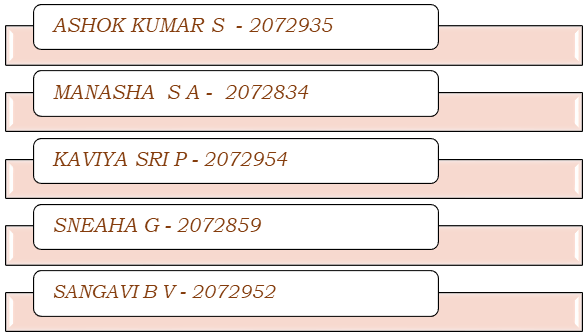
***HACKATHON DOCUMENTATION***

***TEAM NAME***: ***MASKS***

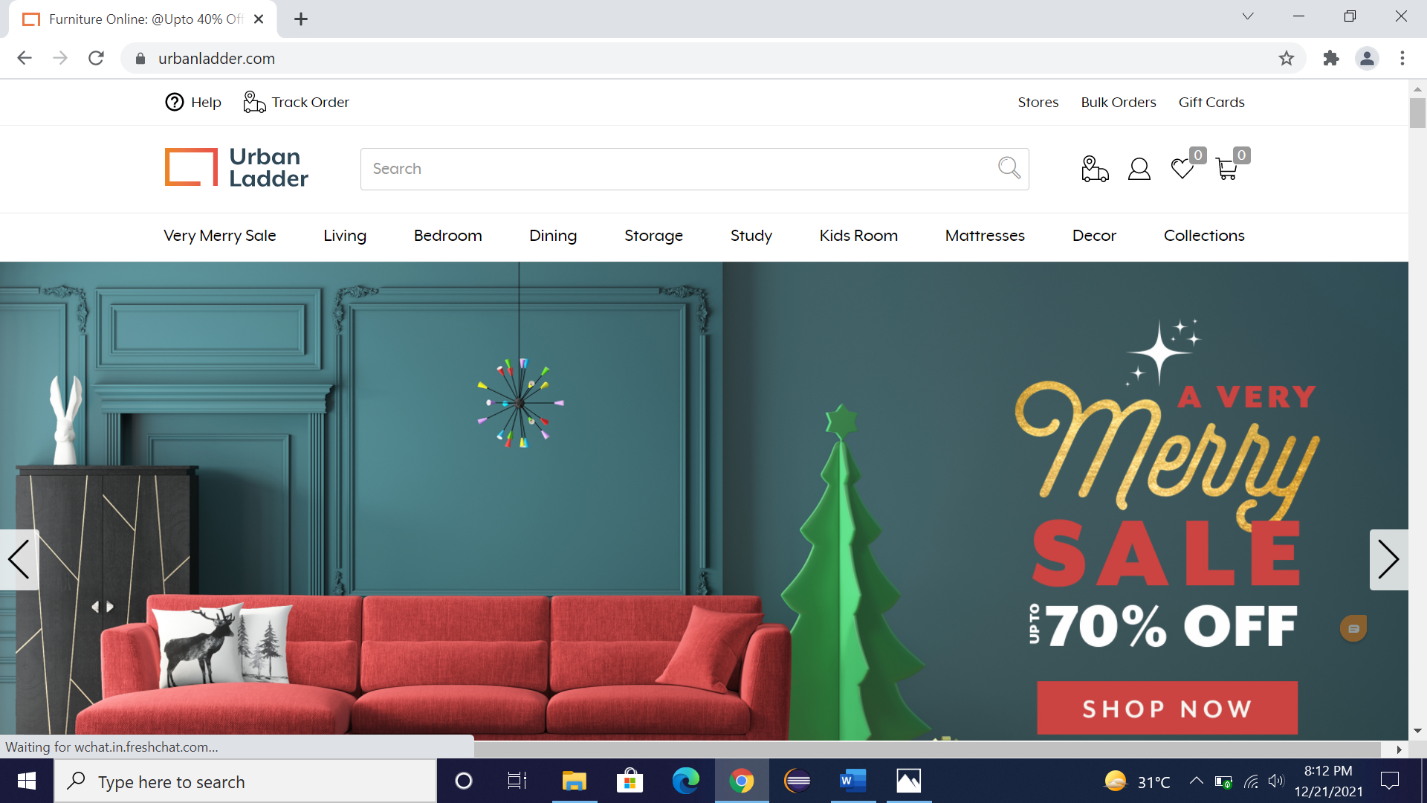
***TITLE***: ***DISPLAY BOOKSHELVES***

TEAM MEMBERS

****

**1.WEBSITE:**

[**https://www.urbanladder.com**](https://www.urbanladder.com)

****

**2.TOOLS AND TECHNOLOGY USED:**

***1.Selenium with Java in Eclipse***: Selenium is an open-source tool that automates web browsers. Eclipse is the most used IDE by the Java developers. And it is used for any programming language for which a plugin is available. Selenium provides a playback tool for authoring functional tests developed using java, the Eclipse platform can be used to develop rich client applications, integrated development environments and other tools**.**

***2.TestNG (Test Next Generation***): TestNG framework is an open-source automated testing framework and here NG means Next Generation. TestNG is inspired from Junit and NUnit, but it has a new functionality that makes this framework more powerful and easier. TestNG provides parallel execution of test methods. It has support for parameterizing test cases using **@** Parameters annotations.

***3.Maven:*** Maven is a build automation tool used primarily for Java projects. It is used to check the compilation issues between framework components whenever multiple test engineer integrates their files into the same framework. It provides ‘build success’ message if no compilation issues in the framework or else provide ‘build failure’ message. Maven has new features like dependency, which is used to download the dependency jar from the internet before the test execution.

***4.Page Factory with POM:*** Pagefactory is a class provided by selenium webdriver to support page object design patterns. The advantage is that it makes your framework more structured and maintainable.

**3.WORK DONE**:

**1**.The code is written in JAVA and is automated in different browsers with the help of Selenium. The project and dependencies are managed with the help of Maven. The project and dependencies are managed with the help of Maven.

**2.**POM is used to create classes for each page therefore reducing usage of same code for various objects.

**3.**Input data is obtained from an excel file with the help of Apache POI.

**KEY AUTOMATION SCOPE:**

* Handling alert, drag & drop, search option
* Extract menu items & store in collections
* Navigating back to home page
* Scrolling down in web page
* Filling form (in different objects in web page)
* Capture warning message

**4.PROJECT STRUCTURE:**

**Graphical user interface, application

Description automatically generated**

**5.STEPS TO EXECUTE:**

***1. Display the name & price of first 3 Bookshelves below Rs. 15000, with Storage type as open & include out of stock***

* Search Bookshelves in **urbanladder.com.**
* Select Bookshelves from category dropdown.
* Adjust the price slider.
* Set the price slider in the range *1,499-15,000.*
* Select the storage type as Open.
* Print the name and price of first 3 Bookshelves.

**A screenshot of a home screen

Description automatically generated with low confidence**

***2. Display the name & price of first 3 Bookshelves below Rs. 15000, with Storage type as open & exclude out of stock***

* Search Bookshelves in **urbanladder.com.**
* Select Bookshelves from category dropdown.
* Adjust the price slider.
* Set the price slider in the range *1,499-15,000*.
* Select the storage type as Open
* Click on Exclude Out of Stock
* Print the name & price of first 3 Bookshelves

A screenshot of a home screen

Description automatically generated with low confidence

***3. From Collections, retrieve all sub-menu items under BestSellers and store in a List; Display the same.***

* In the Homepage, Click on Collections.
* Display all the sub-menu items under Best sellers.

Graphical user interface

Description automatically generated

***4.Display the name & price of first 3 highest recommended Study Chairs.***

* In the search bar, enter Study Chair.
* Click on Search.
* Select category as Study Chair.
* Display the Study Chair based on the recommendation.
* Print the name & price of first 3 Study Chair.

Graphical user interface

Description automatically generated

***5. Gift cards - choose "Birthday/ Anniversary"; fill customize the gift card; fill from to details with any one input invalid (example: email); Capture the error message.***

* In the Homepage, Click on Gift Cards.
* Click on ***Birthday /Anniversary***.
* Click on the price *1000* in “Now you customize your gift card”.
* Enter the amount as 1500 in the next text box.
* Choose the year and month as Jan (2022) and Date as 22.
* Click on next.
* Enter the Recipient’s Name and invalid email Id.
* Enter all the details in “Your name, Your email and Your mobile”.
* Click on ***CONFIRM***.
* An error message is displayed on the screen.
* The screenshot of the error message is captured.

Graphical user interface, text, application

Description automatically generated

***Once Everything is Setup:***

* Go to Testng.xml.
* Right click on anywhere on the editor>>Run As>>Testng Suite.
* The automation will take place, the reports will be generated and screenshot is captured.

***Steps to view the Screenshot:***

* Right click on the screenshots folder.
* Click on Refresh button.
* Click on .png file.

**6.PREREQUISITES:**

1.Initially we need to create a maven project.

2.We have included the following maven dependencies:

* Selenium Java>>3.141.59, Apache POI>>3.17
* Apache POI Based on OPC and OOXML Schemas>>3.17

3.We have created test scenarios and test cases.

4.We created methods in the project.

5.Data driven concept is used for the pages to read data.

6.These are the basic requirements of the project.

**7.DEVELOPMENT PROGRESS:**

1. We completed the functional testing part in the first two days.

2. We divided our test cases into smoke and regression based on their functionality on the next day.

3. For the next two days, we implemented the different parts of the automation separately.

4. After the next two days, we executed the entire automation inside a single main function.

5. On day 8, We executed the same existing code inside Maven.

6. On day 9, We implemented the same code using TestNG.

7. On day 10 and 11, We implemented POM.

8. We used Apache POI in the next day.

8. The whole project integration part was completed by day13.

9. We completed the report part on day 14.

**8.EXTENT REPORT:**

**A screenshot of a computer

Description automatically generated**

**9.CHALLENGES FACED:**

1. The website had dynamic Xpath for some of the webelements.

2. We had some issues while handling the popup alerts.

3. We also faced some issues while capturing the screenshot of the error message.

**10.SUGGESTIONS:**

While giving valid email address/Phone number in the Gift cards section, it is not validating by sending an OTP to the customer.

It should verify by sending an OTP to the customer email/Phone number.