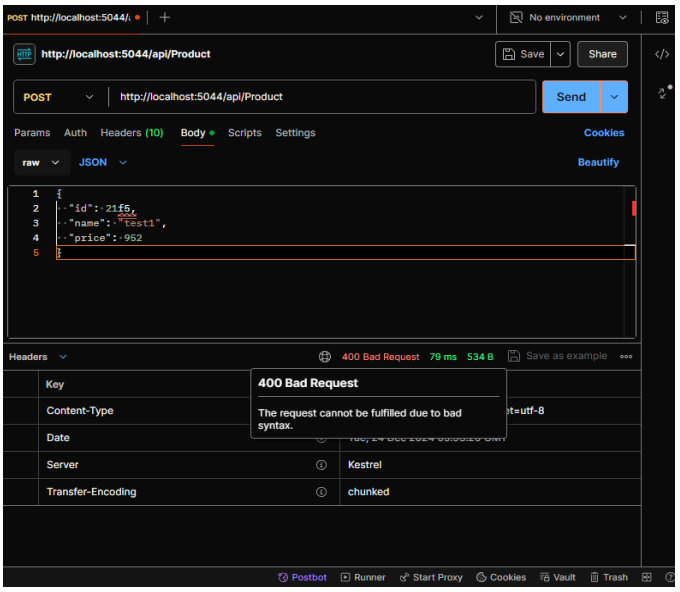
**Product deleted**
https://order-products-images.s3.amazonaws.com/16d44b4c-ca65-431e-81fb-24b1eb835d54

**The test result was carried out using Postman and was successful. A product was correctly displayed when added, updated successfully, and subsequently deleted.**

**2.2. Verify error handling for invalid product data**

- **Objective: Ensure the system handles invalid product data properly.**
- **Steps:**
    1. Attempt to create or update a product with invalid data (e.g., missing required fields, incorrect format).
    2. Verify the response includes an appropriate error message.

**Result: Test was successful. Status code 400 invalid data. In this case bad syntax**

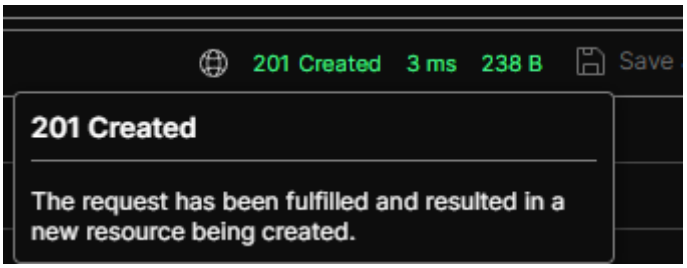


**PART 3 QA CHALLENGE: Andrés Vergara**

**2.3. Verify product listing display**

- **Objective: Ensure that all products are displayed correctly in the listing/dashboard.**
- **Steps:**
    1. Create multiple products.
    2. Access the product listing/dashboard**.**

**Result: Multiple products created - Test was successful**
https://order-products-images.s3.amazonaws.com/ed18f78e-4dae-42b1-b30e-ac777d699e26




**2.4. Verify product update with valid data**

- **Objective: Ensure that an existing product is successfully updated with valid data.**
- **Steps:**
    1. Select a product from the listing/dashboard.
    2. Edit its details (e.g., update price or stock).
    3. Save the changes.

**Result: The product among multiple products was successfully updated. The testing was successful.**
https://order-products-images.s3.amazonaws.com/1a46ee1e-f6ed-46d8-9a01-63107e254bfc

**2.5. Verify product deletion functionality**

- **Objective: Ensure that products can be successfully deleted.**
- **Steps:**
    1. Select a product from the listing/dashboard.
    2. Perform the delete action.
    3. Confirm the deletion.

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/e9186911-5ef2-4b6d-a086-8189e55c3e6e

**2.6. Test user dashboard data display and refresh**

- **Objective: Ensure that the user dashboard displays the correct data and refreshes properly.**
- **Steps:**
    1. Log in and access the user dashboard.
    2. Verify that the displayed data is correct.
    3. Check for data refresh upon interaction.

**Result: Accurate data is displayed "Welcome to the user dashboard!" message is displayed**
https://order-products-images.s3.amazonaws.com/4ff26fcf-ef10-4978-a739-b3c78074dcce

**2.7. Verify successful order creation with valid details**

- **Objective: Ensure an order is created successfully when all required details are valid.**
- **Steps:**
    1. Add valid product(s) to the cart.
    2. Provide valid user and payment information.
    3. Submit the order.

**Result: The system confirms order creation and generates a unique order ID. The order appears in the user's order history.**
https://order-products-images.s3.amazonaws.com/1550400a-ef95-4508-9d7e-2f63ecd2bff4

**2.8. Verify Correct Display of Orders in Order History**

- **Objective: Ensure all orders are displayed accurately in the user's order history, including details like order ID, product names, quantities, prices, status, and timestamps.**
- **Steps:**
    1. Create multiple orders with varying details (e.g., different products, quantities, and statuses).
    2. Navigate to the user's order history page.
    3. Verify that each order is displayed with the correct details
    4. Check displayed data

**Result: All orders are displayed accurately**
https://order-products-images.s3.amazonaws.com/e15bd140-6f52-46e9-a0ff-22a48bfe3ffa

## Performance between modules - FRONTEND

**3.0. Measure Login Page Load Time**

- **Objective: Verify that the login page loads within acceptable time limits.**
- **Steps:**

**PART 3 QA CHALLENGE: Andrés Vergara**

1. Measure the time taken to fully load the login page under normal network conditions.
2. Repeat the test under slow (3G) and high-latency networks.

**Result: The testing was successful.The login page should loads within 113ms on high-speed networks and within 4 seconds on slow networks.**
https://order-products-images.s3.amazonaws.com/3ca20b81-cbd2-4f81-bf14-d6c64c3dadca

**3.1. Test Product Listing Page Load Time**

- **Objective: Ensure the product listing page loads efficiently under normal conditions.**
- **Steps:**
    1. Navigate to the product listing page.
    2. Measure the time taken to load the page fully, including images, product details, and filters.

**Result: The product listing page - In fast connections, it takes no more than 1 second, but in low connections, it keeps retrying and reconnecting, and it would never finish loading due to automatic retries**
https://order-products-images.s3.amazonaws.com/e9ecb9da-2ce4-4f42-8cfd-869c9e3e289b

**3.2. Verify Performance of Time Dashboard**

- **Objective: Assess the dashboard's ability to handle data without performance degradation.**
- **Steps:**
    1. Log in to the dashboard.
    2. Check the performance of  the data displayed on the dashboard.

**Result: Here, the same case occurs: in fast connections, the page loads in less than 200ms, but in slow connections, it keeps reconnecting, which could affect performance.**
https://order-products-images.s3.amazonaws.com/ea707deb-95fc-4012-b521-d78be8a48598

**3.3. Measure Redirection Speed from Login to Dashboard**

- **Objective: Verify the time taken to go from the login page to the dashboard**
- **Steps:**
    1. Log in with valid credentials.
    2. Measure the time taken to navigate from the login page to the fully rendered dashboard.

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/5d976b80-a63e-4a0e-83fa-cbcfe35e993a

### 3.4. Evaluate Redirection Speed from Dashboard to Product Listing

- **Objective:** Ensure quick navigation from the dashboard to the product listing page.
- **Steps:**
    1. Log in and access the dashboard.
    2. Click on the "Products" section to navigate to the product listing page.

**Result:The testing was successful.**
https://order-products-images.s3.amazonaws.com/a1849ca5-1e9a-45bc-9a94-9324b269b08f

### 3.5. Test Redirection Speed from Product Listing to Order Creation

- **Objective:** Verify smooth redirection from the product listing page to the order creation workflow.
- **Steps:**
    1. Navigate to the product listing page.
    2. Measure the time taken to the order page.
    3. Test under varying network conditions.

**Result:The testing was successful.**
https://order-products-images.s3.amazonaws.com/8d922055-1878-4f5a-ba59-e218604256d2

### 3.6. Evaluate Redirection Speed Between Orders in the Dashboard

- **Objective:** Verify navigation performance when switching between different order views.
- **Steps:**
    1. Log in and access the dashboard.
    2. Switch between tabs or sections.
    3. Measure the time taken for redirection and data loading.

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/ec2d90f4-6582-47bf-bb1d-f1b24deb060d

### 3.7. Test Redirection from Dashboard to Login

- **Objective:** Ensure efficient redirection when logging out and returning to the login page.
- **Steps:**
    1. Log in to the dashboard.
    2. Measure the time taken to redirect to the login page.
    3. Verify no delay under varying network speeds.

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/38e9093b-0a6f-4bcb-82a0-0df1ffa02abe

**PART 3 QA CHALLENGE: Andrés Vergara**

CHICKS GOLD

### 3.8. Validate Data Refresh Efficiency on Dashboard

- **Objective:** Ensure dashboard data refreshes efficiently when triggered manually.
- **Steps:**
    1. Load the dashboard.
    2. Trigger a manual data refresh.
    3. Measure the time taken for data to update.

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/c4945f99-c490-4ed0-9a93-516261ef9c68

### 3.9. Verify Login Page Load Time

- **Objective:** Ensure the login page loads within acceptable time limits.
- **Steps:**
    1. Navigate to the login page.
    2. Measure the page load time under different network conditions

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/4429c320-a77a-43c9-b35a-2bd5790c3b7e

### 3.10.Check UI Responsiveness for Different Screen Sizes

- **Objective:** Verify UI responsiveness on devices with different resolutions.
- **Steps:**
    1. Test the application on desktop and mobile devices.
    2. Ensure layouts adjust dynamically without performance degradation.

**Result: The testing was successful.**
https://order-products-images.s3.amazonaws.com/fafe9930-339a-4a0b-8282-abad2ec4a67b

https://order-products-images.s3.amazonaws.com/24748490-e645-42b9-88b0-b7794d8433bf

**FINDINGS**

**Test Case 1.3: Test Token Generation and Expiration**

- **Severity Level: High**
- **Current Result:**
  **The authentication token generated during login is not dynamic. This could lead to the user being logged out each time the page is refreshed, resulting in an unstable user experience. This issue is critical as it impacts user navigation and session continuity right from the start of the login process.**
  https://order-products-images.s3.amazonaws.com/7b9c96d2-7ac1-4173-ba8a-83d34a6bea8f

**PART 3 QA CHALLENGE: Andrés Vergara**

- **Expected Result:**
  **The authentication token should be dynamic, with proper expiration handling. The user's session should persist even after refreshing the page, ensuring a stable navigation experience without interruptions.**

**Test Case 1.5: Verify Account Lockout After Multiple Failed Login Attempts**

- **Severity Level: Critical**
- **Current Findings:**
  **After multiple failed login attempts, the system does not lock the account as expected. This could expose the application to security vulnerabilities, allowing brute-force attacks.**
  https://order-products-images.s3.amazonaws.com/2eff74c2-1bff-4d89-b61f-470db2d6fa3f
- **Expected Result:**
  **After a defined number of consecutive failed login attempts, the system should lock the account and display a proper message indicating the lockout. The account should remain locked for a predefined period or until the user resets the password.**

**Test Case 1.7: Verify Login with Incorrect Password**

- **Severity Level: Moderate**
- **Current Findings:**
  **The system correctly prevents login when an incorrect password is entered, but the displayed error message remains as "Logging failed." This message is not informative enough and could confuse users.**
  https://order-products-images.s3.amazonaws.com/6f5cc793-7201-4b38-9417-10a33e3c1961
- **Expected Result:**
  **The error message should be updated to "Username not found" or "Incorrect password," providing clearer feedback to the user about the specific issue.**

**Test Case 1.8: Verify Login with Incorrect Username**

- **Severity Level: Moderate**
- **Current Findings:**
  **The system correctly prevents login when an incorrect username is entered, but the displayed error message remains as "Logging failed." Similar to the previous issue, this message is not specific enough.**
  https://order-products-images.s3.amazonaws.com/845298d5-f7c6-4f75-9ab7-435655ba798a
- **Expected Result:**
  **The error message should be updated to "Incorrect username" to give the user clearer information about the issue.**

**Test Case 1.9: Verify Login with Both Username and Password Empty**

**PART 3 QA CHALLENGE: Andrés Vergara**

- **Severity Level: Moderate**
- **Current Findings:**
  **The login process correctly prevents login when both fields are empty, but the error message remains as "Logging failed." A more specific message would improve the user experience.**
  https://order-products-images.s3.amazonaws.com/180e940a-9b34-4541-8a90-9b346bf94029
- **Expected Result:**
  **The error message should be updated to "Username and password are required," providing more specific feedback to the user.**

**Test Case 1.10: Verify Login with Leading or Trailing Spaces in Username**

- **Severity Level: Low**
- **Current Findings:**
  **The system does not allow login if the username contains leading or trailing spaces. While this behavior is correct, it can cause user frustration if not handled properly.**
  https://order-products-images.s3.amazonaws.com/d21542df-66bd-48f2-bbf3-8a78992a0f84
- **Expected Result:**
  **The system should automatically trim leading and trailing spaces from the username and allow login if the credentials are correct.**

**Test Case 1.11: Verify Login with Leading or Trailing Spaces in Password**

- **Severity Level: Low**
- **Current Findings:**
  **The system does not allow login if the password contains leading or trailing spaces, even when the credentials are correct. This could cause usability issues.**
  https://order-products-images.s3.amazonaws.com/bd35dbd6-e164-4366-b6a3-7b89ba9beedb
- **Expected Result:**
  **The system should automatically trim leading and trailing spaces from the password and allow login if the credentials are correct.**

**Test Case 3.0: Measure Login Page Load Time**

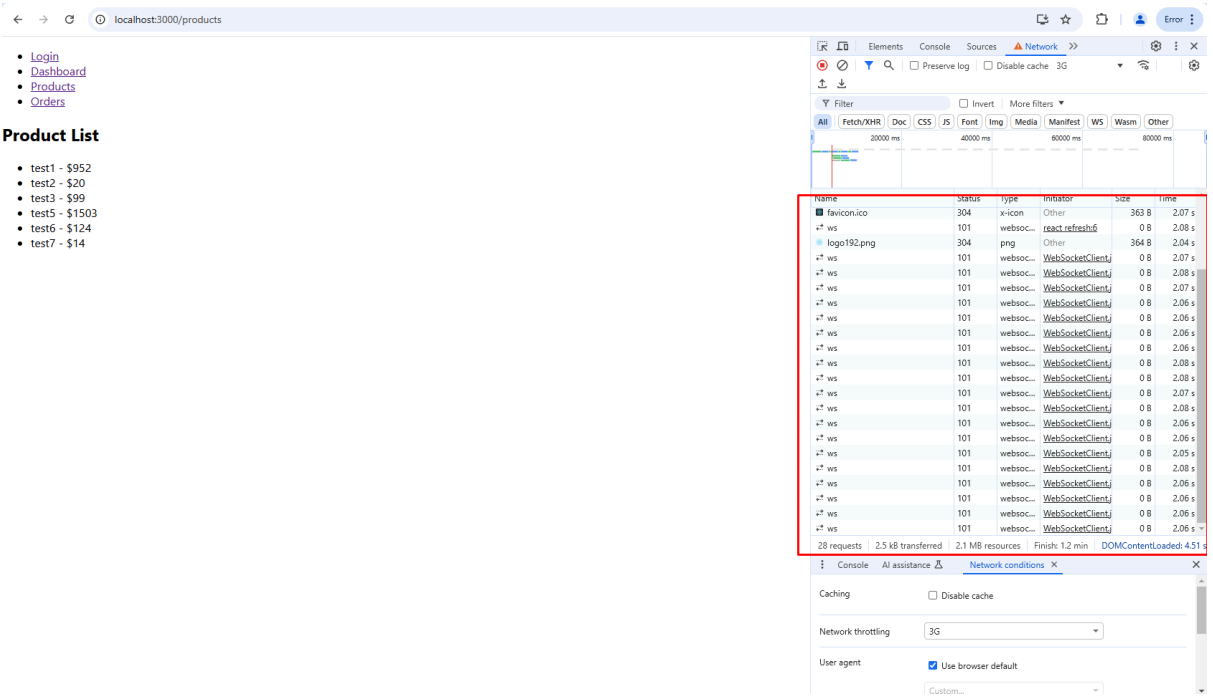
- **Severity Level: High**
- **Current Findings:**
  **The login page loads within 113ms on high-speed networks and within 4 seconds on slow networks. However, on extremely low-speed connections, the page continuously retries and never completes the loading process.**
  https://order-products-images.s3.amazonaws.com/3ca20b81-cbd2-4f81-bf14-d6c64c3dadca
- **Expected Result:**
  **The login page should load efficiently on all network speeds. The page should**

**PART 3 QA CHALLENGE: Andrés Vergara**

**complete loading within 4 seconds on slow connections, without continuous retries or failed loading attempts.**

## Test Case 3.1: Test Product Listing Page Load Time

- **Severity Level: High**
- **Current Findings:**
  **On high-speed networks, the product listing page loads within 1 second, as expected. However, on low-speed networks, the page continuously retries and never finishes loading. This issue can negatively impact user experience in slower network environments.**




- **Expected Result:**
  **The product listing page should load efficiently on all network speeds, completing the load process within 4 seconds on slow connections without retrying repeatedly.**

## Test Case 3.2: Verify Performance of Time Dashboard

- **Severity Level: High**
- **Current Findings:**
  **On high-speed networks, the dashboard page loads in less than 200ms, which is good. However, on slow connections, the page keeps reconnecting and retrying, negatively impacting performance.**

**PART 3 QA CHALLENGE: Andrés Vergara**

https://order-products-images.s3.amazonaws.com/ea707deb-95fc-4012-b521-d78be8a48598

- **Expected Result:**
  **The dashboard should load efficiently on all network speeds, completing within 3 seconds on slow connections without repeated retries or reconnecting issues.**

**Additional Findings:**

- **Product Listing Requests on Empty Products:**
  **Even when there are no products available, the system continues to send requests to fetch products, possibly searching for them in the background. This may lead to unnecessary load and inefficient resource use.**

**Expected Result:**
**The system should stop sending requests when there are no products available.**

- **Lack of Restrictions on Adding and Removing Products via Postman:**
  **Products can be added and removed without restrictions via Postman, which could pose a serious security risk. Any user could potentially manipulate the product database without proper authorization.**

**Expected Result:**
**Proper validation should be implemented to ensure that product management actions (adding/removing products) are only allowed via authenticated, authorized processes.**

- **Duplicate Product IDs:**
  **It was found that the system allows the creation of multiple products with the same ID. This could lead to inconsistencies and errors in the system.**

**Expected Result:**
**The system should prevent the creation of products with duplicate IDs, ensuring each product has a unique identifier.**

- **Order Prices Not Displayed:**
  **Orders are being processed and shown without any pricing information, which could lead to confusion and errors in order management.**

**Expected Result:**
**The order details should include accurate pricing information for each order.**

- **Missing Reset Password Option on Login Page:**
  **The login page does not provide an option for users to reset their password if they forget it. This could hinder users' ability to regain access to their accounts.**

**PART 3 QA CHALLENGE: Andrés Vergara**
**Expected Result:**
**A "Reset Password" link should be added to the login page, allowing users to recover their accounts.**