Food Orderig Application System.

Technological Requirements: HTML, CSS, JAVASCRIPT, BOOTSTRAP, SQL Server, .NET Framework, ASP.NET

- ## System Purpose and Scope:

  ➢ To enhance the food ordering experience, streamline operations for restaurant owners, and provide a seamless platform for both customers and restaurant owners and to improve customer convenience in ordering food

  ➢ To enable restaurant owners to efficiently manage orders and customer interactions and to provide an integrated platform for order management, payment processing, and customer feedback

Presented By: Elvin and Ananya Kamalakar Naik

- **Functional Requirements:**

1. **Data Management:**

   ➢ Collect, read, retrieve, and store customer and order data securely and efficiently.

   ➢ Facilitate data entry and updates by authorized users.

   ➢ Enable data search and filtering based on various criteria (e.g., cuisine type, restaurant location).

   ➢ Implement robust data backup and disaster recovery procedures.

2. **Clinical Support :**

   ➢ Provide an intuitive interface for customers to browse menus, place orders, and make payments.

   ➢ Enable order tracking and status updates.

   ➢ Facilitate communication between customers and restaurant owners for order modifications and inquiries.

   ➢ Allow for customer reviews and ratings.

3. **· Administrative Functions:**

   ➢ Manage restaurant profiles and menu items.

   ➢ Facilitate order processing and real-time order updates.

   ➢ Handle payment processing and transaction records.

   ➢ Generate reports for sales analysis and business insights.

   ➢ Track inventory and manage stock levels.

DLithe ®

Be Agile..Be in Demand

- **Non-Functional Requirements:**

    1. **Performance**

        ➢ Ensure fast response times and efficient data access.

        ➢ Optimize system performance to handle high traffic volumes during peak times.

    2. **Cost :**

        ➢ Focus on development and implementation costs.

        ➢ Consider maintenance and data storage costs.

    3. **Usability**

        ➢ Design a user-friendly interface with comprehensive documentation and demo for easy onboarding.

        ➢ Ensure the application is accessible to all users based on their roles and permissions.

    4. **Scalability:**

        ➢ Ability to handle an increasing number of users and orders.

        ➢ Scalable architecture to support future growth and additional features.