**Food Delivery System Requirements Document**

**Project Overview**

This system is a comprehensive food delivery platform that connects customers with restaurants and delivery drivers. The platform supports restaurant management, real-time order tracking, delivery coordination, and comprehensive user account management for customers, restaurant owners, and drivers.

**Functional Requirements**

**1. User Management**

* **Multi-Role Registration**: Users can register as Customers, Restaurant Owners, or Drivers with role-specific verification.
* **Profile Management**: Users can manage personal information, preferences, and notification settings.
* **Role Switching**: Restaurant owners can also order food as customers.

**2. Restaurant Management**

* **Restaurant Registration**: Owners can register their restaurants with detailed information (cuisine type, hours, location).
* **Menu Management**: Dynamic menu creation with categories, ingredients, and availability tracking.
* **Operating Hours**: Flexible scheduling with special holiday hours support.
* **Real-time Status**: Toggle open/closed status with automatic closing based on hours.

**3. Food Catalog & Search**

* **Advanced Filtering**: Search by cuisine, dietary restrictions (vegetarian, vegan, gluten-free), price range, and ratings.
* **Geolocation-based Results**: Show restaurants based on user's delivery address.
* **Seasonal & Special Items**: Mark items as seasonal or limited-time offers.

**4. Cart & Order Management**

* **Multi-Restaurant Cart**: Support for ordering from multiple restaurants in single transaction.
* **Order Customization**: Special instructions for food preparation.
* **Real-time Price Updates**: Reflect changes in menu prices or availability.
* **Order Scheduling**: Schedule orders for future delivery.

**5. Payment System**

* **Multiple Payment Methods**: Credit cards, mobile payments, cash on delivery.
* **Split Payments**: Support for dividing payment among multiple users.
* **Tip Management**: Pre-set and custom tip options for drivers.
* **Refund Processing**: Automated refunds for canceled or problematic orders.

**6. Delivery Management**

* **Driver Allocation**: Automated driver assignment based on location and availability.
* **Real-time Tracking**: Live GPS tracking of delivery drivers.
* **Delivery Time Estimates**: Accurate ETAs based on distance and traffic.
* **Contactless Delivery**: Special instructions for delivery preferences.

**7. Review & Rating System**

* **Multi-dimensional Ratings**: Rate food quality, delivery time, and overall experience.
* **Photo Reviews**: Customers can upload photos of their food.
* **Owner Responses**: Restaurant owners can respond to reviews.
* **Review Moderation**: Automated and manual review filtering.

**8. Event & Promotion System**

* **Restaurant Events**: Special events, promotions, or tasting menus.
* **Loyalty Program**: Points system for frequent customers.
* **Targeted Promotions**: Location-based and preference-based offers.

**9. Inventory Management**

* **Ingredient Tracking**: Real-time inventory management for restaurants.
* **Stock Alerts**: Notifications for low stock items.
* **Seasonal Menu Planning**: Tools for seasonal menu changes.

**10. Analytics & Reporting**

* **Restaurant Analytics**: Sales reports, popular items, customer demographics.
* **Driver Performance**: Delivery times, customer ratings, earnings.
* **Admin Dashboard**: System-wide metrics and business intelligence.

**Non-Functional Requirements**

**1. Performance**

* **Response Time**: < 2 seconds for most operations
* **Concurrent Users**: Support 10,000+ simultaneous users
* **Order Processing**: Handle 1,000+ orders per minute during peak hours

**2. Scalability**

* **Horizontal Scaling**: Support for multiple server instances
* **Database Sharding**: Geographic-based data partitioning
* **CDN Integration**: For static assets and images

**3. Reliability**

* **99.9% Uptime**: With redundant systems
* **Data Consistency**: ACID compliance for financial transactions
* **Backup Systems**: Real-time database replication

**4. Security**

* **PCI DSS Compliance**: For payment processing
* **Data Encryption**: End-to-end encryption for sensitive data
* **API Security**: Rate limiting and DDoS protection

**5. Usability**

* **Mobile-First Design**: Responsive across all devices
* **Accessibility**: WCAG 2.1 compliance
* **Multi-language Support**: Internationalization ready

**6. Maintainability**

* **Microservices Architecture**: Independent service deployment
* **Comprehensive Logging**: Centralized log management
* **API Versioning**: Backward compatibility support

**Enhanced API Endpoints Structure**

**1. Authentication & Authorization**

/api/v1/auth

* POST /register - Multi-role registration
* POST /login - JWT token generation
* POST /refresh - Token refresh
* POST /logout - Secure logout
* POST /forgot-password - Password reset request
* POST /reset-password - Password reset confirmation
* POST /verify-email - Email verification
* POST /switch-role - Role switching for multi-role users

**2. User Management**

/api/v1/users

* GET /profile - Get user profile
* PUT /profile - Update profile
* GET /preferences - Get user preferences
* PUT /preferences - Update preferences
* GET /notifications - Notification settings
* PUT /notifications - Update notification settings
* POST /deactivate - Deactivate account
* GET /order-history - User order history
* GET /favorite-restaurants - User's favorite restaurants

**3. Restaurant Management**

/api/v1/restaurants

* GET / - Search restaurants with filters
* POST / - Register new restaurant (Owner)
* GET /{id} - Get restaurant details
* PUT /{id} - Update restaurant (Owner/Admin)
* PUT /{id}/status - Update open/closed status
* GET /{id}/menu - Get restaurant menu
* POST /{id}/menu/categories - Add menu category
* POST /{id}/menu/items - Add menu item
* PUT /{id}/menu/items/{itemId} - Update menu item
* GET /{id}/analytics - Restaurant analytics (Owner)
* GET /{id}/reviews - Restaurant reviews
* POST /{id}/events - Create restaurant event

**4. Food & Menu Management**

/api/v1/foods

* GET / - Search foods with advanced filters
* GET /{id} - Get food details
* GET /{id}/nutrition - Nutritional information
* GET /{id}/reviews - Food-specific reviews
* POST /{id}/favorite - Add to favorites
* DELETE /{id}/favorite - Remove from favorites

**5. Cart Management**

/api/v1/cart

* GET / - Get current cart
* POST /items - Add item to cart
* PUT /items/{itemId} - Update cart item
* DELETE /items/{itemId} - Remove cart item
* POST /apply-coupon - Apply discount coupon
* DELETE /coupon - Remove coupon
* POST /clear - Clear cart
* GET /delivery-estimate - Get delivery time estimate

**6. Order Management**

/api/v1/orders

* POST / - Create new order
* GET / - List user orders
* GET /{id} - Get order details
* PUT /{id}/cancel - Cancel order
* POST /{id}/reorder - Reorder previous order
* GET /{id}/tracking - Real-time order tracking
* POST /{id}/rate - Rate order experience
* PUT /{id}/delivery-instructions - Update delivery instructions

**7. Payment Management**

/api/v1/payments

* POST /methods - Add payment method
* GET /methods - List payment methods
* DELETE /methods/{id} - Remove payment method
* POST /{orderId}/process - Process payment
* GET /{orderId}/status - Payment status
* POST /{orderId}/refund - Request refund
* GET /transactions - Payment history

**8. Delivery Management**

/api/v1/delivery

* GET /drivers/available - Available drivers
* POST /{orderId}/assign-driver - Assign driver (Admin)
* GET /{orderId}/tracking - Live delivery tracking
* PUT /{orderId}/status - Update delivery status
* POST /{orderId}/contact-driver - Contact driver
* PUT /{orderId}/tip - Add/update tip

**9. Review & Rating System**

/api/v1/reviews

* POST / - Submit review (order required)
* GET /restaurant/{id} - Restaurant reviews
* GET /food/{id} - Food item reviews
* GET /driver/{id} - Driver reviews
* PUT /{id} - Update review
* DELETE /{id} - Delete review
* POST /{id}/report - Report inappropriate review

**10. Address Management**

/api/v1/addresses

* GET / - List user addresses
* POST / - Add new address
* PUT /{id} - Update address
* DELETE /{id} - Delete address
* POST /{id}/set-primary - Set primary address
* GET /delivery-zones - Check delivery availability

**11. Driver Management**

/api/v1/drivers

* POST /register - Driver registration
* GET /profile - Driver profile
* PUT /profile - Update profile
* PUT /availability - Set availability status
* GET /earnings - Driver earnings report
* GET /deliveries - Delivery history
* POST /location - Update driver location
* GET /performance - Performance metrics

**12. Event & Promotion System**

/api/v1/events

* GET / - List active events
* GET /restaurant/{id} - Restaurant events
* POST / - Create event (Owner/Admin)
* PUT /{id} - Update event
* DELETE /{id} - Delete event

**13. Admin Dashboard**

/api/v1/admin

* GET /dashboard - System overview
* GET /users - User management
* GET /restaurants/pending - Pending restaurant approvals
* PUT /restaurants/{id}/approve - Approve restaurant
* GET /orders - System orders
* GET /reports/sales - Sales reports
* GET /reports/performance - System performance
* POST /notifications - Send system notifications

**14. Search & Discovery**

/api/v1/search

* GET /restaurants - Restaurant search
* GET /foods - Food search
* GET /suggestions - Search suggestions
* GET /popular - Popular items
* GET /nearby - Nearby restaurants
* GET /trending - Trending foods

**15. Notification System**

/api/v1/notifications

* GET / - User notifications
* PUT /{id}/read - Mark as read
* POST /preferences - Notification preferences
* GET /unread-count - Unread notifications count

**Technology Stack Enhancement**

**Backend Services**

* **Framework**: Spring Boot 3.x with Java 17+
* **Security**: Spring Security 6.x with OAuth2
* **Database**: PostgreSQL (primary), Redis (caching)
* **Search**: Elasticsearch for advanced search
* **Message Queue**: RabbitMQ/Kafka for async processing

**External Integrations**

* **Payment**: Stripe, PayPal, Mobile Money APIs
* **Maps**: Google Maps/MapBox for geolocation
* **SMS**: Twilio for SMS notifications
* **Email**: SendGrid/Amazon SES
* **Storage**: AWS S3 for media files

**Monitoring & Analytics**

* **Monitoring**: Prometheus + Grafana
* **Logging**: ELK Stack (Elasticsearch, Logstash, Kibana)
* **APM**: New Relic/Datadog
* **Testing**: JUnit 5, TestContainers, Mockito

**Deployment & DevOps**

* **Containerization**: Docker + Kubernetes
* **CI/CD**: GitHub Actions/Jenkins
* **Cloud**: AWS/Azure/Google Cloud
* **CDN**: CloudFront/Cloudflare

**Security Considerations**

**Authentication & Authorization**

* JWT tokens with refresh mechanism
* Role-based access control (RBAC)
* API rate limiting and throttling
* Secure password policies

**Data Protection**

* GDPR compliance for user data
* Encryption at rest and in transit
* Regular security audits
* PCI DSS compliance for payments

**API Security**

* HTTPS enforcement
* CORS configuration
* Input validation and sanitization
* SQL injection prevention