

**BILLING SYSTEM**

**TEAM - 3**

**RITHESH**

**NUTHAN**

**SANJAY**

**ARCHITH**

**MEGHANA**

**PRAMOD**

**December 2, 2022**

Table Of Contents:

1. PROBLEM STATEMENT…………………………………………2

2. INTRODUCTION………………………………………………….2

3.OBJECTIVES AND GOALS……………………………………….3

4. REQUIREMENTS………………………………………………….4

5. DESIGNING……………………………………………………….5-8

6. FUNCTIONALITIES………………………………………………9-11

7. WORKING PROCEDURE…………………………………………12

8. TESTING……………………………………………………………13

9. WORKFLOW…………….…………………………………………19

**BILLING SYSTEM**

Problem Statement:

Nowadays, in shopping malls we must wait for the billing in a long queue, it accrues traffic in a mall, and it is a very time-consuming process. So, there is a need to design a system that will reduce the time of billing procedure.

Introduction:

Nowadays shopping has become a daily activity in today’s world. One can see a huge rush at shopping malls on holidays, peak hours of the day and on special discount days. People purchase different items, put them in trolleys, and go to the billing counter for payments. So, now the problem is that the customers must wait in long queues to complete the payment process to check out their products. Hence, it is tedious and time consuming. To provide the customer with enhanced shopping experience we have designed a smart and effective way to shop at the supermarket. The system designed is technology oriented, time saving and commercial oriented system. After getting the products from the store, customers can generate their own bill using the web app. In the web app they can select the product which they have collected in the store and complete the payment process where they’ll get bill.

Goals:

* User friendly
* Simple and Fast
* Low cost and effective

Objectives:

* To reduce the time of the billing procedure
* To reach maximum customers at the right time
* To increase sales and profitability of the business

System Requirements:

* Spring Tool suite
* Java,
* Oracle SQL
* HTML , CSS , Js
* Tom cat server
* Maven
* JIRA

Spring Tool suite:

Spring Tool Suite (STS) is **a java IDE tailored for developing Spring-based enterprise applications**. It is easier, faster, and more convenient. And most importantly it is based on Eclipse IDE

Java:

Java is **the official language for Android mobile app development**.

Oracle SQL developer:

Oracle SQL Developer is a free graphical tool that **enhances productivity and simplifies database development tasks**.

HTML , CSS , JS:

* HTML is the code that is used **to structure a web page and its content**
* CSS is **the language for describing the presentation of Web pages, including colors, layout, and fonts**
* JS is used by programmers across the world **to create dynamic and interactive web content like applications and browsers**.

Tom Cat server:

Apache Tomcat is a web container. It **allows the users to run Servlet and JAVA Server Pages that are based on the web-applications**.

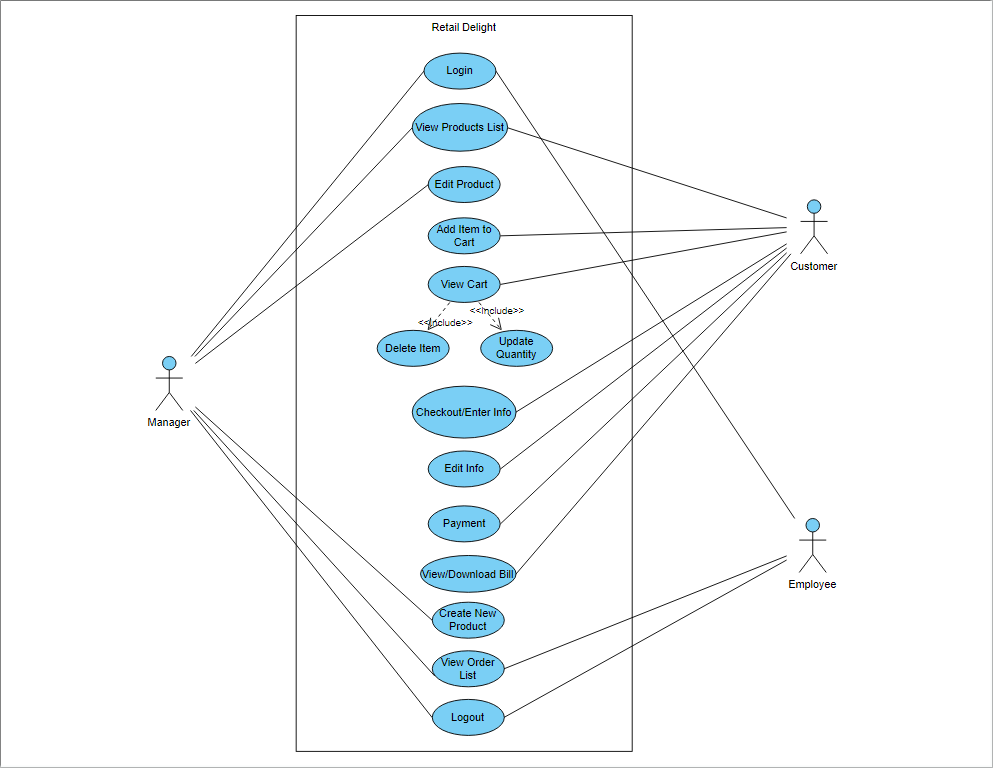
Maven:

Maven is used for building**, publishing, and deploying several projects**.

JIRA: Jira **helps teams plan, assign, track, report, and manage work and brings teams together for everything from agile software development and customer support to start-ups and enterprises**

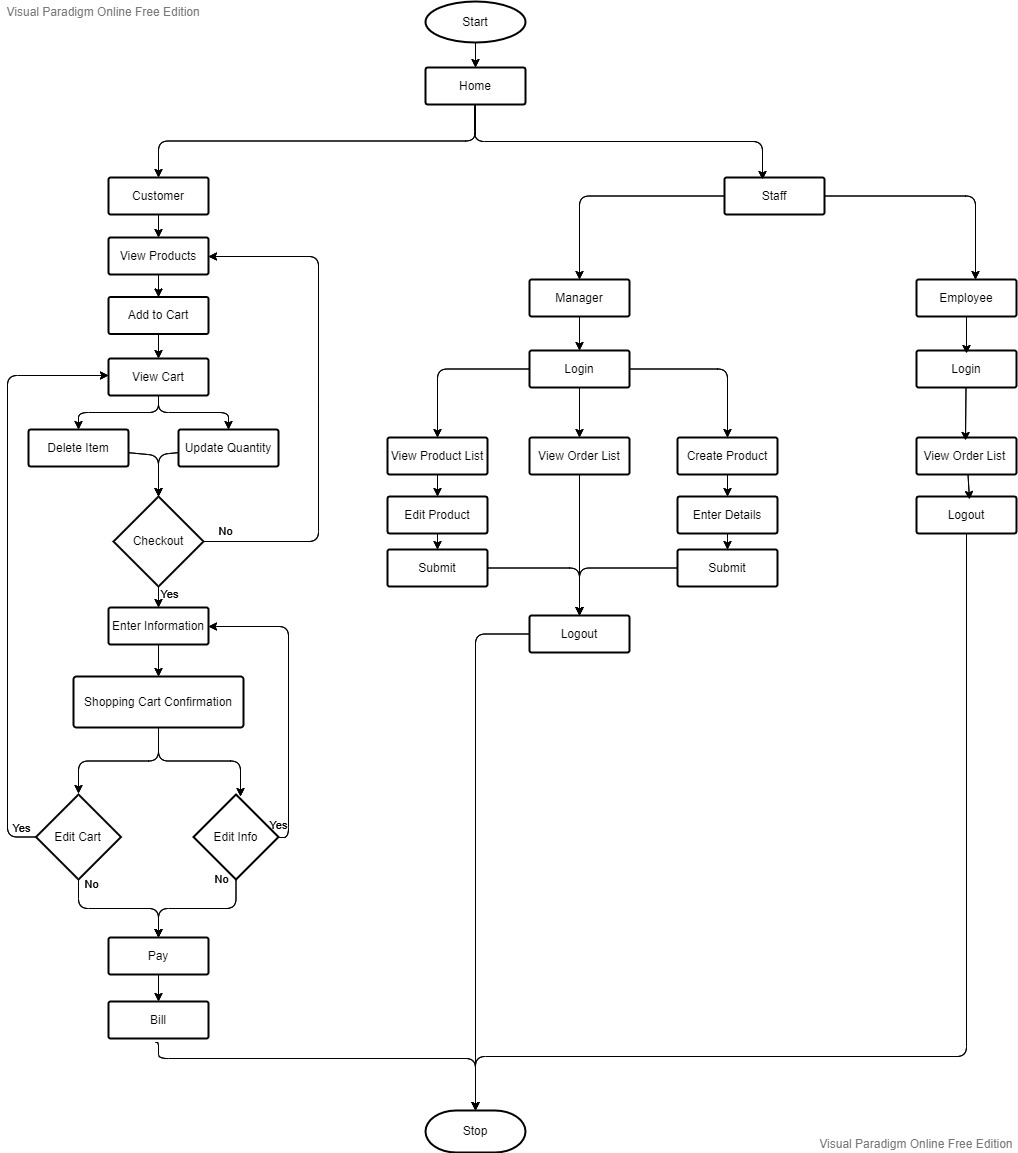
DESIGN PATTERN:

**Use Case** is a written description of how users will perform tasks on your website.



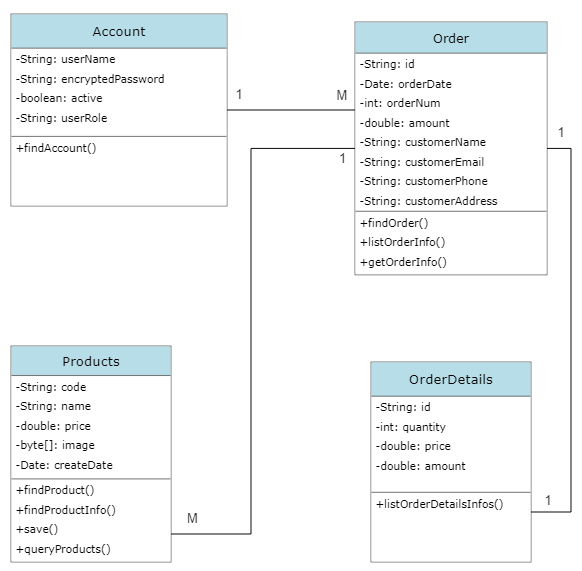
FLOW CHART:

**Flowchart** is a picture of the separate steps of a process in sequential order.



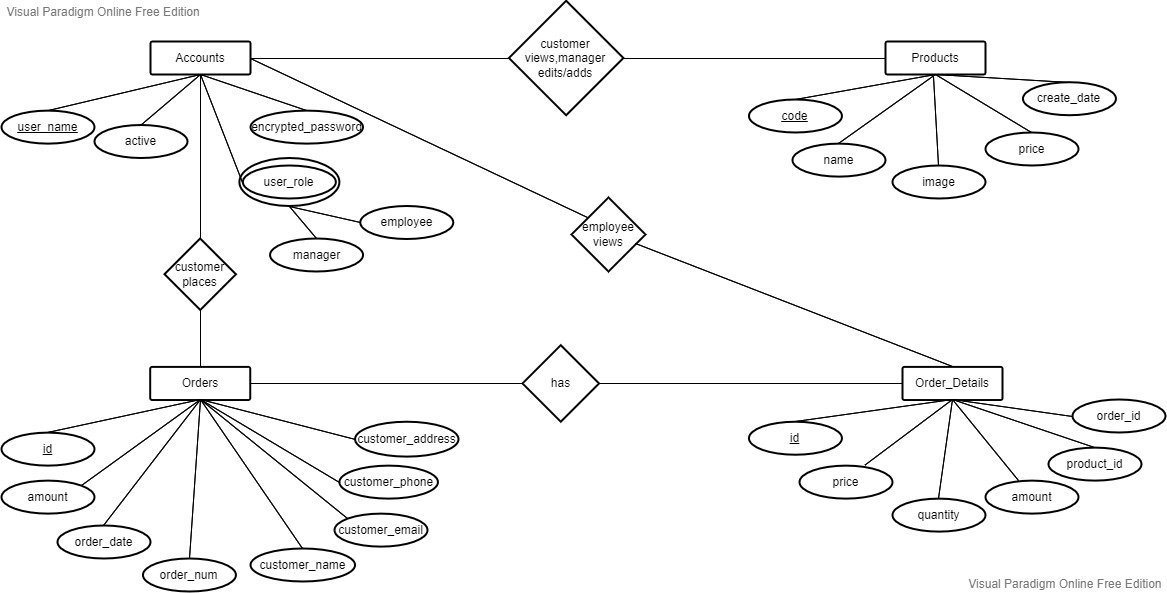
UML CLASS:

**Class** diagrams are the blueprints of your system or subsystem.



ER DIAGRAM:

**ER** diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.



FUNCTIONAL REQUIREMENT:

This subsection presents the identified functional requirements for the subject Billing System. Initially, general requirements that pertain to the whole system are given.

# Customer

[Table](#_bookmark0) presents the identified functional customer requirements that directly relate to the customers of the Billing System.

# Manager Role:

|  |  |
| --- | --- |
| **REQUIREMENT** | **DESCRIPTION** |
| 01 | A manager has to login to the application |
| 02 | A manager shall be able to view the list of options like, Product List, View the order and update products |
| 03 | A manager shall be able to edit the existing products |
| 04 | A manager shall be able to update the new products |
| 05 | A manager shall be able to remove the products from the cart |
| 06 | A manager shall be able to view the order List |
| 07 | A manager shall be able to view the Generated Bill of the customer |

# Employee Role:

|  |  |
| --- | --- |
| **REQUIREMENT** | **DESCRIPTION** |
| 01 | An employee has to login to the application |
| 02 | An employee shall be able to view the option like View the order |
| 03 | An employee shall be able to view the order List |
| 04 | A manager shall be able to view the Generated Bill of the customer |

# Customer Role:

|  |  |
| --- | --- |
| **REQUIREMENT** | **DESCRIPTION** |
| 01 | A customer able to view the product page |
| 02 | A customer able to add the products to the cart |
| 03 | A customer shall be able to view the cart and items |
| 04 | A customer shall be able to payment after updating his information |
| 05 | A customer shall be able to view the order details |
| 06 | A customer shall be able to view the Bill |

USECASE DESCRIPTIONS:

Table presents the Login use case description to show the interaction of Actors when logging into the system.

|  |  |
| --- | --- |
| Use Case | Log In |
| Primary Actor | Manager, Employee |
| Goal In Context | Enable Actors access to the system through an app |
| Preconditions | The Actors must provide a valid username and password to get logged in |
| Scenario | 1. The Actors selects ‘Log-In’ from the app menu 2. The app prompts the user for their username and password 3. The Actors enters their username and password 4. The app enables access to the system according to access control |

**Table: Log In Use Case Description**

Table presents the manager functionality use case description to show the interaction of Actors when viewed the system.

|  |  |
| --- | --- |
| Use Case | Updating and Bill |
| Primary Actor | Manager |
| Goal In Context | Enable Actors access to the functionality through an app |
| Preconditions | The Actor must be able to access the actions available |
| Scenario | 1)The manager able to edit the product list on selecting update product  2)The manager able to add the product to the list  3)The manager able to view the order List of the customer |

**Table: Manager Use Case Description**

Table presents the employee functionality use case description to show the interaction of Actors when viewed the system.

|  |  |
| --- | --- |
| Use Case | Checking the Products purchased |
| Primary Actor | Employee |
| Goal In Context | Enable Actors to view the orders |
| Preconditions | The Actor must be able to access the actions available |
| Scenario | 1. The employee must be able to view the order list 2. The employee shall be able to view the order details of the particular custokmer |

**Table: Employee Use Case Description**

Table presents the products functionality use case description to show the interaction of Actors to view the products

|  |  |
| --- | --- |
| Use Case | PRODUCTS LIST |
| Primary Actor | Customer |
| Goal In Context | Enable Actors to view the products listed |
| Preconditions | The customer has been set up the applications |
| Scenario | 1)The customer brings up a products list  2)The customer selects 'Products' from the List   1. The customer selects 'Add to Cart’ from the Listing page |

**Table: Products List Use Case Description**

Table presents the’ Cart ‘ use case description to show the interaction of Actors to view the products that are been selected from the List

|  |  |
| --- | --- |
| Use Case | Cart |
| Primary Actor | Customer |
| Goal In Context | Enable Actors to view the selected products from the products list. |
| Preconditions | A customer has selected the products to the cart |
| Scenario | 1. The customer brings up a products list in the cart 2. The customer can increase or decrease the quantity of 'Products' from the List 3. The customer selects ‘checkout’ to fill the details. |

**Table: Cart Use Case Description**

Table presents the ’ Checkout‘ use case description to show the interaction of Actors to add the details

|  |  |
| --- | --- |
| Use Case | Checkout |
| Primary Actor | Customer |
| Goal In Context | Enable Actors to update the details |
| Preconditions | A customer has selected the checkout |
| Scenario | 1)The customer shall be filling the require details  2)The customer shall be submit the details |

**Table: Checkout Use Case Description**

Table presents the ‘Pay now’ use case description to show the interaction of Actors to pay the bill .

|  |  |
| --- | --- |
| Use Case | Pay now |
| Primary Actor | Customer |
| Goal In Context | Pay the bill |
| Preconditions | A customer has selected the Pay now to pay the Bill |
| Scenario | 1. The customer selects the payment mode 2. The customer fills in the payment details to pay. |

**Table: Pay Now Use Case Description**

Table presents the ‘Generate Bill’ use case description to show the interaction of Actors to view the generated bill .

|  |  |
| --- | --- |
| Use Case | Bill |
| Primary Actor | Customer |
| Goal In Context | Generate Bill |
| Preconditions | A customer has selected the Bill after payment |
| Scenario | 1. The customer shall the order details 2. The customer shall selects and view the details of the products |

**Table: Bill Generation Use Case Description**

WORKING PROCEDURE:

Steps : -

1. Customer will enter Supermarket and take Products in trolley which they want to buy and will open Supermarket’s App.

2. On clicking product listing Customer will be taken to Product Page which has list of Products, and they will add Products to Cart which they have taken in trolley by clicking add to cart button below each Product

3. After adding Products, Customer will click on Cart button which will show Products added in cart.

4. After checking Cart, Customer will click on Checkout Section button and fill the customer’s details

5. After checking Cart, Customer will click on Payment Section button and payment page will appear.

6. There the customer can also edit the cart and customer information

7. Customer can pay either using Card.

8.After entering credentials, Customer will click on Pay Now button.

9.After Successful completion of payment, the Order number has generated.

10.The customer can view the bill details and view the order detail of that particular bill

11.As a manager he can login and has the access to update the existing product and add new products .

12.Manager can add the new products with adding the respective images and details.

13.Manager can view the order details of the customers.

14.Employee can view the order details of the customer login through his credentials.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | 1 | **Test Case Description** | Test the Functionalities in Supermarket Application | | |
| **Created By** | Team 3 | **Reviewed By** |  | **Version** |  |
|  | | | | | |
| **QA Tester’s Log** |  |  | | | |
|  | | | | | |
|  | | | | | |
| **Tester's Name** | Team 3 | **Date Tested** | 29-11-2022 | **Test Case (Pass/Fail/Not Executed)** | Pass |

TEST CASES:

|  |
| --- |
|  |
|  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | | |
| **Test Scenario 01:** | To test home page | |  |  |  |  |  | | |
|  |  |  |  |  |  |  |  | | |
| **Step #** | **Step Details** | | **Expected Results** | | Actual Results | **Pass / Fail / Not executed / Suspended** | |
|  |
| 1 | Enter http://localhost:8080 | | Home page should open | | Home Page has successfully opened | Pass | |  |
| 2 | On click of Product List | | Has to navigate to Product page | | Product page has opened successfully | Pass | |  |
| 3 | Click on Login | | Customer should be able to navigate to Login page | | Customer has been navigated to the Login page successfully | Pass | |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 02:** | To verify the Login for employee and Manager | | |  |  |  |  | |
|  |  |  |  |  |  |  |  | |
| **Step #** | **Step Details** | | **Expected Results** | | Actual Results | **Expected Results** | |
|  | |
| 1 | Navigate to http://localhost:8080/login | | Login page should open | | sucessfully done | Pass | |  | |
| 2 | Enter the Employee's username and password | | Should be able to enter the credentials | | sucessfully done | Pass | |  | |
| 3 | Enter the Employee's username and password | | Should be able to enter the credentials | | sucessfully done | Pass | |  | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 03:** | To verify ther manager Functionality | | |  |  |  | |  |
|  |  |  |  |  |  |  | |  |
| **Step #** | **Step Details** | | **Expected Results** | | Actual Results | **Expected Results** |
|  | | |
| 1 | on click Product List | | On clicking the product listing the manager has to able to view theproduct list | | Succesfully done | Pass |  | | |
| 2 | on click Product List has to edit | | On clicking the edit the manager has to edit the products with the details | | Successfully done | Pass |  | | |
| 3 | On click the order | | On clicking the view the order details he has to view the order list | | Successfully done | Pass |  | | |
| 4 | On Click the view | | On clicking the view he has to view that particular customers details | | Successfully done | Pass |  | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 04:** | To verify the Employee functionality | | |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | Actual Results | **Expected Results** | |
|  |
| 1 | Login using Wrong credentials and correct credentials | | Should be denied if wromng credentials else gets logged in. | | sucessfully done | Pass | |  |
| 2 | On click the order | | On clicking the view the order details he has to view the order list | | sucessfully done | Pass | |  |
| 3 | On Click the view | | On clicking the view he has to view that particular customers details | | sucessfully done | Pass | |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 05:** | To verify the functionality of the products listing in the home page | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | **Pass / Fail / Not executed / Suspended** | | |
|  |
| 1 | Listing of all Products | | Products should be listed according to their categories along with the description and price | | Successfully Done | | Pass | | |  |
| 2 | User has to add items to the cart | | Products should be added to the cart | | Successfully Done | | Pass | | |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 06:** | To verify the functionality of the cart | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | **Pass / Fail / Not executed / Suspended** | | |
|  |
| 1 | Adding products to the cart | | The selected products should be added to the cart based on the customers prefernce | | Successfully Done | | Pass | | |  |
| 2 | Click on Cart | | The cart items should be displayed along with the quantity of the products added | | Successfully Done | | Pass | | |  |
| 3 | Enter the required Quantity | | The quantity of the selected product in the cart should be increased or decreased based on the needs | | Successfully Done | | Pass | | |  |
| 4 | Update Quantity | | On nclick the quantity has to be addded and the price has to be modified | | Successfully Done | | Pass | | |  |
| 5 | Delete | | On Clicking the Delete the item has to be deleted | | Successfully Done | | Pass | | |  |
| 6 | Continue Shopping | | On clicking continue shoppping it has to return the producyt list | | Successfully Done | | Pass | | |  |
| 7 | Checkout | | On clicking checkout user detail page has tom be viewed | | Successfully Done | | Pass | | |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 07:** | To verify the functionality of Customer info page | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | **Pass / Fail / Not executed / Suspended** | | |
|  |
| 1 | Enter Name | | Enter name if name is not there it should show exception | | Both are Successfully Done | | Pass | | |  |
| 2 | Enter Email | | Enter emailId, if emailid is not there it should show exception | | Both are Successfully Done | | Pass | | |  |
| 3 | Enter the Phone | | Enter phone number, if phone number is not there it should show exception | | Both are Successfully Done | | Pass | | |  |
| 4 | Enter the Address | | Enter phone number, if phone number is not there it should show exception | | Both are Successfully Done | | Pass | | |  |
| 5 | Reset | | On Clicking the reset the details has to be cleared | | Successfully Done | | Pass | | |  |
| 6 | Submit | | On clicking Submit has to view the cart summary page | | Successfully Done | | Pass | | |  |

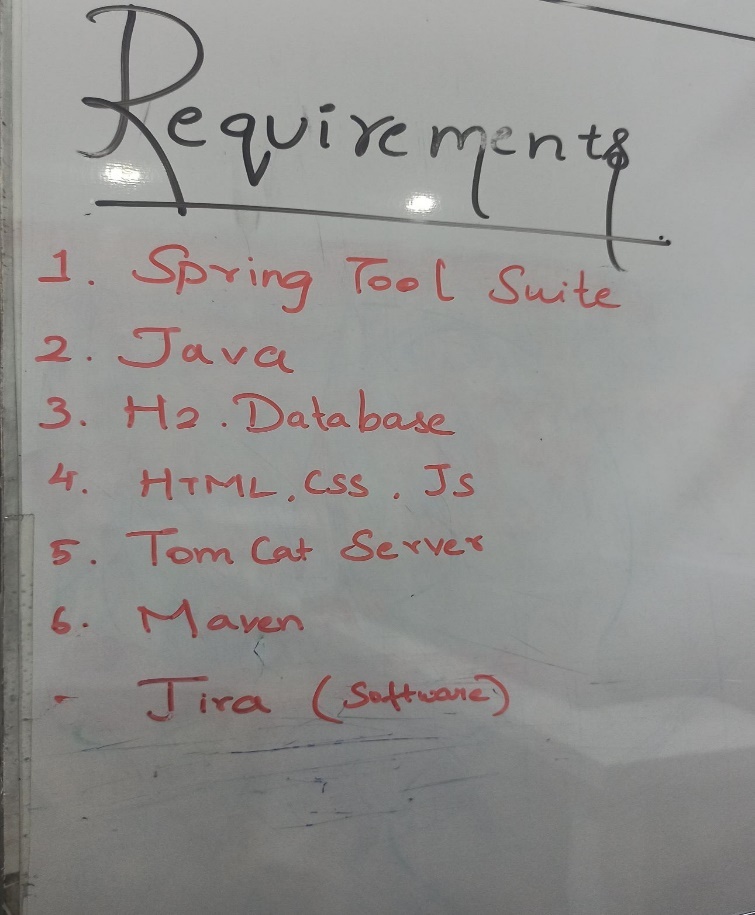
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 08:** | To verify the functionality of Cart Summary Page | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | **Pass / Fail / Not executed / Suspended** | | |
|  |
| 1 | Cart Summary Page | | The cart summary page has the Customer info and the cart info | | Successfully Done | | Pass | | |  |
| 2 | Edit Cart | | On click of edit cart has to be able to the edit cart items | | Successfully Done | | Pass | | |  |
| 3 | Edit customer Info | | On click of edit customer info has to be able to the edit customer info | | Successfully Done | | Pass | | |  |
| 4 | Payment | | Payment option has to be viwed on the page | | Successfully Done | | Pass | | |  |
| 5 | Payment Details | | Enter the payment details | | Successfully Done | | Pass | | |  |
| 6 | Pay | | On clicking Pay It has to be redirected | | Successfully Done | | Pass | | |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 09:** | To verify the Cart Finalize functionality | | |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | Actual Results | **Expected Results** | |
|  |
| 1 | Cart Finalize Page | | The cart finalize page has the bill number for the order placed | | sucessfully done | Pass | |  |
| 2 | On click the Bill | | On clicking the Bill it has redirect to the bill page | | sucessfully done | Pass | |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 10:** | To verify the Bill functionality | |  |  |  |  |  | |  |  |
|  |  |  |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | Actual Results | **Expected Results** | | |  |  |
|  |  |
| 1 | Bill Page | | The Bill page has to show the oreder details of the particular customer | | sucessfully done | Pass | | |  |  |
| 2 | On click view | | On clicking the view it has to show the details of that particular order id | | sucessfully done | Pass | | |  |  |
|  |  |  |  |  |  |  |  | |  |  |
| **Test Scenario 11:** | To verify the functionality of the Logout option in the home page | | | |  |  |  | |  |  |
|  |  |  |  |  |  |  |  | |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | **Pass / Fail / Not executed / Suspended** |
|  | | | |
| 1 | Click On LogOut | | The Customer should be able Logout from the application | | Successfully Done | | Pass |  | | | |

WORKFLOW:

**Requirements:**

****

**UserStories:**

Text, letter

Description automatically generated

Text, letter

Description automatically generated

Text, letter

Description automatically generated

Text, letter

Description automatically generated

Text, letter

Description automatically generated

Text, letter

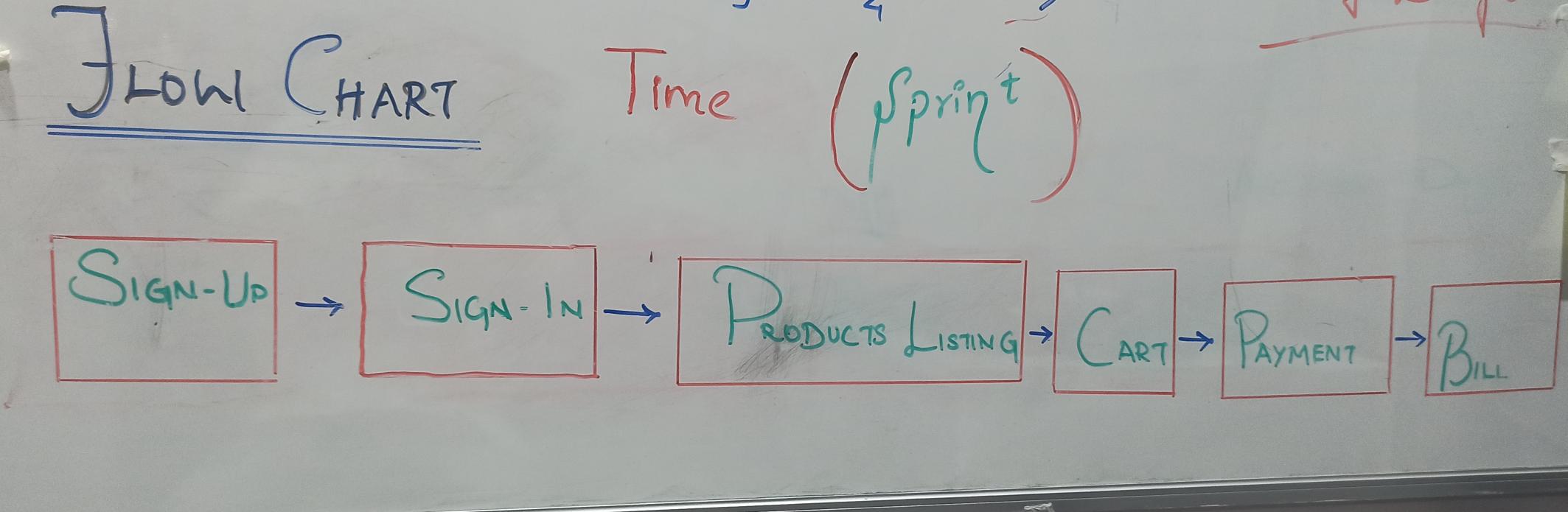
Description automatically generated

Text, letter

Description automatically generatedText, letter

Description automatically generated

**Flow Chart:**



**Task Board:**

Text

Description automatically generated

**Burn Down :**

A white board with writing on it

Description automatically generated

A white board with writing on it

Description automatically generated

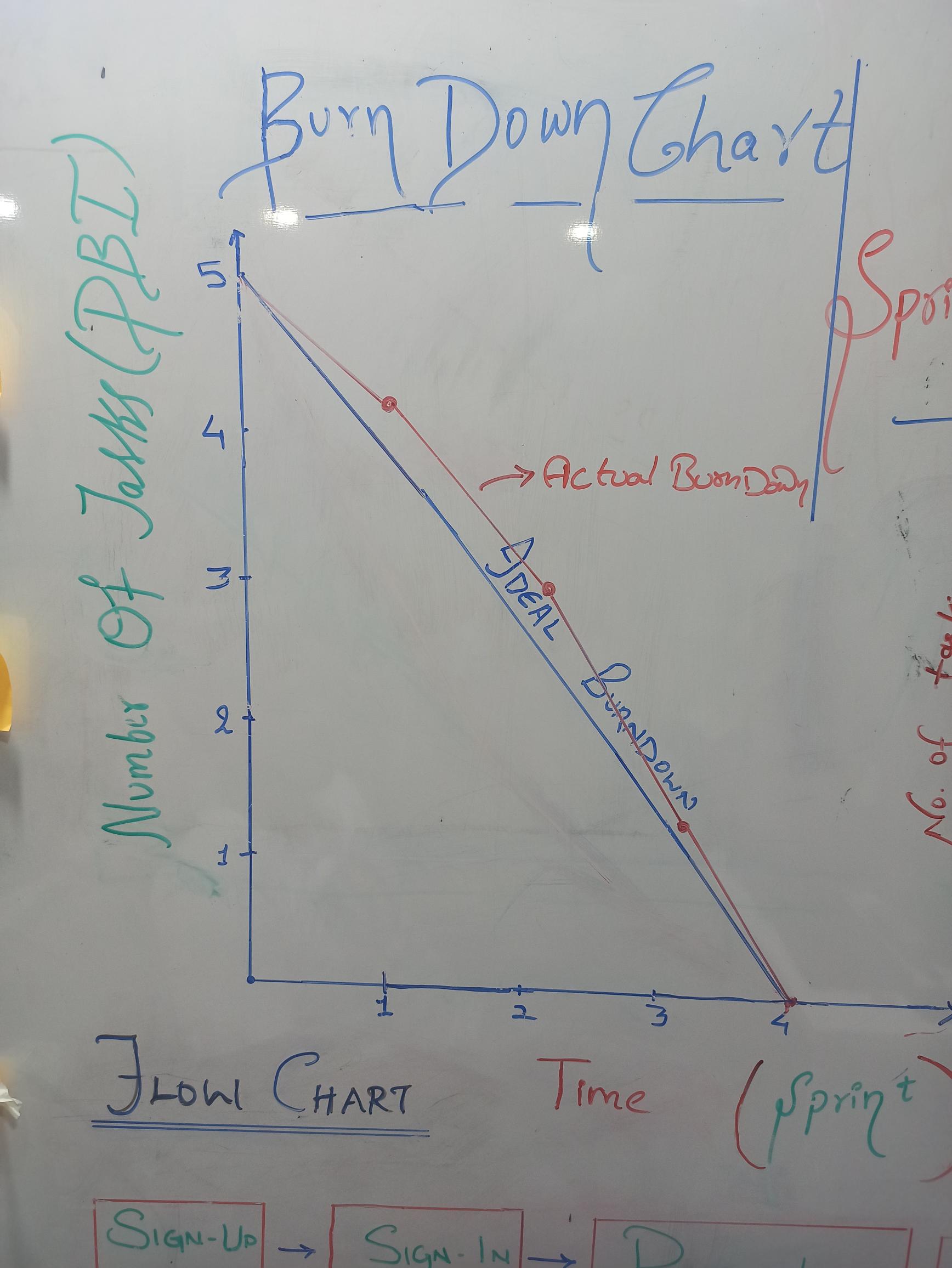
A white board with writing on it

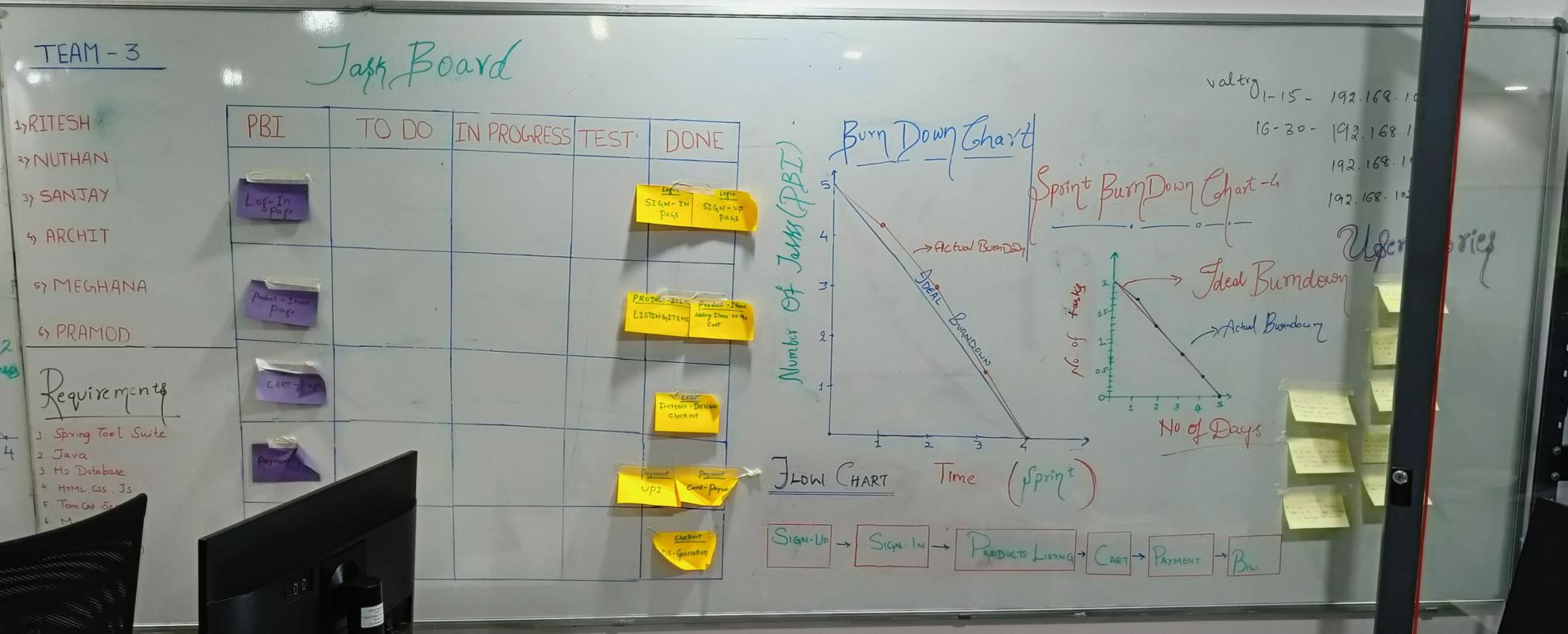
Description automatically generated

A white board with writing on it

Description automatically generated

**Burn Down Chart**





Text, letter

Description automatically generated