

1. Introduction

1.1 Purpose

The purpose of a food delivery system is to provide a convenient and efficient way for customers to order food online and have it delivered to their desired location .A food delivery system is a software application that allows customers to order food online and have it delivered to their homes or offices. The system typically allows customers to browse a digital menu, place and pay for their order online, and receive the order details via the chosen online food ordering system. The restaurant or food outlet then produces the order ready for delivery or customer pick up .The system also ensures the safety of data in the database and the privacy of personal data.

1.2 Document Conventions

➤ Entire document should be justified.

➤ Convention for Main title

Font face: Times New Roman

Font style: Bold

Font Size: 14

➤ Convention for Sub title

Font face: Times New Roman

Font style: Bold

Font Size: 12

➤ Convention for body

Font face: Times New Roman

Font Size: 12

1.3 Scope of Development Project

The scope of food delivery project development includes various aspects of the food delivery process. This includes deciding whether to develop an order-only app or an order-delivery app.The scope of food delivery project development includes determining what features and functionalities to include in the app.The user registration, menu browsing, order placement, payment processing, and order tracking.Professional development team: A professional mobile app development team is required to build a custom food delivery. hiring for roles such as mobile app developers, project managers, and quality assurance testers.

1.4 Definitions, Acronyms and Abbreviations

MOU: Memorandum of Understanding

MRE: Meal Ready to Eat

CSOS: Controlled Substances Ordering System

NRA: National Restaurant Association

QSR: Quick Service Restaurant

CIAA: Confederation of the Food and Drink Industries of the EU

GDA: Guideline Daily Amount

JAVA : platform independence

SQL:Structured query Language

ER: Entity Relationship

UML : Unified Modeling Language

IDE:Integrated Development Environment

SRS: Software Requirement Specification

1.5References

1.) Kumar .H., Jain, M., & Bajwa, M. S. (2021). Online Food Delivery App ‘Foodie’. Journal of University of Shanghai for Science and Technology, 23(8), 761-771.
2.)Smith, L., & Blake, H. (2022). Investigating Experiences of Frequent Online Food Delivery Service Use: A Qualitative Study in UK Adults. BMC Public Health, 22(1), 1-12. .

2.Overall Descriptions

2.1 Product Perspective

1. Market Research and Analysis:

Conduct thorough market research to understand your target audience, competition, and emerging trends in the food delivery industry.

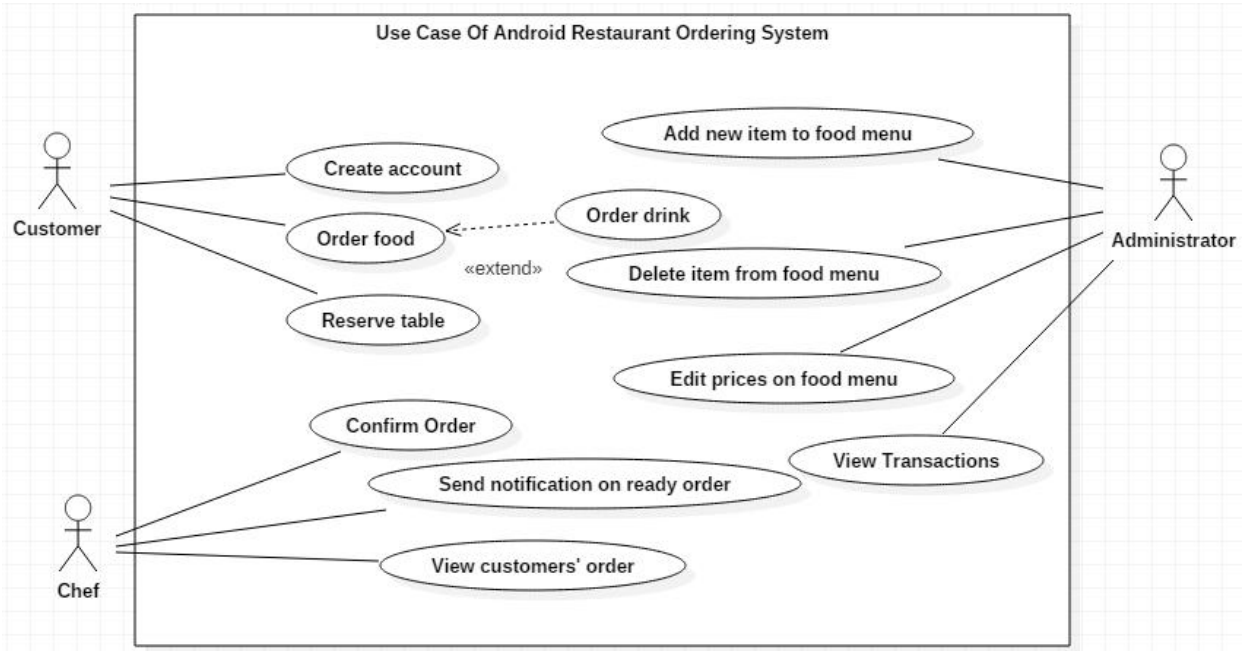
Identify your unique selling points and differentiators.

2.Business Strategy:

Define your business model (commission-based, subscription-based, or a combination).

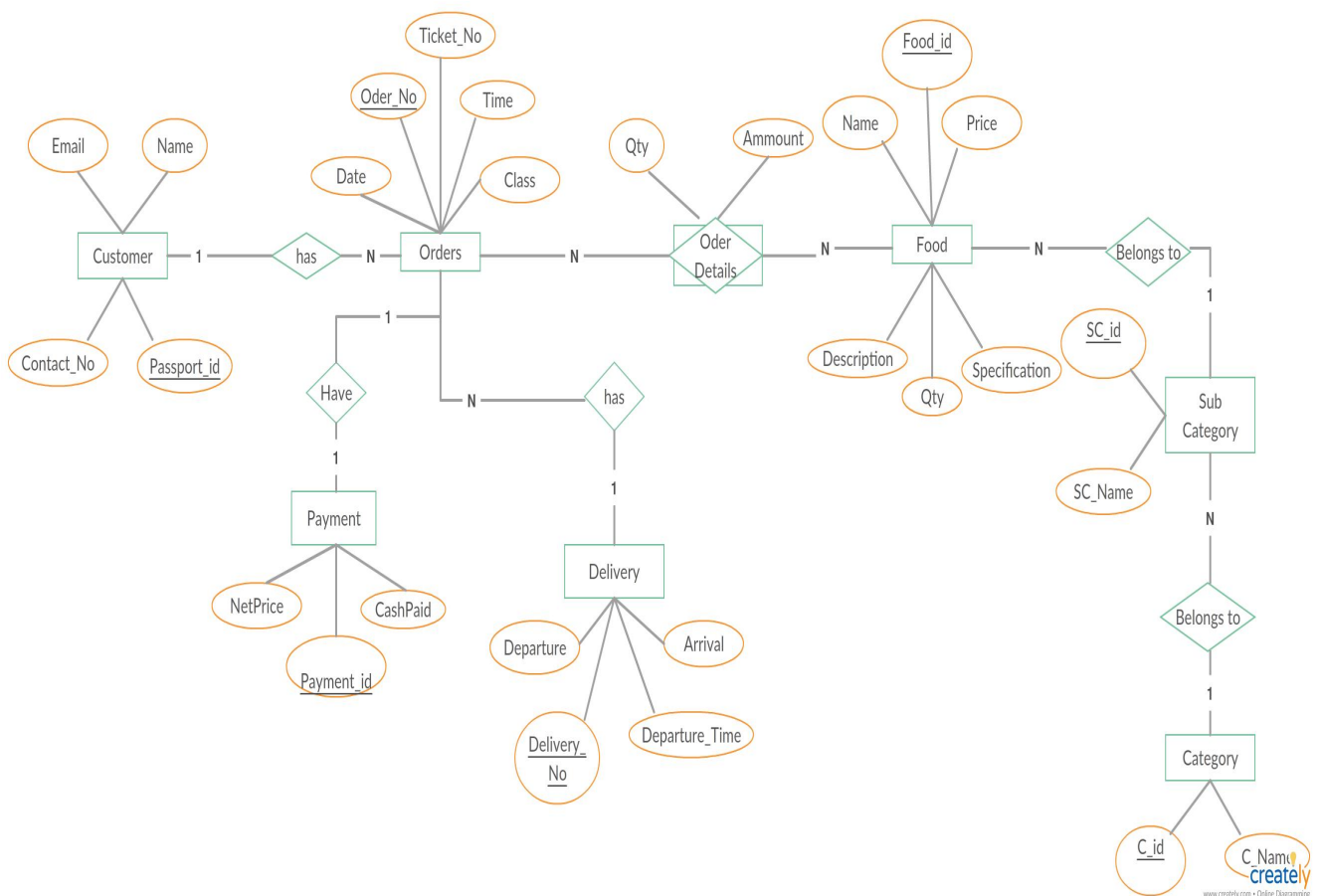
Determine pricing strategies for restaurants and delivery charges for customers.Create a marketing and growth strategy.

Use Case Diagram of food delivery system



2.2 Product Function

Entity Relationship Diagram of food delivery website and application



Customer: This entity represents the person who places an order. It could have attributes such as name, email, phone number, and passport.

Order: This entity represents the details of an order, such as the items ordered, the total cost, and the delivery time. It could have a many to one relationship with the customer entity, as multiple orders can be placed by the same customer.

Food: This entity represents the food items available for ordering. It could have attributes such as name, description, price, and category (e.g., appetizers, entrees, desserts). The menu item entity could have a many to many relationships with the order entity, as described above.

2.3 User Classes and Characteristics

1. Customers:

Characteristics:

- They are individuals looking to order food for delivery or pickup.
- They may have varying dietary preferences or restrictions.
- They may seek a user friendly interface for browsing menus, placing orders, and tracking deliveries.
- They may want to save their preferences, payment information, and delivery addresses for convenience.

Needs:

- Easy to use app or website for browsing restaurants and menus.
- Secure and convenient payment options.
- Ability to track orders in real time.
- Access to reviews and ratings for restaurants.

2. Restaurants:

Characteristics:

- They are businesses offering food for delivery or pickup.
- They may have different cuisines, menu sizes, and specialties.
- They may want to manage their menu, prices, and availability.
- They may want to receive and process orders efficiently.

Needs:

- Easy to use dashboard for managing menus, prices, and orders.
- Reliable notification system for new orders.
- Integration with a secure payment gateway.

3. Delivery Drivers:

Characteristics:

- They are individuals responsible for delivering orders from restaurants to customers.
- They need to be reliable and punctual.
- They may use their own vehicles for delivery.

Needs:

- Access to a mobile app for receiving and managing delivery requests.
- GPS navigation to reach customers' locations.
- Clear instructions and order details for efficient delivery.

4. Administrators/Moderators:**Characteristics:**

- They are responsible for managing the platform.
- They need access to administrative features for monitoring and controlling the system.
- They may handle issues related to user accounts, payments, and content moderation.

Needs:

- Comprehensive dashboard for managing users, restaurants, and drivers.
- Tools for handling customer support and disputes.
- Analytics and reporting tools for business insights.

5. Customer Support:**Characteristics:**

- They assist customers and resolve any issues they encounter.
- They may need access to user accounts and order details for troubleshooting.

Needs:

- Access to a customer support dashboard with user information and order history.
- Communication tools for responding to customer inquiries.

2.4 Operating Environment**1. Technological Infrastructure:**

- Servers and Hosting: The platform requires reliable servers and hosting services to ensure the website and app are accessible 24/7 without downtime.
- Database Management: A robust database system is needed to store user accounts, restaurant information, menus, orders, and other relevant data.
- Content Delivery Network (CDN): A CDN can enhance the speed and performance of the website by caching content and distributing it across multiple servers geographically.
- Payment Gateway Integration: Secure and reliable payment processing systems must be integrated to handle online transactions securely.
- Security Protocols: Firewalls, SSL certificates, and encryption protocols are crucial for protecting user data and maintaining the security of the platform.

- **Mobile Development Platforms:** Depending on the app's platform (iOS, Android, or both), appropriate development tools and frameworks are required.

2. Software and Applications:

- **Operating Systems:** The platform must be compatible with popular operating systems like Windows, mac OS, iOS, and Android to cater to a wide user base.
- **Web Browsers:** The website should be compatible with major web browsers such as Chrome, Firefox, Safari, and Edge for optimal user experience.
- **Mobile Devices:** The mobile app should be designed to work on various devices with different screen sizes and resolutions.

3. Networking and Connectivity:

- **Internet Connectivity:** A stable and high speed internet connection is crucial for users to access the platform and place orders without interruptions.
- **GPS and Location Services:** The app may rely on GPS and location services for tracking orders and providing accurate delivery information.

4. Regulatory Compliance and Legal Considerations:

- **Data Privacy and Security:** The platform must comply with data protection laws and implement measures to safeguard user information.
- **Payment Processing Regulations:** Adherence to financial regulations and secure handling of payment information is essential.
- **Food Safety and Hygiene:** If the platform handles food preparation or delivery, compliance with local health and safety regulations is necessary.

5. Physical Environment (for In Person Operations):

- **Restaurant Partners:** Establishing partnerships with local restaurants for food preparation and pickup is crucial.
- **Delivery Fleet:** If the platform manages its own delivery service, a fleet of drivers and vehicles may be necessary.

2.5 Assumptions and Dependencies

Assumptions:

- 1. Assumption:** Users have access to a compatible device and reliable internet connectivity.

Rationale: The platform relies on internet access for users to browse, place orders, and receive updates.

2. Assumption: Users are familiar with basic smartphone and web navigation.

Rationale: The platform assumes a level of digital literacy to interact with the website and app.

3. Assumption: Restaurants have the capacity to prepare and fulfill orders in a timely manner.

Rationale: The success of the platform depends on restaurants' ability to handle order volumes.

4. Assumption: Delivery drivers have a valid driver's license and a suitable mode of transportation.

Rationale: For platforms managing their own delivery fleet, drivers' eligibility and reliability are crucial.

Dependencies:

1. Dependency: Integration with Payment Gateways.

Rationale: The platform depends on secure and reliable payment processing to facilitate transactions.

2. Dependency: Reliable Internet Service Providers (ISP s) and Hosting Services.

Rationale: The platform relies on stable internet connections for both users and back end operations.

3. Dependency: GPS and Location Services.

Rationale: Accurate location information is crucial for tracking orders and providing accurate delivery estimates.

4. Dependency: Restaurant Partnerships and Menu Integration.

Rationale: The availability of a wide range of restaurants and menus is essential for user choice and satisfaction.

2.6 Requirement

1. User Authentication and Registration: Users should be able to create accounts, log in, and securely manage their profiles.

- 2. Restaurant Listings and Menus:** Display a list of partnered restaurants with their menus, including categories, item descriptions, prices, and images.
- 3. Search and Filters:** Provide robust search and filtering options to help users find specific cuisines, dishes, or restaurants easily.
- 4. Order Placement:** Allow users to add items to their cart, review their order, and proceed to checkout.
- 5. Payment Integration:** Integrate secure payment gateways to allow users to pay for their orders online. This may include credit/debit cards, digital wallets, and other payment methods.
- 6. Order Confirmation and Tracking:** Send order confirmation notifications to users and provide real time tracking of their orders.
- 7. Delivery and Pickup Options:** Offer users the choice between delivery and pickup, with options for specifying delivery addresses and pickup times.
- 8. Delivery Fee and Minimum Order Amount:** Clearly communicate any applicable delivery fees and minimum order amounts.
- 9. Reviews and Ratings:** Allow users to leave reviews and ratings for restaurants and their food, helping others make informed decisions.

2.7 Data Requirement

1. User Data:

Customer Information:

Name, email address, phone number, and password for account creation and authentication.

Delivery Address:

Addresses for delivery, and potentially multiple addresses per user for convenience.

Payment Information:

Credit/debit card details, digital wallet information, or other payment methods for processing orders.

Order History:

Records of past orders for users to review and reorder.

Preferences:

Favorite restaurants, cuisines, or specific dishes for personalized recommendations.

2. Restaurant Data:

Basic Information:

Name, address, contact information, and hours of operation.

Menu Details:

List of dishes, including names, descriptions, prices, and images.

Cuisine Type:

Categorization of restaurants by cuisine (e.g., Italian, Chinese, etc.).

Reviews and Ratings:

User generated feedback on restaurant quality.

3. Menu Item Data:

Name and Description:

Clear names and detailed descriptions of each menu item.

Price:

Cost of each item.

Availability:

Indicate if an item is currently available or not.

4. Order Data:

Order Details:

List of items in the order, along with quantities.

Total Cost:

The overall cost of the order.

Order Status:

Information on whether the order is pending, in progress, out for delivery, or completed.

Timestamps:

Time of order placement, confirmation, and delivery.

5. Delivery Data:

Driver Information:

Details about the assigned driver, including name and contact information.

Delivery Status:

Real time tracking information and updates on delivery progress.

Delivery Confirmation:

Timestamp and confirmation that the order has been delivered.

3.External interface requirements

3.1 GUI

The software provides good and graphical interface for the user and administrator can operate on the system, performing required task such as Home, Menu, Orders, Account.

- A visually appealing homepage with a prominent search bar. Display featured restaurants, deals, and promotions.
- Organized menu items with prices and descriptions. High-quality images for each menu item.
- Visible shopping cart with itemized list and total cost. Options to edit or remove items in the cart.
- User profile pages with personal information and order history. Ability to edit profile details and saved addresses.
- Secure payment gateway integration. Multiple payment options (e.g., credit card, PayPal, cash on delivery). Order confirmation with details and estimated delivery time.

Search and Filters:-

- Robust search functionality with auto-suggestions.
- Filters for dietary preferences (e.g., vegetarian, vegan, gluten-free).

Notifications:-

- Push notifications for order updates, promotions, and discounts.
- Email notifications for order confirmation and receipts.

Help and Support:-

- Access to FAQs, customer support, and contact information.
- Live chat or chat bot for immediate assistance.

Security:

- Implement security measures to protect user data and transactions.

4. System Features

The users of the system should be provided the surely that their account is secure. This is possible by providing :-

- User Registration and Profile Management is a user account creation with email, phone number, or social media login. Profile editing with personal information, delivery addresses, and payment methods.
- Search for restaurants based on location, cuisine, or restaurant name. Filtering options (e.g., ratings, delivery time, etc.,..). View restaurant listings with images, descriptions, and ratings.
- Access to restaurant menus with item details, prices, and images. Special dietary icons (e.g., vegetarian, vegan, gluten-free). Add items to the cart with quantity selection.

5. Other Non-functional Requirements

5.1 Performance Requirements

Food delivery are critical to ensure a smooth and responsive user experience. The website and app should load quickly, with pages appearing within a few seconds. Load times for images, menus, and other content should be optimized. Some key performance-related considerations and requirements.

- The user interface should respond promptly to user interactions (e.g., clicks, taps, scrolling). Ensure that buttons and links are highly responsive without noticeable delays.
- The system should be able to handle an increasing number of concurrent users during peak times. Back end servers should scale dynamically to manage increased traffic.
- Security measures, such as authentication and authorization checks, do not significantly impact performance. Conduct security audits to identify and address performance-related security issues.

5.2 Safety Requirement

Safety requirements in a food delivery are crucial to protect user data, ensure secure transactions, and maintain the trust of both customers and restaurant partners.

5.3 Security Requirement

- Implement strong user authentication mechanisms, including multi-factor authentication (MFA) options.
- Comply with Payment Card Industry Data Security Standard (PCI DSS) requirements when handling credit card information.
- Utilize secure, third-party payment gateways to process transactions.
- Secure API 's used for communication between the website, mobile app, and server, implementing proper authentication and authorization.
- Regularly update all components of the system (web server, database, libraries, and frameworks) to patch security vulnerabilities.
- Manage user sessions securely, ensuring proper timeout periods and secure storage of session tokens.

5.4 Requirement Attributes

- A unique identifier for each requirement, which aids in tracking and referencing them.
- A concise and descriptive title or name for the requirement, making it easily identifiable.
- Indicate the importance or priority of the requirement, such as high, medium, or low, to help with prioritization during development.
- Specify the person or team responsible for overseeing and managing the requirement's implementation.

5.5 Business Rules

Business rules play a critical role in the operation of a food delivery website and application. They govern various aspects of how the platform functions and

interacts with users, restaurants, and delivery partners. Users must create accounts and authenticate themselves to place orders. Password complexity rules and account lockout policies may apply.

5.6 User Requirements

User requirements for a food delivery are essential to creating a platform that meets the needs and expectations of your target audience. Understanding these requirements is crucial for designing a user-centric experience. Users should be able to create accounts easily using email, phone numbers, or social media profiles. Provide options for secure password recovery and account management.

Some common user requirements for such a platform:

- User Registration and Authentication
- User profile management i.e. Edit their profiles, including names, contact details, and delivery addresses.
- Browsing and Searching i.e. Search for restaurants and cuisine types based on location.
- Restaurant Information i.e. Access to detailed restaurant information, including menus, prices, ratings, reviews, and operating hours.
- Order Placement i.e. Where the ability to add and remove items from the cart.
- Checkout and Payment i.e. Checkout process with multiple payment options (credit/debit cards, digital wallets, cash on delivery).
- Order Tracking i.e. Track the status of their orders in real-time.

6. Other Requirements

6.1 Data and Category Requirement

Food delivery services require a wide range of data and categories to effectively operate and provide a seamless experience to their customers. First and foremost, they need comprehensive information about the available restaurants and eateries in the area, including their menus, pricing, and location data. This data should be regularly updated to reflect any changes in offerings or operating hours.

6.2 Appendix

A:availability,admin,B:budget,C:cart,customer,compliance,D:documentation,M :menu item,N:non functional requirement, O:order,order tracking, P:payment gateway, R: review rating, T:time frame, U:user,usability

6.3 Glossary

User: Any person who interacts with the food delivery website or mobile application.

Customer: A registered user who places orders for food delivery.

Menu Item: A specific food item available for ordering.

Cart: A virtual container where customers add items before placing an order.

Order: A request for food delivery, including selected items and delivery details.

Order Tracking: The feature allowing customers to monitor the status of their food delivery.

Reviews and Ratings: User-generated feedback and ratings for food and service.

Admin Panel: A secure portal for restaurant staff to oversee orders and menu management.

Performance: The system's responsiveness and speed in handling user requests.

Security: Measures implemented to protect user data and payment information.

Availability: The system's ability to operate 24/7 without significant downtime.

Usability: The degree to which the system is user-friendly and easy to navigate.

Scalability: The system's capability to handle an increasing number of users.

Payment Gateway: A third-party service used for secure online payment processing.

Compliance: Adherence to legal and regulatory requirements related to food delivery and data privacy.

Documentation: Manuals and guides explaining system usage and administration.

Budget: The allocated financial resources for the development and maintenance of the system.

Time frame: The specified period within which the project must be completed.

6.4 Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the

data and the operations that can be performed on instances (i.e. objects) of the data.

A class of

data has a name, a set of attributes that describes its characteristics, and a set of operations that

can be performed on the objects of that class. The classes' structure and their relationships to

each other frozen in time represent the static model. In this project there are certain main classes

which are related to other classes required for their working. There are different kinds of

relationships between the classes as shown in the diagram like normal association, aggregation,

and generalization. The relationships are depicted using a role name and multiplicities.

