Document Outline for an Online Customer KYC Platform Development

1. **Executive Summary**

Know Your Customer (KYC) is a process that banks and other financial institutions implement to verify the identity of their customers. The primary goal of KYC is to ensure that the services provided by the institution are not misused for illegal activities, such as money laundering, fraud, or terrorist financing. KYC is a fundamental component of a bank's risk management and compliance efforts.

The main objective of the project is to develop online KYC procedures for banks which aim to establish a transparent and secure relationship with their customers, mitigate the risk of financial crimes, and comply with regulatory requirements. The KYC process contributes to the overall integrity and stability of the financial system.

2. **Problem Statement & Business Requirements**

Problem Statement:

To develop the KYC Module to manage the customer KYCs. It will be managed by admin and customer will be able to see his KYC.

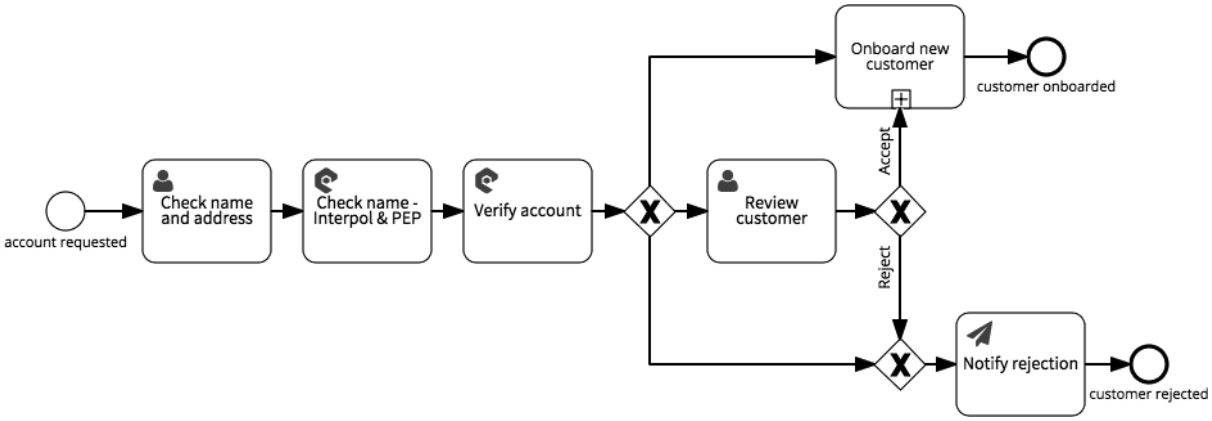
Business Requirements:

- Admin can manage the KYC
- Admin can edit/delete the KYC
- Admin can see the list of all KYC
- Customer can see his KYC

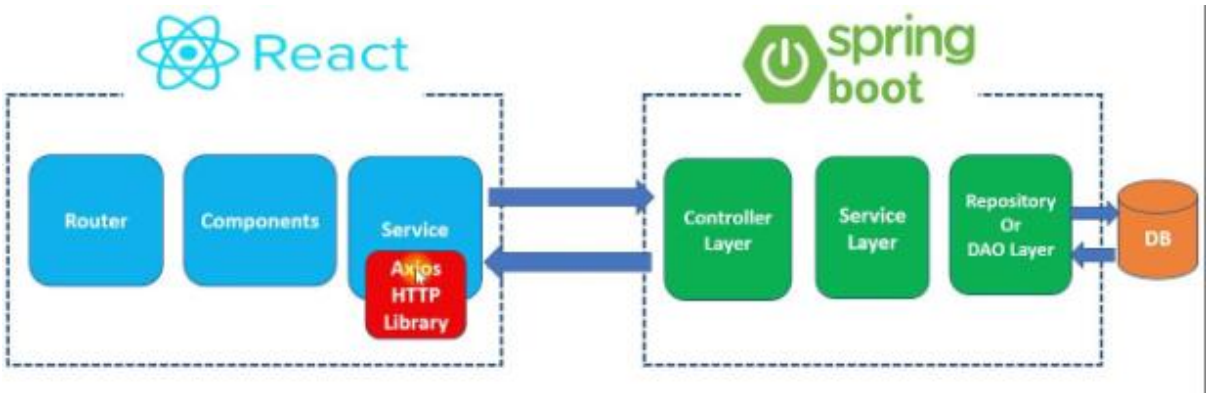
3. **Proposed System Features**

- Login features for Admin and Customer
- Manage KYC screen to adding, updating and deleting the KYC
- Customer View Screen to see the status of his KYC
- Report of All KYCs(Daily, Weekly, Monthly, Yearly)
- Filters for the KYC based on the different categories of submitted KYC document

4. **Proposed Workflow**



5. Application Architecture



- Frontend: Reactjs/Angular for building a responsive and interactive web interface.
- Backend: Springboot for building RESTful APIs and business logic.
- Database: MySQL for data storage.

6. Tool Chain

- Development Tools
  - HTML : Page layout has been designed in HTML
  - CSS : CSS has been used for all the designing part
  - JavaScript : All the validation task and animations has been developed by JavaScript
  - Java Spring Boot : All the business and backend API logic has been implemented in Java Spring Boot
  - MySQL : MySQL database has been used as database for the project
  - React JS : All the frontend logic has been implemented over the React JS
  - Visual Studio Code : For React JS IDE, we have used Visual Studio Code
  - STS : We have used STS (Spring Tool Suite) for developing all spring boot API's
  - Tomcat : Project will be run over the Tomcat server
- Testing Tools: Junit.
- Project Management Tools: Github for end-to-end project management.

7. Development Flow

- Requirement Gathering: In-depth analysis of customer needs and business objectives.
- System Design: Architectural design focusing on scalability and security.
- Development Phases: Agile development with sprints focusing on different features.
- Testing and Quality Assurance: Comprehensive testing strategy including automated and manual tests.
- Deployment: Cloud deployment using AWS.

8. Process Requirement

The following are the discussions that describe how a user uses a system to accomplish a particular goal.

Functionality performed by Admin user:

These are the functionality performed by the admin users.

- Login For Admin
- Forgot password for Admin
- Edit Profile For Admin
- Change Password For Admin
- Logout Functionality
- Dashboard for Admin User
- **Manage Customer**
  - Adding New Customer
  - Edit the Exitinng Customer
  - View details of the Customer
  - Listing of all Customer
- **Manage Banking Service**
  - Adding New Banking Service
  - Edit the Exiting Banking Service
  - View details of the Banking Service
  - Listing of all Banking Service
- **Manage Transaction**
  - Adding New Transaction
  - Edit the Exiting Transaction
  - View details of the Transaction
  - Listing of all Transaction
- **Manage KYC**
  - Adding New KYC
  - Edit the Exiting KYC
  - View details of the KYC
  - Listing of all KYC
- **Reports of the project Online Banking System**
  - Report of all Customers
  - Report of all Banking Services
  - Report of all Transactions
  - Report of all KYCs

Functionality performed by Customer user:

- **Customer Registration :** Customer can register on website using the registration form.
- **Customer Login :** This is the login form, from where customer can login into the system
- **Customer Beneficiary :** This is add customer beneficiary form.
- **Customer Feedback :** This is customer feedback form where customer can give feedback of the Banking Services.
- **Change Password :** This is the change password module from where customer change his account password.

9. Methodology

- Agile Methodology: Implementation using Agile principles for flexibility and adaptability.
- Continuous Integration/Continuous Deployment (CI/CD)
- Documentation: Extensive documentation covering system design, API usage, and user manuals.

10. Testing

1. We need the following fields in the domain/entity class name,surname and age
2. Using Test Driven Development(TDD)cycle let CustomerUnitTest drives the creation of Customer bean.
3. spring-boot-starter-test is used as it comes in with a lot of userful libraries for testing such as JUnit4/5,AssertJ,Hamcrest, Mockito,JSONAssert,JsonPath
4. DataJpaTest is used to test JPA layer,by default this annotation will do following this for us:
   scan all classes annotated with @Entity
   configures 'Spring data JPA repositories`
   configures embedded database if it exists
5. Endpoint Testing WebMvcTest or @AutoConfigure WebTestClient auto configures Spring MVC infrastructure.(auto configures MockMVC) limited to a single controller and is used in combination with mockBean to provide mock implementation
6. Customer creation end point is defined as: api/create

11. Expected Deliverables

- KYC Management Platform: A fully functional and tested online platform.
- Technical Documentation: Detailed documentation for development, deployment, and maintenance.
- User Training Material: Comprehensive guides and training sessions for end-users.

12. Risk Management and Compliance

- Security and Privacy: Stringent security measures to protect sensitive financial data.
- Regulatory Compliance: Ensuring adherence to financial regulations and standards.

- Performance and Scalability: System designed to handle high volumes of transactions and data.

13. Conclusion

The Online KYC Platform is a strategic initiative for DEF Bank to modernize its customer KYC services. Using spring boot, reactjs and mysql this platform is set to offer a superior customer experience, enhanced financial planning capabilities, and drive growth for the bank.