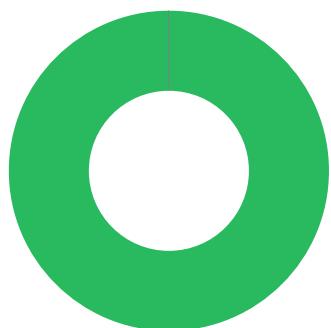


# Regression Run - v1.0 (Release Candidate)

Full regression suite execution. Scope includes End-to-End (E2E) validation for Web UI (SauceDemo) and Functional API testing (Booking Management & Trello Framework).

## Completion chart



## Completion stats

- Untested (0)
- Passed (24)
- Failed (0)
- Blocked (0)
- Retest (0)
- Skipped (0)
- Invalid (0)

## Completion rate

100%

## Started by

Marcie Jimenez

## Start time

2026-01-21 18:43:49

## Estimated

00:03:59.916

## Time spent

00:00:20.656

## Environment

## Milestone

# Login with a standard user

Status	Time spent	Assignee
Passed	00:00:02	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:02.739	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:12

### Steps

Step	1
Action	Enter standard_user in the "Username" field.
Step	2
Action	Enter secret_sauce in the "Password" field.
Step	3
Action	Click the "Login" button.
Expected result	The system redirects the user to the Products Inventory page ( <a href="https://www.saucedemo.com/inventory.html">https://www.saucedemo.com/inventory.html</a> ). The "Products" header and the shopping cart icon are clearly visible in the header.

# Login with a locked out user

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.919	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:14

### Steps

Step	1
Action	Enter locked_out_user in the "Username" field.
Step	2
Action	Enter secret_sauce in the "Password" field.
Step	3
Action	Click the "Login" button.
Expected result	The system denies access and remains on the login page. A red error message is displayed at the bottom of the form stating: "Epic sadface: Sorry, this user has been locked out."

## Sort products by Name (Z to A)

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.867	Marcie Jimenez	-

**Finish time**  
2026-01-21 18:44:16

### Steps

Step	1
Action	Click on the "Filter" icon/dropdown in the subheader (default is "Name (A to Z)").
Step	2
Action	Select the option "Name (Z to A)".
Expected result	The product grid refreshes immediately. The products are sorted alphabetically in descending order. The first product on the top-left is "Test.allTheThings() T-Shirt (Red)".

# Remove item from the shopping cart

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.968	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:18

### Steps

Step	1
Action	Click the "Remove" button associated with "Sauce Labs Backpack".
Expected result	The "Sauce Labs Backpack" container is removed from the list immediately. The shopping cart badge in the header updates correctly (the number decreases or the badge disappears if the cart becomes empty).

# Successful checkout process with a single item

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.754	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:20

### Steps

Step	1	
Action	Click the "Checkout" button.	
Step	2	
Action	Fill in the "Checkout: Your Information" form with valid data:	
Input data	<ul style="list-style-type: none"> <li>First Name: John</li> <li>Last Name: Doe</li> <li>Zip/Postal Code: 12345</li> </ul>	
Step	3	
Action	Click the "Continue" button.	
Step	4	
Action	On the "Checkout: Overview" page, scroll down and click the "Finish" button.	
Expected result	The system redirects to the "Checkout: Complete!" page. The header displays "Thank you for your order!" and the "Back Home" button is visible.	

# Checkout - Missing Zip/Postal Code Validation

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.788	Marcie Jimenez	-

**Finish time**  
2026-01-21 18:44:22

### Steps

Step	1
Action	Enter Jane in the "First Name" field.
Step	2
Action	Enter Doe in the "Last Name" field.
Step	3
Action	Leave the "Zip/Postal Code" field <b>empty</b> .
Step	4
Action	Click the "Continue" button.
Expected result	The system validates the input and displays a red error button. The message "Error: Postal Code is required" appears. The user is NOT redirected to the Overview page.

## [GET] Health - Ping

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.845	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:23

### Steps

Step	1
Action	Send GET request to {{baseUrl}}/ping
Expected result	<ol style="list-style-type: none"><li>1. Status Code 201 Created.</li><li>2. Response time is below 1000ms.</li></ol>

## [POST] Auth - CreateToken

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.780	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:25

### Steps

Step	1
Action	Send POST request to {{baseUrl}}/auth with Body: {   "username": "admin",   "password": "password123" }
Expected result	1. Status code is 200 OK. 2. Response body contains a valid "token" string (length > 0). 3. Token is successfully saved to the "token" environment variable.

## [POST] Booking - CreateBooking

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.720	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:27

#### Steps

Step	1
Action	Send POST request to {{baseUrl}}/booking with Body: <pre>{   "firstname" : "Marcie",   "lastname" : "Jimenez",   "totalprice" : 150,   "depositpaid" : true,   "bookingdates" : {     "checkin" : "2026-01-01",     "checkout" : "2026-01-10"   },   "additionalneeds" : "Breakfast" }</pre>
Expected result	<ol style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Response body contains a unique "bookingid" (number).</li> <li>3. Validated Data: "firstname" is "Marcie" and "depositpaid" is true.</li> <li>4. "bookingid" is successfully saved to environment variables.</li> </ol>

**[GET] Booking - GetBookingById**

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

**Results****Result 1**

Status	Time spent	User	Defects
Passed	00:00:00.824	Marcie Jimenez	-

**Finish time**

2026-01-21 18:44:28

**Steps**

Step	1
Action	Send GET request to {{baseUrl}}/booking/{{bookingId}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Data Consistency: "firstname" is "Marcie" and "lastname" is "Jimenez".</li> <li>3. Date Validation: Checkin date matches YYYY-MM-DD format (Regex).</li> <li>4. Type Validation: "totalprice" is a valid number.</li> </ul>

## [PUT] Booking - UpdateBooking

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.872	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:30

### Steps

Step	1
Action	Send PUT request to {{baseUrl}}/booking/{{bookingId}} Header: Cookie: token={{token}} Body: { "firstname": "Marcie", "lastname": "Jimenez", "totalprice": 200, "depositpaid": true, "bookingdates": { "checkin": "2026-02-01", "checkout": "2026-02-05" }, "additionalneeds": "Full Update Test" }
Expected result	1. Status code is 200 OK. 2. Update Verified: "totalprice" is 200 and "additionalneeds" is "Full Update Test". 3. Persistence Verified: "firstname" remains "Marcie".

# [PATCH] Booking - PartialUpdateBooking

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.813	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:32

### Steps

Step	1
Action	<p>Send PATCH request to {{baseUrl}}/booking/{{bookingId}}</p> <p>Header: Cookie: token={{token}}</p> <p>Body:</p> <pre>{   "firstname": "Marcie QA",   "lastname": "Jimenez Master" }</pre>
Expected result	<ol style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Patched Fields: "firstname" is "Marcie QA"\, "lastname" is "Jimenez Master".</li> <li>3. Regression Check: "totalprice" remains 200 (unchanged from previous state).</li> </ol>

**GET] Booking - GetBookingIdsByFilter**

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

**Results****Result 1**

Status	Time spent	User	Defects
Passed	00:00:00.751	Marcie Jimenez	-

**Finish time**

2026-01-21 18:44:33

**Steps**

Step	1
Action	Send GET request to {{baseUrl}}/booking?firstname=Marcie QA&lastname=Jimenez Master
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Search Validation: Response is an Array with at least one item (length &gt; 0).</li> <li>3. Structure Check: The found item contains a "bookingid".</li> </ul>

# [DELETE] Booking - DeleteBooking

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.687	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:35

### Steps

Step	1
Action	Send DELETE request to {{baseUrl}}/booking/{{bookingId}} Header: Cookie: token={{token}}
Expected result	1. Status code is 201 Created. 2. Response Body: Text equals "Created". 3. Performance: Response time is below 1000ms.

## [GET] Auth - Check Credentials

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.733	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:37

#### Steps

Step	1
Action	Send GET request to {{baseUrl}}/members/me?key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Response contains User ID and Username.</li> <li>3. Performance: Response time &lt; 1000ms.</li> </ul>

## [POST] Boards - CreateBoard

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.801	Marcie Jimenez	-

#### Finish time

2026-01-21 18:44:39

#### Steps

Step	1
Action	Send POST request to {{baseUrl}}/boards/?name=Postman-Workshop&key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Response body contains a valid "id".</li> <li>3. Automation: "boardId" is successfully saved to environment variables.</li> </ul>

## [POST] Lists - CreateList

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.801	Marcie Jimenez	-

### Finish time

2026-01-21 18:44:40

### Steps

Step	1
Action	Send POST request to {{baseUrl}}/lists?name=To Do&idBoard={{boardId}}&key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Response body contains a valid "id".</li> <li>3. Automation: "listId" is successfully saved to environment variables.</li> </ul>

## [POST] Cards - CreateCard

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

## Results

### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.680	Marcie Jimenez	-

### Finish time

2026-01-21 18:44:42

### Steps

Step	1
Action	Send POST request to {{baseUrl}}/cards?idList={{listId}}&name=My First Automated Card&key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Performance: Response time is below 800ms.</li> <li>3. Data Integrity: "name" equals "My First Automated Card".</li> <li>4. Structure: Response contains a valid "id".</li> <li>5. Safety Check: Response does not contain "error" property.</li> <li>6. Automation: "cardId" is successfully saved.</li> </ul>

## [PUT] Cards - UpdateCard

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.802	Marcie Jimenez	-

#### Finish time

2026-01-21 18:44:43

#### Steps

Step	1
Action	Send PUT request to {{baseUrl}}/cards/{{cardId}}?desc=Updated description via API&pos=top&key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Data Validation: "desc" is updated to "Updated description via API".</li> <li>3. Position check: "pos" is not null.</li> <li>4. Immutability Check: "id" matches the known {{cardId}}.</li> <li>5. Safety: Response does not contain "error".</li> </ul>

## [GET] Cards - GetCard

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.695	Marcie Jimenez	-

Finish time  
2026-01-21 18:44:45

#### Steps

Step	1
Action	Send GET request to {{baseUrl}}/cards/{{cardId}}?key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Data Persistence: "desc" equals "Updated description via API".</li> <li>3. Structure Check: Response contains "id" and "name".</li> <li>4. Data Validation: "name" is not empty.</li> </ul>

## [POST] Cards - AddComment

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.667	Marcie Jimenez	-

#### Finish time

2026-01-21 18:44:46

#### Steps

Step	1
Action	Send POST request to {{baseUrl}}/cards/{{cardId}}/actions/comments?text=Automation testing is fun!&key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Content Verification: "data.text" matches "Automation testing is fun!".</li> <li>3. User Verification: "idMemberCreator" is not null.</li> <li>4. Structure: Response contains a valid "id".</li> <li>5. Sanity Check: Object is not marked as "deleted".</li> </ul>

## [POST] Checklists - CreateChecklist

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.773	Marcie Jimenez	-
<b>Finish time</b>			
2026-01-21 18:44:48			
<b>Steps</b>			
Step	1		
Action	Send POST request to {{baseUrl}}/cards/{{cardId}}/checklists?name=QA Tasks&key={{key}}&token={{token}}		
Expected result	1. Status code is 200 OK. 2. Data Verification: "name" equals "QA Tasks". 3. Relationship Check: "idCard" matches the environment variable {{cardId}}. 4. Initial State: "checkItems" is an empty array.		

## [DELETE] Cards - DeleteCard

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.571	Marcie Jimenez	-

#### Finish time

2026-01-21 18:44:50

#### Steps

Step	1
Action	Send DELETE request to {{baseUrl}}/cards/{{cardId}}?key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Operation Check: Response confirms successful deletion (Body is OK/Null).</li> <li>3. Performance: Response time is below 800ms.</li> </ul>

## [DELETE] Boards - DeleteBoard

Status	Time spent	Assignee
Passed	00:00:00	Marcie Jimenez

### Results

#### Result 1

Status	Time spent	User	Defects
Passed	00:00:00.806	Marcie Jimenez	-

#### Finish time

2026-01-21 18:44:51

#### Steps

Step	1
Action	Send DELETE request to {{baseUrl}}/boards/{{boardId}}?key={{key}}&token={{token}}
Expected result	<ul style="list-style-type: none"> <li>1. Status code is 200 OK.</li> <li>2. Response Validation: Body contains property "_value".</li> <li>3. Performance: Response time is below 2000ms.</li> </ul>