**Software Requirements Specification (SRS) for Kisan Connect**

**1. Introduction**

**1.1 Purpose**

Kisan Connect is a web-based platform aimed at replacing the orthodox APMC model for crop selling in India. The platform enables representatives to assess crops, certify them, and list them for auction. Bulk buyers can bid on crops, ensuring fair market pricing for farmers. The platform generates revenue through a commission-based model.

**1.2 Scope**

The platform will facilitate seamless transactions between farmers and buyers while ensuring quality verification through representatives. It will include:

* A manual crop assessment and certification process.
* An auction-based system for bulk purchases.
* An escrow-based payment system to ensure secure transactions.
* Role-based dashboards for different user types.
* Future enhancements like logistics tracking and automated notifications.

**1.3 Stakeholders**

* **Farmers (Kisans):** Crop sellers.
* **Buyers:** Bulk purchasers (retailers, wholesalers, businesses).
* **Representatives:** Conduct field inspections and upload crop details.
* **Administrators:** Manage auctions, handle conflict resolution, and oversee operations.
* **Logistics Partners (Future Enhancement):** Manage transportation and delivery of purchased crops.

**2. Functional Requirements**

**2.1 User Roles and Dashboards**

**2.1.1 Representative Dashboard**

* Upload crop details (quantity, quality, images, location, certifications).
* Edit or update crop information before listing.
* View a list of crops assessed and their statuses.

**2.1.2 Kisan (Farmer) Dashboard**

* View listed crops and their auction status.
* Receive notifications about auctions and payments.
* Track past transactions and earnings.

**2.1.3 Buyer Dashboard**

* View available auctions and place bids.
* Track bids and purchase history.
* Make payments through an integrated escrow system.
* Download receipts and invoices for purchases.

**2.1.4 Admin Dashboard**

* Approve or reject listed crops.
* Start, schedule, and close auctions.
* Manage user roles and permissions.
* Handle dispute resolution between buyers and farmers.
* Oversee the escrow payment system.

**2.2 Auction System**

* Dynamic bidding system where buyers place bids on listed crops.
* Auctions close after a predefined duration.
* Winning buyer makes payment to an escrow account.
* Notifications sent to all participants after auction closure.

**2.3 Payment Processing**

* **Escrow System:** Payment is held until successful delivery.
* Payment gateways like Razorpay, PayU, or Stripe for transactions.
* Refunds and dispute handling via admin intervention.

**2.4 Dispute Resolution (Future Enhancement)**

* Buyers can raise disputes for quality mismatch, shortfall, or delivery issues.
* Admin investigates and mediates disputes.
* Escrow holds payments until disputes are resolved.

**2.5 Logistics Partner Dashboard (Future Enhancement)**

* Assign transportation for purchased crops.
* Track deliveries in real-time.
* Update delivery status.

**2.6 Notification & Communication System (Future Enhancement)**

* Email, SMS, or in-app notifications for:
  + Auction start/end alerts.
  + Payment confirmations.
  + Dispute updates.
  + Delivery status updates.

**3. Non-Functional Requirements**

**3.1 Performance**

* The system should handle concurrent bidding without lag.
* Auctions should update in real-time.

**3.2 Security**

* Secure login with OTP-based authentication.
* Role-based access control (RBAC) for different users.
* End-to-end encrypted transactions.

**3.3 Scalability**

* Cloud-based deployment for scalability.
* Support for future integration with AI-based crop assessment.

**3.4 Usability**

* Mobile-friendly responsive web app.
* Simple and intuitive UI for farmers with minimal tech literacy.

**3.5 Compliance**

* Compliance with Indian e-commerce and agriculture laws.
* Data protection following GDPR and IT Act regulations.

**4. Technology Stack**

* **Frontend:** React.js, Bootstrap/Tailwind CSS
* **Backend:** Node.js with Express.js
* **Database:** MongoDB/MySQL (TBD based on scalability needs)
* **Payment Gateway:** Razorpay, PayU, or Stripe
* **Hosting:** AWS/GCP/Digital Ocean

**5. Future Enhancements**

* AI-based automated crop assessment.
* Multilingual support for regional farmers.
* Mobile app version for Android and iOS.

**6. Conclusion**

Kisan Connect aims to bring efficiency, transparency, and fairness to the agricultural market. The initial phase will focus on core auction and payment functionalities, with future upgrades planned for logistics and dispute resolution. This SRS will serve as a foundation for the platform’s development and scaling.