International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 03 | March - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Food Ordering System

Dr.K.Anandan[1] , P.Deepakpandi[2]

[1]Assistant Professor , Department of Computer Applications,

Nehru college of Management , Coimbatore , Tamilnadu , India

[2]Student of II MCA, Department of Computer Applications,

Nehru college of Management, Coimbatore, Tamilnadu, India

ABSTRACT:

online food ordering systems have revolutionized the way people order and enjoy meals. This project presents a Food Ordering System developed using Django as the backend framework and MongoDB as the database. The system enables users to browse restaurant menus, place orders, make payments, and track order statuses in real time. It also provides restaurant administrators with a user-friendly interface for managing menus, tracking orders, and handling customer interactions. By leveraging Django's robust framework and MongoDB's flexible NoSQL data structure, the system ensures scalability, efficiency, and seamless user experience.

KeyWords: Django, MongoDB, Tracker.

I.INTRODUCTION:

The increasing reliance on digital platforms for everyday activities has significantly impacted the food industry, leading to the widespread adoption of online food ordering systems. Traditional food ordering methods, such as phone calls or walk-in reservations, often result in delays, miscommunication, and inefficiencies. This project proposes a Django-based food ordering system with MongoDB, offering a streamlined, automated, and user-friendly solution for customers and restaurants alike. The system simplifies food ordering by allowing users to browse menus, place orders, and make payments online, while restaurant owners can efficiently manage their business operations.

Django, a high-level Python web framework, is chosen for its robust security, scalability, and rapid development capabilities. It provides a structured backend, handling user authentication, order management, and business logic effectively. MongoDB, a NoSQL database, is used for its flexibility and scalability, allowing the system to handle dynamic data structures, such as menu changes, customer preferences, and real-time order updates. The combination of Django and MongoDB ensures fast query performance, real-time data processing, and efficient handling of large datasets.

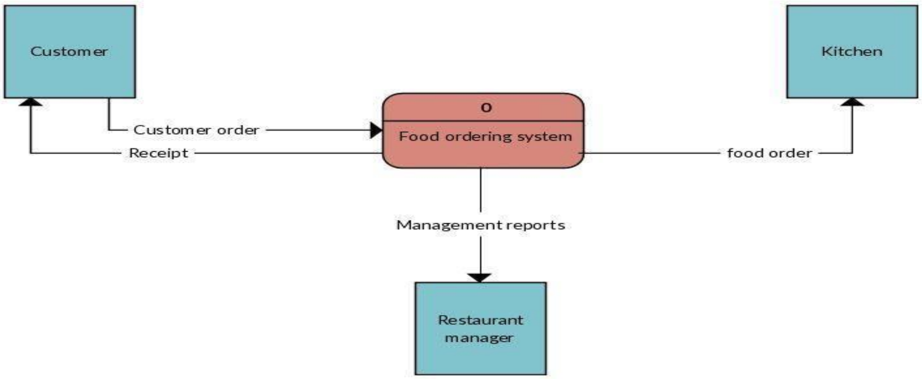


Fig.1 Work Flow

II. SCOPE OF STUDY:

The Food Ordering System using Django and MongoDB focuses on enhancing the efficiency of digital food ordering and restaurant management. The scope of this study covers three primary areas: customer interaction, restaurant management, and system scalability. The system enables customers to browse menus, place orders, make payments, and track order status in real-time. Additionally, it provides restaurants with tools to manage their menus, update food availability, and process customer orders seamlessly. From a technical perspective, the study explores the integration of Django with MongoDB, emphasizing its benefits in handling dynamic data structures, real-time updates, and user authentication. Django's built-in security mechanisms help ensure data privacy, while MongoDB's NoSQL architecture provides flexibility and scalability to support a growing number of users and transactions. The study also considers the implementation of RESTful APIs for seamless communication between the frontend and backend, enhancing the overall performance of the system. The study is limited to web-based platforms and does not cover mobile application development, though future enhancements may include a mobile-friendly interface. It primarily focuses on small to medium-sized restaurants but can be scaled for larger enterprises with additional features such as AI-driven recommendations and predictive analytics. The research also evaluates user experience, security measures, and system efficiency, ensuring that the proposed solution meets industry standards for modern food delivery services.

III. METHODOLOGY:

DESIGN:

The design of the Food Ordering System using Django and MongoDB focuses on a modular, scalable, and user-friendly architecture. The system is structured into three primary components: the frontend (user interface), backend (Django framework), and database (MongoDB).

1. SYSTEM ARCHITECTURE:

The system follows a Model-View-Controller (MVC) architecture, ensuring clear separation of concerns between data handling, business logic, and user interface.

- ➢ Frontend (UI/UX Design) Built using HTML, CSS, JavaScript.
- ➢ Backend (Django Framework) Handles business logic, authentication, order processing, and communication between the database and frontend.
- ➢ Database (MongoDB) A NoSQL document-based database that stores user information, menus, orders, and transaction history efficiently.

2. DATABASE DESIGN:

The system leverages MongoDB's flexible schema to store dynamic data, including:

- ➢ Users Collection: Stores user details such as name, email, password, and order history.
- ➢ Restaurants Collection: Contains restaurant information, including menus, pricing, and availability.
- ➢ Orders Collection: Tracks order status, delivery time, payment details, and assigned restaurant.
- ➢ Payments Collection: Records transaction details, payment status, and method of payment.

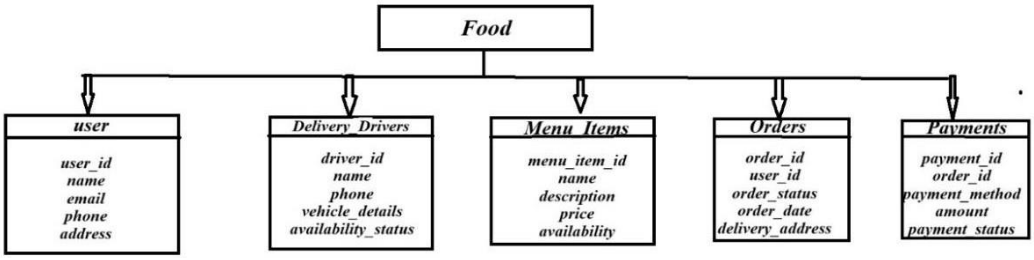


Fig.2 DataBase Design

1. **User Table:**
   - Stores information about customers who use the platform to place orders**.**
   - Attributes: user_id (Primary Key), name, email, phone, and address.
   - Each user can place multiple orders.

2. **Delivery Driver Table:**
   - Maintains details of delivery personnel responsible for delivering food.
   - Attributes: driver_id (Primary Key), name, phone, vehicle_details, and availability_status.
   - The availability_status helps in assigning deliveries.

3. **Menu Item Table:**
   - Contains details of the food items available for ordering.
   - Attributes: menu_item_id (Primary Key), name, description, price, and availability.
   - Each menu item can be ordered multiple times by different users.

4. **Orders Table:**
   - Stores records of food orders placed by users.
   - Attributes: order_id (Primary Key), user_id (Foreign Key referencing User Table), order_status, order_date, and delivery_address.
   - Each order belongs to a specific user, and the order status helps track progress.

5. **Payment Table:**
   - Manages transactions related to food orders.
   - Attributes: payment_id (Primary Key), order_id (Foreign Key referencing Orders Table), payment_method, amount, and payment_status.
   - Ensures that payments are tracked and linked to corresponding orders.

IV. FEATURE AND FUNCTIONS :

USER AUTHENTICATION:

User authentication is a critical component of the Food Ordering System to ensure secure access for customers, restaurant owners, and administrators. The system employs Django's built-in authentication framework, enhanced with

JWT (JSON Web Token) authentication for secure and scalable login sessions. The authentication module is designed to handle user registration, login, role-based access control, and session management.

MENU MANAGEMENT:

The Menu Management Module is a crucial component of the Food Ordering System that allows restaurants to efficiently manage their menus. This module enables restaurant owners to add, update, delete, and categorize food items, ensuring that customers have access to an up-to-date menu. Built using Django as the backend framework and MongoDB as the database, this module ensures flexibility and scalability, allowing real-time menu updates without affecting system performance.

ORDER PROCESSING:

The Order Processing Module is a core component of the Food Ordering System, responsible for handling the entire lifecycle of a food order, from placement to delivery. This module ensures efficient order management, real-time tracking, and seamless communication between customers and restaurants. The module is designed using Django as the backend framework and MongoDB as the database for flexible and scalable data storage.

PAYMENT INTEGRATION:

The Payment Integration Module in the Food Ordering System using Django and MongoDB ensures a seamless, secure, and efficient transaction process for users. It allows customers to make payments for their orders using multiple payment methods, ensuring flexibility and ease of use. This module is designed to integrate with various payment gateways such as PayPal, Stripe, Razorpay, and UPI-based transactions to provide multiple options for online payments.

V.RESULT:

Views and URLs are then set up to handle the display of food items, the process of placing an order, and the confirmation of successful orders. Templates are used to render the food menu, the order placement form, and a success message after the order is placed.

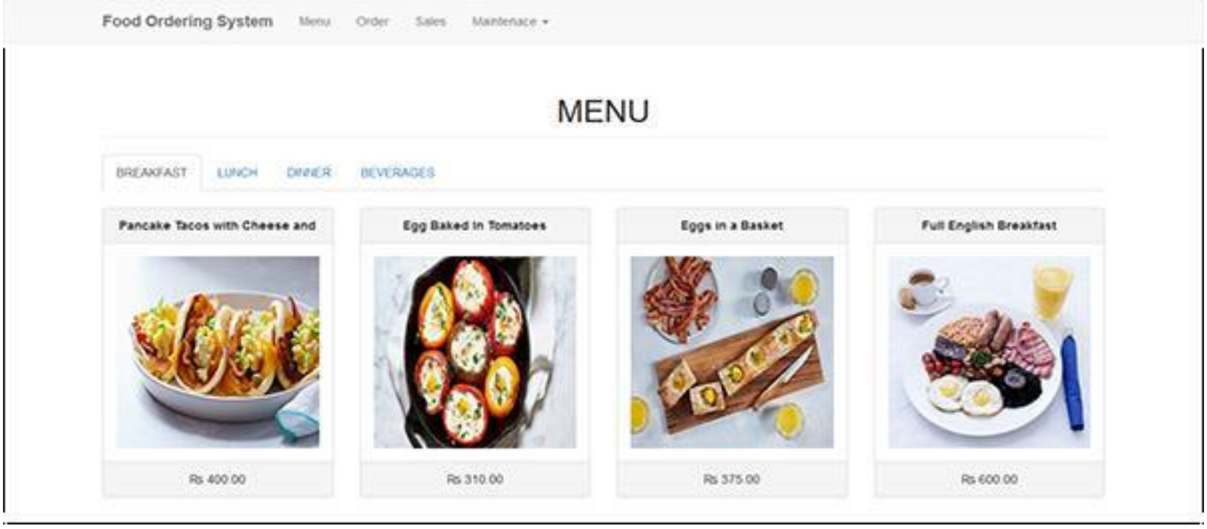
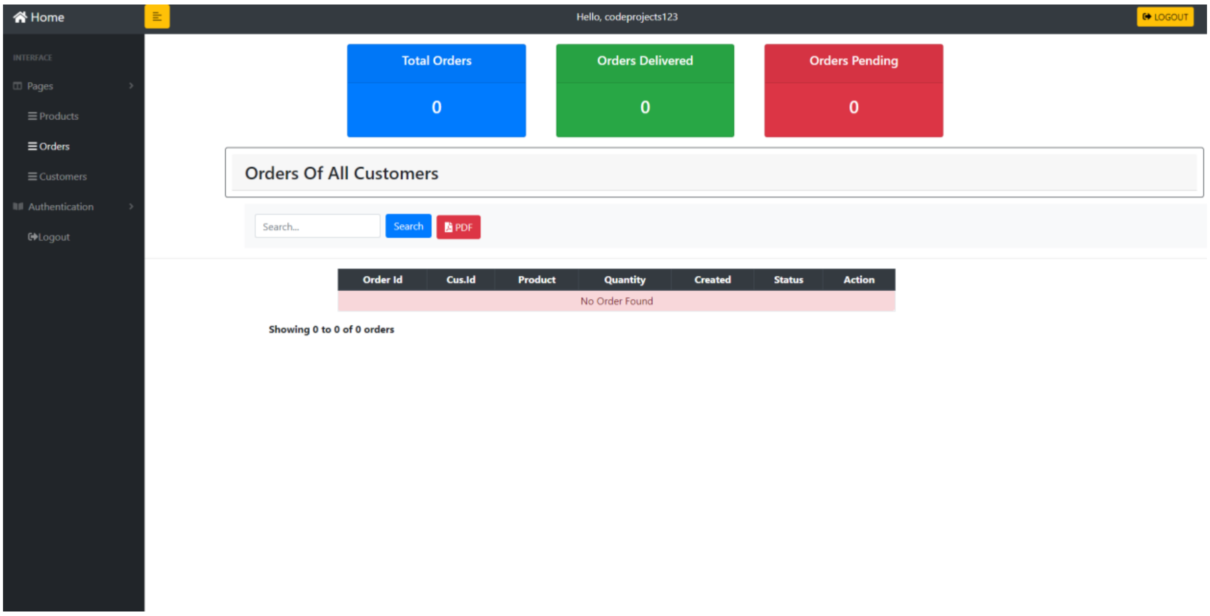


Fig.3 Sample Photo



Orders Of All Customers

Order Id	Cus.Id	Product	Quantity	Created	Status	Action
No Order Found						

Showing 0 to 0 of 0 orders

Fig.4 Admin Panel

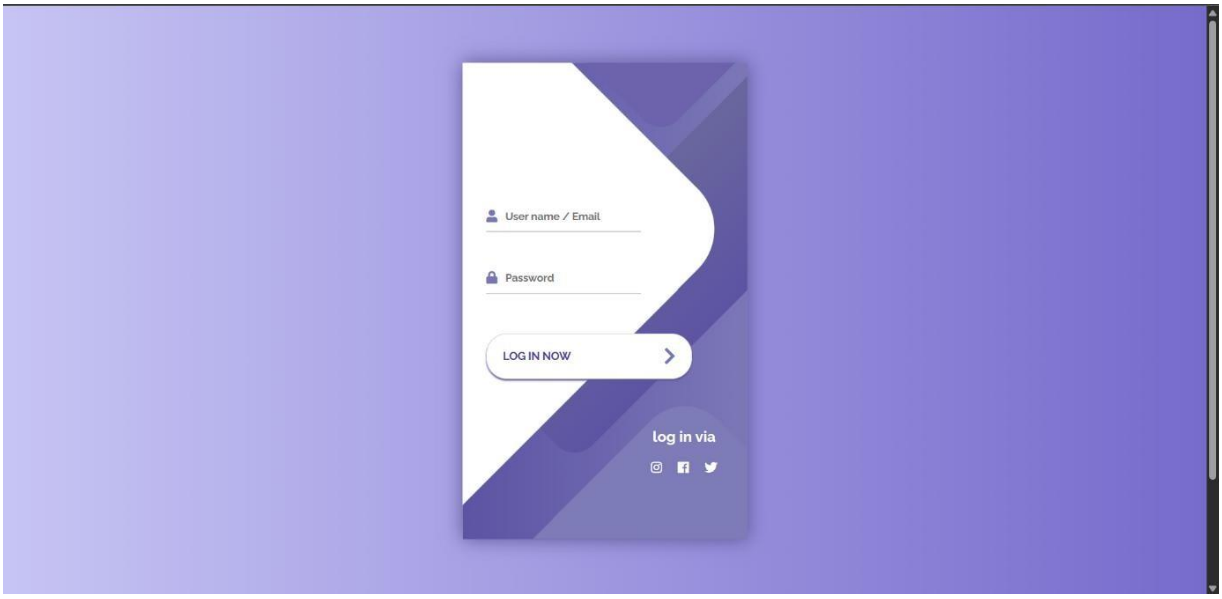


Fig.5 Login Panel

VI. CONCLUSION:

In conclusion, this project demonstrates the potential of web-based food ordering systems in transforming the traditional food industry. By reducing order mismanagement, enhancing efficiency, and providing a seamless digital experience, the system meets the growing demands of online food services. Future enhancements may include AI-driven recommendations, automated chatbots, and mobile app extensions, further improving the platform's usability and expanding its reach to a broader audience.

REFERENCES:

1. R. Adithya, A. Singh, S. Pathan and V. Kanade, (2017). Online food ordering system. International Journal of Computer Applications [Online]. Available: https://doi.org/10.5120/ijca2017916046

2. N. N. Li, X. F. Zhang, Y. F. Wang, and R. Zhang, "Design and Implementation of Campus Dining Application Based on Android," Applied Mechanics and Materials, vol. 556–562, pp. 5250–5254, May 2014, doi: 10.4028/www.scientific.net/amm.556-562.5250. [Online]. Available: http://dx.doi.org/10.4028/www.scientific.net/amm.556-562.5250

3. R. Aulia, A. Zakir, H. Dafitri, D. Siregar, and Hasdiana, "Mechanism of Food Ordering in A Restaurant Using Android Technology," Journal of Physics: Conference Series, vol.930, p. 012030, Dec. 2017, doi: 10.1088/1742-6596/930/1/012030. [Online]. Available: http://dx.doi.org/10.1088/1742- 6596/930/1/012030

4. Dr. Vinayak, V. Ranjan, N. Masiwal, and N. Verma, "e-Restaurant: Online Restaurant Management System for Android," International Journal of Advanced Computer Science and Applications, vol. 3, no. 1, 2013, doi: 10.14569/specialissue.2013.030108. [Online]. Available: http://dx.doi.org/10.14569/specialissue.2013.030108

5. Mayur Kumar Patel. "Online Food Order System for Restaurants," Computer Information Systems, Grand Valley State University, ScholarWorks@GVSU, December 2015.

6. Anitta Abraham. "A Study on the effectiveness Of Online Food Applications on Registered Restaurants," International Journal of Creative Research Thoughts (IJCRT) ISSN: 2320-2882, Volume 9, Issue 1 January 2021.

7. Prof Upendra More, Prof Ria Patnaik, Reema Shah." "A Study on Online Food delivery services during the COVID -19 in Mumbai", Thakur Global Business School & Thakur Institute of Management Studies & Research, friend arch's journal of archaeology of Egypt, PJAEE, 18 (7) (2021).