**SOFTWARE REQUIREMENTS SPECIFICATION**

URBAN LADDER AUTOMATION PROJECT

|  |  |  |  |
| --- | --- | --- | --- |
| Details | Prepared By | Reviewed By | Approved By |
| Name | Code Crafters | P Devathilagai | P Devathilagai |
| Role | Quality Assurance Interns | Batch Trainer |  |

Contents

[1. Introduction 2](#_Toc203659090)

[1.1 Intended Audience 3](#_Toc203659091)

[2. Overall Description 4](#_Toc203659092)

[2.1 Operating Environment 4](#_Toc203659093)

[3. Functional Requirements 5](#_Toc203659094)

[3.1 Bookshelves Filter and Extraction Automation 5](#_Toc203659095)

[3.2 Cart Item Verification Automation 6](#_Toc203659096)

[3.3 Store States Extraction Automation 6](#_Toc203659097)

[3.4 Login Authentication Validation 7](#_Toc203659098)

[3.5 Social Media Link Verification 8](#_Toc203659099)

[3.6 Auto-suggestion Capture Automation 9](#_Toc203659100)

[3.7 Help Page FAQ Extraction Automation 10](#_Toc203659101)

[3.8 Being at Home Category Collection Automation 11](#_Toc203659102)

[3.9 Sofas and Recliners Category Collection Automation 12](#_Toc203659103)

[3.10 Track Order Functionality Automation 13](#_Toc203659104)

[Testing Approach 14](#_Toc203659105)

[Tools & Technologies 15](#_Toc203659106)

[Documentation & Deliverables 15](#_Toc203659107)

[Scope Boundaries 15](#_Toc203659108)

# 1. Introduction

This document outlines the Software Requirements Specification (SRS) for the **Urban Ladder Website Automation Project**. The objective is to automate key functionalities of the Urban Ladder e-commerce platform using **Behavior Driven Development (BDD)** with the **Cucumber framework**, **TestNG**, and **data-driven testing** via **Excel**.

The automation suite is designed to validate user-centric workflows such as product filtering, cart operations, login authentication, social media link verification, FAQ extraction, and order tracking. The system ensures that critical business functionalities are tested efficiently and reliably, improving test coverage and reducing manual testing efforts.

## 1.1 Intended Audience

* QA Engineers
* Automation Testers
* Business Analysts
* Project Managers
* Technical Leads
* Product Owners

**1.2 Project** Scope

This document defines the functional and technical requirements for automating workflows on the **Urban Ladder Web Application**. It captures the business logic, validation rules, and data dependencies necessary to ensure robust and scalable automation.

The scope includes the automation of 10 key workflows, covering:

Product filtering and extraction (e.g., Bookshelves under ₹15,000)

Cart operations and item verification

Login functionality using multiple credential scenarios

Social media link validation

Auto-suggestion capture from the search bar

FAQ extraction from the Help section

Category collection from various product sections

Order tracking functionality

The automation framework is built using **Selenium WebDriver**, **Cucumber**, and **TestNG**, with **Excel-based data input/output** to support data-driven testing. The test cases are modeled using Gherkin syntax to ensure clarity and maintainability.

This initiative aims to enhance the **reliability**, **scalability**, and **efficiency** of the Urban Ladder testing process by automating repetitive and critical user journeys. It also ensures faster feedback cycles and improved test coverage across the platform.

# 2. Overall Description

## 2.1 Operating Environment

Windows Operating System  
• Web Browser (Chrome, Firefox)  
• Selenium WebDriver  
• Cucumber Framework  
• TestNG  
• Excel/XML for Data Input

# 3. Functional Requirements

## 3.1 Bookshelves Filter and Extraction Automation

* Req0101: Navigate to the Urban Ladder home page.
* Req0102: Handle any popup that appears on the screen.
* Req0103: Navigate to the Bookshelves section from the homepage.
* Req0104: Apply a price filter to display bookshelves under ₹15,000.
* Req0105: Select the "Open Storage" type from the bookshelf categories.
* Req0106: Filter the results to show only "In Stock" items.
* Req0107: Extract the top 6 bookshelf products along with their prices.
* Req0108: Write the extracted product details to the Excel sheet named "Bookshelves".

A screenshot of a computer

AI-generated content may be incorrect.

## 3.2 Cart Item Verification Automation

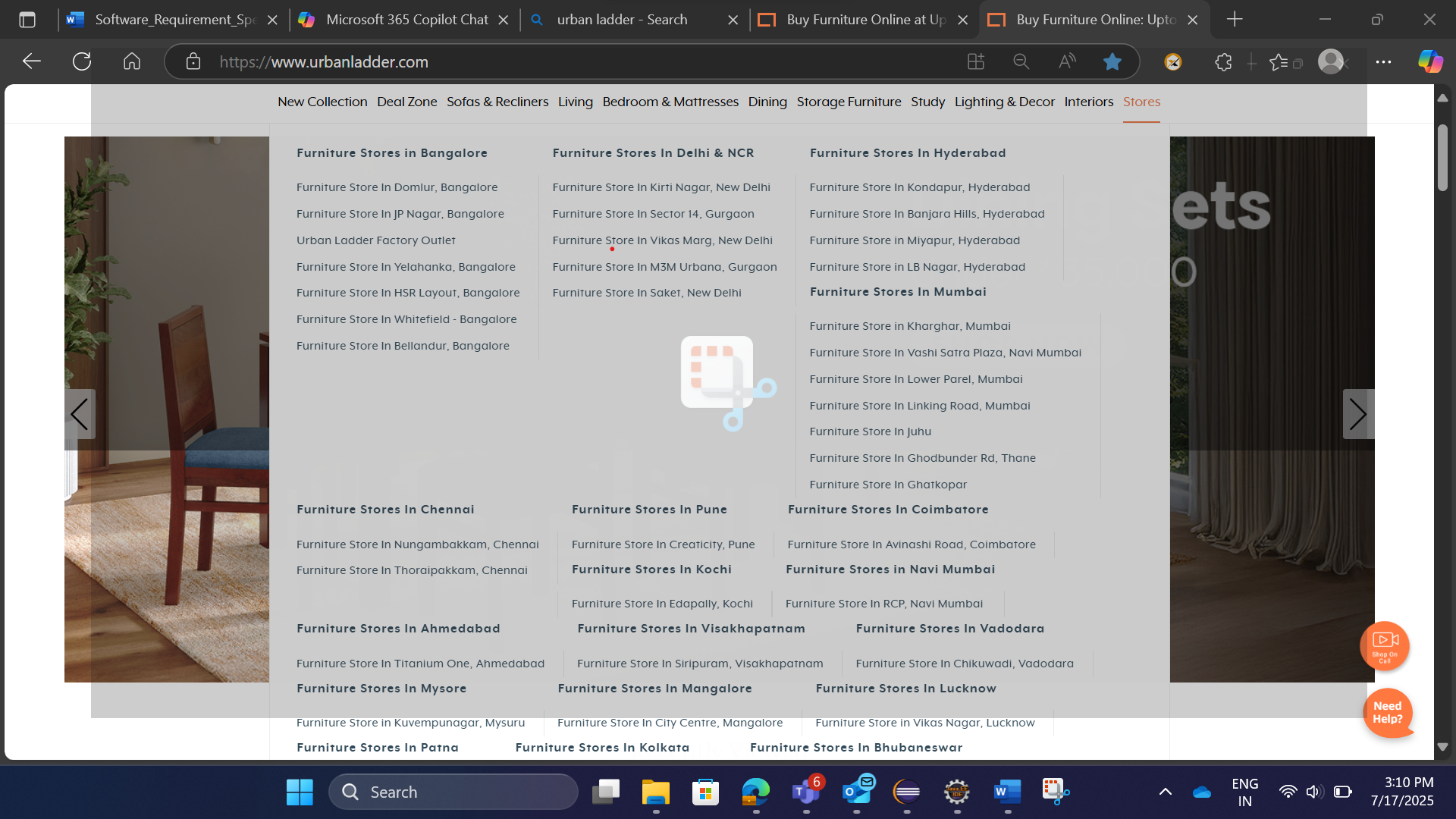
* **Req0201**: Launch the Urban Ladder application and navigate to the home page.
* **Req0202**: Search for the keyword "chair" using the search bar.
* **Req0203**: Click on the first item in the search results and switch to the newly opened product window.
* **Req0204**: Add the selected item to the shopping cart.
* **Req0205**: Verify that the item name in the cart matches the originally selected product.
* **Req0206**: Retrieve and print the item price for confirmation.

A screenshot of a computer

AI-generated content may be incorrect.

## 3.3 Store States Extraction Automation

* **Req0301**: Navigate to the Urban Ladder home page.
* **Req0302**: Hover over the "Stores" tab to reveal the dropdown menu.
* **Req0303**: Scroll through and retrieve the list of store states displayed.
* **Req0304**: Write the extracted store state names to the Excel sheet named "Cities".



## 3.4 Login Authentication Validation

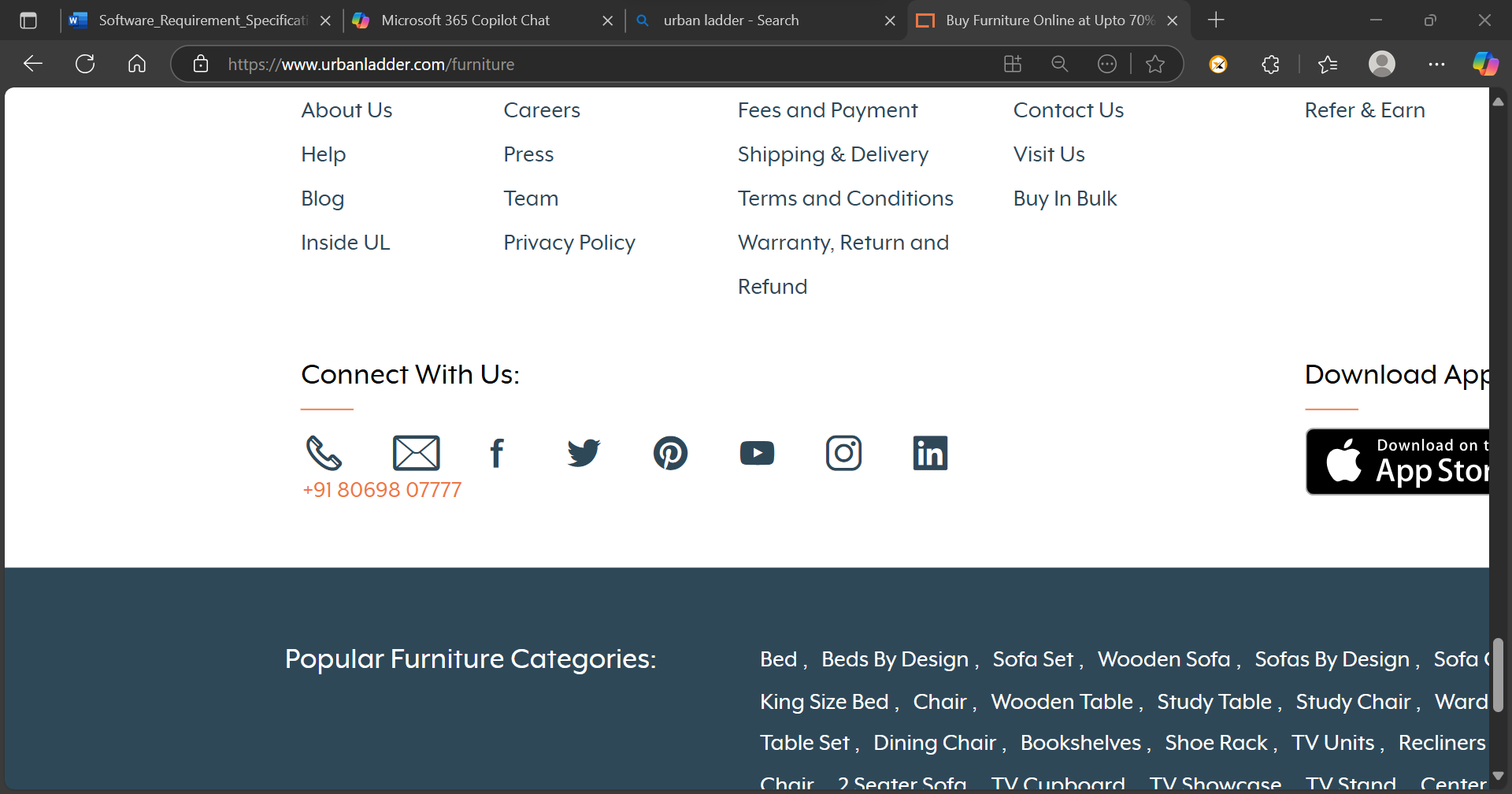
* **Req0401**: Retrieve login credentials (email and password) from an external Excel sheet.
* **Req0402**: Execute login attempts using the retrieved credentials for multiple scenarios:
  + Missing email
  + Missing password
  + Invalid credentials
  + Valid credentials
* **Req0403**: Validate whether login is successful or capture and log error messages for failed attempts.

A screenshot of a website

AI-generated content may be incorrect.

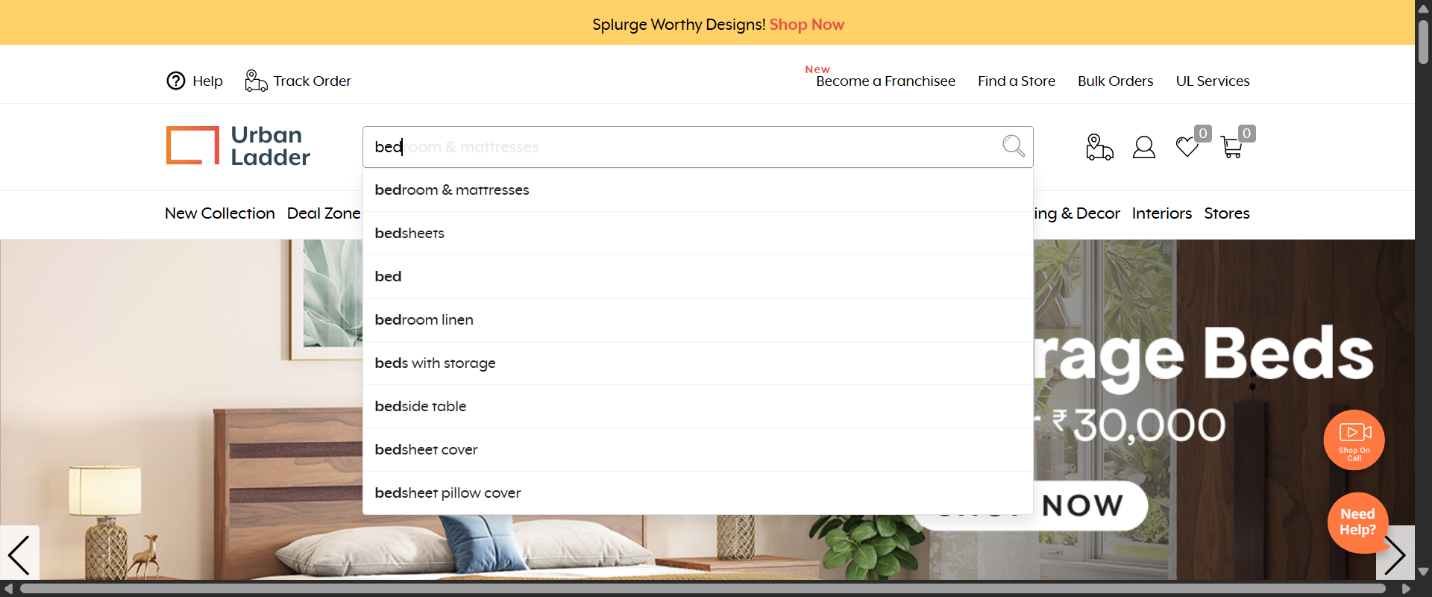
## 3.5 Social Media Link Verification

* **Req0501**: Navigate to the Urban Ladder home page.
* **Req0502**: Scroll to the footer section of the page.
* **Req0503**: Verify the presence of social media icons (e.g., Facebook, Twitter, Instagram).
* **Req0504**: Click each social media icon and ensure it redirects to the correct platform.
* **Req0505**: Retrieve and print the URLs of each social media link to the console.
* **Req0506**: Save the URLs to the Excel sheet named "SocialMedia".



## 3.6 Auto-suggestion Capture Automation

* **Req0601**: Launch the Urban Ladder application and close any popup if present.
* **Req0602**: Enter the keyword "bed" into the website’s search field.
* **Req0603**: Wait for the auto-suggestion dropdown to appear.
* **Req0604**: Capture all visible auto-suggestion texts.
* **Req0605**: Write the captured suggestions to an Excel sheet for reference and validation.



## 3.7 Help Page FAQ Extraction Automation

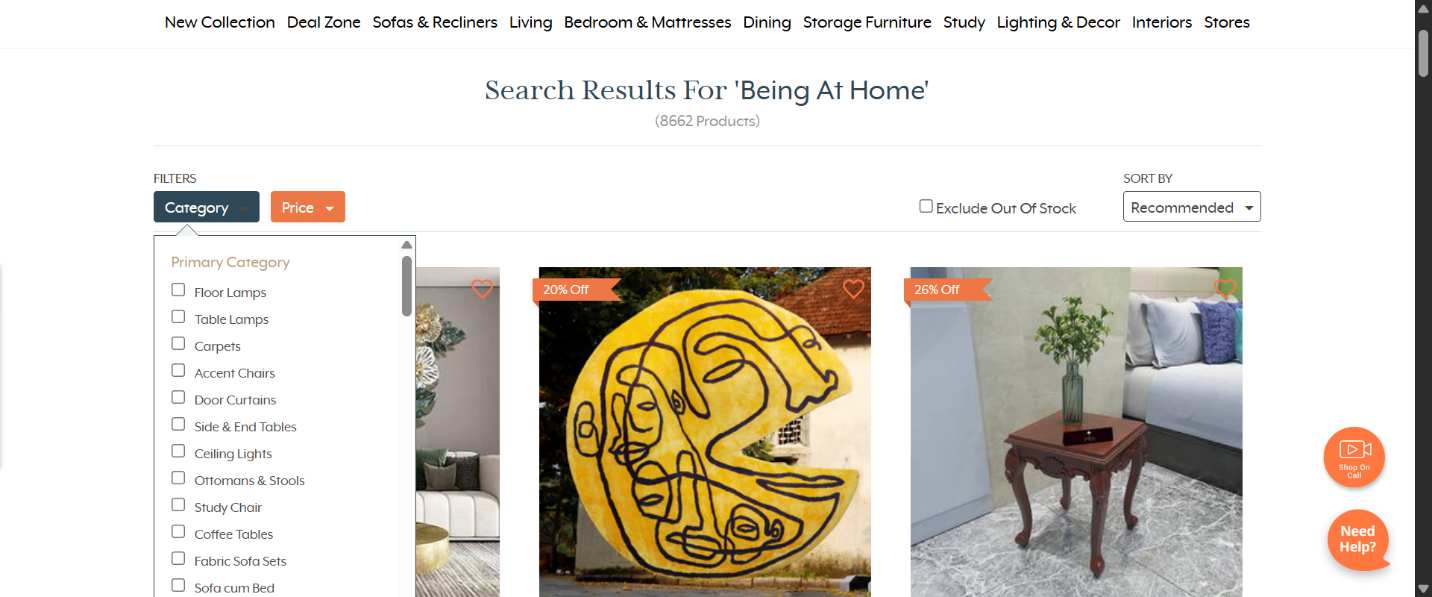
* **Req0701**: Navigate to the Help section of the Urban Ladder website.
* **Req0702**: Expand all available help questions listed on the page.
* **Req0703**: Extract all visible FAQ questions and their corresponding answers.
* **Req0704**: Save the extracted FAQ content into an Excel sheet for documentation and validation.

A screenshot of a computer

AI-generated content may be incorrect.

## 3.8 Being at Home Category Collection Automation

* **Req0801**: Search for the term "being at home" using the website’s search functionality.
* **Req0802**: Redirect the user to the "Being at Home" landing page.
* **Req0803**: Click on the category option available on the page.
* **Req0804**: Display the list of categories under the "Being at Home" section.
* **Req0805**: Collect and store all category names for reporting and validation.



## 3.9 Sofas and Recliners Category Collection Automation

* **Req0901**: Navigate to the Urban Ladder home page.
* **Req0902**: Hover over the "Sofas and Recliners" section in the main navigation menu.
* **Req0903**: Display all subcategories and items listed under the "Sofas and Recliners" section.
* **Req0904**: Collect and store the names of all subcategory items for documentation and validation.

A screenshot of a computer

AI-generated content may be incorrect.

## 3.10 Track Order Functionality Automation

* **Req1001**: Navigate to the Urban Ladder home page.
* **Req1002**: Click on the "Track Order" option available on the website.
* **Req1003**: Redirect the user to the order tracking page.
* **Req1004**: Enter the required order number and phone number into the respective fields.
* **Req1005**: Submit the form and retrieve the tracking response or status update.
* **Req1006**: Display or log the tracking information for validation and reporting.

A screenshot of a computer

AI-generated content may be incorrect.

# Testing Approach

Use of the **Cucumber BDD framework** for scenario-driven automation that mirrors user behavior.

Integration with **TestNG** for robust execution, reporting, and test lifecycle management.

Implementation of **data-driven testing** to verify multiple input conditions using spreadsheet and XML formats.

Use of **Excel sheet writing/reading** for test output validation and result storage.

# Tools & Technologies

Automation was built with **Selenium WebDriver** for browser-based interaction.

Testing scripts managed using **Java**, integrated with Maven/Gradle as needed.

Compatibility across major browsers like Chrome and Firefox on Windows OS.

# Documentation & Deliverables

Feature files defining Gherkin-based test scenarios.

Excel/XML templates for data input and output.

Screenshots and logs for error messages, form validations, and visual checkpoints.

Well-organized test reports and summaries generated post-execution.

# Scope Boundaries

The project does not include non-web applications, native mobile testing, or backend API automation.

Only UI-driven workflows relevant to the defined features will be automated at this stage.

Third-party integrations (e.g., payment gateways, external hospital data providers) are considered out of scope unless explicitly required.