**Software Requirement Specification (SRS) for Blinkit Clone**

**1. Introduction**

1.1 **Purpose**  
This document provides a detailed description of the functionality and requirements for developing a Blinkit-like grocery delivery platform, which allows users to order groceries and have them delivered to their doorstep within a short timeframe.

1.2 **Scope**  
The application will allow users to browse groceries, add them to their cart, select a delivery time slot, and make payments through various modes. The platform will consist of a mobile app for customers and a dashboard for admin operations.

1.3 **Definitions, Acronyms, and Abbreviations**

* **User**: End-user or customer who orders groceries.
* **Admin**: The administrator managing products, inventory, and orders.
* **Rider**: Delivery personnel assigned to deliver orders.

**2. Overall Description**

2.1 **Product Perspective**  
The product will function as an on-demand grocery delivery app, similar to Blinkit, operating via a mobile application and a web dashboard for administration.

2.2 **User Classes and Characteristics**

* **Customers**: Users who register and use the platform to order groceries.
* **Admin**: The platform manager responsible for adding products, managing inventory, and handling customer complaints.
* **Riders**: Delivery personnel who deliver groceries to the customers.

2.3 **Operating Environment**

* **Mobile Application**: Android and iOS.
* **Admin Panel**: Web-based interface compatible with modern browsers.
* **Rider App**: Separate mobile app for riders to view and track delivery assignments.

2.4 **Design and Implementation Constraints**

* Must be scalable to support high traffic.
* Should integrate with popular payment gateways (e.g., PayPal, Stripe, UPI).
* Real-time tracking of orders and riders using GPS.
* Should support multiple warehouses and delivery zones.

**3. Functional Requirements**

**3.1 User Registration and Login**

* Users should be able to sign up using their email, phone number, or social media accounts.
* Users should be able to log in using their registered credentials.
* Password reset functionality should be available.

**3.2 Product Catalog**

* Users can browse through categories of grocery items.
* Product search functionality must be available with filtering (price, brand, category).
* Each product should display a name, description, price, available stock, and an image.

**3.3 Shopping Cart**

* Users can add or remove items from the cart.
* The cart should display the total price dynamically based on the items added.
* Users should be able to view and edit the cart before checkout.

**3.4 Checkout and Payment**

* Users should be able to enter/select their delivery address.
* Payment options should include credit/debit cards, net banking, wallets, and UPI.
* Users should be able to apply discount coupons at checkout.

**3.5 Order Tracking**

* Users should receive notifications about order status (e.g., "Order Placed," "Out for Delivery").
* Real-time GPS tracking of the rider should be available once the order is out for delivery.

**3.6 Delivery Management**

* Riders should receive notifications of new orders assigned to them.
* Riders should have a map view to track the delivery route.
* Upon delivery completion, the rider should be able to mark the order as "Delivered."

**3.7 Order History**

* Users should be able to view their past orders with details such as items, amount, and delivery status.

**3.8 Admin Features**

* Admin should be able to add, update, or remove products from the catalog.
* Inventory management should be integrated to notify the admin when stock is low.
* Admin can view all orders and update their status.
* Admin can view and manage user complaints and feedback.

**4. Non-Functional Requirements**

**4.1 Performance Requirements**

* The system should be able to handle a minimum of 10,000 concurrent users.
* Real-time order and delivery updates should be processed within 2-3 seconds.

**4.2 Security Requirements**

* All user data must be stored securely and encrypted.
* Payment information should be processed through a secure payment gateway compliant with PCI-DSS standards.
* Two-factor authentication should be available for both users and admin.

**4.3 Usability Requirements**

* The mobile app should be intuitive and easy to navigate for all age groups.
* Users should be able to place an order in under 3 minutes from the moment they open the app.

**4.4 Availability**

* The application should have 99.9% uptime, with provisions for automatic backups and failover.

**4.5 Scalability**

* The system should be able to scale dynamically to support a large number of users during peak hours (e.g., festivals or sales).

**5. External Interface Requirements**

**5.1 User Interfaces**

* **Mobile App**: User-friendly UI for iOS and Android.
* **Web Admin Panel**: Easy-to-use dashboard for managing products and orders.
* **Rider App**: Simple UI to allow riders to accept and manage deliveries.

**5.2 Hardware Interfaces**

* Mobile devices for customer and rider applications.
* Web browsers for the admin panel.

**5.3 Software Interfaces**

* Integration with payment gateways (e.g., PayPal, Stripe, UPI).
* GPS for real-time tracking.
* SMS and email services for notifications.

**5.4 Communication Interfaces**

* The app should integrate with third-party APIs for services such as payments, location tracking, and notifications.

**6. System Features**

**6.1 Push Notifications**

* Users should receive notifications on order updates, promotions, and new arrivals.

**6.2 Referral System**

* Users can refer friends and earn credits for future purchases.

**6.3 Delivery Slot Selection**

* Users can choose their preferred delivery slot based on availability.

**7. Other Requirements**

**7.1 Data Privacy**

* User data must be compliant with local data protection laws (e.g., GDPR).

**7.2 Error Handling**

* In case of failed payments, the app should provide clear instructions for retrying or using an alternate method.

This SRS document outlines all necessary requirements to build a Blinkit clone. It can be expanded further based on business-specific customizations or additional features.