

Practical 1

Aim :To Select the project title and assign requirement engineering to the project title.

➤ **Project Profile:**

Project Definition	Cafeteria Ordering System
Objective	Main Objective of this system is to view and provide payment and delivery and discounts on all the Café products.
Front-end	WordPress
Back-end	MySQL

➤ **Requirement Engineering:**

- This Website provides all the items of café shop which you can easily filter out. Then, you can easily select which can help in faster delivery.
- You can see all the Discounts and Special offers in all the cafes and then can select your order based on your interest.

Practical 2

Aim: To perform the system analysis: Requirement analysis, SRS.

SRS:

➤ Problem Definition:

- To allow the probable customers to make search and match the items available in the café shop and make order for that.
- It will help us to find the required food items with its most efficient time to get delivered faster.

➤ Purpose:

- Cafeteria Ordering System is designed & suitable for coffee shop and cafes either operating as individual shop or part of any bigger shops. It covers all process from showing all the items which are currently available and showing current discounts and offers available on that, giving customer all the payment options and guarantees the faster delivery to any part of the city.
- Anybody can use this website from local resident of that city to non residents as well who want to deliver that order to their mentioned address.

➤ Scope:

- The scope of the specification includes the following scenarios:
 1. The Cafeteria Ordering System will permit Customers to order meals from the cafeteria online to be delivered to specified locations.
 2. Big Order for special functions and events (for example - office party, birthday party, housewarming party, baby shower, weddings, etc.).
 3. Emergency delivery of food.
 4. Management of payment and refund of returned or wrong delivery of order.

➤ **User Requirements:**

There are two users involved in this project of Cafeteria Ordering System (COS) and they are

- User
- Administrator

● **User**

1. To be able to view the café food items and drink items, including where and when they will be delivered, and the results for each, to learn of their total cost.
2. To be able to view and update their personal information, including name, contact address, and phone number, to keep their information record up-to-date.
3. To be notified of the purchase of their order, to know the success of their delivery.
4. To be able to give feedback through the website so that the administrator can evaluate and improve the current existing system.

● **Administrator**

1. To be able to update, delete, category and items in order to manage information.
2. To be able to create, update, delete, and retrieve records to manage information about deliveries made.
3. To be able to deposit money into account when payments are made.
4. To be able to withdraw money from the account.
5. To be able to keep a record of available items to always keep count of the available menu items of cafe.
6. To be able to create, update, delete, and query records in order to manage database information.
7. To be able to send confirmations to customers for orders through the system.

8. To be able to respond to customer feedback about website and delivery of orders.

9. To be able to update the existing system and software for upgrading it to latest software.

➤ Functional Requirements

-This section provides requirements overview of project. This project will be required WORDPRESS and PHP as a frontend and the backend MySQL database will be running.

ID	Title & Description
FR1	<p>Title: Login</p> <p>Desc: The system provides security features through username-password matching where only authorized user can access the system with different authorization level.</p>
FR2	<p>Title: Admin</p> <p>Desc: Input:-Username, Password Output: - Invalid or Update User Details, logout</p>
FR3	<p>Title: Customer Profile Registration</p> <p>Desc: This allows healthy public to register as customer. Input:- Customer Id, Name, Date of Birth, Sex, Address, Contact Number, Email Address ,Aadhar Card No. Output: - Successfully Registered.</p>
FR4	<p>Title: Order and Delivery Management</p> <p>Desc: The administrator can manage the available items database collection, to items screening, processing,viewing, storage and updation through his system. Each process or work-flow can be traced from the database. The system will also raise alert to the administrator whenever the item quantity is below its par level or when the item from the menu is finished.</p>

FR5	<p>Title: Customer Management</p> <p>Desc: The records of all customers and their history are kept in one centralized database and thus reducing duplicate data in the database. The record of payments is maintained by the system.</p> <p>Input:-Order and Payment Type</p> <p>Output:-No. of searched items Available</p>
FR6	<p>Title: Report</p> <p>Desc: The system is able to generate pre-defined reports such as the list of Customers, orders placed, orders successfully delivered, Total revenue generated, the food quantity available in the cafe and charts.</p> <p>Input:-Admin Username, Admin Password</p>

➤ Non Functional Requirements

- **Security** : The security can be achieved in this Cafeteria Ordering System (COS) by validating that only registered customers are able to place and receive the order.
- **Logging** : The authenticated user or customers by the administrator must be able to login and place and receive the order after the registration only and verified by admin of the website.
- **Storage** : The customer information must be stored and maintained in database so that can be record can be easily retrieve whenever necessary. The database must be updated with each old and new record and maintained by the administrator.
- **Configuration**: The website must be arranged with all elements required and must contain various diagrams such as data flow, case , class diagrams etc.
- **Performance** : The website must be optimized in the performance and also the website must appear flexible a per the user needs.
- **Cost** : Reasonable amount of time and money is invested in this.
- **Disaster recovery** : If the failure of the database or error occurs than proper plan must be implemented by administrator to cop up with such problems.

- **Accessibility** : Only the administrator must be allowed to access or check required data from data base also the access permission must be allotted to other entities or elements of system by the administrator only. Hence the main access of the system must be handled by administrator.

This system will present an alert system to the selected café shop about requirement of the order to a person need and will also provide online status of item wise availability of a given item.

➤ **Hardware and software Requirements:**

For the Cafeteria Ordering System (COS) the estimated functional requirements include the software and hardware requirements as below:

Software Requirements:

- Operating System : Windows Xp, Windows 7/8/10 ,ubuntu- linux OS
- Front End : Online website making using Wordpress
- Back End : My Sql
- Technology : xampp Server

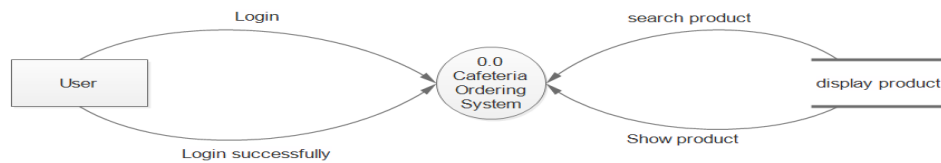
Hardware Requirements:

- Processor: Inbuilt in Wordpress
- Hard Disk : 5 GB
- Memory : 1GB RAM

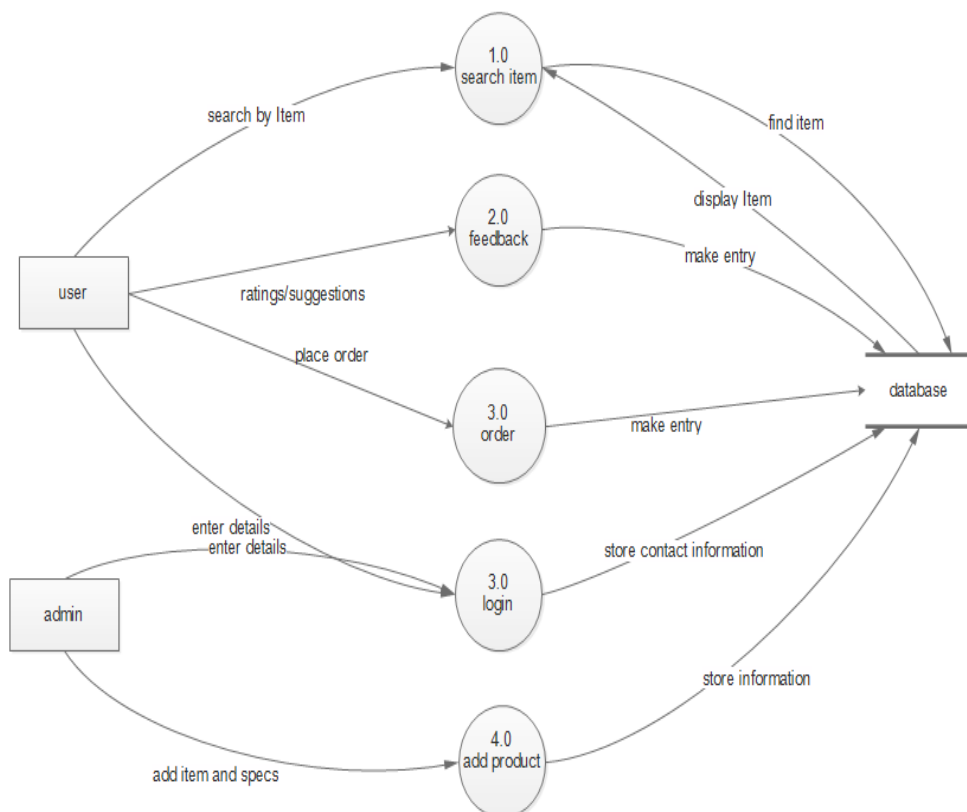
Practical 3

Aim:To perform function oriented diagram: DFD and Structured chart.

DFD 0:

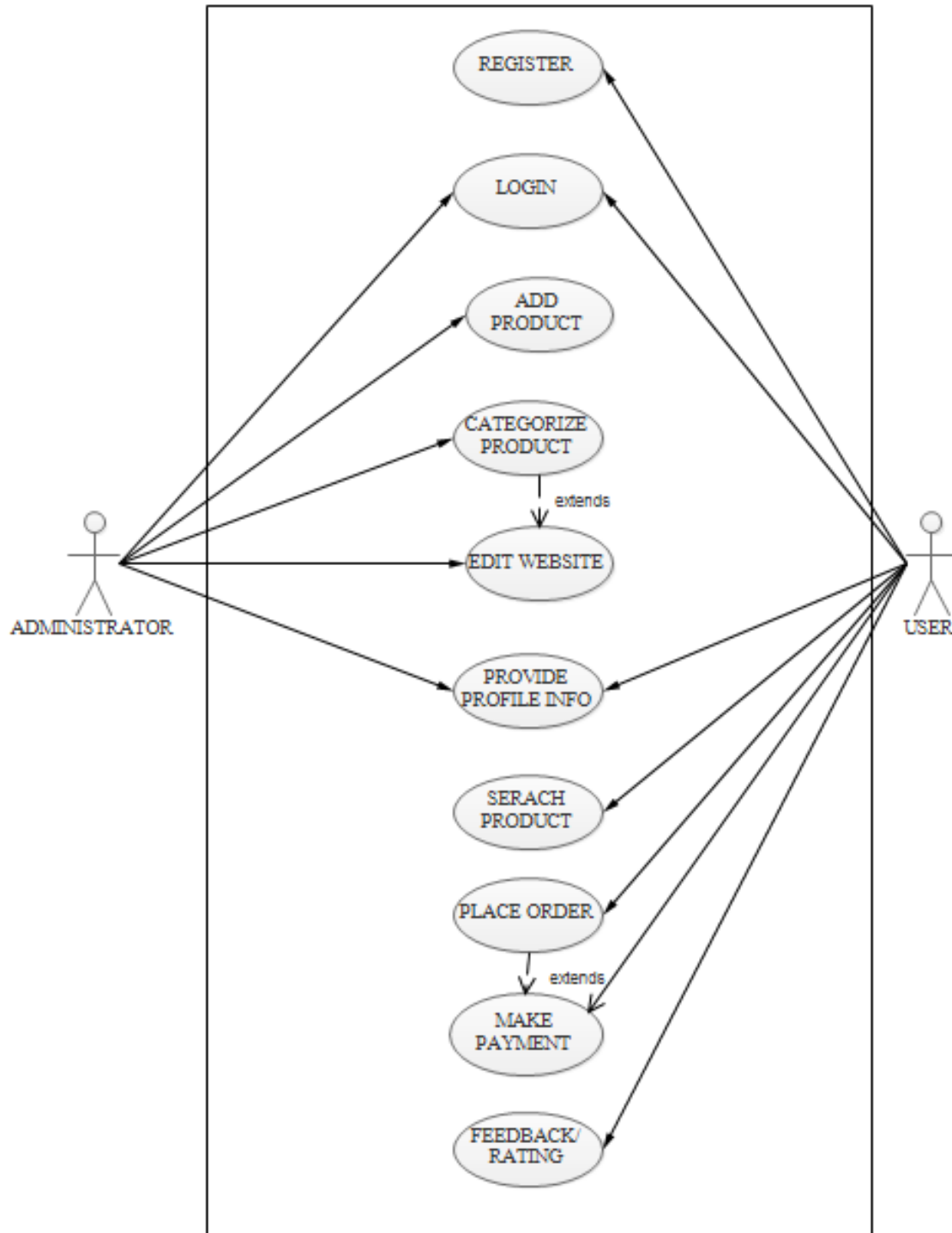


DFD 1:



Practical 4

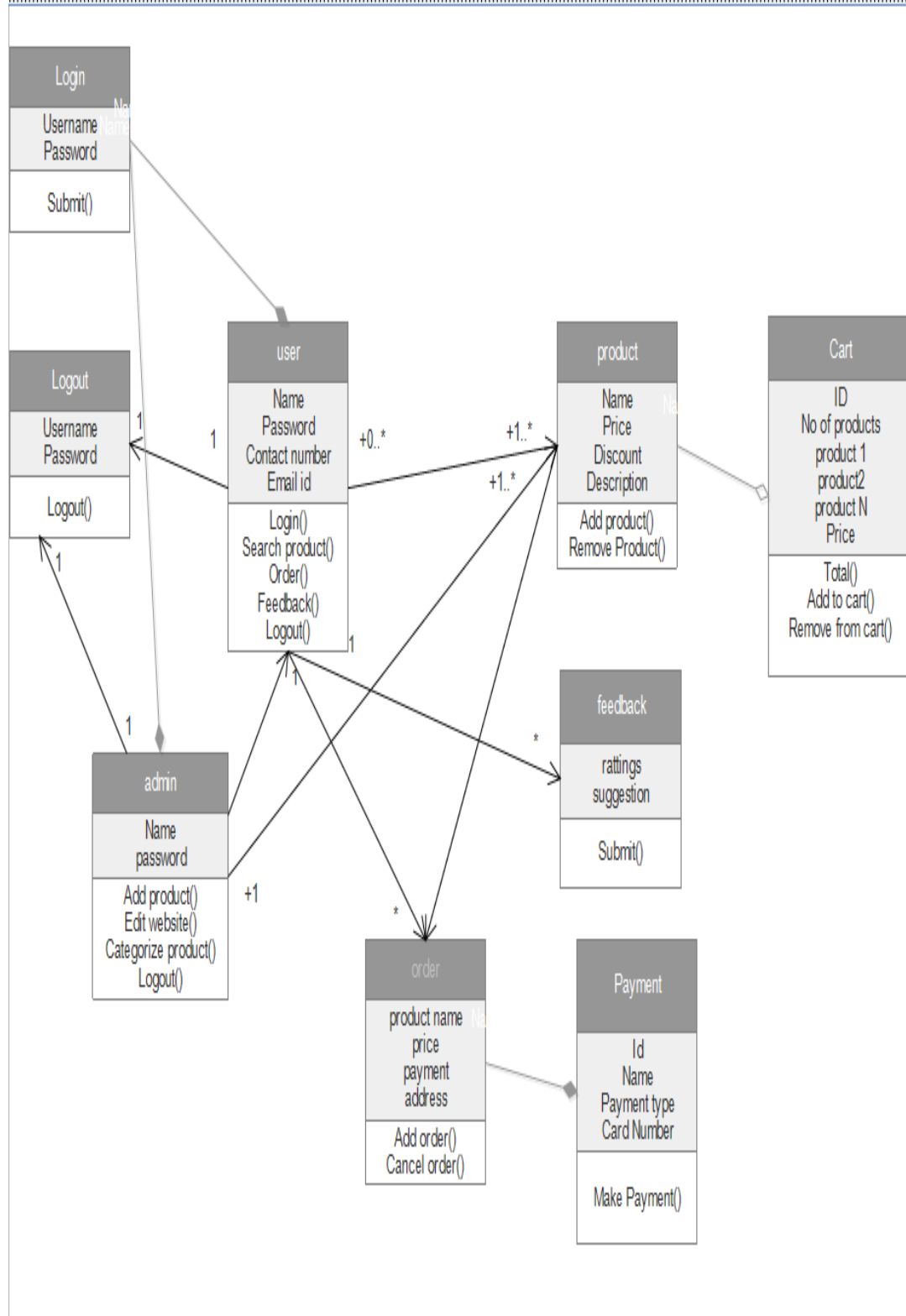
Aim: To perform the user's view analysis: Use case diagram.



Practical 5

Aim:To draw the structural view diagram: Class diagram.

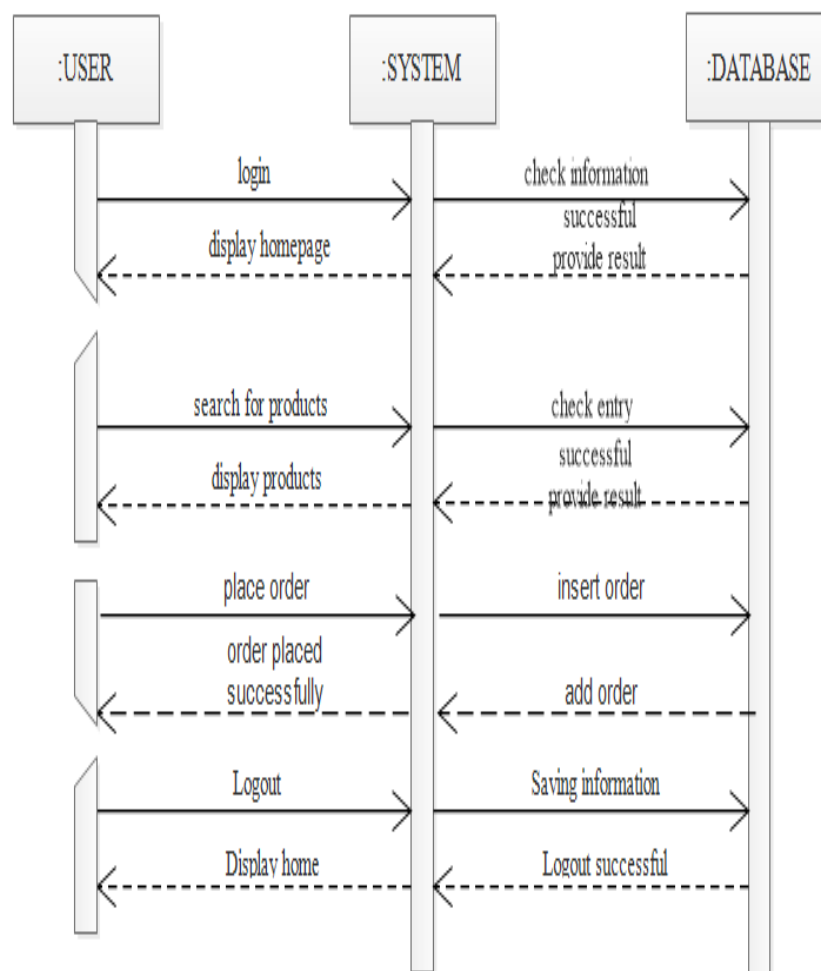
Class Diagram:



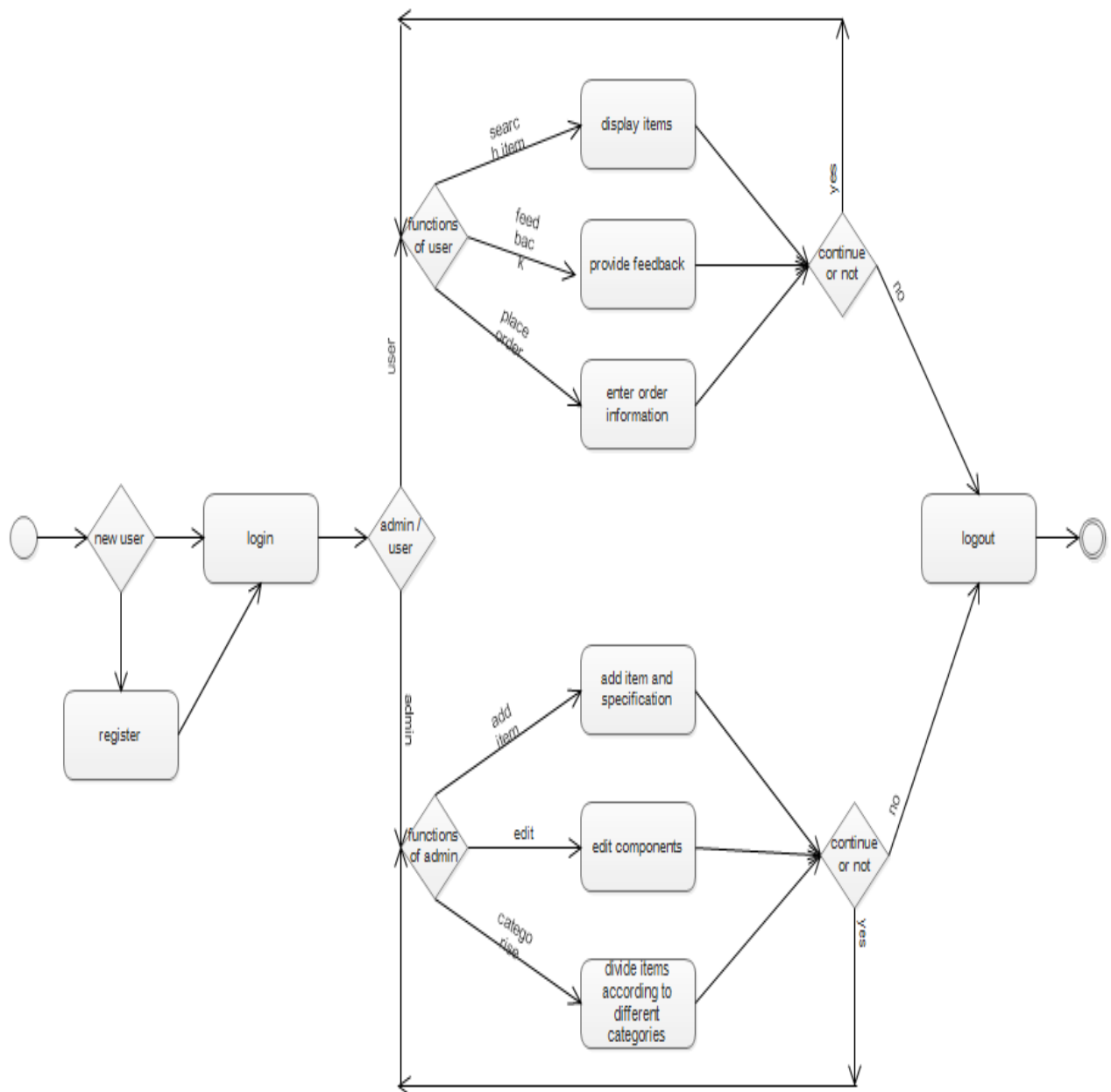
Practical 6

Aim:To draw the behavioural view diagram: Sequence diagram, Activity diagram.

Sequence Diagram:



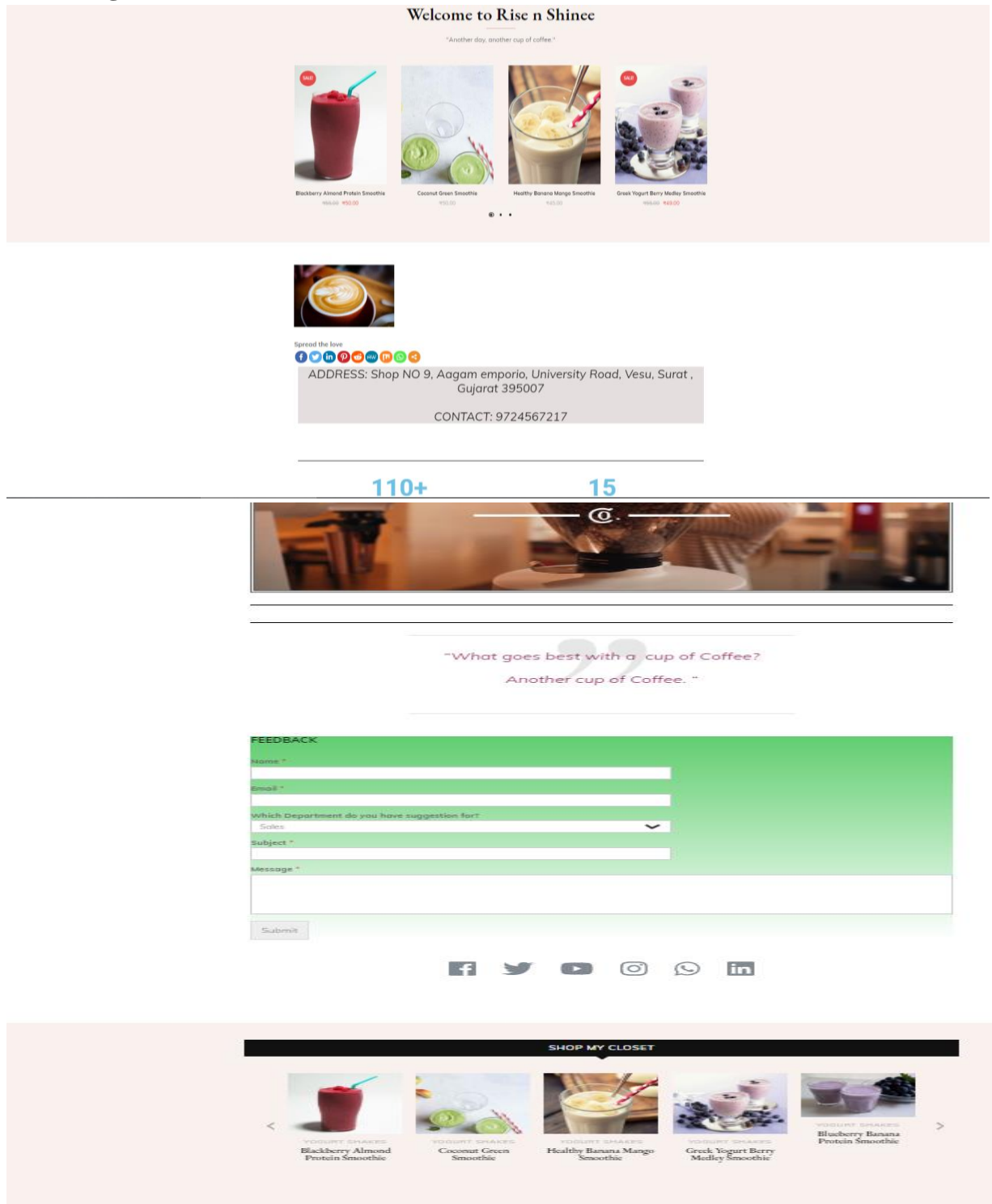
Activity Diagram:

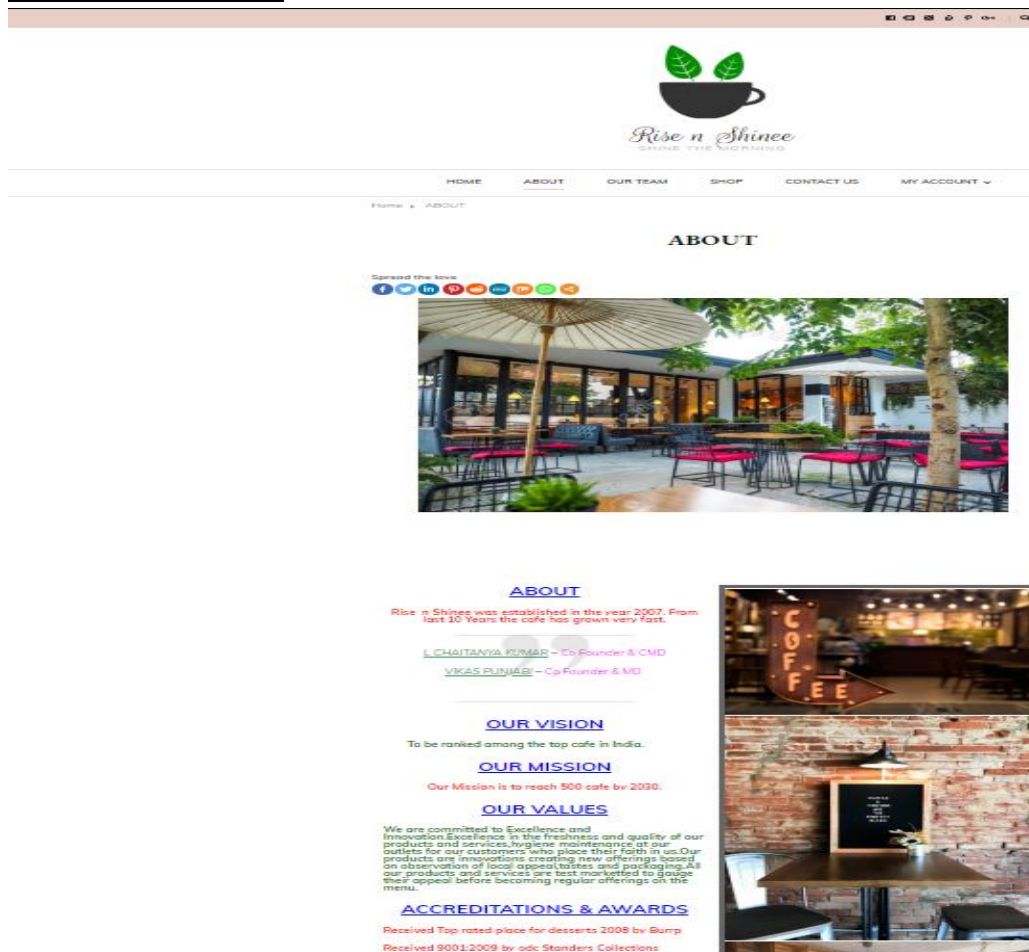
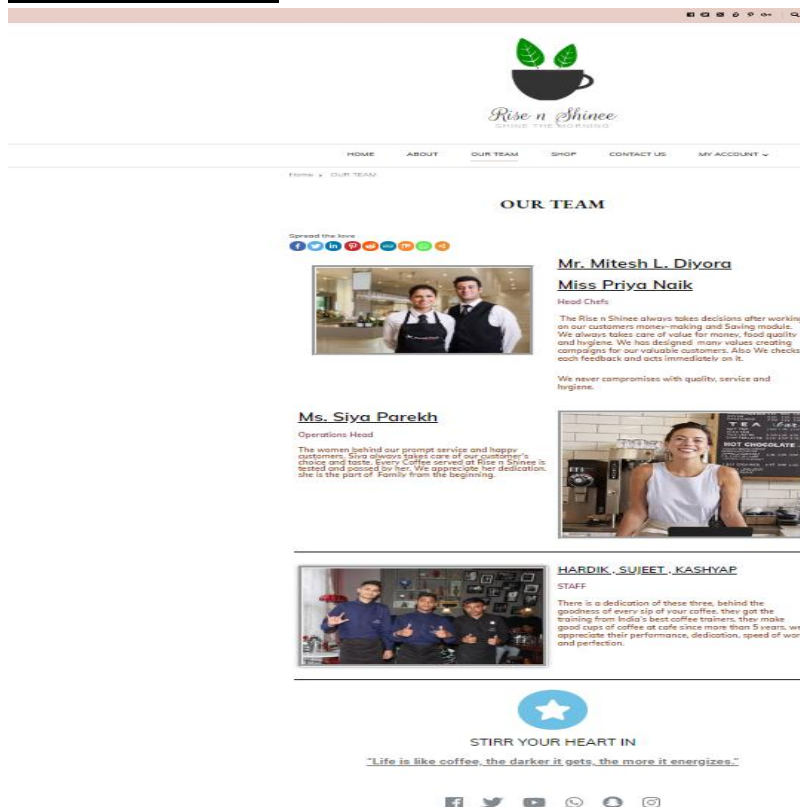


Practical 7

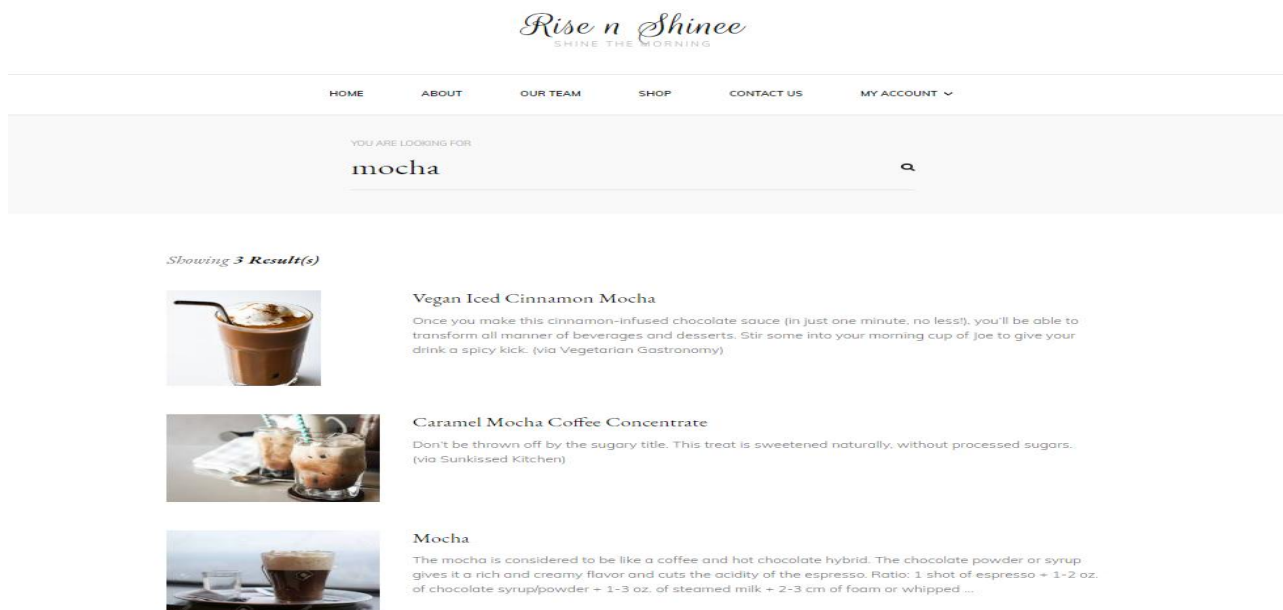
Aim: Implementation of the project.

➤ **Home Page.**



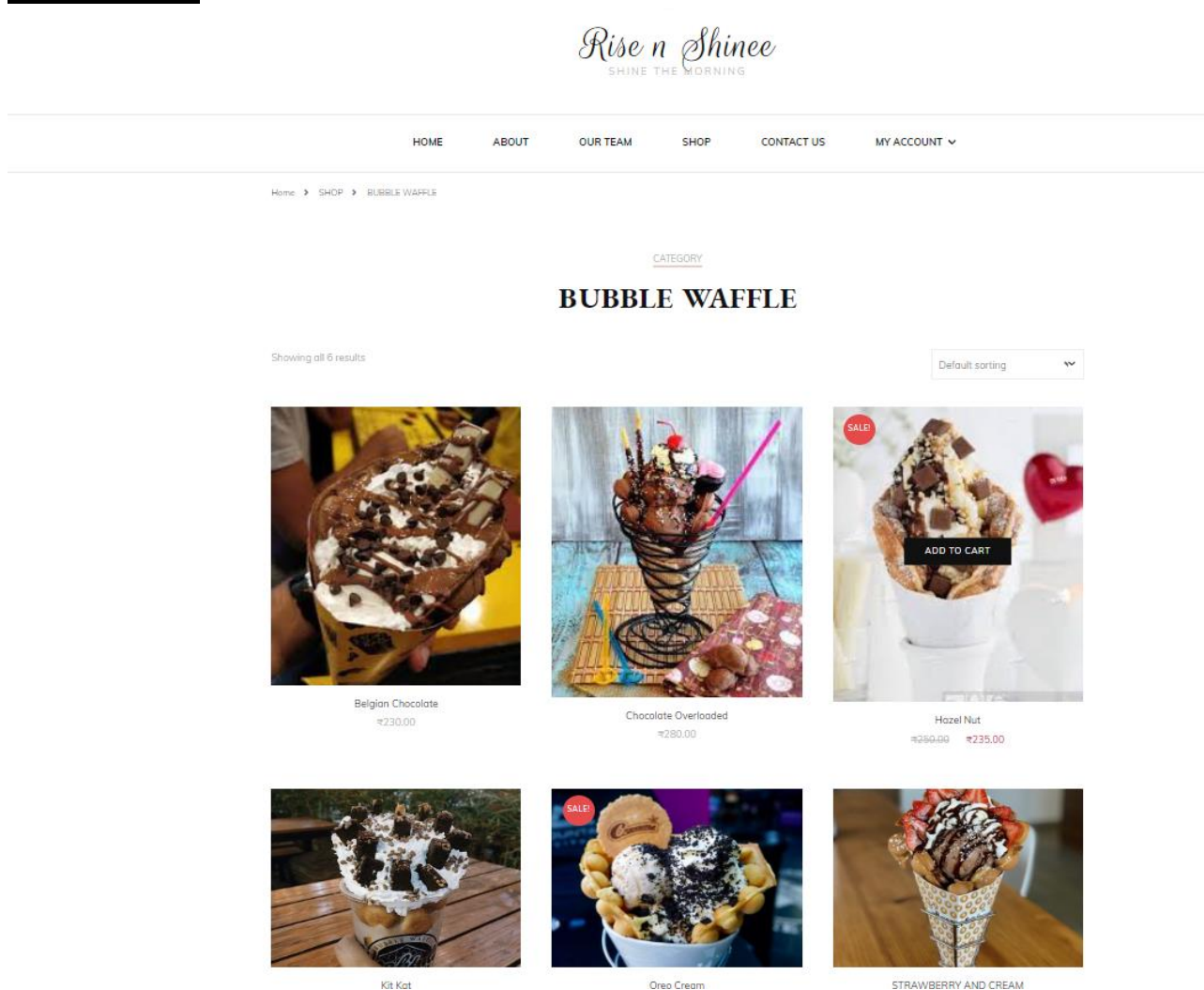
➤ Display About cafe .➤ Display About Staff

➤ Search Different Items based on keywords.

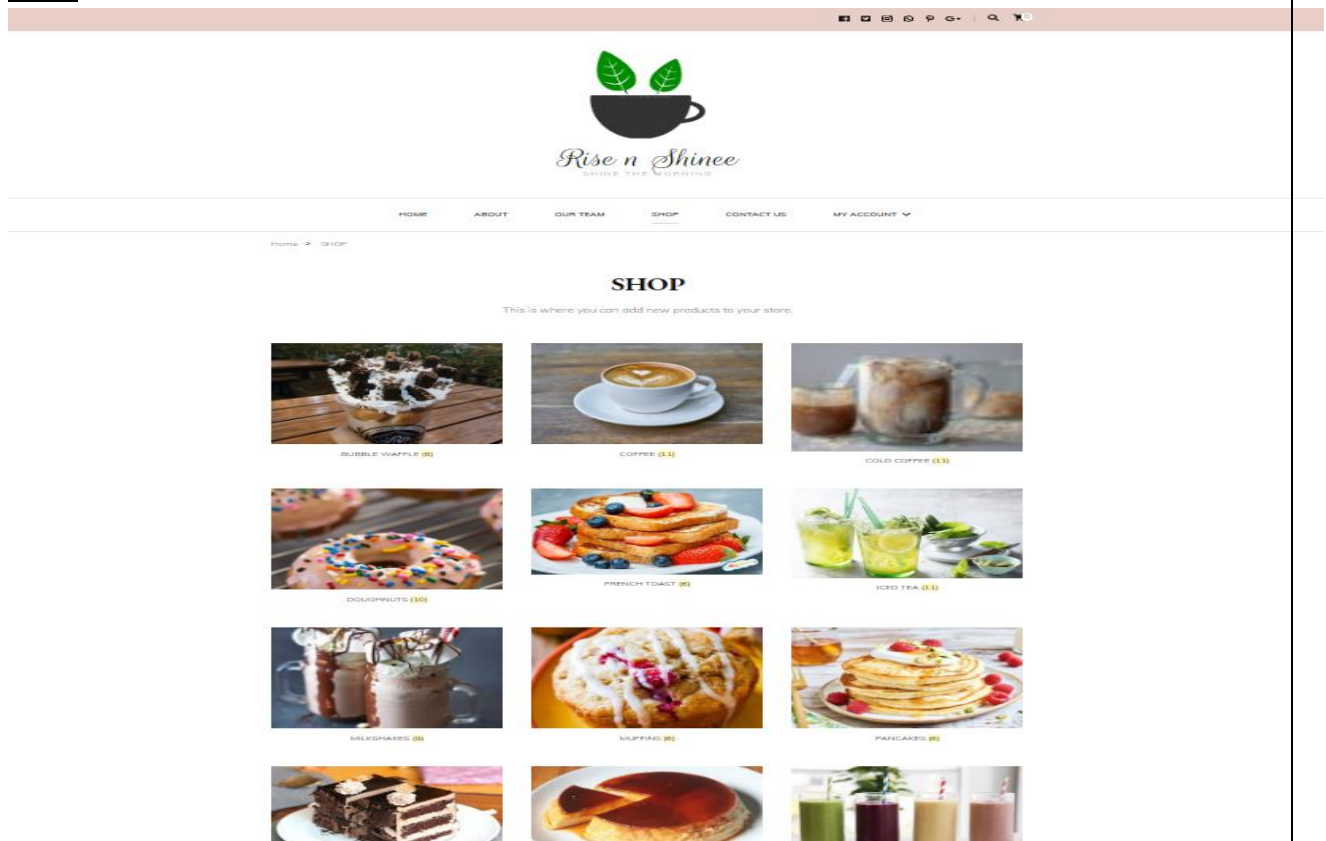


© Copyright 2020 Rise n Shinee. All Rights Reserved. Fashion Diva | Developed By Blossom Themes. Powered by WordPress.

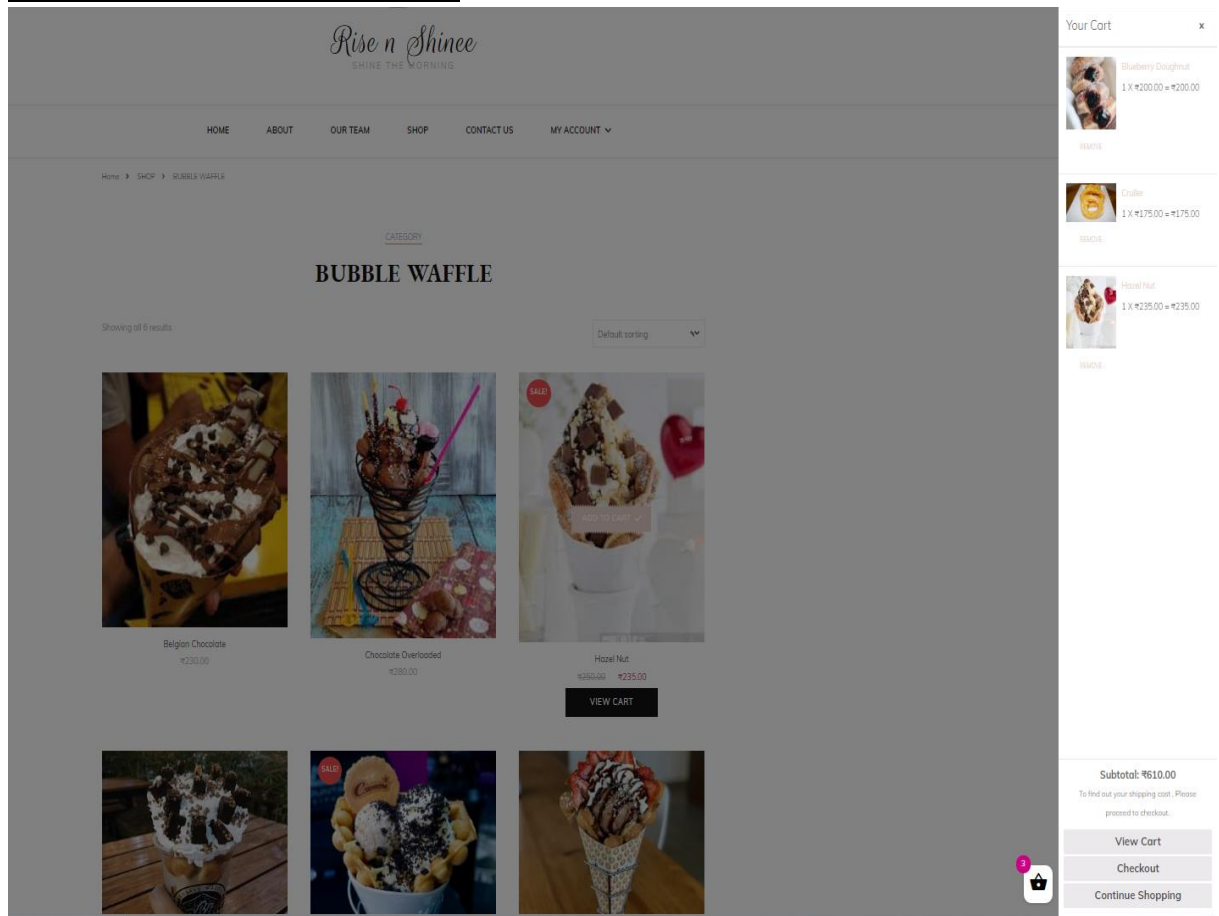
➤ Image Gallery.




➤ Shop



➤ Now Add Different Items to cart.



➤ **Now generating Total Bill.**




SHINE THE MORNING




[HOME](#) [ABOUT](#) [OUR TEAM](#) [SHOP](#) [CONTACT US](#) [MY ACCOUNT ▾](#)

Home > CART

CART

Spread the love



	Product	Price	Quantity	Subtotal
✕	 Blueberry Doughnut	₹200.00	1	₹200.00
✕	 Cruller	₹175.00	1	₹175.00
✕	 Hazel Nut	₹235.00	1	₹235.00

APPLY COUPON


UPDATE CART

Cart totals

Subtotal	₹610.00
Total	₹610.00

PROCEED TO CHECKOUT

➤ **Login**




Rise n Shinee
SHINE THE MORNING


[HOME](#) [ABOUT](#) [OUR TEAM](#) [SHOP](#) [CONTACT US](#) [MY ACCOUNT ▾](#)

Home > Login

Login



Spread the love



Username or E-mail

Password

☐ Keep me signed in

Login

Register

Forgot your password?

Practical 8

Aim: To study various testing tools.

Selenium is a portable [software-testing framework](#) for [web applications](#). Selenium provides a playback (formerly also recording) tool for authoring tests without the need to learn a test [scripting language](#) (Selenium IDE). It also provides a test [domain-specific language](#) (Selenese) to write tests in a number of popular programming languages, including [C#](#), [Groovy](#), [Java](#), [Perl](#), [PHP](#), [Python](#), [Ruby](#) and [Scala](#). The tests can then run against most modern [web browsers](#). Selenium deploys on [Windows](#), [Linux](#), and [macOS](#) platforms. It is [open-source software](#), released under the [Apache 2.0 license](#): web developers can download and use it without charge.

Components

Selenium is composed of several components with each taking on a specific role in aiding the development of web application test automation.

- **Selenium IDE**

Selenium IDE is a complete integrated development environment (IDE) for Selenium tests. It is implemented as a Firefox Add-On, and allows recording, editing, and debugging tests. It was previously known as Selenium Recorder. Selenium-IDE was originally created by Shinya Kasatani and donated to the Selenium project in 2006. It is little-maintained and is compatible with Selenium RC, which was deprecated.^[3]

Scripts may be automatically recorded and edited manually providing autocompletion support and the ability to move commands around quickly. Scripts are recorded in *Selenese*, a special test scripting language for Selenium. Selenese provides commands for performing actions in a browser (click a link, select an option), and for retrieving data from the resulting pages.

The Selenium IDE for Firefox stopped working^[4] after the Firefox 55 upgrade and will be no longer maintained. However users can run Selenium IDE on some older Firefox versions (pre-Firefox 55) or try other alternative solutions.^[5]

- **Selenium client API**

As an alternative to writing tests in Selenese, tests can also be written in various programming languages. These tests then communicate with Selenium by calling methods in the Selenium Client API. Selenium currently provides client APIs for Java, C#, Ruby, JavaScript and Python.

With Selenium 2, a new Client API was introduced (with *WebDriver* as its central component). However, the old API (using class *Selenium*) is still supported.

- **Selenium Web Driver**

Selenium Web Driver is the successor to Selenium RC. Selenium WebDriver accepts commands (sent in Selenese, or via a Client API) and sends them to a browser. This is implemented through a browser-specific browser driver, which sends commands to a browser, and retrieves results. Most browser drivers actually launch and access a browser application (such as Firefox, Chrome, Internet Explorer, Safari, or Microsoft Edge); there is also an HtmlUnit browser driver, which simulates a browser using the headless browser HtmlUnit.

Unlike in Selenium 1, where the Selenium server was necessary to run tests, Selenium WebDriver does not need a special server to execute tests. Instead, the WebDriver directly starts a browser instance and controls it. However, Selenium Grid can be used with WebDriver to execute tests on remote systems (see below). Where possible, WebDriver uses native operating system level functionality rather than browser-based JavaScript commands to drive the browser. This bypasses problems with subtle differences between native and JavaScript commands, including security restrictions.^[6]

In practice, this means that the Selenium 2.0 API has significantly fewer calls than does the Selenium 1.0 API. Where Selenium 1.0 attempted to provide a rich interface for many different browser operations, Selenium 2.0 aims to provide a basic set of building blocks from which developers can create their own Domain Specific Language. One such DSL already exists: the Watir project in the Ruby language has a rich history of good design. Watir-webdriver implements the Watir API as a wrapper for Selenium-Webdriver in Ruby. Watir-webdriver is created entirely automatically, based on the WebDriver specification and the HTML specification.

As of early 2012, Simon Stewart (inventor of WebDriver), who was then with Google and now with Facebook, and David Burns of Mozilla were negotiating with the W3C to make WebDriver an internet standard. In July 2012, the working draft was released. Selenium-Webdriver (Selenium 2.0) is fully implemented and supported in Python, Ruby, Java, and C#.