

# Food Consignment Website

Yokesh Saravanan M<sup>1</sup>, Mahalakshmi L<sup>2</sup>

<sup>1</sup>Student, Department of Computer Applications, Hindusthan College of Engineering and Technology, Coimbatore.

<sup>2</sup>Assistant Professor, Department of Computer Applications, Hindusthan College of Engineering and Technology, Coimbatore.

[mahalogu@gmail.com](mailto:mahalogu@gmail.com)

## Abstract

This project is entitled as “FOOD CONSIGNMENT WEBSITE” a web and mobile-based application. This application has been developed using HTML, CSS, and JAVASCRIPT as the front end and PHP, and SQL as the back end. The project aims to provide customers with a way to place an order at a restaurant over the web application. It Overcomes the Disadvantage of the Traditional Queueing system. The main objective of this website is to manage the details of the Item Category, Food, Delivery Address, Orders, Shopping cart, and Item Category. The project is built at the administrative end thus, only the administrator is guaranteed access.

**Keyword:** Food Ordering, Food Website, Hotel Automation.

## Objective

The objective of this project is to develop a Door Delivery System for Hotel Vikram. The main reason behind implementing this project is to automate and facilitate the whole process of Ordering through the internet. People need a system where they can order products easily and reliably through online. This system will be implemented to replace the traditional purchasing method with online purchasing since the internet and e-commerce online purchase plays an important role in business and it has been a popular means of selling products. Thus providing an increase in the number of sales by making use of attractive web pages. The client-side needs to be a user-friendly interface, CSS is entirely used with HTML and JavaScript. The server-side program stores all required data in the database using MySQL database with PHP which can easily retrieve and store the data in the database through MySQL.

## Existing system

<https://ijrtte.com>

The existing system is a traditional food ordering system. In which customers go in personally to the hotel for ordering and buying food. Time spent by the customer is more if it is a peak hour the condition is worse. The main problem here is, to eat or order food, the customer needs to travel a long way in spite of a fine work schedule. The traditional food ordering procedure is not efficient enough for hotels and restaurants, as they have to deal with crowds in their restaurant

### **Disadvantage**

1. The main problem is that customers need to wait and buy food. Could get there and the store doesn't have the item you're looking for.
2. Traditional food ordering system is very cumbersome and time-consuming. More difficult to compare prices from one store to the next.
3. Customer faces waste of time in case of the queue.

### **Proposed system**

In the proposed system order is made through the internet, and the time spent on traveling is removed or rectified. So, the customer need not travel all a for buying food. The proposed system is an online food consignment system that allows customers to check for various foods available at the website and order online. This system is a dynamic website that will make online purchasing easy.

### **Advantage**

1. Customers can browse through the food menus and order online rather than visiting the restaurant. more options to choose
2. All the data related to the purchase are stored in the database thus eliminating the risk of maintaining data via paper works.
3. This system helps to save time.

## **System specification**

### **Hardware specification**

- Processor: Intel Core i5 8<sup>th</sup> Gen
- RAM: 8GB
- HDD: 256GB
- Monitor type: 15 Inch VGA
- Keyboard: 110Keys Keyboard

### **Software specification**

- Operating System: Windows 10
- Client : Any Browser (Compatible)
- Frontend : HTML,CSS,JAVA script
- Backend : PHP
- Database: MYSQL
- Server : Apache
- Code Behind: Visual Studio.

## **About front end**

### **Html**

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

### **CSS**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain Extensible Markup Language (XML), Scalable Vector Graphics (SVG) and XML User Interface Language (XUL), and is

applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging web pages, user interfaces for web applications, and user interfaces for many mobile applications.

### **Javascript**

JavaScript is often abbreviated as JS is a high-level interpreted programming language. It is a language that is also characterized as dynamic, weakly typed, prototype-based, and multi-paradigm. Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles.

### **PHP**

PHP stands for **PHP: Hypertext Preprocessor**. Hypertext refers to files linked together using hyperlinks, such as Hyper Text Markup Language (HTML) files. Pre-processing is executing instructions that modify the output. PHP is a scripting language originally designed for producing dynamic web pages. It has evolved to include a command line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License; however, it is incompatible with the GNU General Public License (GPL), due to restrictions on the usage of the term PHP.

PHP is a widely-used general-purpose scripting language that is especially suited for web development and can be embedded into HTML. It generally runs on a web server, taking PHP code as its input and creating web pages as output. It can be deployed on most web servers and on almost every operating system and platform free of charge. PHP is installed on more than 20 million websites and 1 million web servers. PHP 5 was released in July 2004 after long development and several pre-releases. It is mainly driven by its core, the Zend Engine with a new object model and dozens of other new features.

### **About back end**

#### **Xampp**

XAMPP is a free and open-source cross-platform web server resolution stack package

developed by Apache Friends, consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), Maria DB (M), PHP (P), and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (Maria DB), and scripting language. (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

### **Apache**

The Apache HTTP Server commonly referred to simply as Apache is a web server notable for playing a key role in the initial growth of the World Wide Web. The majority of all web servers using Apache are Linux web servers. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. The application is available for a wide variety of operating systems, including UNIX, FreeBSD, Linux, Solaris, Novell NetWare, Mac OS X, Microsoft Windows, OS/2, TPF, and E-Com Station. Released under the Apache License, Apache is characterized as free software and open-source software.

Apache is primarily used to serve both static content and dynamic Web pages on the World Wide Web. Many web applications are designed expecting the environment and features that Apache provides. Apache is the web server component of the popular LAMP web server application stack, alongside MySQL, and the PHP/Perl/Python (and now also Ruby) programming languages.

Apache is redistributed as part of various proprietary software packages including the Oracle Database or the IBM WebSphere application server. Mac OS X integrates Apache as its built-in web server and as support for its WebObjects application server. It is also supported in some way by Borland in the Kylix and Delphi development tools. Apache is included with Novell NetWare 6.5, where it is the default web server. Apache is also included with many Linux distributions.

Apache is used for many other tasks where content needs to be made available in a secure and reliable way. One example is sharing files from a personal computer over the Internet. A user who has Apache installed on their desktop can put arbitrary files in the Apache's document root which can then be shared.

Programmers developing web applications often use a locally installed version of Apache in order to preview and test code as it is being developed. Microsoft Internet Information Services (IIS) is the main competitor to Apache, trailed by Sun Microsystems' Sun Java System Web Server and a host of other applications such as Zeus Web Server. Some of the biggest websites in the world are run using Apache. Google's search engine front end is based on a modified version of Apache, named Google Web Server (GWS).

## **MySQL**

MySQL is a relational database management system (RDBMS) that has more than 11 million installations. The program runs as a server providing multi-user access to a number of databases. MySQL is owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now a subsidiary of Sun Microsystems, which holds the copyright to most of the codebase. The project's source code is available under terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL is popular for web applications and acts as the database component of the LAMP, BAMP, MAMP, SAMP, and WAMP platforms and for open-source bug-tracking tools like Bugzilla. Its popularity for use with web applications is closely tied to the popularity of PHP, which is often combined with MySQL. Several high-traffic websites (including Flickr, Facebook, Wikipedia, Google, Nokia, and YouTube) use MySQL for its data storage and logging of user data.

### **Project description**

#### **Problem definition**

The problem with the current system is that people need to drive around, find parking, and get stressed out in traffic. Customers have to travel a certain distance for the Food. Sometimes our ordered item may not be available in the hotel. Crowds may annoy some people and standing around in queue is a waste of time.

## Overview of the project

The proposed system is a website that can be used by anyone who has an account in the website. The customer can browse the Food and can buy the same online. The administrator/manager can manage all the details regarding the products and the orders of the customers. The data is stored in the database and can be retrieved later for future reference. The proposed website can be used to buy foods it including veg, non-veg, and fast food items. It

also stores all the details of the products and the customers in the database. The admin can view the orders placed by each customer. Once the order is received, the respective customer will get a call from the restaurant to ensure their delivery address and the Food is delivered at their doorstep. The delivery person updates the delivery status manually, once the ordered food is delivered.

## Module description

### User Registration Module

People who wish to order food from the hotel have to do registration. Only registered users can make an order through this site. The system automatically rejects unauthenticated users who try to make an order. An unauthenticated user can view only the website.

### Login Module

People who have completed the registration process can use their username and password to log in to the website.

### Order Module

This module is for ordering the Food items available in the hotel. Only those who have registered as Users can place their order for purchasing. The menu displays pictures of the Food items, their price, description, and quantity. The 'Add to cart' option is available for the Users who have logged in to add selected items to their cart. This places the order.

### Purchase Module

Registered customers can log in, can add the items to the cart, and then buy the same. This module deals with information about the customers and their orders.

### Add to Cart

The customer wishes to buy a product, they can click on the add to cart box, and now

the product will be added to the cart. When the cart has been selected the products stored in the cart can be seen here we can place the product

### Admin Module

The admin module manages all the details regarding the food items and the orders of the customers. Admin should log in with an authorized username and password. Admin can view the order details and enquire about the same for delivering the items to their doorstep.

### Data flow diagram

A Data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system. DFD can also be used for the visualization of data processing (structured design). On a DFD, data items flow from an external data source or an internal data store or an external data sink, via an internal process.

### Level 0

The level 0 DFD shows the complete flow of data through the proposed system (Online shopping cart). Admin will add the items in the store that will be purchased by the customer. Admin can control and view all the information. Flow of data is described in Figure 1, 2 and 3.

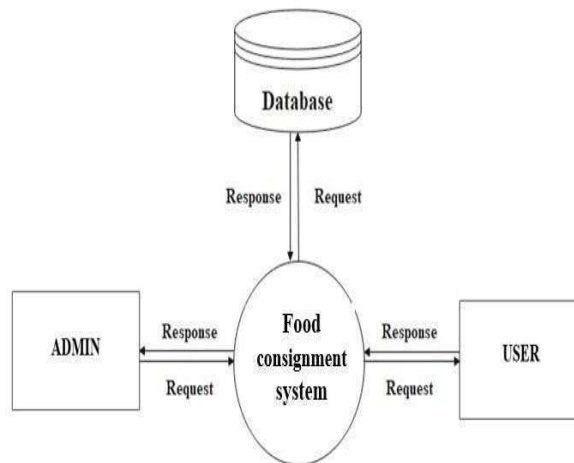


Figure 1: Level 0 DFD



**Level 1 admin**

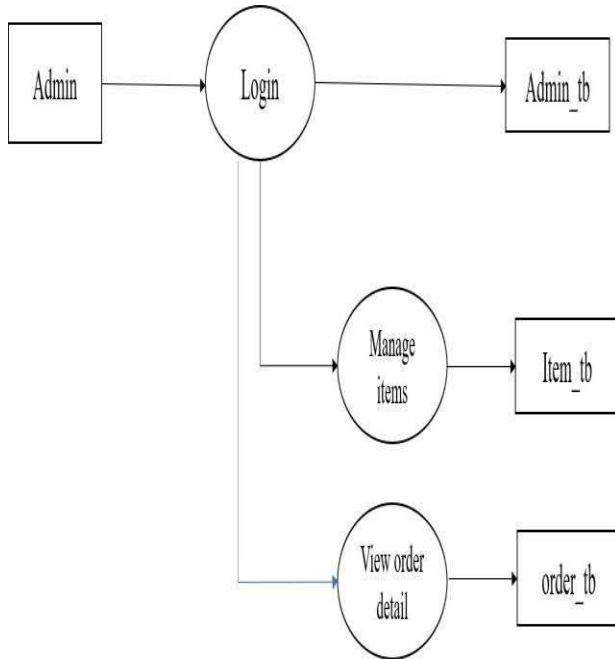


Figure 2 : Level 1 DFD - Admin

**Level 1 user**

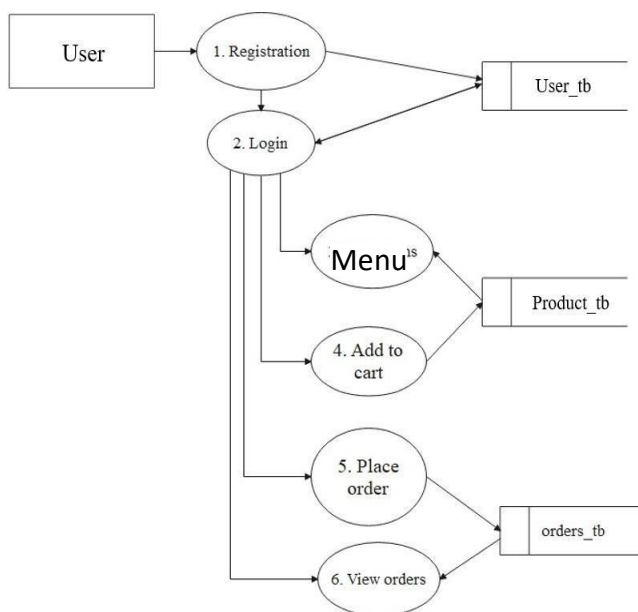


Figure 3 : Level 1 DFD - User

**Registration**

User have to register by entering required information and the User details are stored into User table (User\_tb) in the database.

**Login**

User can login to the system by entering their username and password and the login details are checked with the customer table(User\_tb).

**View Items**

The User can view the items available in the restaurant that have been added by the Admin. The details regarding the items are retrieved from product table (Products\_tb).

**Add To Cart**

User can add the items into the cart one by one and purchase them altogether at the end.

### Place Order

The User can place the order after adding the items into the cart. All the details regarding the orders are stored in the orders table (orders\_tb).

### View Order

The User can view their previous order whenever necessary. All the details regarding the previous orders are retrieved from the order table (orders\_tb).

### ER Diagram

In software engineering, an entity– relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them. Entity relationship is shown in Figure 4.

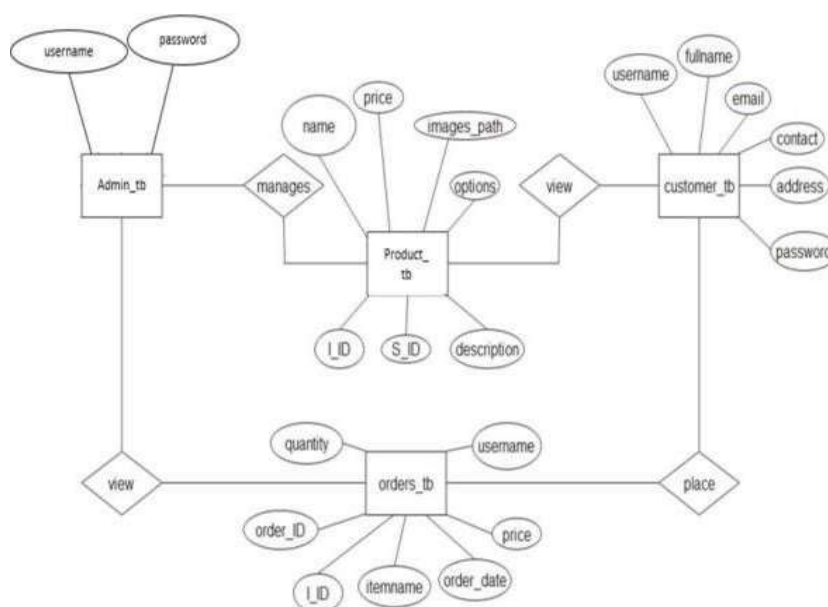


Figure 4:E-R Diagram

### Input design

Input design is the process of converting a user-oriented description to a system-understandable format. Here both mobile phones and PC involved in the input design.

## **User Registration**

Users can give input values such as name, address, contact, email id, and password.

## **Login Details**

A login form will allow the Admin and the User to give input values such as username and password.

## **Item Details**

Item details form will ask the Admin to give input values such as item name, item price, item description, item image etc.

## **Output design**

The output design is GUI based activity; almost from the beginning of the project this project provided a best user interface model. Effects and well define an output design improves the relationship of system and the user, thus facilitating decision making for both seeker and provider.

## **Item Zone**

All the Items will be displayed for the customer.

## **Cart Details**

All the items that have been added to the cart will be displayed.

## **Order Details**

All the order details will be displayed.

## **Item Details**

All the item details will be displayed for the admin.

## **System testing**

### **Software testing**

A primary purpose of testing is to detect software failures so that defects may be discovered and corrected. Testing cannot establish that product functions properly under all conditions but can only establish that it does not function properly under specific conditions.

The scope of software testing often includes an examination of code as well as execution of

that code in various environments and conditions as well as examining the aspects of code: does it do what it is supposed to do and do what it needs to do. In the current software development culture, a testing organization may be separate from the development team. There are various roles for testing team members. Information derived from software testing may be used to correct the process by which software is developed.

Software faults occur through the following processes. A programmer makes an error (mistake), which results in a defect (fault, bug) in the software source code. If this defect is executed, in certain situations the system will produce wrong results, causing a failure. Not all defects will necessarily result in failures. For example, defects in dead code will never result in failures. A defect can turn into a failure when the environment is changed. Examples of these changes in environment include the software being run on a new computer hardware platform, alterations in source data, or interactions with different software. A single defect may result in a wide range of failure symptoms.

### **Unit testing**

Unit testing provides a sort of living documentation of the system. Developers looking to learn what functionality is provided by a unit and how to use it can look at unit tests to gain a basic understanding of the unit's API. Unit test cases embody characteristics that are critical to the success of the unit. These characteristics can indicate appropriate/inappropriate use of a unit as well as negative behaviors that are to be trapped by the unit. A unit test case, in and of itself, documents these critical characteristics, although many software development environments do not rely solely upon code to document the product in development.

Testing will not catch every error in the program, since it cannot evaluate every execution path in any but the most trivial programs. The same is true for unit testing. Additionally, unit testing by definition only tests the functionality of the units themselves. Therefore, it will not catch integrations or broader systems.

Level errors (such as functions performed across multiple units, or non-functional test areas such as performance). Unit testing should be done in conjunction with other software testing activities, as they can only show the presence or absence of particular errors; they

cannot prove a complete absence of errors. In order to guarantee correct behavior for every execution path and every possible input, and ensure the absence of errors, other techniques are required, namely the application of formal methods to proving that a software component has no unexpected behavior.

### **Customer Registration**

All the fields in the customer registration form is inputted with relevant data, once the create account button is clicked all the data is stored in the database.

### **Login Details**

All the fields in the manager/customer login form is inputted with relevant data, once the Submit button is clicked it will switch to their account.

### **Product Details**

All the fields in the add item form is inputted with relevant data, once the add item button is clicked all the data is stored in the database.

### **Stock Details**

All the fields in the add store form is inputted with relevant data, once the add store button is clicked all the data is stored in the database.

### **Integration testing**

**Bottom Up Testing** is an approach to integrated testing where the lowest level components are tested first, then used to facilitate the testing of higher level components. The process is repeated until the component at the top of the hierarchy is tested.

**Top-Down Testing** is an approach to integrated testing where the top integrated modules are tested and the branch of the module is tested step by step until the end of the related module.

### **Customer Registration**

Once all the field in the Customer registration form is inputted and submitted, then it will show a message called “Your Account Has Been Created. Login From Here’ and navigates to login form.

### **Login Details**

All the fields in the manager/customer login form is inputted with relevant data, once the Submit button is clicked, it navigates to their account.

### **Store Details**

Once all the fields in the add store form is inputted with relevant data and add store button is clicked, it will navigates to add item page.

### **System implementation**

System implementation is the practice of creating or modifying a system to create a new business process or replace an existing business process. Technology integration is the practice of integrating multiple system activities to interact and generate information efficiently. In both cases, the system development lifecycle (SDLC) is heavily integrated in these practices. A CPA professional may be involved in designing a system, performing end-user testing, or training others how to utilize the financial system.

As with the other phases of the system development life cycle, the system implementation phase has multiple components or sub-phases. Up to now in the life cycle, the system has been planned, it has been analysed, and the system has been designed. Now, the system needs to be developed, which is known as coding and is the first sub-phase. The second sub-phase is to test the newly coded system on multiple levels. The third-phase is the planning and implementation of the new system. Finally, post-implementation is the final sub-phase of system implementation.

Once the system has been coded, the system goes through different degrees of testing. It is important to make sure the system works and performs the duties the system was intended to perform. The rest of this section deals with the quality assurance process. Based upon Kendal and Kendal, quality should be tested on internal and external factors (2005). The internal factors deals with the duties the systems should perform. The external factors are the perceptions of the customers or other external parties such as suppliers.

## **Conclusion And Future Enhancement**

### **Conclusion**

An online food consignment website is developed for a relative restaurant to simplify their routine managerial and operational task and to improve the shopping experience of the customers. This application nullifies the need of going personally to a restaurant and standing in queues to buy food. The advantage is that in a crowded restaurant, there will be chances of overloaded orders and they are unable to meet the requirements of the customer in a satisfactory manner. Therefore, by using this application, users can directly place an order for the items online without wasting their time by physically wandering around the restaurant. This also helps the store owners develop healthy customer relationships by providing reasonably good services. This application software has been computed & implemented successfully and was also tested successfully. It is user-friendly and has required options, which can be utilized by the user to perform the desired operations. The goals that are achieved by the software are:

1. Instant access.
2. Improved productivity.
3. Optimum utilization of resources.
4. Efficient management of records.
5. Simplification of the operations.
6. Less processing time and getting required information.
7. User friendly.

### **Future Enhancements**

This system is aimed at developing an online purchasing system which can be used in small stores firstly and then on large scale. The future plan of this project is to improve design, implementation and documentation in such a way that anyone can use this project for better performance. This system can be developed more dynamically and the database work as well. The following section describes the work that will be implemented with future releases of the software.

1. Enhance User Interface by adding more user interactive features.
2. Online transaction system.



3. Payment Options: Add different payment options such as Google Pay, Paytm, PayPal, Gift Cards etc.
  4. More security in the system.
- The outcome of the project is shown in Figure 5 to 17.

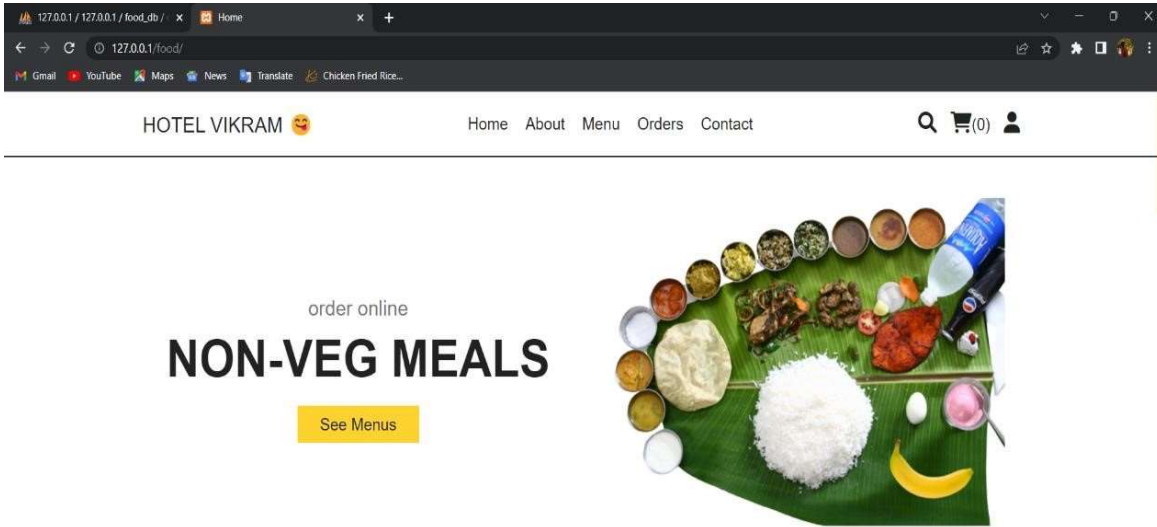


Figure 5: Homepage

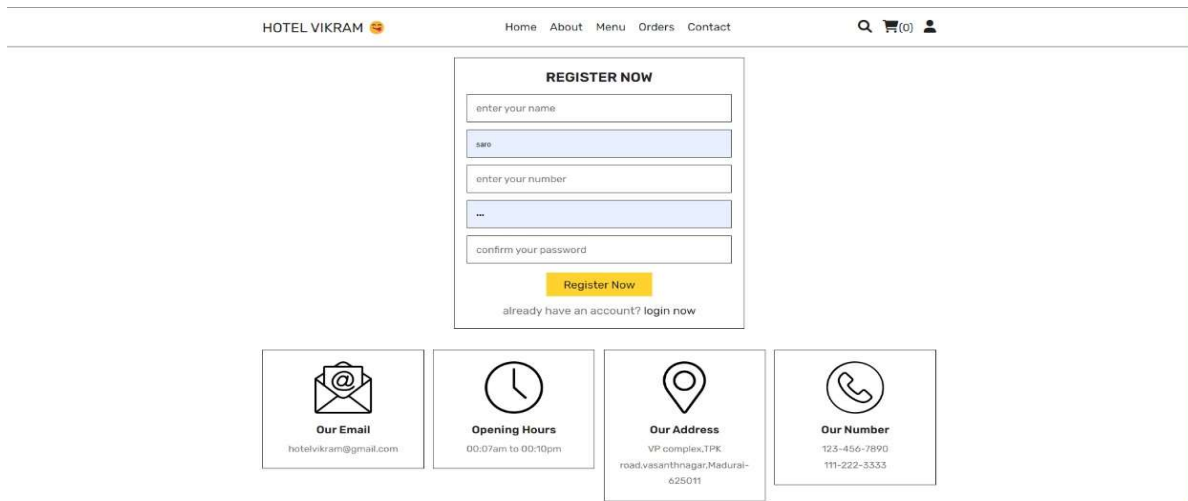


Figure 6: User Registration

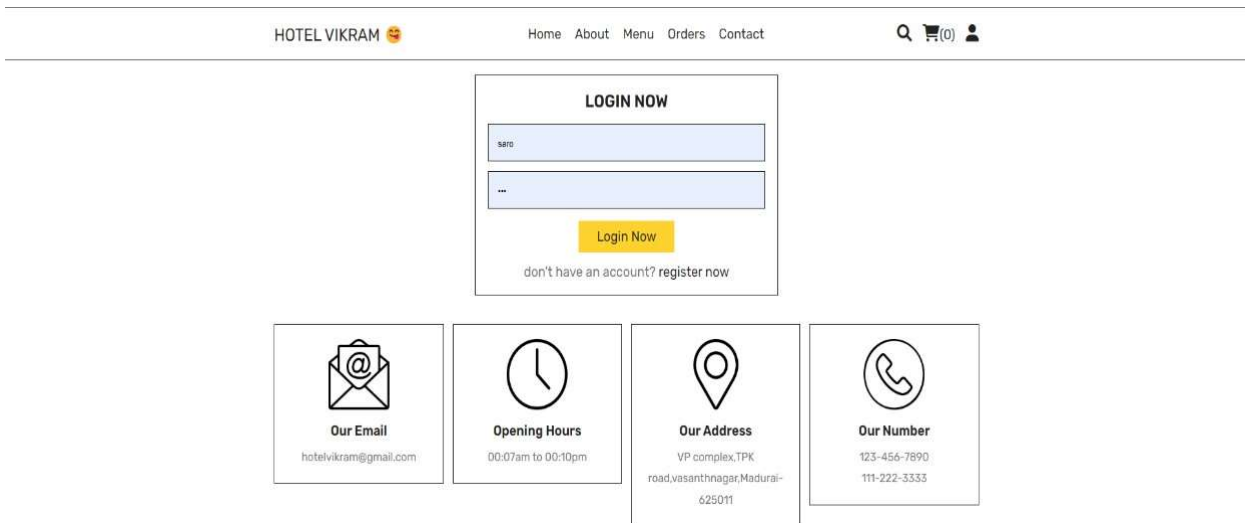


Figure 7: User Login

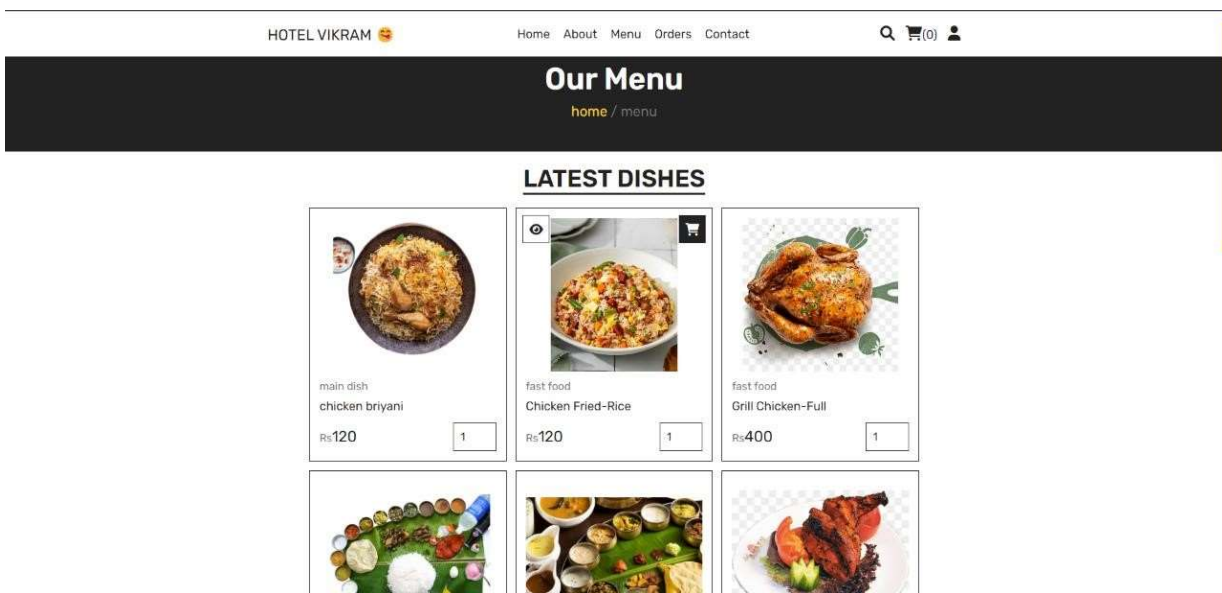


Figure 8: Menu

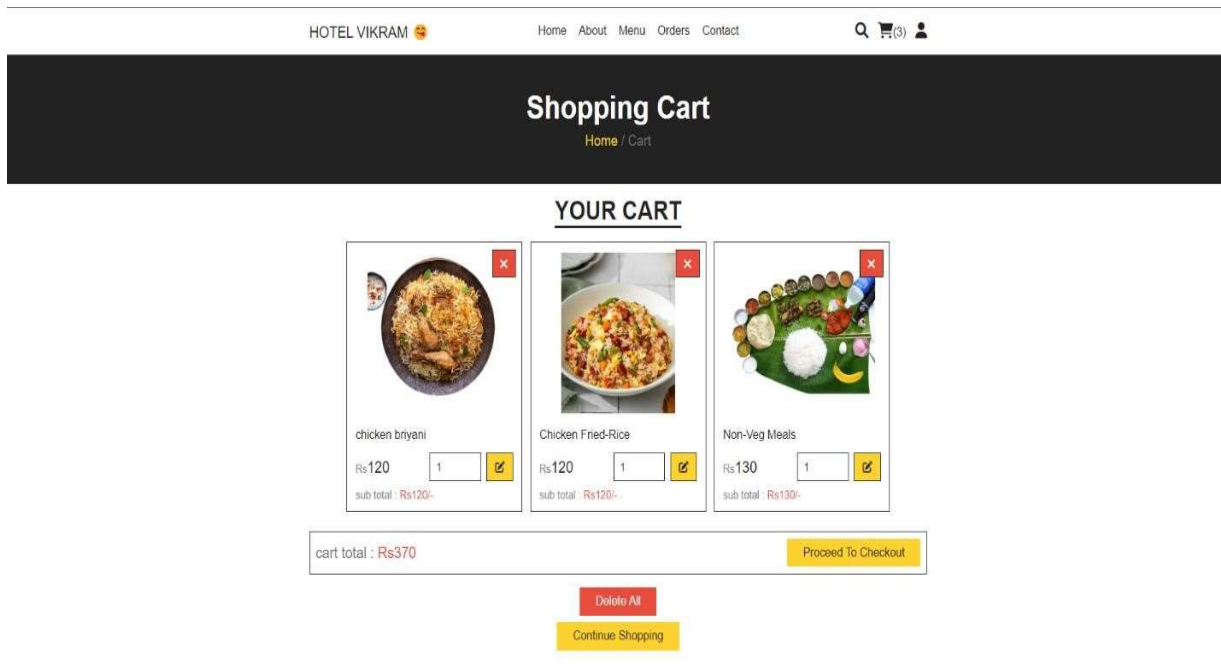


Figure 9: Cart

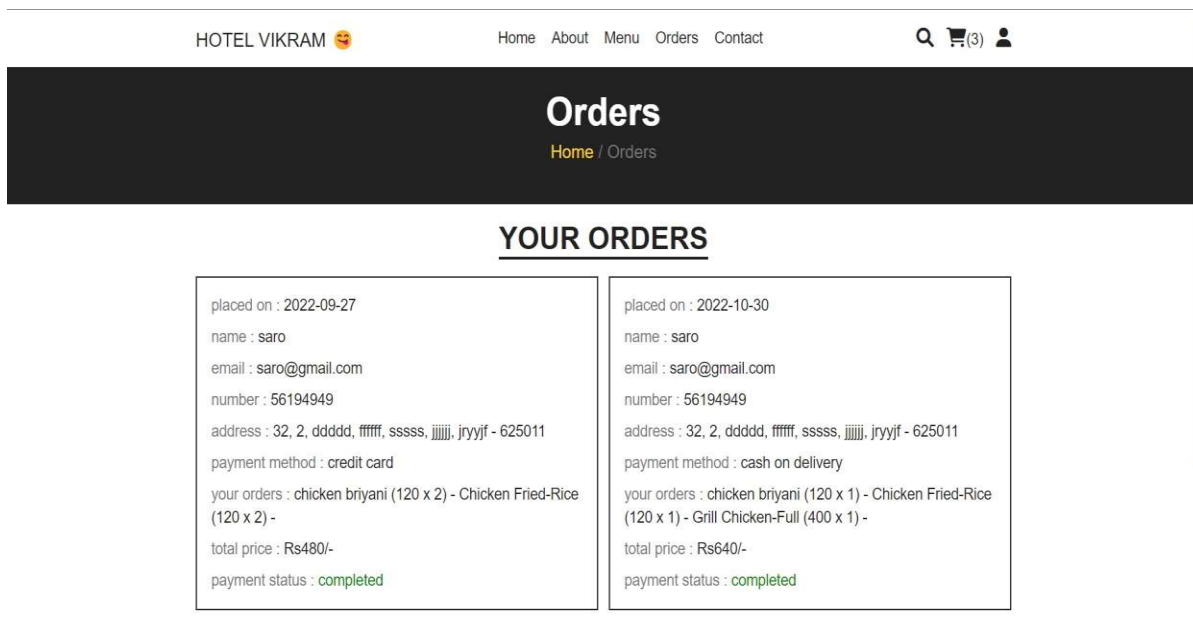


Figure 10: Previous Orders

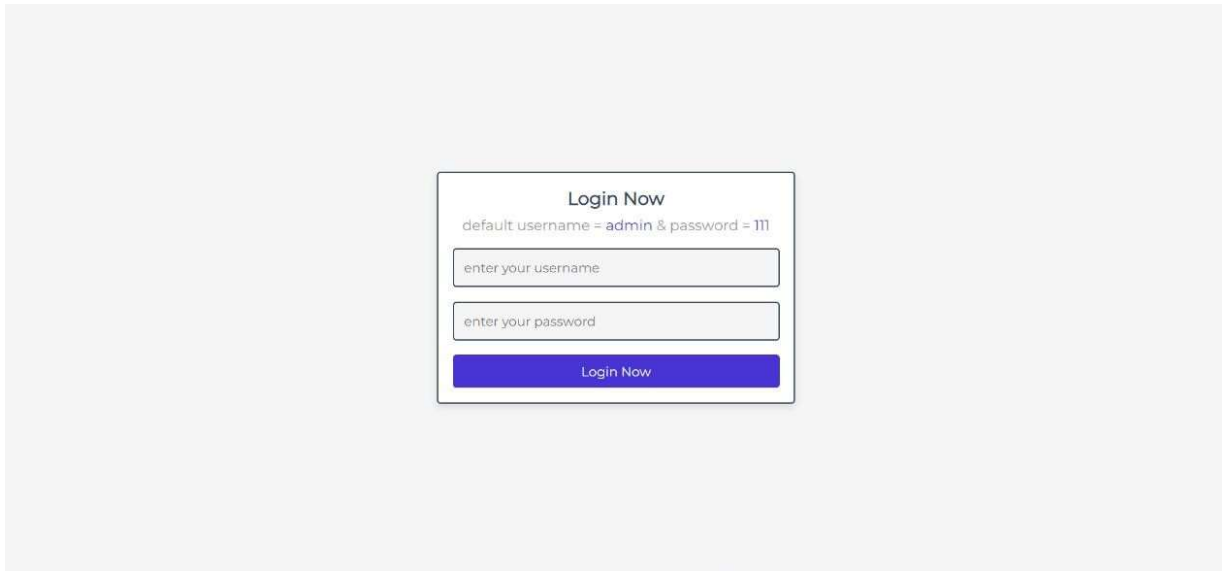


Figure 11: Admin Login

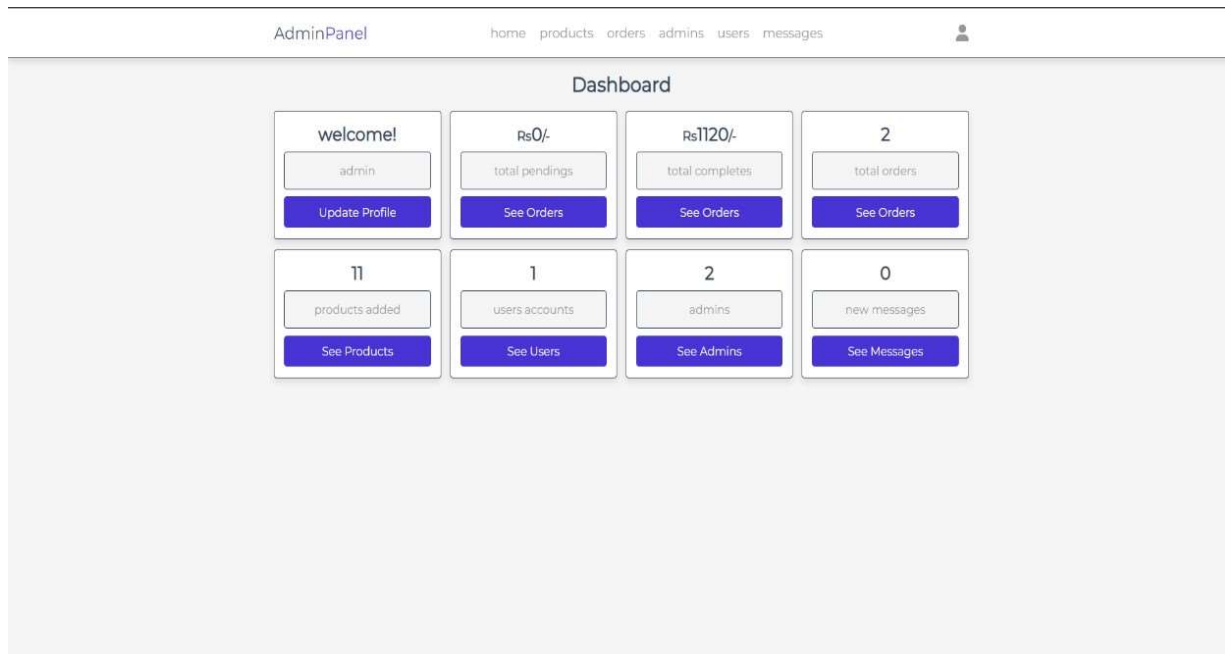


Figure 12: Admin Dashboard

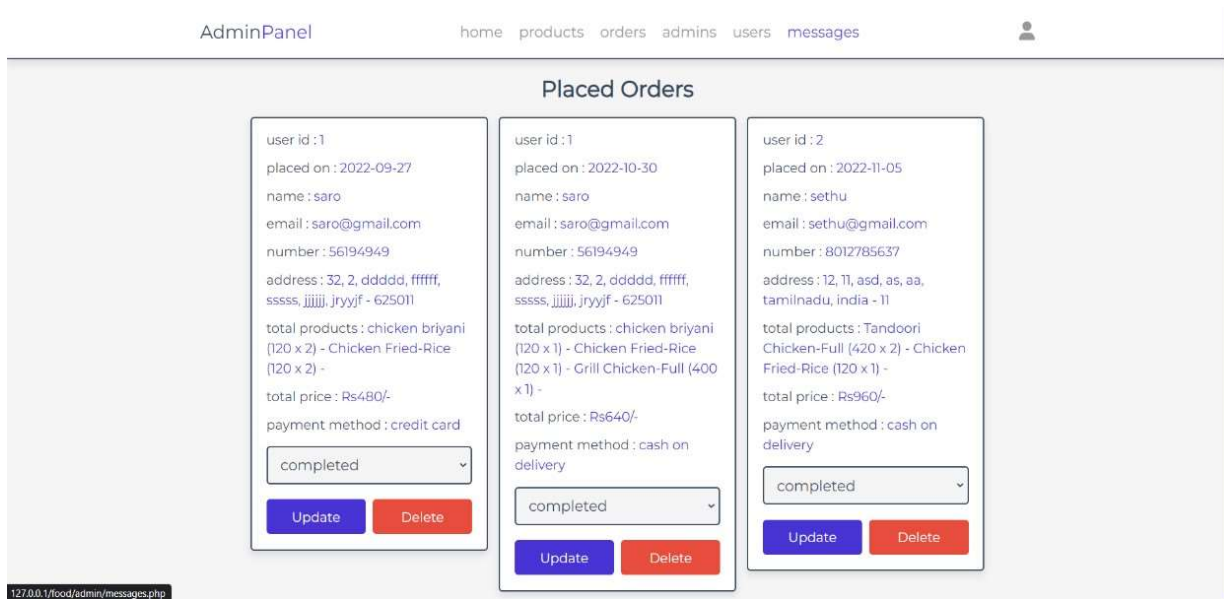


Figure 13: Placed orders

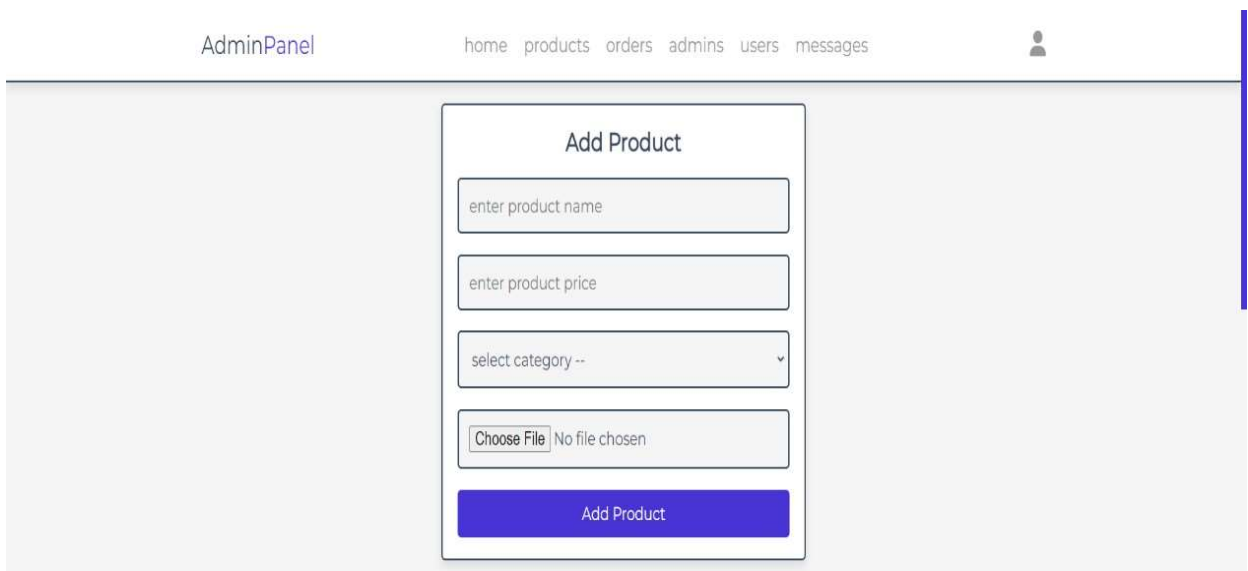


Figure 14: Product updates

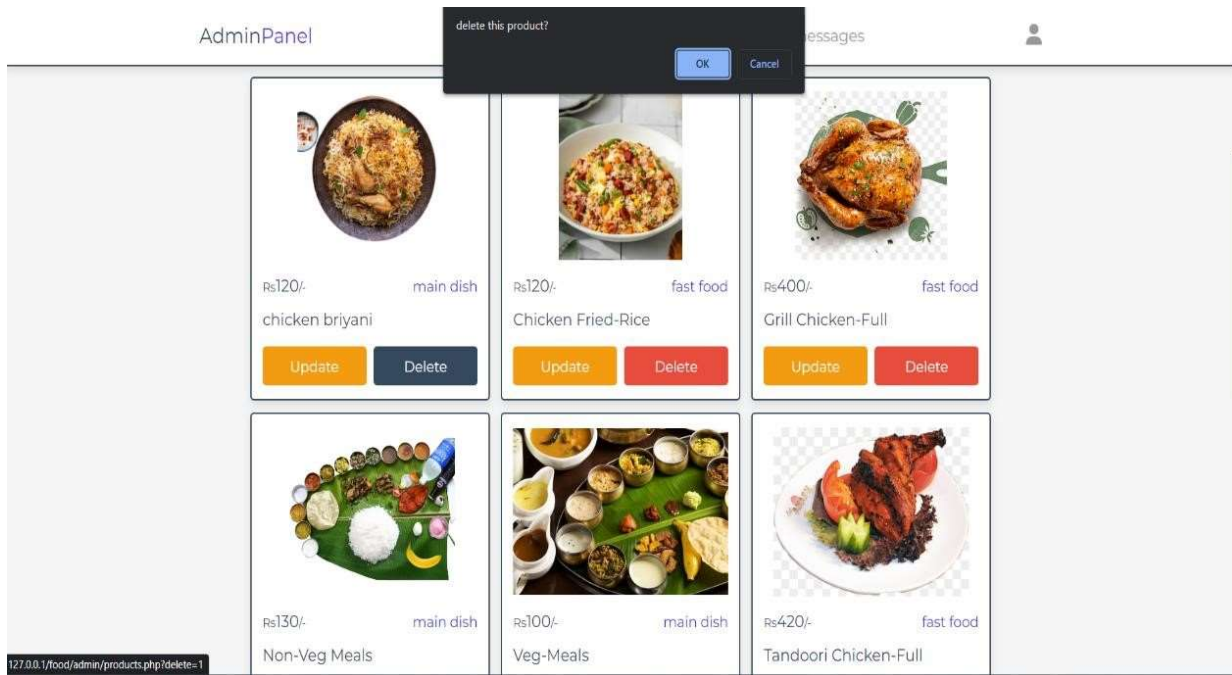


Figure 15: Delete products

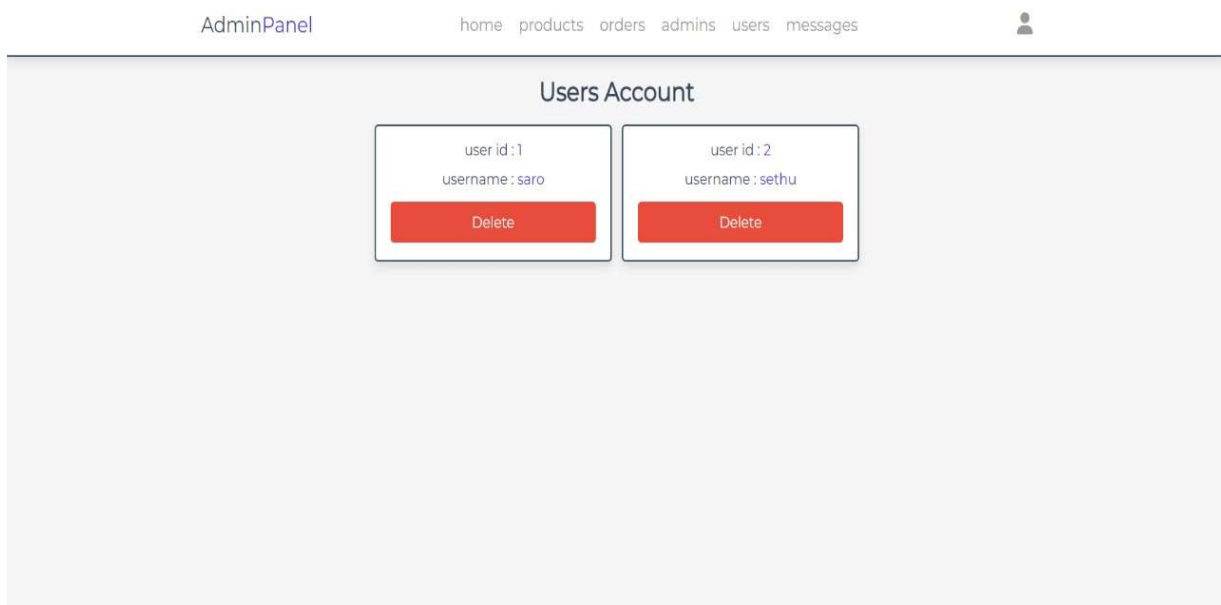


Figure 16: User Accounts

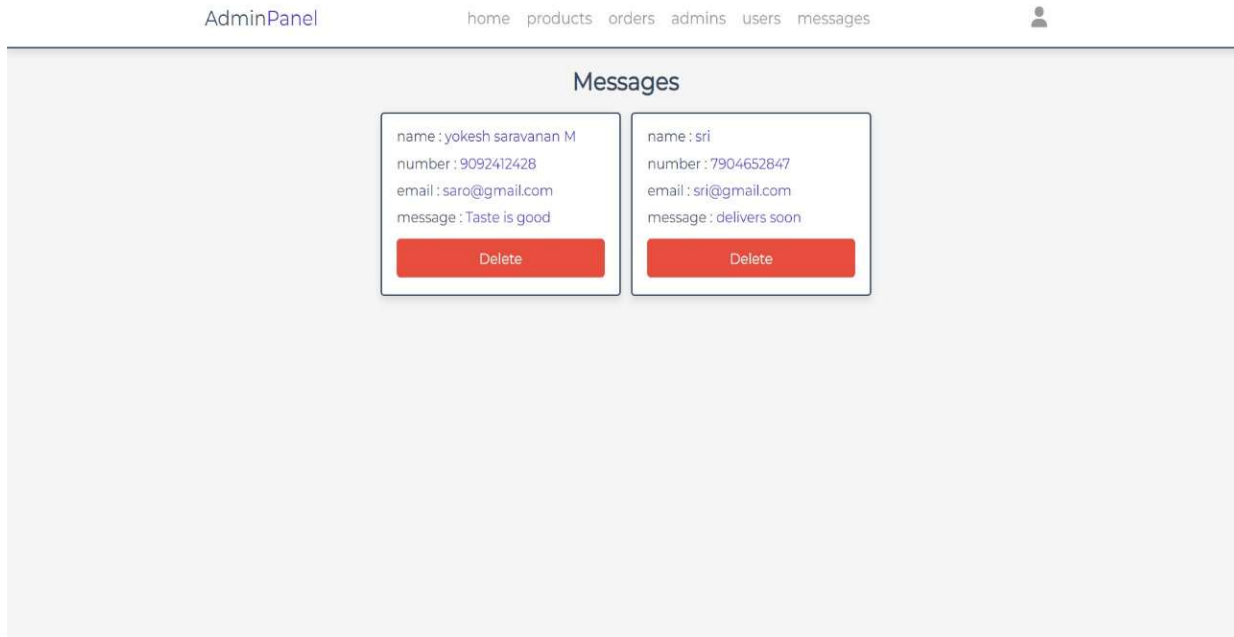


Figure 17: Messages

## References

### Bibliography

- [1] Robin Nixon, Learning PHP, MySQL, JavaScript, and CSS , 4<sup>th</sup> Edition, O'Reilly Media, 2014-11-21
- [2] Tom Butler, Kevin Yank, PHP & MySQL: Novice to Ninja, 6th Edition, SitePoint, October2017
- ISBN: 9780994346988.

### Webliography

- [1]<https://www.w3schools.com> [2]<https://www.tutorialspoint.com>
- [3]<https://youtu.be/uDxZj9V0MAs>@Tutor Joe's Stanley