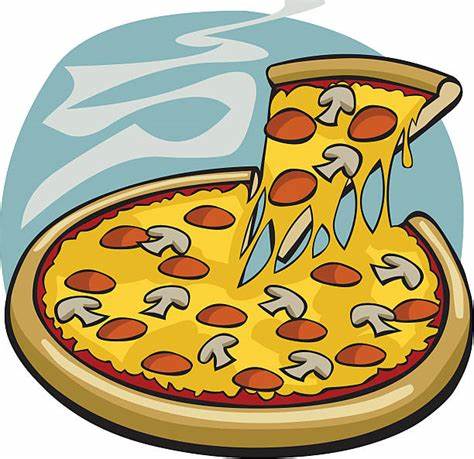
**PIZZA ORDERING SYSTEM**



**The project report is documented by-**

**Abiram M J**

**Joannes Jacob Mathew**

**Pilla rishi Vinay Kumar**

**Chuppala Venkata Tarun**

**PIZZA ORDERING SYSTEM**

Abstract

The purpose of pizza ordering system is to help customers to order and eat pizza, with the help of web application, just by sitting in their homes so their valuable time and efforts aren’t wasted. The required software and hardware is easily available and easy to work with. This system can help you to have an error free, secure and reliable services directly from the vendor.

It can assist the user to concrete on their other activities since they are saving time by not going to restaurant and waiting for the waiter to place their order. It will also help the restaurant in better utilization of its resources. The aim is to automate its existing manual system by the help of computerized equipment and full-fledged computer application, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same.

There by users can now easily assist themselves getting the pizza from the store with the same efficiency but with less efforts and faster responses. Customers will be satisfied to fullest as they are being treated with utmost attention and respect even if they aren’t showing up to the store.

Objective:

* To develop an online system to help the users to order pizza.
* To manage orders for the restaurant effectively.

**Functionalities provided by pizza ordering system are as follows**:

* Admins can add food items by category and also update them, which will be reflected in the user page.
* Shows the information and description related to the food items.
* Adding and removing food items from the cart.
* We can create an account as both admin and user.
* Our application is secure and safe as the user details are encrypted in the data base

**Introduction:**

Pizza ordering System is a web-based application developed with the help of Java programming language. It is mainly focused on receiving orders from the customer and delivering it to the vendor. This system helps in maintaining the user records and orders, more efficiently. Admin and customers all have a different interface and different privileges according to their needs.

**Scope:**

Our project aims at Business process automation i.e.; we have tried to computerize the process of ordering and serving food.

Scope of any software depends upon the following things:

1. It satisfies the user requirement.

2. Be easy to understand by the user and operator.

3. Be easy to operate.

4. Have a good user interface.

5. Be expandable.

6. It helps the vendor in expanding business.

7.It can be more profitable in terms of both money and time.

We have tried to make the such type of software, which satisfy the above given requirements.

**Significance:**

• Easy to update information.

• Work becomes speedy.

• Access of any information individually.

• Decrease the load of the person involve in existing manual system.

• Well-designed reports.

• Easy & fast retrieval of information.

• Accuracy in work.

. • It contains better storage capacity.

• Robust database back-end.

• Creating and changing data at ease.

**Modules:**

Our system has only two main interfaces. One is for the admin and the other one is for the user, here user is a customer.

Along with these, we have other interfaces which can be discussed as below:

**Register Module:**

Through this module new user can easily register himself on the website. In this module, the user register's itself by entering the necessary data. This module additionally confirms the new user registration with the previously enrolled users.

**Login Module:**

After registration, one can log in to the system as the end-user of the system on the behalf of the user. The user will get only those privileges that are given to the user for which one has registered. For example, if a user has registered as a customer, then the user only has the privileges to view the data and cannot make any changes to the data that is shown.

**User Module:**

In the user module, customer can choose his desired food item see the details of the food items aa add the food item to the cart. the customer can also the customer can also remove items from the cart. They can find the total amount to be paid.

**Shop Module:**

We can select the product of our wish and add them to the cart. We can select the category of the product. We can see more details of the product in the view product. And then add them to the cart.

**Cart Module:**

We can add products to the cart. We can also remove products from the cart. We can see the total price and we can check out the products at once. After clicking check out, page will be redirected for filling the address details of the user. Finally, payment can be done.

**User Interface Design:**

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

The following steps are various guidelines for User Interface Design:

• The system user should always be aware of what to do next.

• The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.

• Use display attributes sparingly.

• Default values for fields and answers to be entered by the user should be specified.

• A user should not be allowed to proceed without correcting an error.

• The system user should never get an operating system message or fatal error.

**Technologies Used:**

**Frontend Used:**

**HTML:**

HTML (Hypertext Markup Language) is the code that is used to structure a web page and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.

**CSS:**

Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML.

**JavaScript:**

JavaScript (JS) is a scripting language, primarily used on the Web. It is used to enhance HTML pages and is commonly found embedded in HTML code. JavaScript is an interpreted language. Thus, it doesn't need to be compiled. JavaScript renders web pages in an interactive and dynamic fashion. This allowing the pages to react to events, exhibit special effects, accept variable text, validate data, create cookies, detect a user’s browser, etc.

**Thyme leaf:**

Thyme leaf is a Java XML/XHTML/HTML5 template engine that can work both in web and non-web environments. It is better suited for serving XHTML/HTML5 at the view layer of MVC-based web applications, but it can process any XML file even in offline environments. It provides full Spring Framework integration.

**Bootstrap:**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

**Backend Used:**

**Java:**

Java is a general-purpose, class-based, object-oriented programming language designed for having lesser implementation dependencies. It is a computing platform for application development. Java is fast, secure, and reliable, therefore. It is widely used for developing Java applications in laptops, data canters, game consoles, scientific supercomputers, cell phones, etc.

**Spring Boot:**

Spring Boot is an open-source micro framework maintained by a company called Pivotal. It provides Java developers with a platform to get started with an auto configurable production-grade Spring application. With it, developers can get started quickly without losing time on preparing and configuring their Spring application.

**Spring Data JPA:**

JPA is a Java specification that is used to access, manage, and persist data between Java object and relational database. It is a standard approach for ORM.

**Spring Security:**

Spring Security is a powerful and highly customizable authentication and access-control framework. It is the de-facto standard for securing Spring-based applications. Spring Security is a framework that focuses on providing both authentication and authorization to Java applications.

**Database Used:**

Oracle: Oracle database is a relational database management system. It is also called Oracle DB, or simply Oracle. It is produced and marketed by Oracle Corporation. It was created in 1977 by Lawrence Ellison and other engineers. It is one of the most popular relational database engines in the IT market for storing, organizing, and retrieving data.

**Hardware Requirements:**

**Table 1.8: Table of Hardware Requirement**

|  |  |  |
| --- | --- | --- |
|  | **Minimum System Requirements** | **Recommended System Requirements** |
| Processor | 7th Gen Intel Core i7 | 9 th Gen Intel Core i7 or better |
| RAM | 4GB | 8GB or more |
| Storage | 256GB SSD | 512GB or more |
| Display | 14-inch FHD (1920 X 1080) | 15.6-inch FHD IPS(1920 X 1080) |
| Graphics | 4GB NVIDIA GeForce GTX 1060 | 8GB NVIDIA GeForce GTX 2070 |
| Battery | Up to 2 hours | Up to 5 hours |

**Project Life Cycle:**

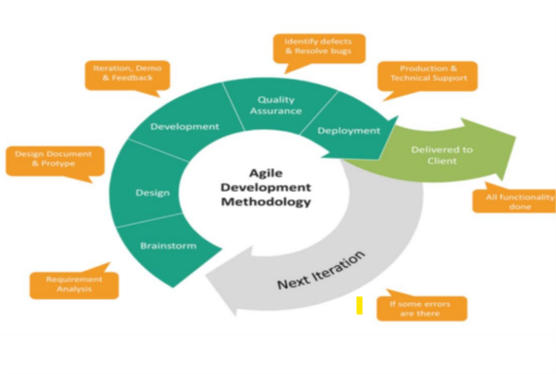
**Agile Software Development Methodology**

The Agile software development methodology is one among the only and effective processes to show a vision for a business need into software solutions. Agile may be a term want to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to vary.

The agile software development emphasizes on four core values.

* Individual and team interactions over processes and tools
* Working software over comprehensive documentation
* Customer collaboration over contract negotiation
* Responding to change over following a plan

**Phases of Agile Methodology:**

****

***Agile Methodology Phases***

**Phase-1: Requirement Analysis: -** In this phase, we gather data and analyses how food ordering system works. Also collected requirements after reviewing earlier papers & websites.

**Phase-2: Design: -** On the basis of gathered information we designed and build a model.

**Phase-3: Development: -** Deliver the working software based on iteration, requirements or feedback.

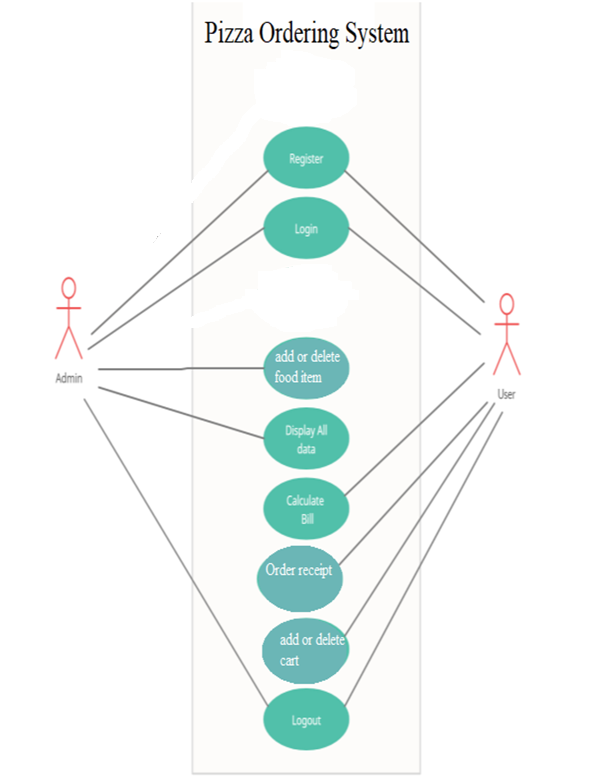
**Phase-4: Quality Assurance: -** This is a testing phase where we test our model.

**Phase-5: Deployment: -** In this phase we deploy our final release of the iteration into production.

**Phase-6: Feedback: -** Accept the user feedback and work it into the requirements of next iteration.

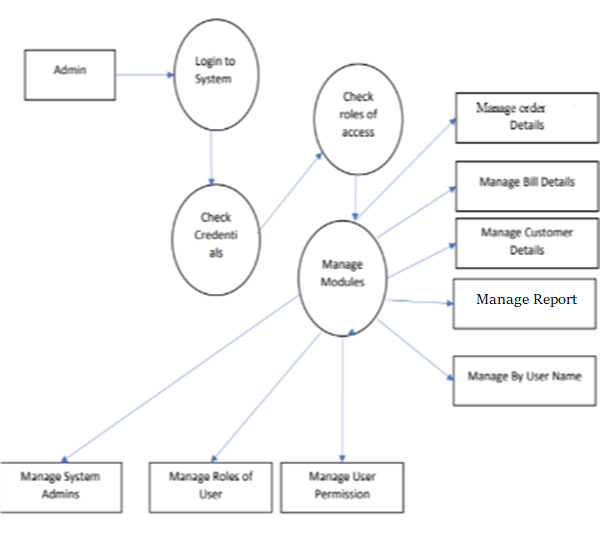
**Use Case Diagram:**

A use case depicts a series of steps that give something significant value to an actor and is drawn as a horizontal ellipse. A use case diagram apprehends the functional features of a system with processes implement in the system. While discussing the functionality and procedures of the framework, you discover important characteristics of the framework that you represent in the use case diagram.

****

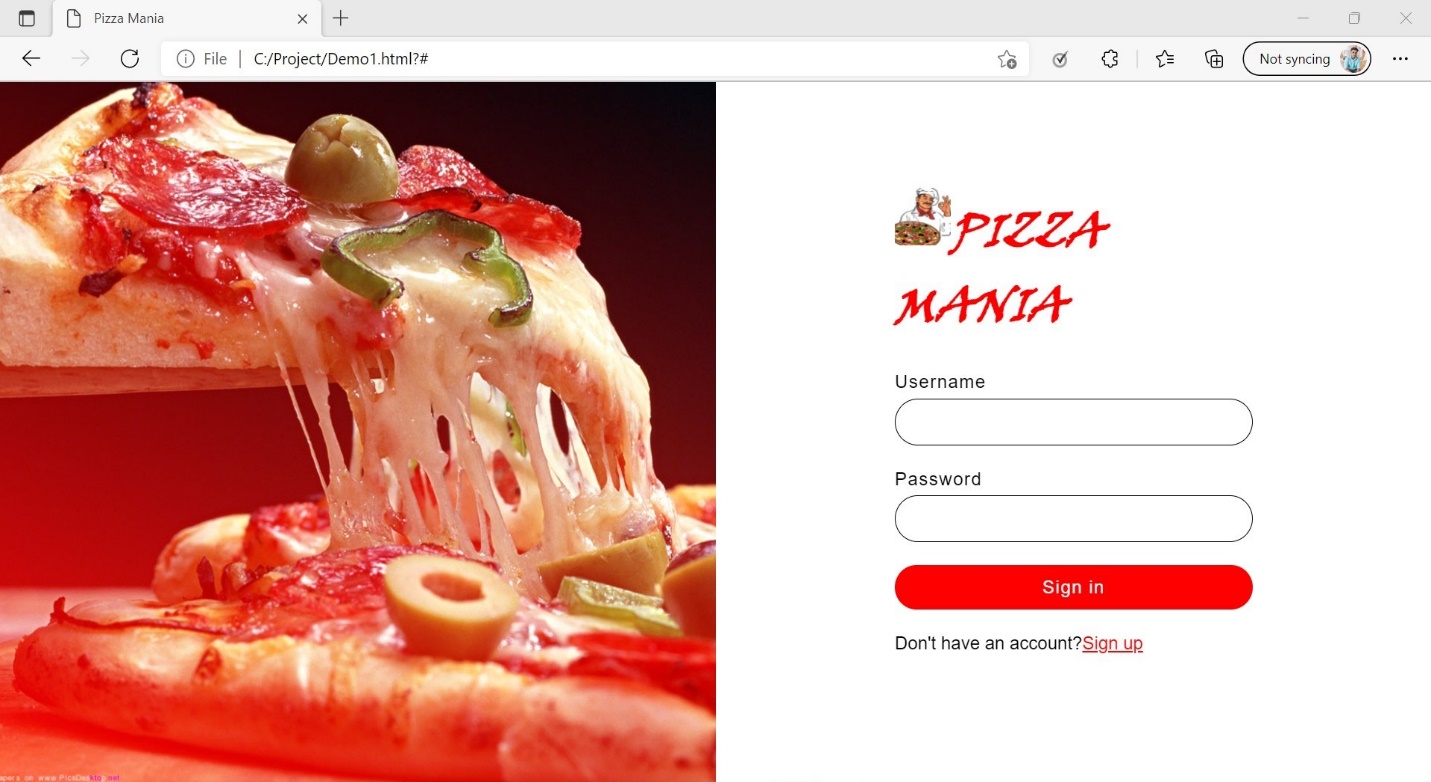
**Dataflow Diagram:**

A data flow diagram should be the first mechanism used by framework analyst to model system fundamental. These fundamentals are the system operations; the information used by this operations and external organization that interact with the framework and the data flows in the system.

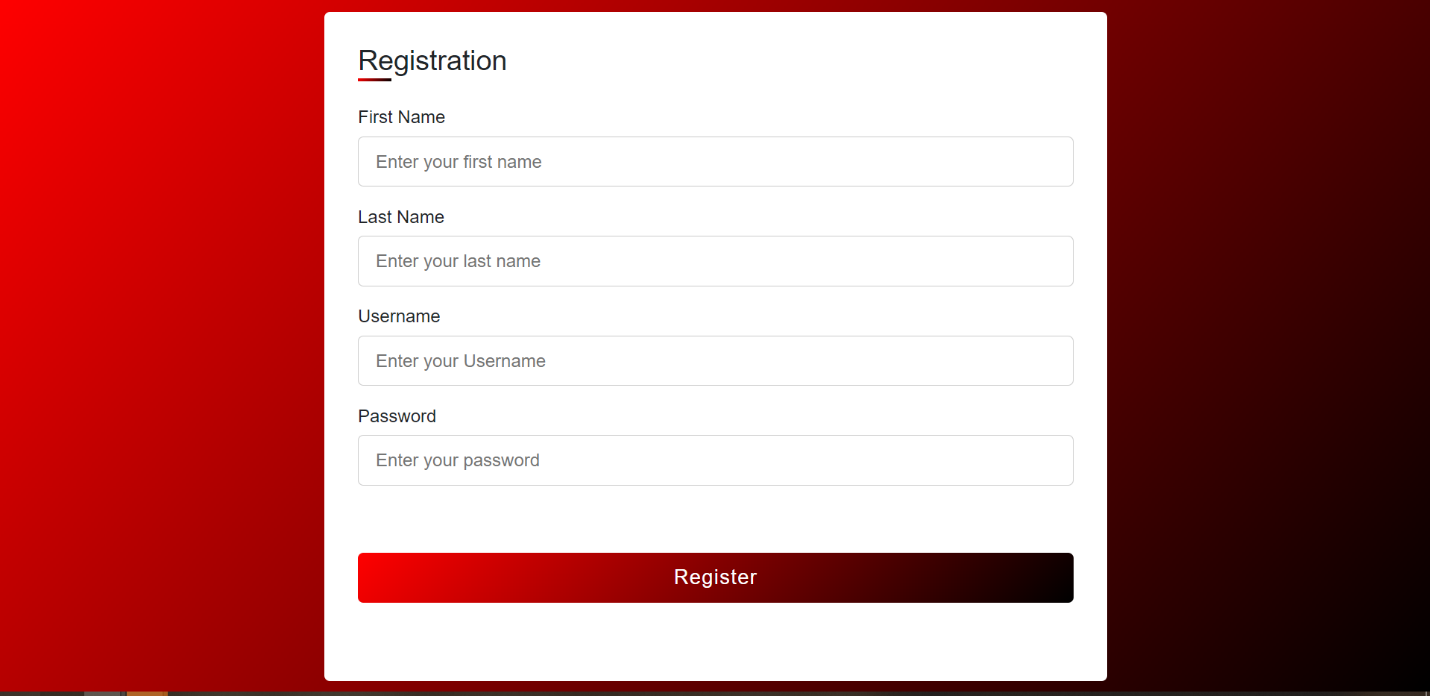
****

***Dataflow Diagram of Pizza Ordering System.***

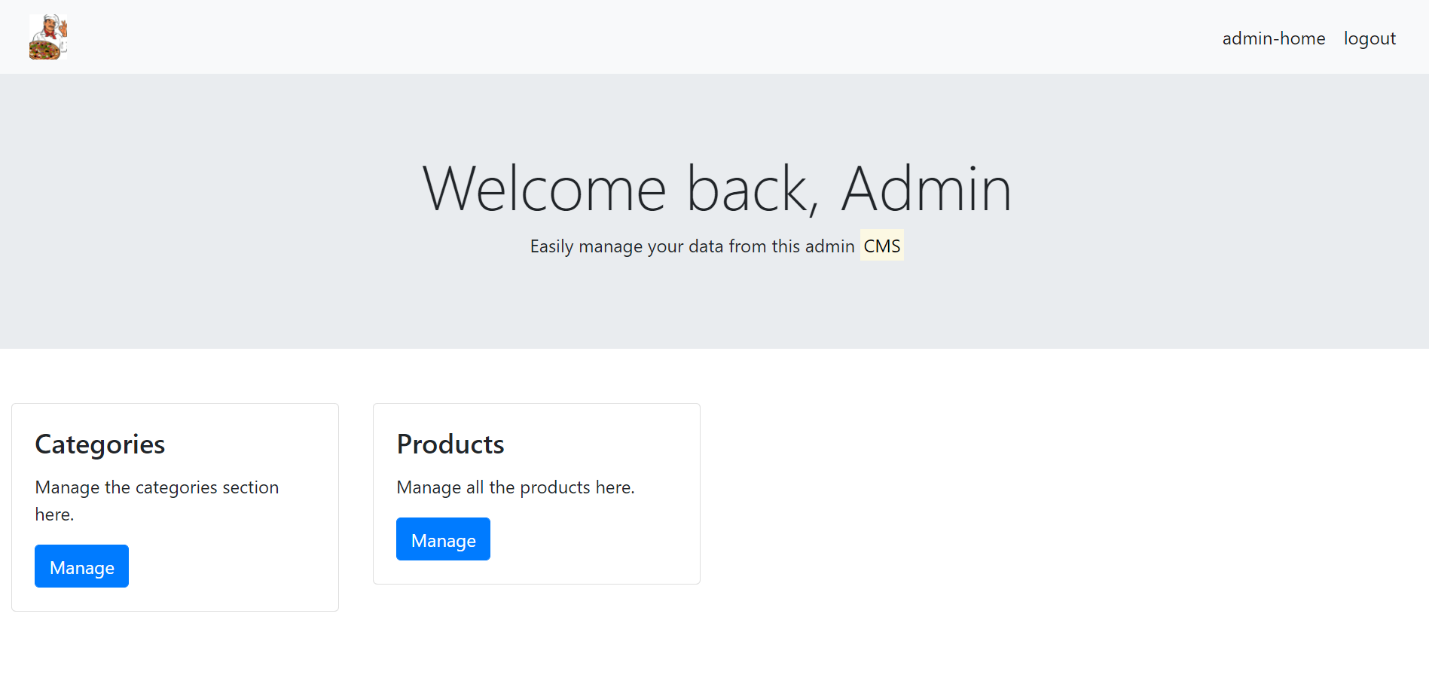
**Snapshots**

**Login Page**

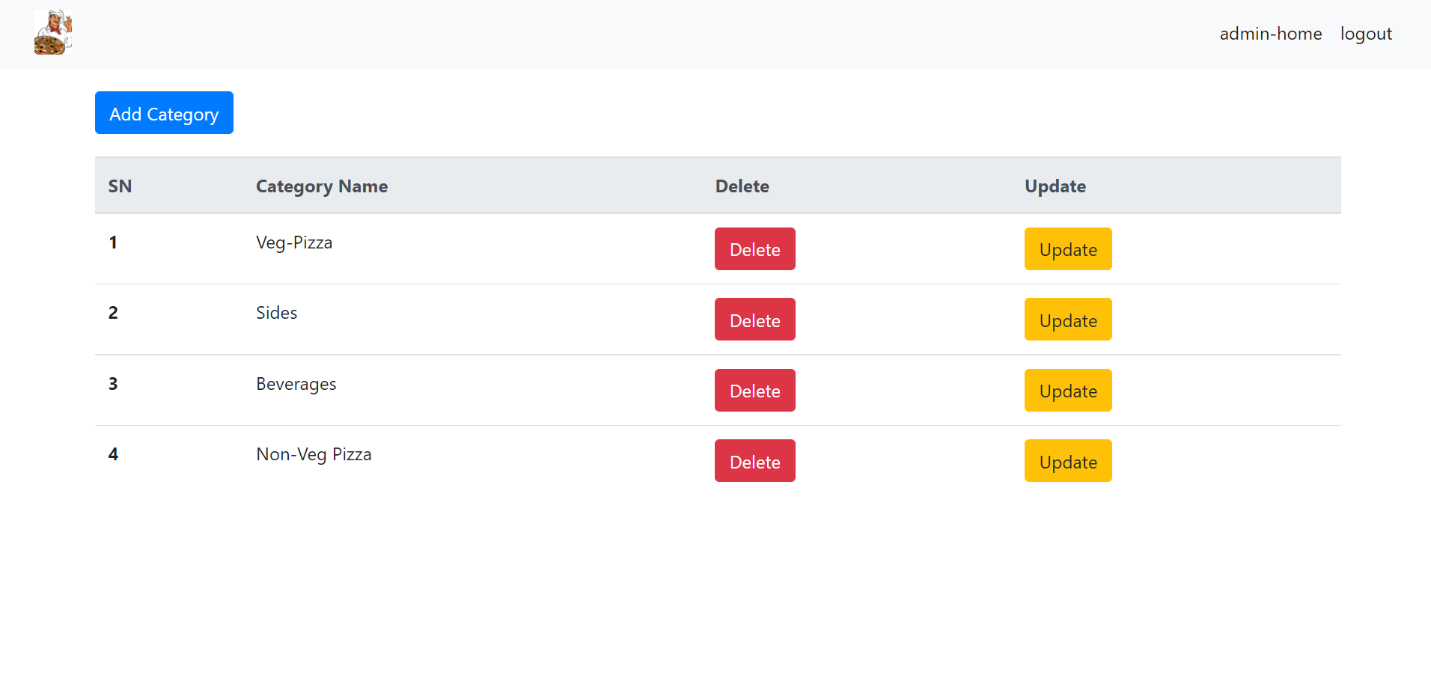
**Registration Page**

****

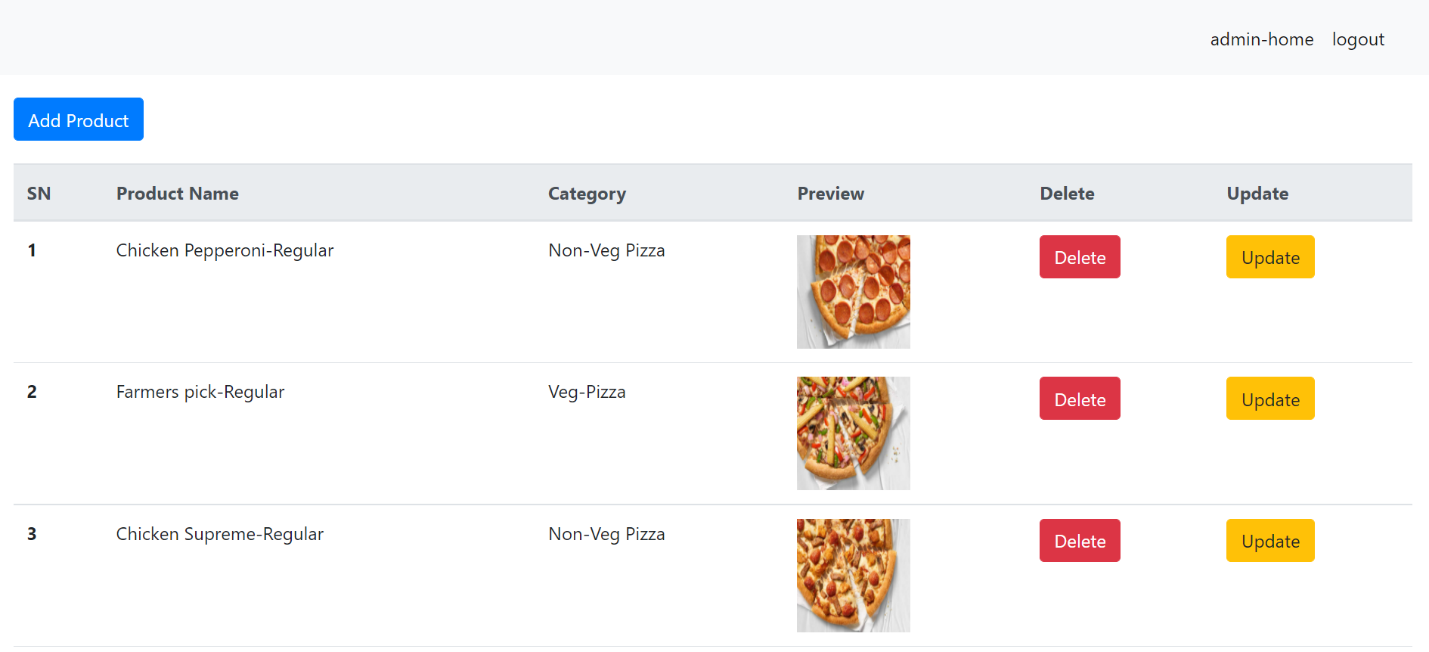
**Admin-Home**

****

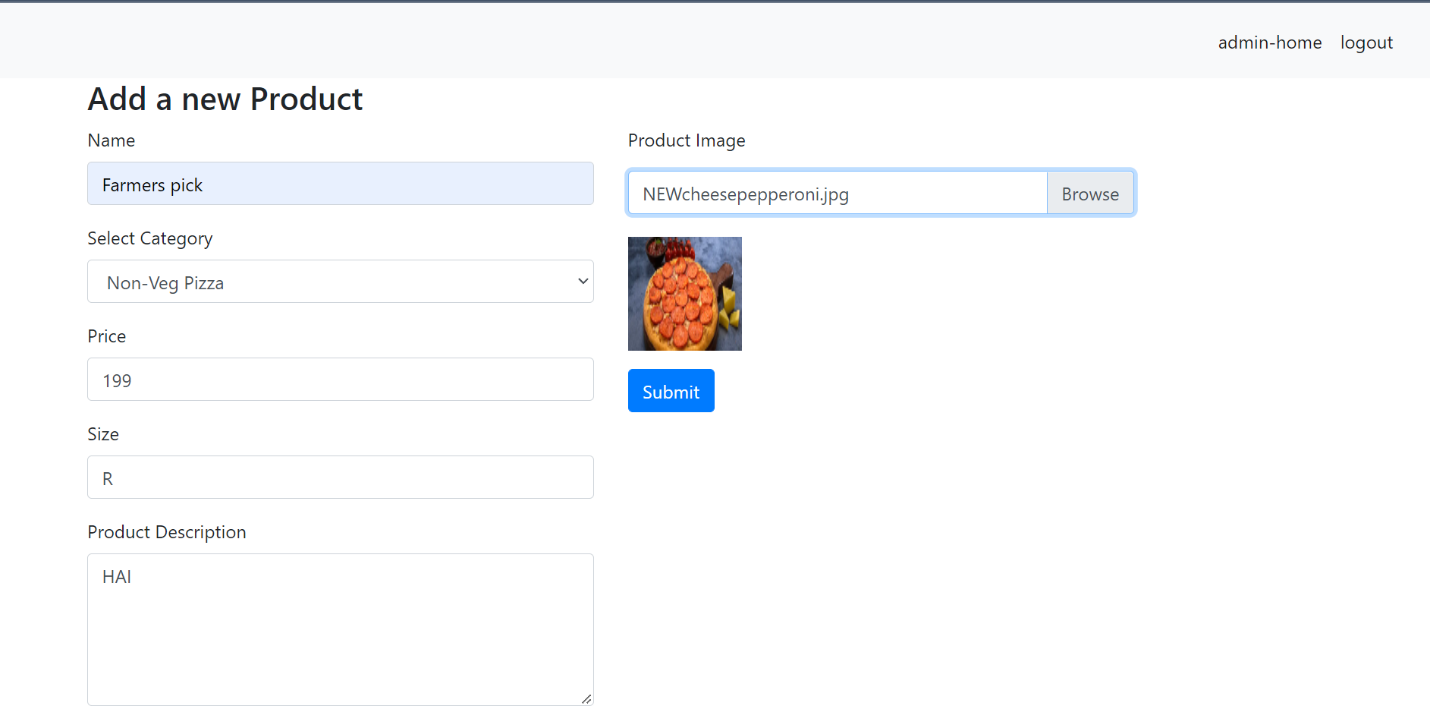
**View, Update, Delete- Category**

****

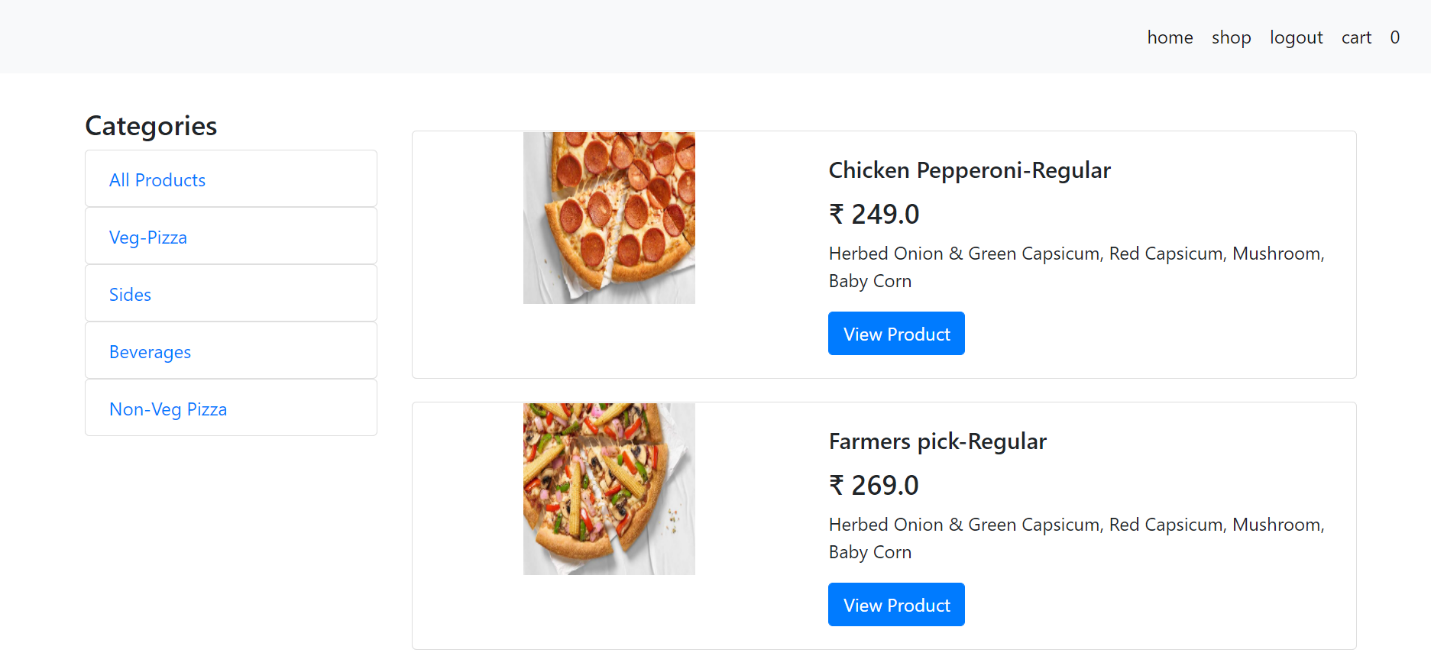
**View, Update, Delete- Product**

****

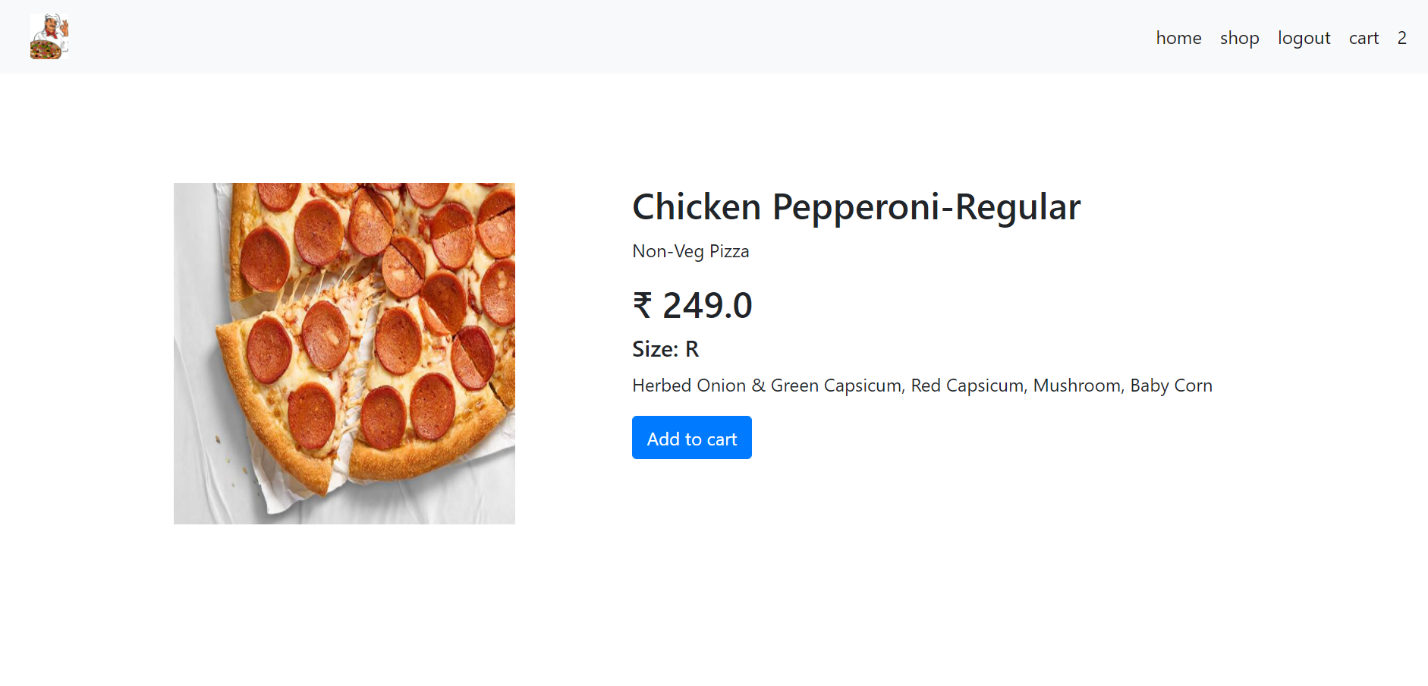
**Add- Product**

****

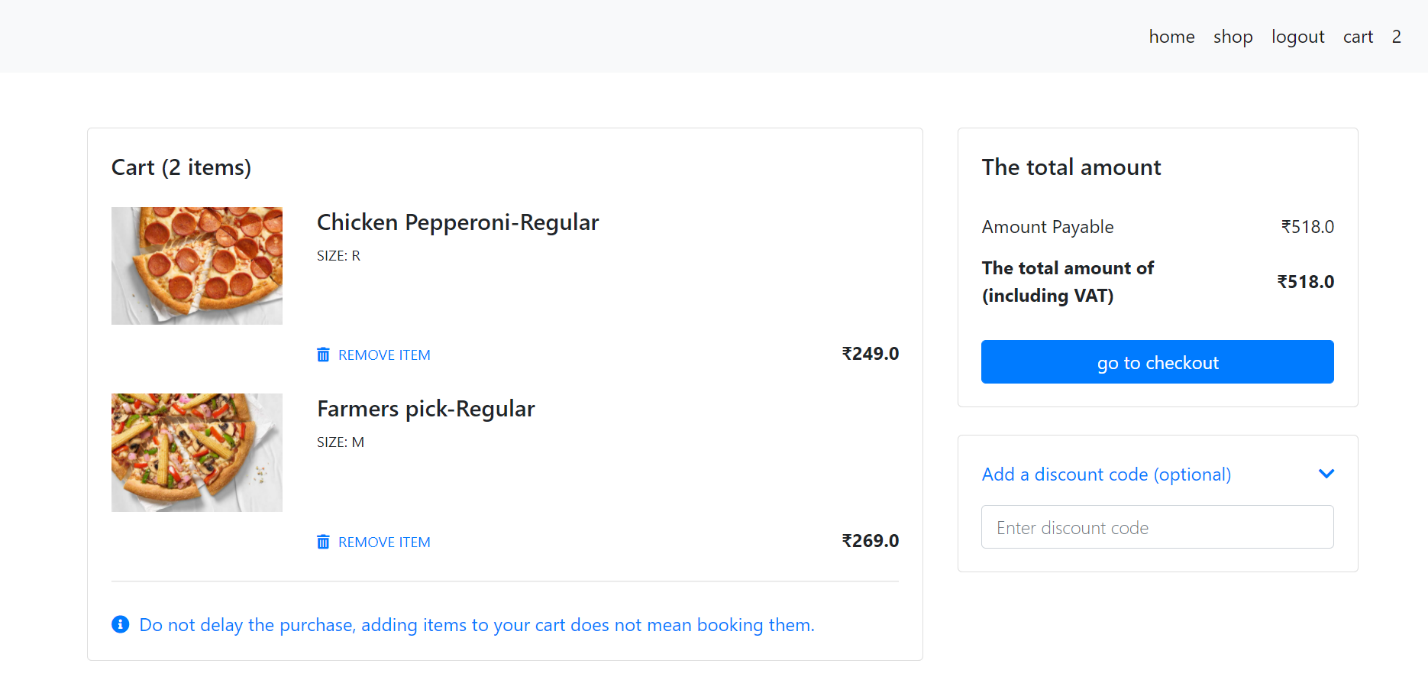
**Shop**

****

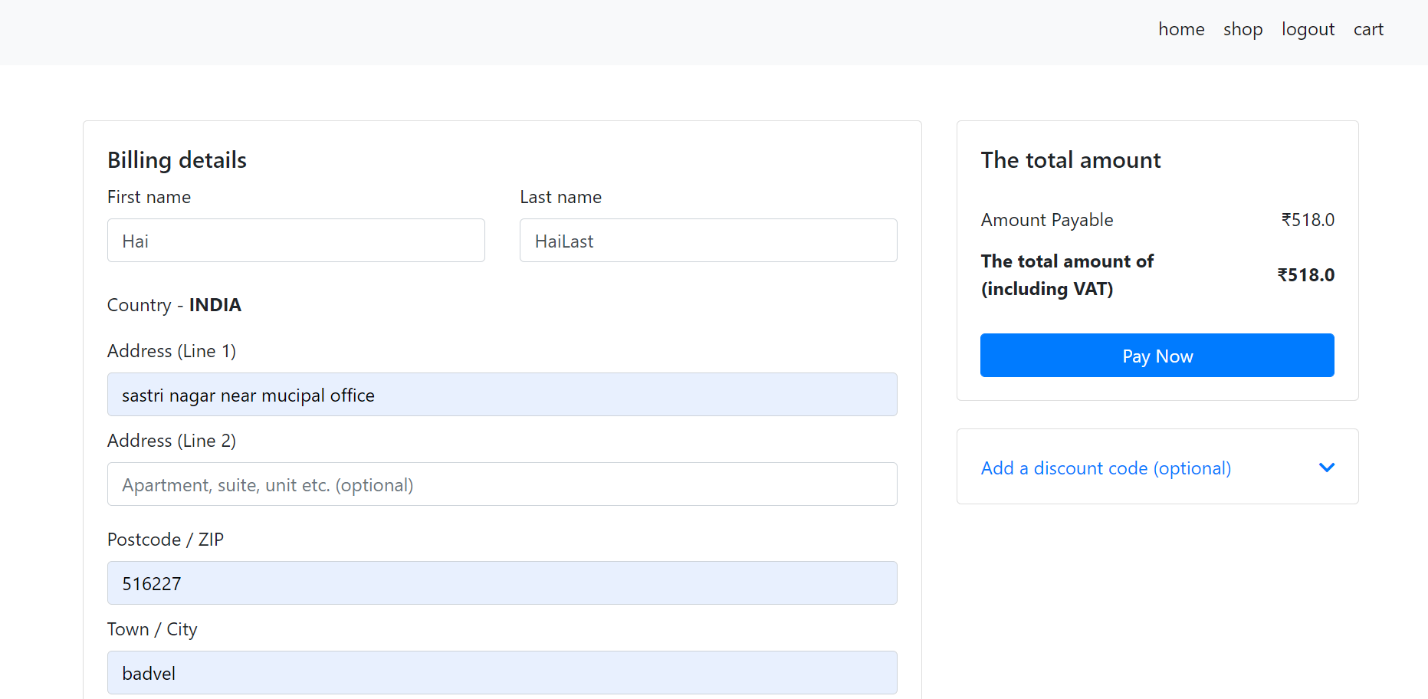
**Product View**

****

**Cart View**

****

**Billing Details**

****