PES UNIVERSITY

Department of Computer Science and Engineering

UE21CS341A: Software Engineering

ASSIGNMENT / PROJECT GUIDELINES

Acknowledgment: Authors of the document

Purpose: As part of the software engineering course, students are expected to do a project incorporating all stages of software development life cycle. As the units will be taught in the classes, phase wise inclusion needs to be done.

Outcomes: Team Details, Project Title, Synopsis, all intermediate SE(SDLC) Documents and a complete Project Report (Formats attached for reference)**.**

Evaluation and Complete Process: The details of how this project must be done and evaluated are as described below:

Guidelines:

1: Project team must identify an application case study ex: e-ticket/Banking/Recruitment System/Trading System.

2: Team must prepare a synopsis to enlist all the features of the chosen project / software application.

3: Prepare a document consisting of requirements, planning, design, implementation, and testing details.

4: Project Report consisting of all the relevant documents must be submitted with all results. **Project**

Team:

The project will be done by a group of 4 students. One or two teams in a class may be an exception with prior approval of the subject teacher. Teams must be among students belonging to the same section. Each team member needs to complete one full functionality. Weekly evaluations will be done based on individual performance and contribution made towards the project. A mid-point review of the project and corresponding document submission is mandatory. This will carry a portion of the total marks awarded for the project.

PES UNIVERSITY
Department of Computer Science and Engineering
UE21CS341A: Software Engineering

Week-Wise Deliverables and Task Details

Week No.	Details of the Task	Deliverables With Specified Format
Week 1	Team formation, problem selection, Project approval after deliberation with the Professor.	• Synopsis submission with functionality of the chosen application (Ref Format 1)
Week 2	Identifying software life cycle model, SRS preparation, Small Prototype	• SRS document preparation • Submit the Requirement Document in the specified format (Ref Format 2)
Week 3	WBS/Scheduling using project planning tool	• Prepare a scheduling chart using Gantt pro/Microsoft project. • Create a project plan as per the template shared. (Ref Format 3)
Week 4	Propose Architectural style /design for the project selected. Design pattern for their problem statement (Architectural)	• Design Diagrams UML diagrams Architectural designs
Week 5	Coding/practices adopted using any PL	
Week 6	Exploring CM in the project for collaborative coding and configuration management.	• Create GitHub accounts and start exploring .
Week 7	Coding using standard practices.	
Week 8	Prepare test strategy and test plan Prepare a test suite for your project Generating test cases designing using tools	Test case document submission (Ref Format 4)
Week 9	Testing of complete project/documentation	Complete Report submission (Ref Format 5)
Week 10	Presentation/Evaluation with the report	Complete Demo of the project.

PES UNIVERSITY
Department of Computer Science and Engineering
UE21CS341A: Software Engineering

Format 1

Synopsis/Project Proposal

Each proposal should contain the following elements:

• Project title, Group number
• Team profile: Individual team members Name, SRN, eMail addresses, Phone nos. Note: It is expected that every team member shall be involved in all project activities. **Proposed Project Description:**

Your project description must include a thorough explanation of your planned project, whether you are proposing a brand-new project or expanding on one of the sample software projects. You can also say whether you already have a customer or someone who is interested in your suggested project, as well as characterise the usual users of your planned system. Describe in a list of bullet points what the user will be able to perform with your system ("functional features") at the conclusion of this section.

Plan Of Work And Product Ownership:

 Your plan of work should list and describe the items that you are planning to accomplish in the short term (next few weeks).
 Split your team into "**who should do what** "
 Functionality: which functional features (from the itemized list at the end of the previous section) each person will contribute. *Example functional features are customer registration, data capture and storage, data processing to extract statistical parameters, etc.*
 User interface, graphic design, database interaction, unit testing, etc. are not functional features.
 Qualitative property, if any, that you will contribute, such as fine tune the system performance to achieve response time under x seconds, or develop and evaluate an easy-to-use user interface for this-and-this specific functional feature, or ensure confidentiality of a specific set of data, etc.

Product ownership is critical to demonstrate that each team member will play a clearly defined role in the proposed project.

Evaluation:

 Proposals that are missing any of the above sections will be returned without review. Each team should submit by given due date a single document for their team project. Only PDF document format will be accepted.

PES UNIVERSITY
Department of Computer Science and Engineering
UE21CS341A: Software Engineering

Format 2

Software Requirements Specification

for

AgileBite Express

(Food Delivery Management System)

Version 1.0 approved

Prepared by:
Anshul Baliga - PES1UG21CS095
Anoosh Damodar - PES1UG21CS092

16-10-23

PES UNIVERSITY
Department of Computer Science and Engineering
UE21CS341A: Software Engineering

Table of Contents

Revision History

Name	Date	Reason for changes	Version

PES UNIVERSITY

BENGALURU

Department of Computer Science and Engineering

Introduction

Purpose

The purpose of this SRS is to outline both the functional and non-functional requirements of the subject Food Delivery Management System.

In addition to said requirements, the document also provides a detailed profile of the external interfaces, performance considerations and design constraints imposed on the subsequent implementation. The document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future.

Our project aims to create a comprehensive, intuitive and user-friendly platform that allows customers to order food from a variety of restaurants and have it delivered to their doorstep. The customer opens the app, browses through the various food items, combos and cuisines available there and goes ahead and selects and purchases the item he or she needs.

It also incorporates Table Reservation functionality and enhances user convenience with In-App Ordering at Restaurants. Payments for such online orders can be made through debit cards, credit cards, cash or card on delivery, or even through digital wallets.

Intended Audience

The primary audience of this SRS document will be the stakeholders, to confirm on all the requirements and features before starting development. Stakeholders include Sponsors, Management team, Product Champion, Q/A staff.

The secondary document audience comprises the development team employed to build and implement the system. It will provide an extensive capacity for project planning and progress assessment.

This document provides a comprehensive overview of the project, including detailed explanations of its features and specifications. It serves as a valuable resource for all stakeholders, enabling users to gain a thorough understanding of the project's objectives.

Additionally, it aids the Quality Assurance (Q/A) team in preparing testing cases aligned with the project's requirements. The document also outlines any limitations, constraints, and operational requirements necessary for the system to function effectively.

Product Scope

The Food Delivery Management System is a sophisticated digital platform designed to revolutionize the way customers interact with restaurants and food establishments. Its primary objective is to provide a seamless and convenient experience for users seeking to order food for delivery or dine-in at a restaurant.
The project will be divided into three phases:

- Phase 1-Initial Phase: Basic Food Delivery Management App/Website
- Phase 2: Online order placement and payment
- Phase 3: Recommendation and Data Analysis

In the initial phase, the system will also introduce a Table Reservation and In-App Ordering feature, allowing customers to seamlessly pre-book tables and place orders directly from their tables, reducing waiting times and enhancing the overall dining experience.

The system acts as a bridge between customers and restaurants, offering a user-friendly interface that allows users to effortlessly browse menus, customize their orders, and securely process payments. By leveraging the power of technology, this platform aims to enhance customer satisfaction and streamline operations for food businesses.

References

1. Data Privacy Regulations:
    - Title: GDPR Compliance Guidelines
    - Author: Legal Team
    - Version Number: 1.0
    - Date: 2023-09-14
    - Source/Location: https://gdpr.eu/

2. User Interface Style Guide:
    - Title: UI Style Guide for Food Delivery Management System
    - Author: Dev Team
    - Version Number: 1.0
    - Date: 2023-10-10
    - Source/Location: https://github.com/anshulbaliga7/AgileBiteExpress

3. System Requirements Specification:
    - Title: System Requirements Specification for Food Delivery Management System
    - Author: IEEE
    - Version Number: 1.0
    - Date: 2023-09-14
    - Source: https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc

4. Contracts:
    - Title: AgileBite Express Project Contract
    - Author: Legal Team
    - Version Number: N/A
    - Date: 2023-09-01

Overall Description

Product Perspective

The Food Delivery Management System outlined in this SRS is a new, standalone product. This innovative solution stems from a deep understanding of customer and restaurant owner needs, aiming to revolutionize the food delivery industry by providing a more streamlined and user-friendly experience. The development of this system is driven by a vision to enhance customer satisfaction and optimize operations for food businesses. Its origin lies in the desire to bridge the gap between consumers and restaurants, enhancing satisfaction and operational efficiency.

The system also provides a unique feature set that includes the Table Reservation and In-App Ordering functionality, distinguishing it as a pioneer in providing a comprehensive solution that caters to the diverse needs of both customers and restaurant owners, a feature yet to be fully realized by other industry players.

Product Functions

Customer Features:

- User Registration and Authentication
- Browse and Search Restaurants and Menus
- Place Orders with Customization Options
- Manage Cart (Add/Remove Items)
- Select Delivery/Pickup Options
- Check for any Offers/Discounts
- Process Payments (Debit/Credit Cards, Cash, Digital Wallets)
- Confirm Orders and Provide Tracking
- Customer Care Support

Super Administrator Features:

- Add/Update/Delete Multiple Restaurant admin accounts
- Manage User, Restaurant, Delivery Driver
- Track monthly report
- Enable User Reviews and Ratings

Delivery Rider Features:

- Details of each order along with Customer Details
- Real-Time Navigation Assistance

Restaurant Administrator Features:

- Table Reservation Functionality and In-App Ordering Convenience
- Updates the menu regularly
- Track monthly report
- Provide Restaurant Owners with Management Dashboard

User Classes and Characteristics

1. Customer

Characteristics:

- Frequency of Use: Frequent usage for ordering food
- Technical Expertise: Basic knowledge of using mobile apps
- Responsibilities: Browsing menus, Placing orders, Customizing orders, Providing payment details, Tracking orders, Leaving reviews and ratings

2. Restaurant Owner/Manager

Characteristics:

- Frequency of Use: Regular usage for managing restaurant operations
- Technical Expertise: Moderate to advanced knowledge of the app's management interface
- Educational Level/Experience: Business owner or management background
- Responsibilities: Managing menu items and prices, Provides In-App Ordering for Dine-in Customers, Provides Table Reservation Features, Responding to customer reviews and ratings

3. Administrator (Admin)

Characteristics:

- Frequency of Use: Occasional usage for system maintenance and user management
- Technical Expertise: Advanced knowledge for system administration
- Educational Level/Experience: Technical background in system administration
- Responsibilities: User account management, System maintenance and updates, Track monthly revenue report

4. Delivery Personnel

Characteristics:

- Frequency of Use: Frequent usage for delivering orders
- Technical Expertise: Basic knowledge of using a mobile app for order management and navigation
- Responsibilities: Receiving and delivering orders, Updating order status, Adding Reviews, Adhering to Safety Protocols, Maintaining Vehicle and Equipment

5. Customer Support

Characteristics:

- Frequency of Use: Regular usage for addressing customer queries and issues
- Technical Expertise: Moderate knowledge for using customer support tools
- Educational Level/Experience: Customer service background
- Responsibilities: Addressing customer inquiries, Resolving order-related issues, Offering Product Guidance, Managing Feedback and Complaints

6. Guest User

Characteristics:

- Frequency of Use: Infrequent usage, without creating an account
- Technical Expertise: Basic knowledge of using the app
- Responsibilities:Browsing menus, Placing orders, Exploring Special Offers, Encouraging Account Creation

Operating Environment

Environment Description:

Hardware Platform:

- Mobile Devices:
  iOS Devices (e.g., iPhones, iPads)
  Android Devices (e.g., smartphones, tablets)

- Administrative Systems:
  PCs in Restaurants
  In-App Ordering Tablets for Customers

- Operating System and Versions:
  iOS: Compatible with iOS 13 and above
  Android: Compatible with Android 8 (Oreo) and above
  Waiter's Devices: Compatible with both iOS and Android platforms
  Customer Devices: Compatible with both iOS and Android platform

PES UNIVERSITY

BENGALURU

Department of Computer Science and Engineering

Design and Implementation Constraints

Constraints:

- **Corporate and Regulatory Policies:**

  The development process must adhere to company policies and industry-specific regulations, particularly regarding data privacy, security, and handling of payment information

- **Hardware Limitations:**

  Developers must account for varying hardware capabilities of different mobile devices, ensuring the app performs well across a range of platforms, including older devices

- **Memory Requirements**:

  The app should be optimized to operate within memory limits, especially on devices with lower RAM

- **Interfaces to Other Applications:**

  Integration with third-party services (e.g. payment gateways, mapping services) must be carefully managed to ensure seamless interaction without causing conflicts or disruptions.

- **Parallel Operations:**

  Developers must implement concurrent processing to handle multiple users interacting with the system simultaneously, particularly during peak usage times

- **Security Considerations:**

  The app must employ robust security measures to protect user data, including encryption of sensitive information, secure authentication, and protection against common security threats (e.g., SQL injection, XSS attacks)

- **Cross-Browser Compatibility:**

  Developers should follow established coding standards for efficient code maintainability, readability, and scalability. Thus, Users may access the system using different browsers such as Google Chrome, Mozilla Firefox, Safari, Microsoft Edge, and others.

2.6 Assumptions and Dependencies

Assumptions:

- **Platform Stability:**

  iOS and Android platforms will maintain stability and compatibility with the app over time.

- **Network Availability:**

  Assumed that users will have access to a stable internet connection for app functionality, including browsing menus and placing orders.

- **User Device Capabilities:**

  Users will have devices with the necessary hardware and software capabilities in terms of memory, to run the app.

- **Language and Localization:**

  Assumed that translation and localization services will be available for content in different regional languages.

- **Regulatory Compliance:**

  Assumed that the app will comply with local and international regulations regarding data privacy, security, and payment processing.

Dependencies:

- **Payment Gateway Integration:**

  Integration with third-party payment gateways for secure transaction processing is crucial for handling payments.

- **Mapping Services:**

  Dependence on mapping services (e.g., Google Maps API) for location-based features and restaurant recommendations.

- **Database System:**

Relying on a backend database system (e.g., MongoDB, MySQL) for storing user accounts, menu details, orders, and payment records.

- **User Authentication and Authorization Services:**

  Relying on a secure authentication and authorization system for user access control and account management.

- **Device Compatibility:**

  Ensuring compatibility with a wide range of iOS and Android devices, factoring in various screen sizes, resolutions, and hardware capabilities.

- **App Store Policies:**

  Adherence to policies and guidelines set by app stores (Apple App Store, Google Play Store) for app submission and updates.

- **Security Certificates:**

  Acquisition and proper implementation of SSL certificates for secure communication between the app and external servers.

External Interface Requirements

User Interfaces

Customer-Facing Mobile App:

- **Logical Characteristics:** The customer-facing mobile app provides an intuitive and user-friendly interface for customers to browse menus, place orders, track deliveries, and manage their accounts. It employs standard mobile app design conventions for both iOS and Android platforms
- **GUI Standards:** The app follows platform-specific design guidelines for iOS devices and Android devices. This includes standardized navigation patterns, consistent iconography, and a visually appealing screen layout with consistent spacing, font sizes.
- **Standard Button and Functions**: Navigation Bar, Search Icon, Cart Icon, Notification and Help/Support Icon, Order Status Indicator and Promotional Banner
- **Error Message Display Standards:** Error messages appear prominently, providing guidance on how to resolve the issue, In-app notifications alert users to errors or issues related to order processing, payments, or account management.
- **Help/Support Function:** In-app chat support and customer service contact details

Restaurant Management Dashboard:

- **Logical Characteristics:** The restaurant management dashboard is designed for restaurant owners and staff to manage orders, update menus, and monitor performance. It offers a clear overview of incoming orders, delivery statuses, and customer reviews
- **GUI Standards:** The dashboard adheres to web application design principles, ensuring responsiveness and usability across different screen sizes with the use of color schemes, fonts for an easy-on-the-eyes and professional appearance
- **Standard Buttons and Functions:** Navigation Menu, Order Summary Panel, Reports Icon, Settings Icon, Help Icon
- **Keyboard Shortcuts:**
    - Ctrl + O: Navigate to the Orders section
    - Ctrl + M: Access Menu Management

Admin Console:

- **Logical Characteristics:** The admin console is accessed by system administrators for overall system management. It provides tools for user management, system configuration, and monitoring of system health
- **GUI Standards:** The admin console follows a web-based interface with a focus on functionality and system controls. It is designed for efficient administration tasks, incorporating standard form layouts, data tables, and interactive charts where applicable
- **Standard Buttons and Functions:** Navigation Bar, System Configuration Icon, Reports Dashboard, Help Icon, User Management Dashboard
- **System Health Panel:** System health metrics, such as server status, database connectivity, and system resource usage
- **Keyboard Shortcuts:**
    - Ctrl + U: Navigate to User Management
    - Ctrl + S: Access System Configuration

In-App Ordering Tablet Interface:

- **Logical Characteristics:** The tablet interface is used by customers within restaurants for in-app ordering at tables. It allows users to browse menus, customize orders, and place them directly from their tables
- **GUI Standards:** The tablet interface aligns with the design principles of the customer-facing mobile app, with a focus on touch-based interactions, but is tailored for a larger screen. It features a simplified layout for ease of use within a restaurant setting
- **Standard Buttons and Functions:** Home Button, Menu Navigation Panel, Order Summary Panel, Help Icon
- **Error Message Display Standards:** Real-time notifications for issues related to order processing, payment, or system connectivity

Software Interfaces

Software Components:

- **React Native Framework:** This is the core framework used for developing the mobile application. It enables cross-platform compatibility between iOS and Android.
- **Node.js :** Node.js provides the runtime environment for running JavaScript on the server-side, enabling server-side logic and handling asynchronous operations.
- **Express.js :** Express.js is a web application framework for Node.js. It is used for routing, handling HTTP requests, and middleware integration.
- **MongoDB :** MongoDB is a NoSQL database used for storing user accounts, menu details, orders, and payment records.
- **Third-Party Payment Gateway :** The app integrates with a third-party payment gateway for secure processing of payments.
- **Google Maps API :** The app integrates with the Google Maps API to provide location-based services, enabling features like restaurant recommendations and order tracking.

Data Flow and Communication:

1. **Incoming Data/Events:**
    - User Input
    - API Requests
    - Notifications

2. **Outgoing Data/Events:**
    - Order Details
    - Payment Information
    - User Notifications

Communications Interfaces

- **HTTP/HTTPS Protocols:**

  The customer-facing mobile app, admin console, and in-app ordering tablets communicate with the server using standard HTTP/HTTPS protocols for data exchange, user authentication and payment processing

- **RESTful API Endpoints:**

  Communication between different components, including the customer-facing mobile app, admin console, and any third-party integrations

- **Email Notifications(SMTP):**

  Order confirmations and password reset instructions

- **Data Transfer Rates**:

  High-speed data transfer between the customer-facing mobile app, admin console, and the server.

- **Security and Encryption:**

  TLS/SSL protocols are used to encrypt data during communication over the network for better confidentiality and integrity

- **Push Notification Services(FCM):**

  Send real-time updates and alerts to users' mobile devices

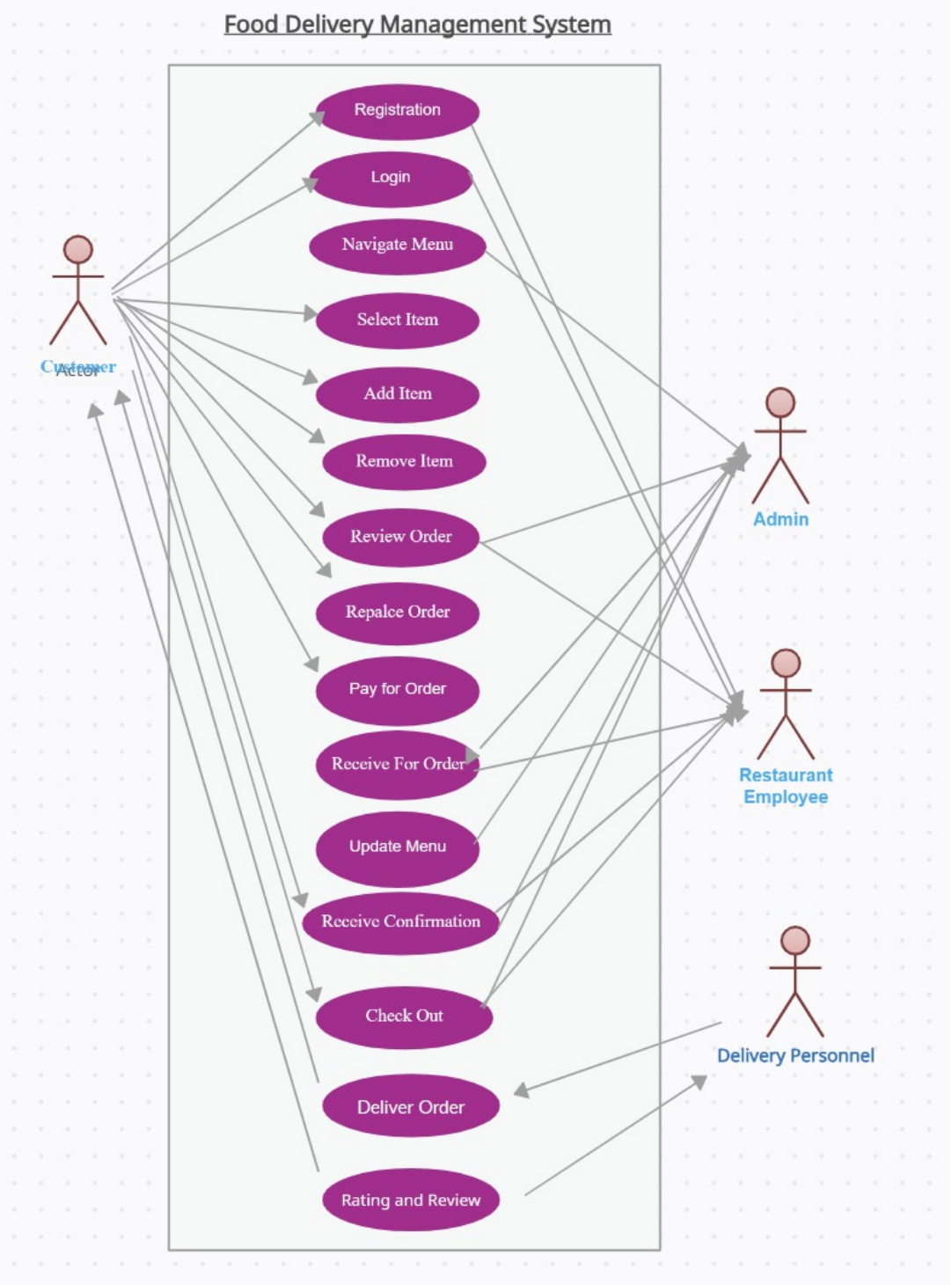
PES UNIVERSITY

BENGALURU

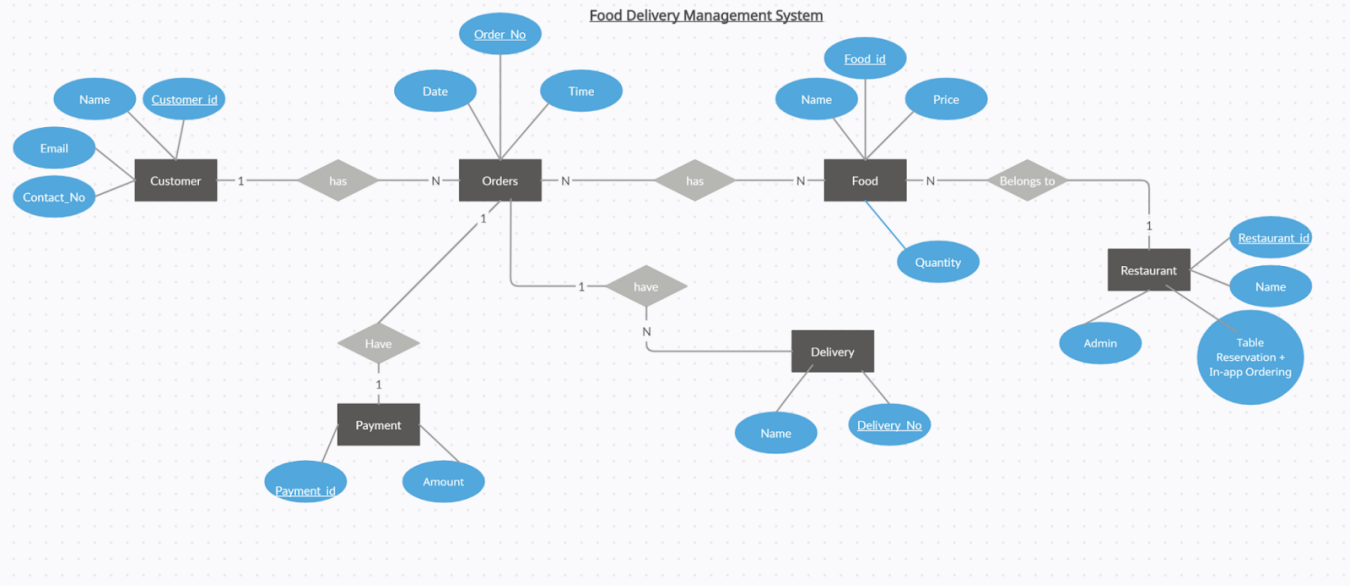
Department of Computer Science and Engineering

Analysis Models

Use-case diagram:



ER Diagram:



System Features

System Feature 1

User Registration and Authentication

Description and Priority:

This feature allows users to create accounts, log in, and ensure secure authentication. It is of high priority.

Stimulus/Response Sequences:

- User clicks on the "Sign Up" button
- User provides necessary details (e.g., name, email, password)
- System validates and stores user information
- Upon successful registration, the system navigates the user to the dashboard
- If a user exists, they click on "LogIn" and enter credentials

Functional Requirements:

- REQ-1: Validate user-provided information during registration

- REQ-2: Hash and securely store user passwords

- REQ-3: Provide appropriate error messages for invalid inputs

- REQ-4: Implement password recovery/reset functionality (TBD)

- REQ-5: Ensure secure session management to keep users authenticated

System Feature 2

Browsing and Searching Restaurants and Menus

Description and Priority:

This feature enables users to browse through a list of available restaurants, view menus, and search for specific items. It is of high priority.

Stimulus/Response Sequences:

- User opens the app and views the list of nearby restaurants.

- User clicks on a specific restaurant to view its menu.

- Users use the search functionality to find specific cuisines or dishes.

Functional Requirements:

- REQ-1: Display a list of restaurants based on the user's location.

- REQ-2: Provide filters for sorting and narrowing down restaurant options.

- REQ-3: Display detailed menus with item names, descriptions, and prices.

- REQ-4: Implement search functionality for restaurants and menu items.

- REQ-5: Allow users to add items to their cart for ordering.

System Feature 3

Placing and Managing Orders

Description and Priority:

This feature allows users to select food items, customize orders, and manage their shopping cart. It is of high priority.

Stimulus/Response Sequences:

- User selects food items and adds them to the cart

- User customized orders (e.g., quantity, special instructions)

- User reviews the cart and proceeds to checkout

Functional Requirements:

- REQ-1: Allow users to customize items (e.g., specify toppings, choose portion size)

- REQ-2: Calculate order totals, including taxes and fees

- REQ-3: Enable users to view and edit their cart contents

- REQ-4: Provide a clear and intuitive checkout process

- REQ-5: Display order summary before finalizing the order

System Feature 4

Processing Payments

Description and Priority:

This feature involves securely processing payments using various methods (e.g., credit/debit cards, cash on delivery). It is of high priority.

Stimulus/Response Sequences:

- User selects their preferred payment method during checkout

- User provides necessary payment details

Functional Requirements:

- REQ-1: Integrate third-party payment gateways for secure transaction processing

- REQ-2: Implement encryption for sensitive payment information

- REQ-3: Provide error handling for failed transactions

- REQ-4: Support multiple payment options (e.g., credit cards, digital wallets)

- REQ-5: Allow users to save payment methods for future orders

Other Non Functional Requirements

Performance Requirements

1. Page Load Time:
- Requirement: The app should load the main menu page within 3 seconds on a standard 4G network.

2. Order Processing Time:
- Requirement: The system should process and confirm an order within 30 seconds of the user finalizing the order.

3. Real-time Order Tracking:
- Requirement: The system must update the order status in real-time with no more than a 15-second delay. (Real-time Tracking)

4. Concurrency Handling:
- Requirement: The system should support at least 500 concurrent users placing orders during peak hours without a significant performance degradation.

5. Database Response Time:
- Requirement: Database queries for retrieving menu items or order details should have a response time of less than 300 milliseconds (Very quick response).

Safety Requirements

- **Payment Security:**

  The system must implement encryption protocols (e.g: SSL) to protect users' payment information during transmission.

- **Data Privacy and Protection:**

  The system must comply with data protection regulations (e.g: GDPR) to safeguard user information.

- **User Authentication and Authorization:**

  Only authenticated and authorized users should have access to sensitive account information and payment details.

- **Order Accuracy and Quality:**

  The system should accurately transmit orders to restaurants and ensure the correct items are delivered to customers.

Security Requirements

- **Data Encryption:**

  Requirement: All data transmission, including user authentication and payment information, must be encrypted using industry-standard protocols (e.g., SSL/TLS).

- **User Authentication:**

  Requirement: Users must authenticate their identity before accessing sensitive information or performing secure actions (e.g., placing orders, accessing payment details).

- **Password Security:**

  Requirement: Passwords must be stored securely using strong cryptographic hashing techniques, and users should be encouraged to create strong, unique passwords.

- **Access Control and Authorization:**

  Requirement: Implement role-based access control (RBAC) to ensure that users only have access to functionalities and information relevant to their role (e.g., customer, restaurant owner).

Software Quality Attributes

Usability: The app should achieve a System Usability Scale (SUS) score of at least 80, indicating good usability.

Reliability: The app should have an uptime of at least 99.9% in any given month. Reliability ensures that the app is consistently available, minimizing disruptions for users.

Availability: The app should be accessible 24/7, with planned downtime for maintenance communicated to users in advance.

Robustness: The app should handle unexpected inputs and edge cases without crashing or causing significant disruptions to the user experience.

Testability: The app should have a test coverage of at least 90% for critical functionalities, as measured by code coverage tools.

Business Rules

Availability and Menu Management:

Restaurant owners can set their availability status (open/closed) and manage their menu by adding, updating, or removing items.

Domain Requirement: Provide a user-friendly dashboard for restaurant owners to manage their restaurant's details.

Payment Processing:

Payment processing will be handled securely using a third-party payment gateway. The system should not store sensitive payment information.

Domain Requirement: Integrate a reliable and secure third-party payment gateway.

Order Cancellation and Refunds:

Customers can cancel an order within a specified time window before the restaurant starts preparing it. Refunds will be issued according to the cancellation policy.

Domain Requirement: Implement a clear cancellation policy and provide functionality for users to cancel orders.

Feedback and Ratings:

After order completion, customers can provide feedback and ratings for both the restaurant and the delivery experience.

Domain Requirement: Provide a feedback system with options for rating and leaving comments.

Other Requirements

Database Requirements:

Requirement: Use a relational database management system (e.g., MySQL, PostgreSQL) to store and manage application data, including user profiles, order history, and restaurant information.

Internationalization and Localization:

Requirement: The app should support multiple languages and currencies to cater to a diverse user base. Users should be able to select their preferred language and currency.

Legal and Regulatory Compliance:

Requirement: The system must comply with all relevant legal and regulatory requirements, including data protection laws, privacy policies, and any industry-specific regulations.

Appendix A: Glossary

1. SRS(Software Requirements Specification):

A document that outlines the functional and nonfunctional requirements of a software system.

2. API(Application Programming Interface):

A set of rules and protocols for building and interacting with software applications.

3. GUI(Graphical User Interface):

The visual interface that allows users to interact with software through graphical elements such as icons, buttons, and windows.

4. SMTP(Simple Mail Transfer Protocol):

A protocol for sending and receiving electronic mail.

5. FCM(Firebase Cloud Messaging):

A cross-platform messaging solution that allows messages to be sent to iOS, Android, and web applications.

6. SSL/TLS(Secure Sockets Layer/Transport Layer Security):

A protocol for establishing secure communication links between clients and servers.

The successor to SSL, providing secure communication over a computer network.

7. HTTPS(Hypertext Transfer Protocol Secure):

An extension of HTTP that adds a layer of security by using encryption through SSL/TLS protocols.

8. GDPR(General Data Protection Regulation):

A data protection law designed to safeguard individuals' privacy by regulating the collection, processing, and storage of their personal data.

9. RBAC(Role-Based Access Control):

A security model that restricts system access to authorized users based on their roles within an organization.

Appendix B: Field Layouts

An Excel sheet containing field layouts, properties/attributes and report requirements.

	Field / Element	Description	Properties / Attributes	Report Requirement
2	Customer Registration Form	Form for customers to create accounts.	- Input fields for name, address, email, and password- Registration button- Social media login options	N/A
3	Restaurant Registration Form	Form for restaurants to register on the platform.	- Input fields for restaurant details- Registration button- Verification process	N/A
4	Food Menu Display	Display of available food items with prices and details.	- Display of food items with images, prices, and descriptions- Search and filter options	High
5	Order Placement	Process for customers to place food orders.	- Select food items, quantity, and delivery details- Order confirmation button- Order history tracking	High
6	Delivery Tracking	Real-time tracking of order delivery status.	- Real-time order tracking- Notifications for order status updates	High
7	Payment Processing	Handling secure payment transactions.	- Secure payment gateway integration- Payment options (credit card, online wallets)	High
8	Delivery Personnel Management	Management of delivery personnel information.	- Database of delivery personnel information- Assignment of orders to personnel	N/A
9	User Profile Management	User profile customization and settings.	- User profile settings for customization- Password change and account deletion options	N/A
10	Discounts and Promotions	Implementation of discounts and promotional offers.	- Application of discounts and promotions- Coupon code input	N/A
11	Inventory Management	Tracking and management of available food inventory.	- Inventory tracking and updates- Low stock notifications	N/A
12	Customer Feedback	System for collecting and analyzing customer feedback.	- Customer feedback form- Ratings and reviews	High
13	Response Time	Application responsiveness for order placement and tracking.	- Response time for order placement and tracking- System response during peak hours	High
14	Scalability	Ability to handle a large number of simultaneous orders.	- Maximum concurrent orders- Scalability resources	High
15	Offline Ordering	Ordering functionality without an internet connection.	- Offline order placement and viewing of past orders- Synchronization on reconnection	N/A
16	Data Synchronization	Synchronization of order data across devices.	- Data synchronization frequency- Conflict resolution methods	N/A
17	User Authentication	Secure user authentication methods.	- Strong password encryption- Two-factor authentication options	N/A
18	Secure Payment Gateway	Integration with secure payment gateways.	- Integration with secure payment gateways (e.g., Stripe)- SSL for secure data transmission	High
19	Protection Against Fraud	Prevention measures against fraudulent activities.	- Fraud detection algorithms- Monitoring of unusual activity	High
20	Security Audits	Regular security assessments and audits.	- Regular security audits- Vulnerability assessment methods	High
21	Usability	User interface ease of use and accessibility.	- User-friendly interface design- Intuitive navigation	High
22	Reliability	Application availability and responsiveness.	- Uptime percentage- Order placement success rate	High
23	Maintainability	Codebase documentation standards.	- Code documentation standards- Modular and maintainable code	High
24	Performance Efficiency	Efficient application performance on various devices.	- Resource consumption optimization- Device compatibility	High
25				

PES UNIVERSITY

BENGALURU

Department of Computer Science and Engineering

Appendix C: Requirement Traceability Matrix

Sl. No	Requirement ID	Brief Description of Requirement	Architecture Reference	Design Reference	Code File Reference	Test Case ID	System Test Case ID
1.	R1	Customer Registration Form	ARCH-001				
2.	R2	Restaurant Registration Form	ARCH-002				
3.	R3	Food Menu Display	ARCH-003				
4.	R4	Order Placement	ARCH-004				
5.	R5	Delivery Tracking	ARCH-005				
6.	R6	Payment Processing and Customer Feedback	ARCH-006				