**CSCE 5430 Software Engineering**

**BON VIVEUR**

**Deliverable 3**

**Phase 1 Development**

Harshitha Mandru

Sravya Bussoli

Sravya Garlapati

Anuradha Vishwapriya

Contents

**[Requirements](#_Toc22591135)** [1](#_Toc22591135)

[**Functional requirement** 2](#_Toc22591136)

[ADMIN 2](#_Toc22591137)

[VENDOR 2](#_Toc22591138)

[CUSTOMER 2](#_Toc22591139)

[**Non-Functional requirements** 3](#_Toc22591140)

[**UML design** 3](#_Toc22591141)

[**Class diagram** 4](#_Toc22591142)

[**Sequence** 4](#_Toc22591143)

[Sequence diagram for Customer Login 4](#_Toc22591144)

[Sequence Diagram for Vendor Login 5](#_Toc22591145)

[Sequence diagram for Customer Registration 6](#_Toc22591146)

[Sequence diagram for Vendor Registration 6](#_Toc22591147)

[Sequence diagram of system 7](#_Toc22591148)

[Use case diagram 8](#_Toc22591149)

[**Test cases** 9](#_Toc22591150)

[**Who wrote what components?** 11](#_Toc22591151)

[**User manual and installation** 11](#_Toc22591152)

[**Test cases scenarios** 13](#_Toc22591153)

[**Feedback of peer review session and actions taken on that feedback.** 16](#_Toc22591154)

# **Requirements**

## **Functional requirement**

ADMIN: Has the centralized authority to the website.

1. Accepts customer orders

Bon Viveur is a website that deals with food orders and delivering it. So, it will be able to accept all the orders made by the customers. A customer can make any number of orders and our website will accept all of them without any issues.

1. Displays all vendor’s information

The website will store the details of any person who registers himself as the vendor. It immediately displays this information so that the customers can view and order from this vendor if they like because we never want to leave a chance to satisfy our customer. So, providing new vendors and dishes will be going on always.

1. Shows payment details

As soon as the order is confirmed and the payment is made, it immediately displays the invoice/receipt so that there will be a proof for customers, vendors and also the admin.

VENDOR: A person who sells the food items

1. Register

Has to enter the details like name, address, phone number and the dish he/she wants to sell.

1. Login

He/she should use the website with the credentials given while registering.

1. Check for new orders

He/she should keep checking if there is any order placed by the customer.

1. Respond

Should give a reply to the customer based on the request made.

1. Update details

They can update their details and change password whenever they want to.

CUSTOMER: A person who places orders

1. Register

Has to enter the details like name, address, phone number and the dish he/she wants to sell.

1. Login

Should use the website with the credentials given while registering.

1. Place an order

Whenever he/she comes across the dish they like, they can immediately place an order.

## **Non-Functional requirements**

1. Usability

It is very easy to use. Any person can easily choose the dishes they like and also make the orders.

1. Performance

It performs any task in less than a second. It allows multiple users to access the website at the same time without downgrading the performance.

1. Maintainability

It is very easy to restore it and this can be done in less time if at all there is a crash because the website also has a backup code designed.

1. Security

It protects the details of the customers and also the banking details. Only admin has the access to view these details.

1. Availability

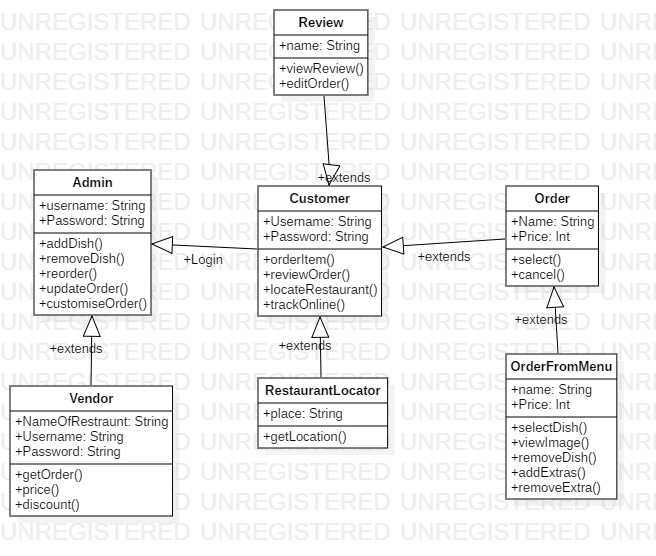
This website is available to the users 24\*7.

1. Extensibility

This website can be further extended for some more requirements if there is a demand.

# **UML design**

## **Class diagram**



## **Sequence**

### Sequence diagram for Customer Login

: Customer

:System

:Database

Login Details

Username & Password

Retrieve Details

Give Access

### Sequence Diagram for Vendor Login

: Vendor

:System

:Database

Login Details

Username & Password

Retrieve Details

Give Access

### Sequence diagram for Customer Registration

:Customer

:System

:Database

New User Request

Registration Form

Submit Registration Form

Verifying Registration Details

Successful Update

Successful Registration as customer

### Sequence diagram for Vendor Registration

:Vendot

:System

:Database

New User Request

Registration Form

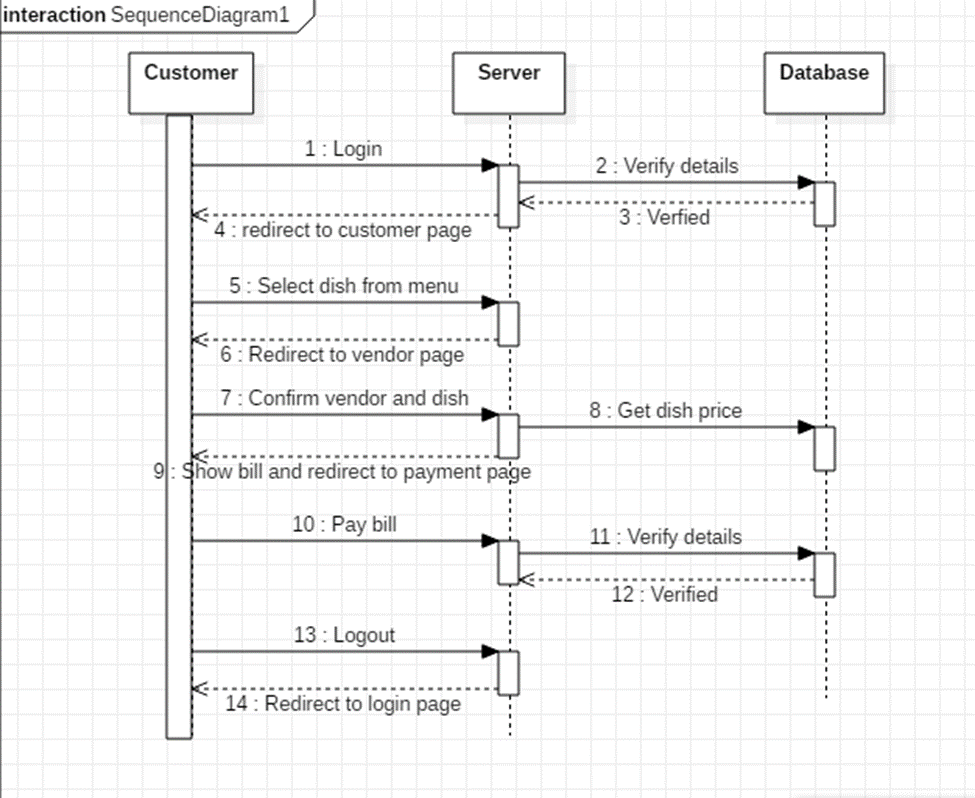
Submit Registration Form

Verifying Registration Details

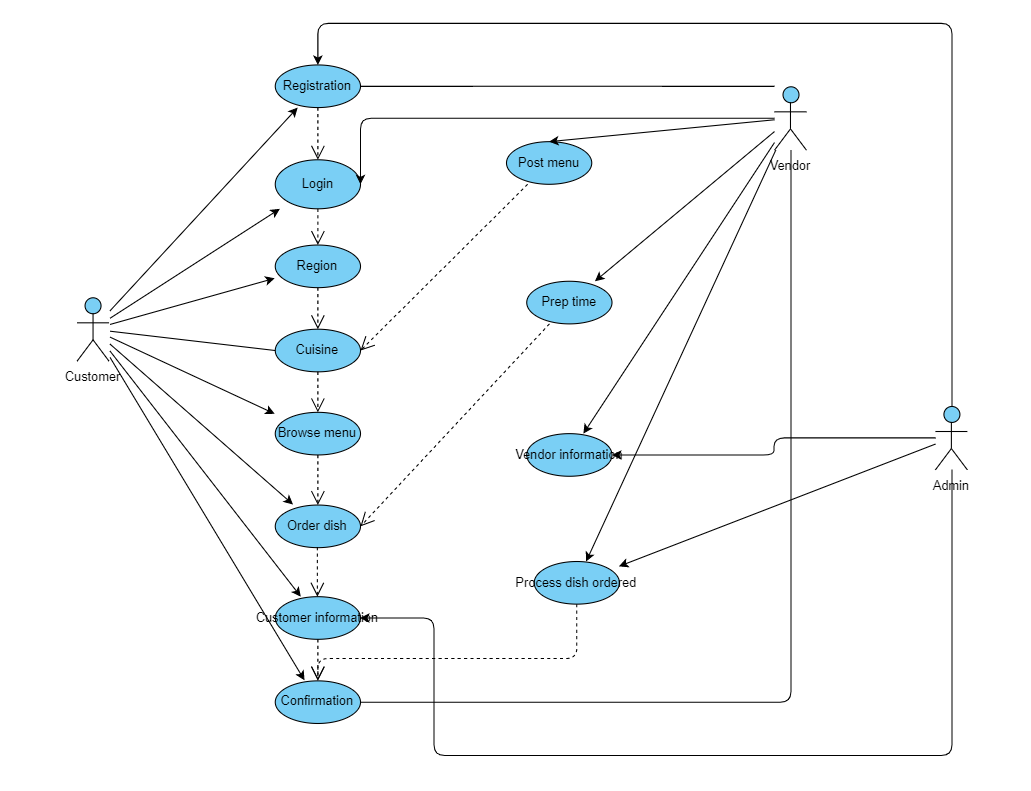
Successful Update

Successful Registration as vendor

### Sequence diagram of system



### Use case diagram



# **Test cases**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case scenario** | | **Registration** | | **Test Priority** | | **High** |
| **Test case description** | | **Positive** | | **Post requisite** | | **NA** |
| **Prerequisite** | | **NA** | |  | |  |
|  | | | | | | |
| **Sl. No** | **Action** | **Input** | **Expected Output** | **Actual Output** | **Test result** | **Test comments** |
| 1 | Launch website | <http://localhost/bonviver3/> | Website Home | Website Home | Pass | Successful |
| 2 | Enter correct email id, password, name, contact no, address and hit register button | Email id: [anurangreji@gmail.com](mailto:anurangreji@gmail.com)  Password: \*\*\*\*\*\*\*\*\*\*  First Name: --------  Last name: -------  Contact no: ---------  Address: -------- | Registration success | Registration success | Pass | Successful |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case scenario** | | **Registration** | | **Test Priority** | | **High** |
| **Test case description** | | **Negative** | | **Post requisite** | | **NA** |
| **Prerequisite** | | **NA** | |  | |  |
|  | | | | | | |
| **Sl. No** | **Action** | **Input** | **Expected Output** | **Actual Output** | **Test result** | **Test comments** |
| 1 | Launch website | <http://localhost/bonviver3/> | Website Home | Website Home | Pass | Successful |
| 2 | Enter incorrect email id, password, name, contact no, address and hit register button | Email id: [anurangrejigmail.com](mailto:anurangreji@gmail.com)  Password: \*\*\*\*\*\*\*\*\*\*  First Name: --------  Last name: -------  Contact no: ---------  Address: -------- | Error message -enter valid email id with ‘@’ | Error message – enter valid email id with ‘@’ | Pass | Successful |
| 3 | Enter correct email id, password, name, contact no, address and hit register button | Email id: [anurangreji@gmail.com](mailto:anurangreji@gmail.com)  Password: \*\*\*\*\*\*\*\*\*\*  First Name: --------  Last name:  Contact no: ---------  Address: -------- | Error message – Enter last name | Welcome. You are registered into the website. Please Login to continue. | Fail | Failure |
| 4 | Enter correct email id, password, name, contact no, address and hit register button | Email id: [anurangreji@gmail.com](mailto:anurangreji@gmail.com)  Password:  First Name: --------  Last name: ----------  Contact no: ---------  Address: -------- | Error message – Enter valid password | Registration page is loaded again | Fail | No error message is displayed |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case scenario** | | **Login** | | **Test Priority** | | **High** |
| **Test case description** | | **Positive** | | **Post requisite** | | **NA** |
| **Prerequisite** | | **A valid user account** | |  | |  |
|  | | | | | | |
| **Sl. No** | **Action** | **Input** | **Expected Output** | **Actual Output** | **Test result** | **Test comments** |
| 1 | Launch website | <http://localhost/bonviver3/> | Website Home | Website Home | Pass | Successful |
| 2 | Enter correct email id, password, and hit login button | Email id: [anurangreji@gmail.com](mailto:anurangreji@gmail.com)  Password: \*\*\*\*\*\*\*\*\*\* | Registration success | Registration success | Pass | Successful |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case scenario** | | **Login** | | **Test Priority** | | **High** |
| **Test case description** | | **Negative** | | **Post requisite** | | **NA** |
| **Prerequisite** | | **A valid user account** | |  | |  |
|  | | | | | | |
| **Sl. No** | **Action** | **Input** | **Expected Output** | **Actual Output** | **Test result** | **Test comments** |
| 1 | Launch website | <http://localhost/bonviver3/> | Website Home | Website Home | Pass | Successful |
| 2 | Enter correct email id, no password, hit login button | Email id: [anurangreji@gmail.com](mailto:anurangreji@gmail.com)  Password: \*\*\*\* | Wrong combination of username and password | Wrong combination of username and password | Pass | Successful |
| 3 | Enter incorrect email id, password, hit login button | Email id: [invalid@gmail.com](mailto:invalid@gmail.com)  Password: \*\*\*\*\*\*\*\*\* | Enter valid email id | Enter valid email id | Pass | Successful |

# **Who wrote what components?**

The project has the following modules and for coding they are distributed among the team members as follows.

1. Registration and login module - Sravya Bussoli and Sravya Garlapati
2. Customer and Vendor accounts - Anuradha Vishwapriya and Harshitha Mandru
3. Cart and payment - Anuradha Vishwapriya and Harshitha Mandru
4. Security features and feedback - Sravya Bussoli and Sravya Garlapati

Till now, we have completed registration and login page for both customer and vendors.

# **User manual and installation**

The user can run the code if he has xampp already installed else he can download xampp and then the following steps can be followed to execute the code.

1. Download and install XAMPP in your system.
2. XAMPP for windows 7.1.32 64 bit version
3. Once XAMPP is installed, start Apache and MySQL.

A screenshot of a computer

Description automatically generated

1. Change the port number on clicking the config button if there is any conflict while starting with Apache and MySQL
2. Place the executable folder i.e., bonviver3 in the C:\xampp\htdocs location

A screenshot of a cell phone

Description automatically generated

1. Open localhost/bonviver3 in the browser which will open the homepage of the website

A screen shot of a social media post

Description automatically generated

1. In the other case, a list of php files may appear
2. Click on the php files to run the php file

# **Test cases scenarios**

The test case screenshots are shown below.

**Registration successful scenario**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

**Invalid email id during registration**

A screenshot of a cell phone

Description automatically generated

**Invalid login password**

**A screenshot of a cell phone

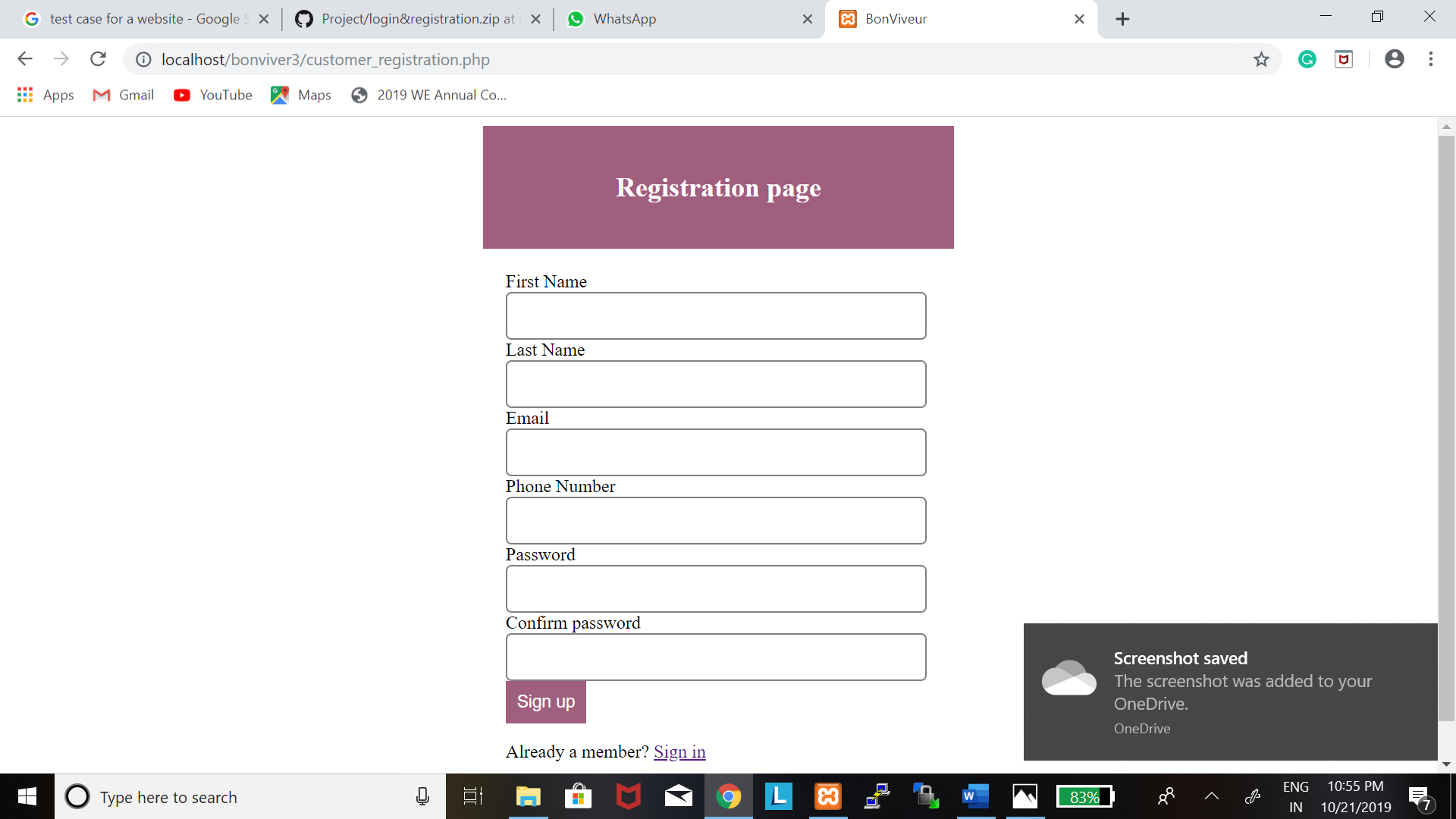
Description automatically generated**

**Invalid password registration**

When the password isn’t entered, it goes back to the registration page.

**A screenshot of a cell phone

Description automatically generated**



# **Feedback of peer review session and actions taken on that feedback.**

The feedback of peer review session was good, and we got to know what mistakes we had in our design like differentiating between the customer registration and vendor registration page and in the database. They also suggested to add a column called “scope” to the users table and in the vendor table as well and can change that value for a user if that user wants to be vendor.