Project 23: Food Mate- Online Food Delivery System

About the Client

Food Mate is a start-up food delivery company in Melbourne. They need an online portal with a wide selection of restaurants and takeouts in local area. Whether customers need fancy pizza, tex-mex, sushi, noodles or a sub; for lunch, dinner or as a treat, Food Mate have this covered.

Customers save money when ordering on Food Mate compared to ordering from rivals food portals. Even when it's the same meal from the same restaurant.

Unlike other food portals, Food Mate don't charge a service fee for ordering. No service fee combined with exclusive offers from restaurant partners mean that you'll be saving money on your meal. Same meal, same restaurant, better deal.

Our long list of restaurant partners is constantly growing thanks to our policy of not charging them a high commission for orders made through Food Mate. Not only are you supporting your local business by using Food Mate, you're benefiting from the exclusive discount they can afford to offer by not paying a high commission to a rival platform.

Project Brief & Business Problem Specifications:

Food Mate has not an integrated online ordering and delivery platform. This problem leads to operational inefficiencies for the business and a poor experience for customers. Following are the issues Food Mate is facing without an online food delivery system:

- Customers cannot easily browse restaurant menus, place orders, or track delivery status.
- Restaurants manually manages orders via phone calls, increasing errors and delays.
- Delivery partners lack coordination tools to accept assignments and update statuses.
- Payment collection is fragmented, with limited online options.

- Admins lack reporting tools to optimise operations or assess performance.

System Modules Requirements

User Management

- User registration and login
- Role management (Customer, Restaurant, Delivery Partner, Admin)
- Profile management (address, phone, preferences)
- Password recovery

Restaurant Management

- Restaurant registration
- Menu creation and updates
- Restaurant profile (logo, hours, location)
- Manage received orders
- Status updates (accept/reject/prepare/ready)

Menu & Product Catalogue

- Category management (Starters, Mains, Desserts, Drinks)
- Add/edit/delete food items
- Price management
- Item availability toggling
- Image uploads

Customer Ordering

- Browse restaurants
- View menus
- Add to cart

- Checkout process
- Order confirmation
- Estimated delivery time

Cart & Checkout

- Cart management (add/remove/update items)
- Delivery address selection
- Special instructions
- Payment method selection (Cash on Delivery, Card, Online Payment API)

Payment Integration

- Payment gateway API
- Transaction tracking
- Refund handling
- Payment status updates

Order Management

- Order placement
- Order tracking (preparing, out for delivery, delivered)
- Notifications (email/SMS)
- Cancel/modify order rules

Delivery Partner Module

- Delivery partner registration/login
- Assigned orders list
- Real-time order status updates
- Mark as picked up/delivered

Admin Dashboard

- User/restaurant management
- Delivery partner management
- Review all orders in the system
- Manage payments and commissions
- View analytics and reports

Ratings & Reviews

- Customers rate restaurants
- Customers rate delivery partners
- Text reviews
- Admin moderation of reviews

Search and Filter

- Search restaurants by name, location, cuisine
- Filter by price, ratings, offers
- Sorting options (fastest delivery, top-rated)

Promotions and Discounts

- Coupon code creation
- Restaurant-specific deals
- Automatic discounts (first order, loyalty)

Notifications System

- Email/SMS notifications
- Order status updates
- Promotions and campaigns

Reports and Analytics

- Daily/weekly sales
- Top-selling items
- Delivery time statistics
- Payment method breakdown

User Modules (User Frontend):

Developers need to research and discuss with the client to finalise the modules and requirements.

UI Design

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

UI Design Requirements

1. The system user should always be aware of what to do next.
2. The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.
3. Message, instructions or information should be displayed long enough to allow the system user to read them.

4. Use display attributes sparingly.
5. Default values for fields and answers to be entered by the user should be specified.
6. A user should not be allowed to proceed without correcting an error.
7. The system user should never get an operating system message or fatal error.

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work.

- Security of data.
- Ensure data accuracy's
- Proper control of the higher officials.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.
- Minimum time required.

Functional Requirements:

1. User Registration and Authentication

- Allow customers, restaurants, delivery partners, and admins to register with valid email and password.
- Provide secure login and logout functionality.
- Enable password reset via email verification.
- Support role-based access control to restrict features based on user type.

2. Customer Features

2.1 Profile Management

- View and edit personal details (name, address, phone).
- Save multiple delivery addresses.

2.2 Restaurant Browsing

- Search restaurants by name, cuisine, location.
- Filter and sort restaurants by rating, price, delivery time.

2.3 Menu Viewing

- View restaurant menus with categories and item details.
- See item images, prices, and availability.

2.4 Cart Management

- Add items to cart.
- Update item quantities or remove items.
- View cart total and delivery charges.

2.5 Checkout and Payment

- Select delivery address.
- Apply discount codes or coupons.
- Choose payment method (Cash on Delivery, Online Payment).
- Complete payment securely.

2.6 Order Tracking

- View current order status (Preparing, Out for Delivery, Delivered).
- Access order history with itemised details.

2.7 Ratings and Reviews

- Submit ratings and text reviews for restaurants.
- Edit or delete reviews.

3. Restaurant Features

3.1 Profile Management

- Manage restaurant details (name, logo, address, hours).
- Set delivery zones and charges.

3.2 Menu Management

- Add, edit, or remove categories.
- Add, edit, or remove menu items with images and prices.
- Mark items as available/unavailable.

3.3 Order Management

- Receive new orders with customer details.
- Accept or reject orders.
- Update order status (Preparing, Ready for Pickup).

3.4 Promotions

- Create and manage discount codes.
- View usage reports for promotions.

4. Delivery Partner Features

4.1 Profile Management

- Register and update personal details.
- View assigned deliveries.

4.2 Order Management

- Accept or reject delivery assignments.
- Update status (Picked Up, Out for Delivery, Delivered).

- View delivery history.

5. Admin Features

5.1 User Management

- View, approve, or suspend customer, restaurant, and delivery partner accounts.
- Reset passwords.

5.2 Restaurant Management

- Approve new restaurant registrations.
- View and edit restaurant details if needed.

5.3 Order Oversight

- Monitor all system orders.
- View order statuses and payment details.

5.4 Payment and Commission Management

- View transaction history.
- Manage restaurant commissions.

5.5 Reviews Moderation

- View and moderate customer reviews.
- Delete inappropriate content.

5.6 Reporting and Analytics

- View sales reports (daily, weekly, monthly).
- Track top-selling items.
- Review delivery performance statistics.
- Export reports to CSV/PDF.

6. Notifications

- Send email/SMS notifications to customers for order confirmation, status updates, and delivery completion.
- Notify restaurants of new orders.
- Notify delivery partners of new assignments.
- Support admin announcements to all users.

7. Payment Integration

- Support secure online payment gateway.
- Track payment status for each order.
- Issue refunds when orders are cancelled (where policy permits).

8. Security Requirements

- Use HTTPS for all data transmissions.
- Validate and sanitise all user inputs.
- Log user activities for auditing.
- Enforce role-based access to all features.

Non Functional Requirements

There are a lot of software requirements specifications included in the non-functional requirements of the system, which contains various processes, namely Security, Performance, Maintainability, and Reliability.

Security:

- Cybersecurity Implementation: Identify ethical risks in database design and implement the actions of mitigation.
- Cybersecurity Implementation: Provide evidence that you have implemented the data encryption and anonymization of data.
- Cybersecurity Implementation: Perform 'Data Protection Impact assessment' to help ensure compliance, facilitate a privacy by-design approach and identify better practice.
- Cybersecurity Implementation: Implement the secure methods for data encryption, data security and data breach to maintain the privacy of end users.
- Ensure average page load times under 2 seconds.
- Support at least 500 concurrent users without performance degradation.
- Enable horizontal scaling of PHP servers to handle increased demand.
- Design MySQL database to allow replication or partitioning for scalability.
- Maintain at least 99.5% uptime excluding planned maintenance.
- Implement reliable backup and restore procedures for disaster recovery.
- Store all user passwords securely using hashing algorithms.
- Encrypt sensitive data (e.g., payment details) in the database.
- Enforce HTTPS for all data transmission between client and server.
- Use prepared statements or ORM to prevent SQL injection.
- Validate and sanitise all user inputs to prevent XSS attacks.
- Apply secure session management with secure cookies and timeout handling.
- Enforce role-based access control for all system features.

- Follow PSR-12 PHP coding standards for maintainable and readable code.
- Use modular architecture to simplify updates and feature additions.
- Normalise database schema while ensuring necessary indexing for performance.
- Enforce referential integrity using foreign keys in MySQL.
- Use database transactions for critical operations (e.g., order placement, payment confirmation).
- Schedule daily backups of the MySQL database and verify their integrity.
- Provide an intuitive and responsive user interface for desktop and mobile.
- Ensure cross-browser compatibility (Chrome, Firefox, Safari, Edge).
- Log all significant system events (logins, orders, payments, errors) securely.
- Provide admin access to system logs and basic monitoring dashboards.
- Include clear privacy policy and terms of service pages.

Hardware Requirement: Should be recommended by the developers.

Software Requirement: Should be recommended by the developers.