***Software Requirements Specification (SRS) Document***

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to provide a comprehensive overview of the requirements for the development of the Banking System Application.

**1.2 Scope**

The Banking System Application aims to provide users with a secure and user-friendly platform to perform various banking transactions, manage accounts, and interact with customer support.

**1.3 Document Conventions**

- Use of UML for diagrams (Use Case, Class, etc.).

- IEEE Standard 830 for SRS document format.

**2. System Overview**

**2.1 System Description**

The Banking System Application is designed to offer a suite of banking services, including account management, transaction processing, and customer support, to both individual and joint account holders.

**2.2 System Architecture**

The system will be built using the MERN stack, with a modular architecture to ensure scalability and maintainability.

**3. Functional Requirements**

**3.1 Administrator**

1. **Manage Users:**

- Add, edit, and delete user accounts.

- Reset user passwords.

2. **Configure System Settings:**

- Set transaction limits.

- Define system parameters.

3. **Generate Reports:**

- Generate user activity reports.

- Create financial reports.

4. **System Maintenance:**

- Perform routine system maintenance tasks.

**3.2 Bank Employee**

1. **Handle Customer Inquiries:**

- Respond to user inquiries and issues.

- Provide support for account-related queries.

2. **Process Transactions:**

- Facilitate deposits and withdrawals.

- Execute internal fund transfers.

3**. Manage User Accounts:**

- Verify and update user information.

- Handle account closures.

4. **Provide Customer Support:**

- Assist users with technical issues.

- Resolve general inquiries.

**3.3 Customer Support**

1. **Assist with Account Issues:**

- Provide support for account-related problems.

- Guide users through troubleshooting steps.

2. **Provide Technical Support:**

- Assist users with technical difficulties.

- Collaborate with other teams for issue resolution.

3**. Handle General Inquiries:**

- Answer general questions about the banking system.

- Provide information about services.

**3.4 User Account Holder (Regular and Joint)**

1. **Account Management:**

- Register and log in/log out.

- Update personal information.

- Change account passwords.

2. **Transaction Operations:**

- View account balance.

- Deposit and withdraw funds.

- Transfer funds internally.

- Request account statements.

3. **Security Measures:**

- Change transaction PIN.

**3.5 Transaction History**

1. **View Detailed Transaction History:**

- Access a detailed log of all transactions.

- Filter transactions by date and type.

- Export transaction history in standard formats.

**3.6 Bill Payment (External)**

**1. Add Payee:**

- Add external entities for bill payments.

**2. Make Bill Payments:**

- Initiate bill payments to external payees.

- Schedule recurring payments.

- Set payment reminders.

3. **View Payment History:**

- Access a history of all bill payments made.

**4. Non-Functional Requirements**

**4.1 Performance**

- The system should handle a minimum of 1000 concurrent users.

- Transactions should be processed within 5 seconds.

**4.2 Security**

- User authentication and authorization should follow industry best practices.

- All data transmission should be encrypted using SSL/TLS.

**4.3 Usability**

- The user interface should be intuitive and user-friendly.

- The system should support accessibility standards.

**5. System Interfaces**

**5.1 User Interfaces**

- The application will have web-based interfaces for both desktop and mobile users.

**5.2 External Interfaces**

- Integration with external systems for bill payment services.

## **6. System Constraints**

**6.1 Hardware Constraints:**

****Server specifications:****

**Minimum:** Quad-core CPU (2.5 GHz), 16 GB RAM, 250 GB SSD

**Recommended:** 8-core CPU (3.0 GHz), 32 GB RAM, 500 GB SSD

**Network infrastructure:**

* + High-speed internet (1 Gbps minimum)
  + Load balancing and failover
  + Secure network segmentation

**Supported devices:**

* Desktops/laptops (modern browsers)
* Mobile (Android/iOS) with dedicated apps

**6.2 Software Constraints:**

**Operating system:** Linux (Ubuntu, Red Hat)

**Database:** High-performance, scalable DBMS (PostgreSQL, Oracle)

**Middleware:** Secure communication protocols (HTTPS, TLS) and API Gateway for external integrations.

**6.3 Performance Constraints:**

* **Response times:** Web: under 3 seconds, Mobile: under 5 seconds
* **Transaction throughput:** **1,000 transactions per second (adjust based on transaction type, volume, and future growth)**
* **Scalability:** Horizontal scaling to accommodate increased load

**6.4 Security Constraints:**

* **Authentication/authorization:** Multi-factor authentication (MFA) and role-based access control (RBAC).
* **Encryption:** Industry-standard encryption (AES-256) for data at rest and in transit.
* **Compliance:** PCI DSS, GDPR, and any additional regional regulations.

**6.5 Legal and Regulatory Constraints:**

* Adherence to local/national banking regulations, data protection laws (e.g., GDPR, CCPA), and relevant financial industry standards.

**6.6 Operational Constraints:**

* **Availability:** 99.9% uptime for critical functions; planned maintenance outside peak hours.
* **Backup and recovery:** Regular backups to secure off-site locations and a disaster recovery plan for quick outage recovery.

**6.7 Cultural and Organizational Constraints:**

* **User training:** Materials on secure system usage and ongoing training for new features and updates.
* **Change management:** Well-defined processes for managing system changes and upgrades.
* **Accessibility:** System accessible to users with disabilities.

**6.8 Environmental Constraints:**

* **Data center:** Secure, reliable data center with appropriate environmental controls.
* **Disaster recovery:** Plan for recovering from natural disasters or other disruptions.
* **Environmental considerations:** System designed to minimize power consumption and heat generation.

**6.9 Budgetary Constraints:**

* Defined project budget to be adhered to.
* Cost-effective solutions for hardware, software, and other resources, while maintaining quality and security.

**6.10 Interoperability Constraints:**

* **Integration:** Ability to integrate with existing financial systems and payment networks.
* **Compliance:** Compatibility with relevant industry standards (ISO, SWIFT).

**7. Appendices**

Include any additional information or supporting documents.

**8. Glossary**

Define any technical terms or acronyms used in the document.