



Food Ordering Website using Flask

ON

Submitted in partial fulfillment of the requirements
of the degree of

**Bachelor of Engineering
(Information
Technology)**

By

Sanket More - Roll No (29)

Under the guidance of

Dipti Karani



**Department of Information Technology
VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF
TECHNOLOGY,**

Chembur, Mumbai 400074

(An Autonomous Institute, Affiliated to University of Mumbai)

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature
) Sanket More 29

Abstract

This project is a full-stack responsive food ordering website developed using Flask and MongoDB. The platform provides users the ability to browse food items, add them to a cart, and place online orders with secure payments through Stripe. It also includes a contact form powered by Formspree for managing custom and bulk orders. The backend is efficiently designed with Flask and MongoDB, while the frontend uses HTML5, CSS3, Bootstrap, and Jinja2 templates for dynamic rendering. The project incorporates secure user authentication using Flask-Login and offers an admin dashboard for managing menu items. With its modern tech stack and thoughtful user-centric design, the project streamlines the food ordering experience and provides a foundation for scalable growth.

Contents

1. Introduction

- 1.1 Introduction
- 1.2 Objective
- 1.3 Feasibility Study
- 1.4 Organization of the report

2. Design and Implementation

- 2.1 Block Diagram
- 2.2 Hardware Requirements
- 2.3 Software Requirements
- 2.4 Front End Development Framework and Libraries
- 2.5 Back End Development and Database Management

3. Results and Discussion

- 3.1 Results of Implementation
- 3.2 Google Analysis

4. Conclusion

- 4.1 Conclusion

Chapter 1:

Introduction

1.1 Introduction

This project presents a fully functional food ordering platform built using **Flask** as the backend and **MongoDB** as the database. The system is designed to serve both customers and admins with features such as menu viewing, cart management, user authentication, payment processing, and order management. A key highlight is the **Formspree integration** that enables efficient handling of custom bulk orders.

1.2 Objectives:

- Develop a fully responsive food ordering website using Flask and MongoDB.
- Implement secure user authentication using Flask-Login.
- Design an intuitive UX/UI using Bootstrap and Jinja2.
- Integrate Stripe for secure online payments.
- Enable communication through Formspree without storing sensitive order data.

1.3 Feasibility Study:

The project is technically and operationally feasible for deployment by small and mid-scale restaurants aiming to digitize their services. It uses open-source technologies, ensuring cost-effectiveness and flexibility. Scalability is built-in through the use of MongoDB and Flask's modularity.

The increasing demand for digital solutions in the food and beverage industry has made it essential for restaurants to adopt online ordering systems. With changing consumer behavior and the growing use of smartphones, customers now expect seamless and convenient food ordering experiences. This project addresses that need by offering a full-stack web application that allows users to order food from the comfort of their homes. By combining modern technologies with practical design, it bridges the gap between restaurants and their customers in the digital age.

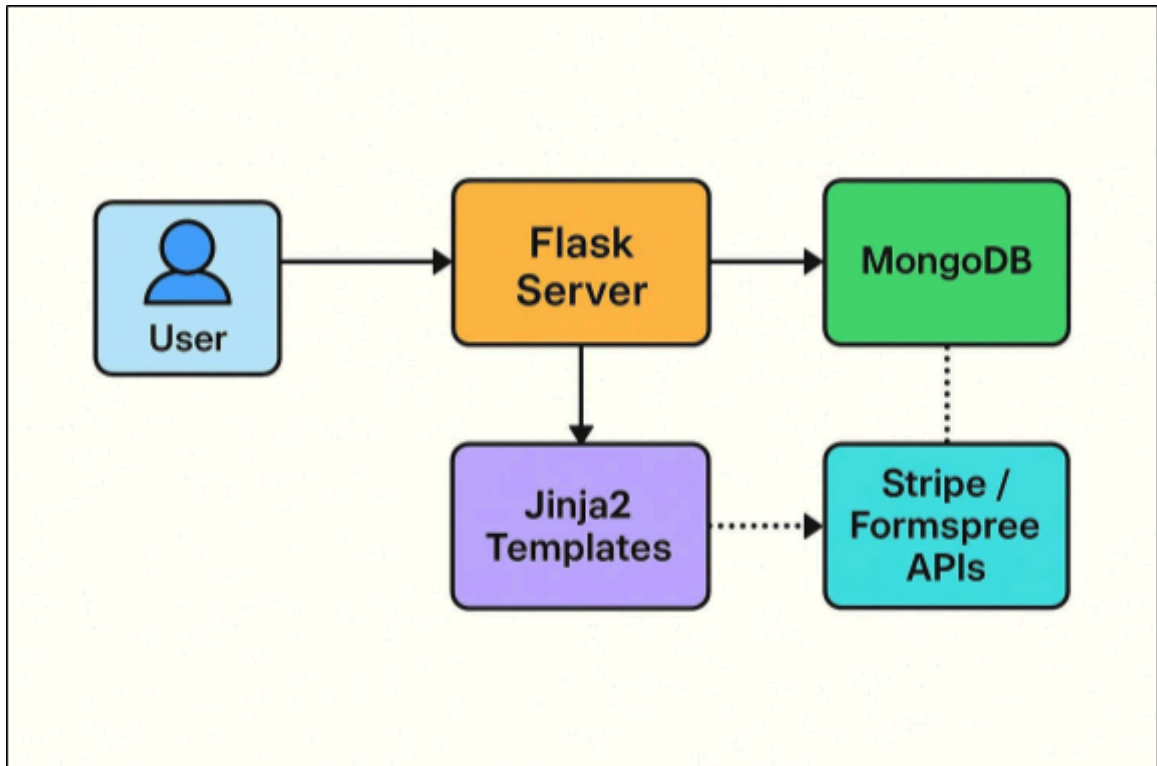
1.4 Organization of the report

The system is not only beneficial for end-users but also provides restaurant administrators with tools to manage their offerings efficiently. Features such as menu management, order tracking, and real-time updates through email ensure that the backend operations remain smooth and organized. This project is a stepping stone toward digital transformation for local restaurants, offering scalability and adaptability for future integrations like mobile apps, AI-based recommendations, and delivery APIs.

Chapter 2:

Design and Implementation

2.1 Block Diagram



1. User Interface (Frontend Layer)

Users (Customers):

They interact with the platform through a web browser. Users can browse the menu, add food items to their cart, and proceed to place an order.

Technologies Used:

- HTML5, CSS3, and Bootstrap for building a responsive and visually appealing interface.
- Jinja2 template engine to dynamically render HTML pages with data passed from the Flask backend.

2. Flask Server (Backend Layer)

Flask Framework:

Acts as the core controller of the system. It manages routing, request-response handling, session management, and acts as the bridge between the frontend and database.

Authentication System:

Implemented using Flask-Login to allow secure user registration, login, logout, and session control. Certain routes are protected and only accessible to authenticated users.

Order Management:

Handles the logic for cart functionality, order processing, and transferring order data to the payment gateway.

Contact Form Integration:

A contact form is provided for users to place bulk orders or inquiries. This is integrated using Formspree, which directly sends submitted queries to the admin's email.

3. Payment Gateway

Stripe API:

Used to handle secure online payments. When a user places an order, payment information is processed through Stripe, ensuring data privacy and security.

4. Database Layer

- **MongoDB:**

A NoSQL database used to store user details, menu items, and order history. It supports flexibility and scalability.

- **MongoDB Compass:**

A GUI tool for managing and visually monitoring collections such as users, food items, and orders

2.2 Hardware Requirements

- **CPU:** Dual-core or Quad-core processor
 - **RAM:** 4 GB or higher
 - **Storage:** SSD with at least 100 GB of free space
 - **Network:** High-speed internet connection
 - **User Devices:** Any modern desktop or laptop with a browser
-

2.3 Software Requirements

Languages and Technologies Used:

- **Frontend:** HTML5, CSS3, Bootstrap, Jinja2
 - **Backend:** Python, Flask
 - **Database:** MongoDB (NoSQL)
-

2.4 Front End Development Frameworks and Libraries:

- **HTML5 & CSS3:** For structuring and styling web pages
 - **Bootstrap:** For responsive layout and component design
 - **Jinja2:** For rendering dynamic content from backend to frontend
 - **JavaScript (Basic):** For enhancing interactivity in UI components
-

Tools and Editors:

- **Visual Studio Code:** Code editor used for development
 - **Python (3.x):** Backend programming language
 - **MongoDB Compass:** GUI for managing and visualizing database content
 - **Git & GitHub:** Version control and project collaboration
 - **Browser:** Google Chrome or any modern browser for testing and running the app
-

2.5 Backend Development & Database Management:

- **Flask:** Python microframework used to build backend routes, logic, and API interactions
 - **MongoDB:** NoSQL database for storing food items, orders, and user data
 - **Flask-Login:** For user authentication and session management
 - **Stripe API:** For integrating secure payment processing
 - **Formspree API:** For handling contact form submissions via email
-

Development and Hosting (Optional for Future Deployment):

- **Hosting Platforms:**
 - **Render / Heroku / PythonAnywhere** (for Flask app deployment)
 - **MongoDB Atlas:** For hosting MongoDB in the cloud
 - **Netlify (optional):** If frontend were separated from Flask

Chapter 3:

Results

Hosted Link : <https://sanket29.onrender.com>

Github Link:
https://github.com/moresanket4003/webx_lab

3.1 Results of Implementation:

Login Page:

Log In

Email

Password

[new user? click here](#)

LOG IN

Create an Account

Name

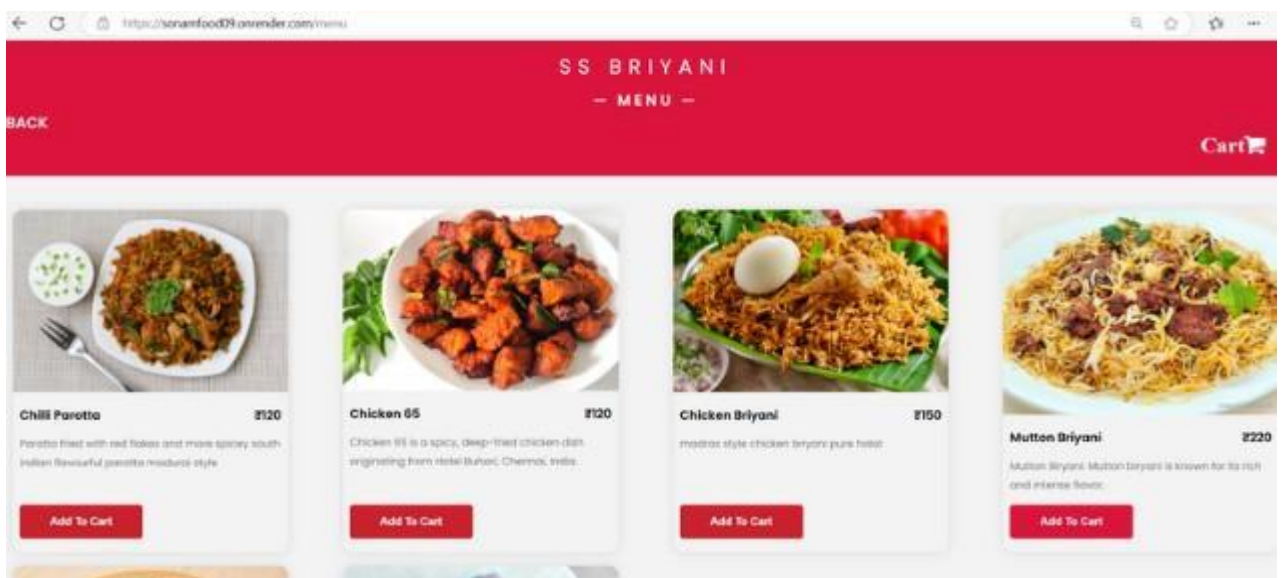
Email

Password

SIGN UP

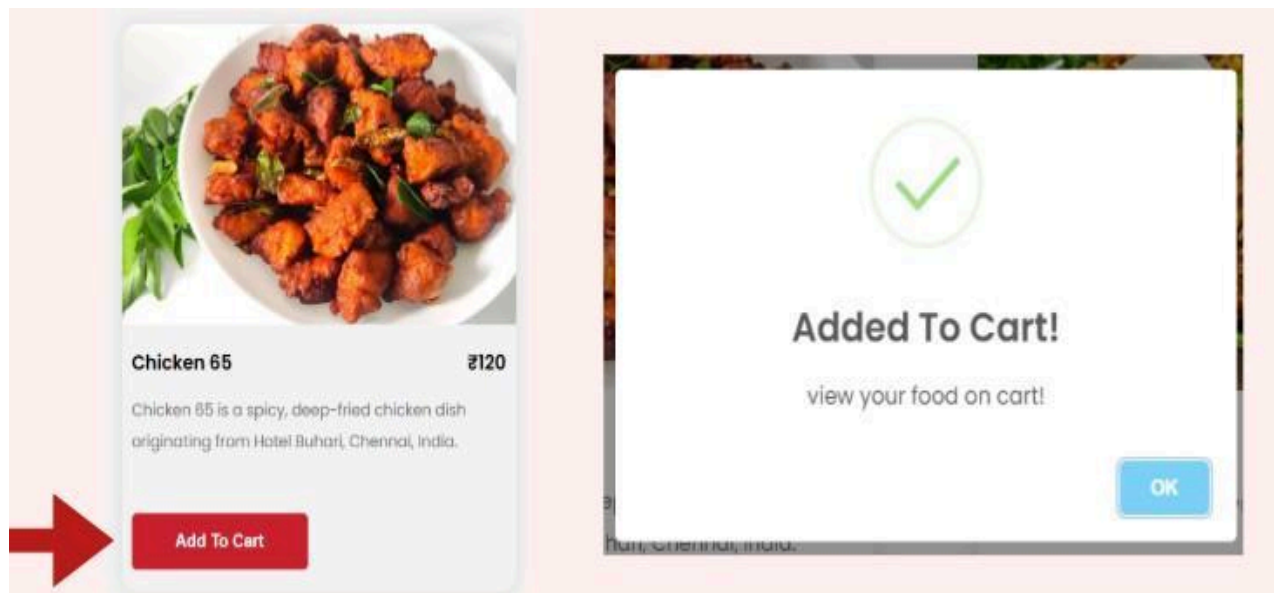
LOGIN SIGN UP

View food items:

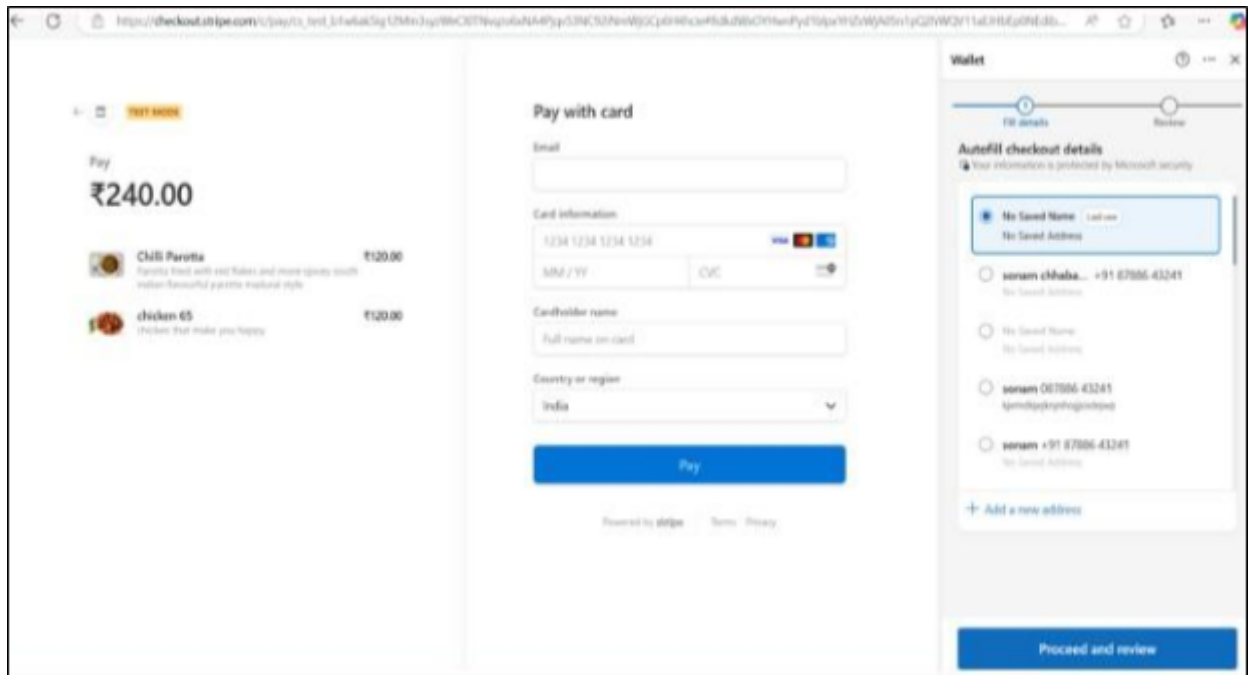




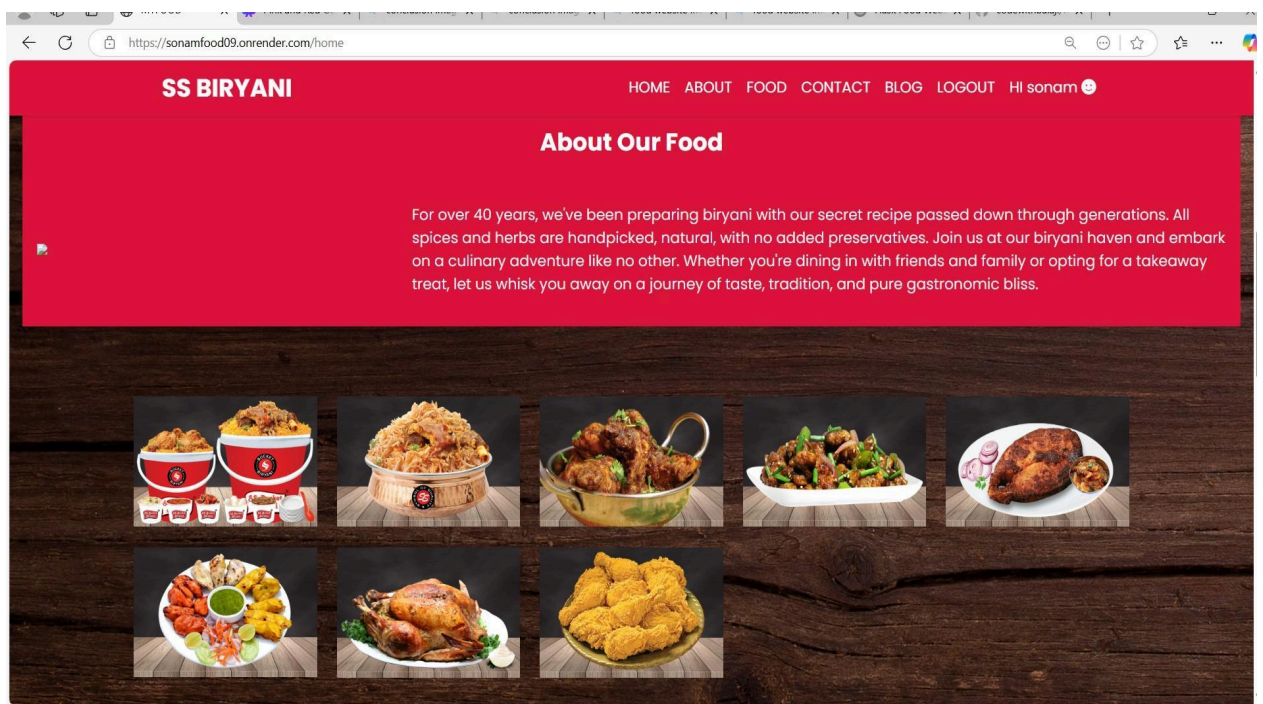
Add items to cart:



Place order and Payment Gateway:



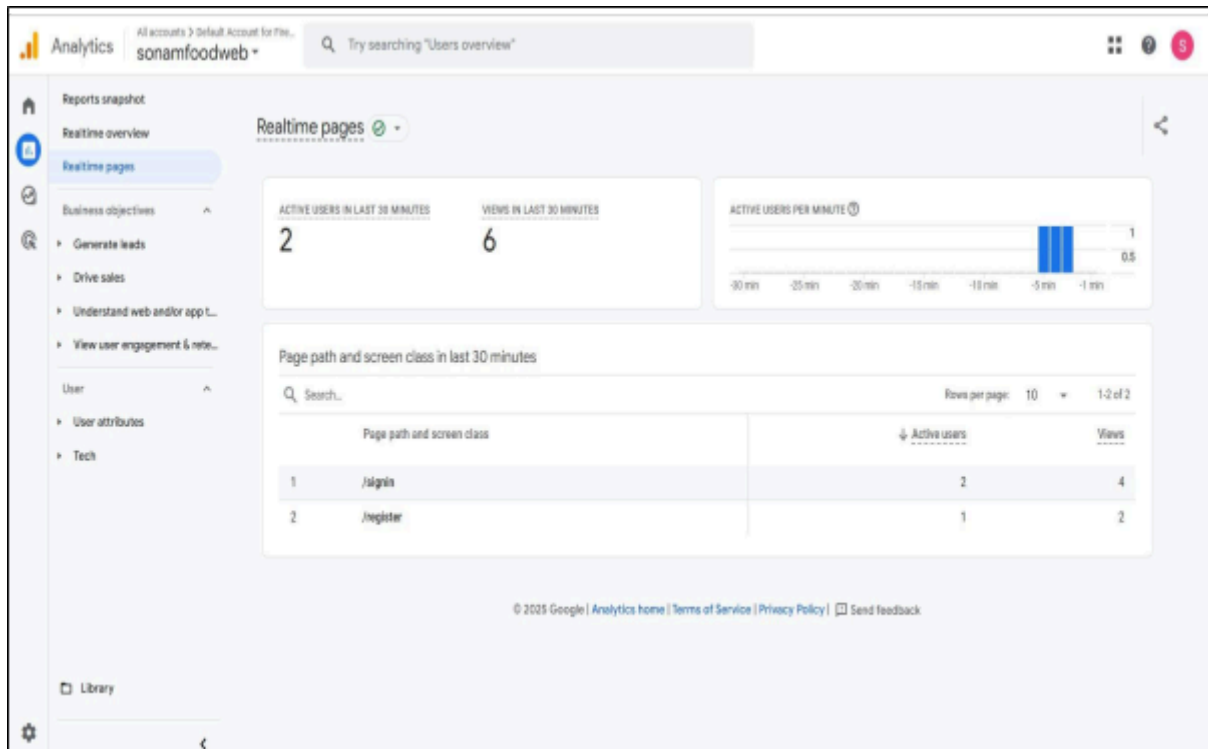
About Us Page:

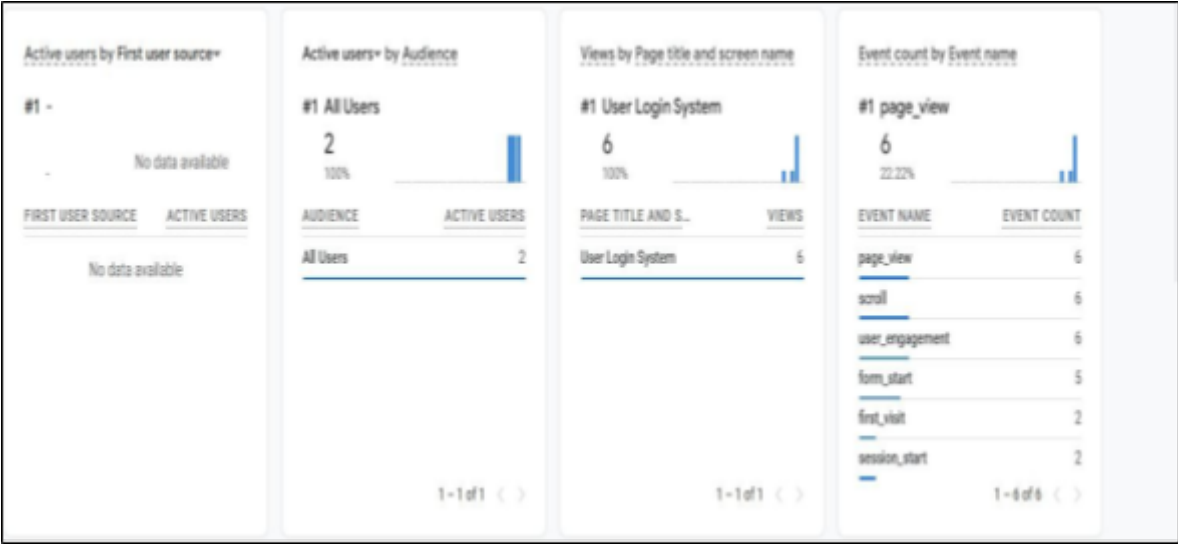


Formspre Form:

The screenshot displays the SS Biryani website. The top navigation bar is red with white text for 'HOME', 'ABOUT', 'FOOD', 'CONTACT', 'BLOG', 'LOGOUT', and 'Hi sonam'. The main content area has a dark wood-grain background. A large red banner reads 'For Bulk Orders & Party Orders'. Below this, on the left, is a white contact form with fields for 'Your Name', 'Enter Your Email...', 'Enter Your Mobile no...', and a text area for 'Write Message Here...'. A red 'SUBMIT' button is at the bottom of the form. On the right is a map of Hyderabad, India, with red pins marking various SS Biryani locations such as 'SS Hyderabad Biryani', 'SS Hyderabad Biryani Kaladpet', 'SS Hyderabad Biryani Royapuram', 'SS Hyderabad Biryani Light House', 'SS Hyderabad Biryani Sadaipet', 'SS Hyderabad Biryani Adityar', 'SS Hyderabad Biryani Chromepet', 'SS Hyderabad Biryani Medavakkam', and 'SS Hyderabad Biryani Kalyandurg'. The map includes a 'View larger map' link and a 'Report a map error' link.

3.2 Google Analytics





Chapter:4

4.1 Conclusion

The Food Ordering Website project demonstrates the effective use of full-stack development to solve real-world problems in the restaurant industry. By integrating technologies such as Flask, MongoDB, Stripe, and Formspree, the project successfully delivers a responsive and user-friendly platform that simplifies the food ordering process for both customers and restaurant administrators.

The system provides essential functionalities including secure user authentication, interactive menu browsing, cart management, online payments, and real-time communication through contact forms. Admins can easily manage menu items, while users benefit from a smooth and intuitive interface. The responsive design ensures that the platform is accessible across all devices, enhancing usability and user engagement.

This project highlights the potential of web development in automating and optimizing business processes. With a solid foundation in place, the website is highly scalable and can be extended in the future to include real-time order tracking, AI-based food recommendations, and a mobile application. Overall, the project is a comprehensive solution that aligns with modern digital expectations and demonstrates strong practical application of web technologies.