**Project Documentation: POS (Point of Sale) System for Electronics Selling Company**

**Objective:**

Develop a comprehensive POS system that streamlines the selling process in an electronic retail environment. This system should facilitate smooth sales transactions, inventory management, and generate various reports for business insights.

**Goal:**

To create an intuitive, efficient, and secure POS system capable of handling sales, customer information, product inventory, payments, and reporting for the company.

**Scope:**

* Sales Transaction Handling
* Inventory Management
* Customer Database
* Payment Gateway Integration
* Reports and Analytics
* User Management (Admin and Staff)

**2. Requirements**

**Functional Requirements:**

1. **Sales Transactions:**
   * Record and manage product sales.
   * Calculate totals, apply discounts, taxes, and shipping fees.
   * Process payments via multiple methods (cash, credit/debit cards, online payments).
2. **Product Inventory Management:**
   * Add, update, and delete products in the system.
   * Track stock levels, and notify the user when inventory is low.
3. **Customer Management:**
   * Create and update customer profiles (name, email, purchase history, etc.).
   * Apply customer discounts and loyalty points.
4. **Reports and Analytics:**
   * Generate daily, weekly, and monthly sales reports.
   * Track product performance and revenue generation.
   * Export reports to CSV or PDF format.
5. **Payment Gateway Integration:**
   * Secure online payment integration for processing credit/debit cards.
   * Provide payment status updates.
6. **User Management:**
   * Admin login with full access to settings, reports, and system configuration.
   * Staff login with restricted access based on roles.

**Non-Functional Requirements:**

* **Usability:** User-friendly interface for both customers and sales staff.
* **Security:** Secure handling of transactions and customer data, including encryption.
* **Scalability:** The system should be scalable to handle multiple stores and a growing product catalog.
* **Reliability:** The system should ensure uptime and consistent operation.

**3. System Architecture**

Provide a diagram and description of the system architecture, including:

* **Frontend:**
  + UI designed with HTML, CSS, and JavaScript.
  + Frameworks like React or Vue.js can be used for dynamic interfaces.
* **Backend:**
  + Server-side logic built in Python, Node.js, or Java.
  + RESTful API for communication between frontend and backend.
* **Database:**
  + Relational database such as MySQL or PostgreSQL for storing transactions, customer data, and inventory.
* **Payment Gateway:**
  + Integration with third-party services like Stripe, PayPal, or local payment solutions.
* **Security:**
  + Authentication via OAuth or JWT.
  + Data encryption (SSL/TLS) for transactions and sensitive data.

**4. Technology Stack**

List the technologies and tools that will be used in the development of the project.

**Frontend:**

* **HTML/CSS:** For static structure and styling.
* **JavaScript (React/Vue):** For dynamic content and interactivity.
* **Bootstrap/Tailwind CSS:** For responsive design.

**Backend:**

* **Node.js or Python (Flask/Django):** For handling server-side logic.
* **Express (if using Node.js):** For routing and middleware.

**Database:**

* **MySQL/PostgreSQL:** Relational database for storing transactional data.

**Payment Gateway:**

* **Stripe/PayPal API:** For payment processing.

**Authentication:**

* **JWT/OAuth:** For user authentication and session management.