Software Requirements Specification (SRS)

For: AML Detection System with NLP Integration

# 1. Introduction

This document specifies the requirements for an Anti-Money Laundering (AML) detection system. The system includes customer-facing, compliance officer, and admin functionalities. It integrates a Natural Language Processing (NLP) module using NLTK for text analysis of transactions, KYC documents, and investigation notes.

# Technology Stack

Spring Boot(JAVA 17+), Angular, JWT, Cloudinary, reCAPTCHA, Email Service, NLP(NLTK For Text Analysis)

# 2. Overall Description

## 2.1 Product Perspective

The system is designed as a multi-tier architecture with a customer portal, backend services, NLP microservice, and compliance officer dashboard. It supports integration with external watchlists and regulator reporting APIs.

## 2.2 Product Features

- Customer onboarding with KYC document upload  
- Transaction monitoring with real-time risk scoring  
- NLP-based suspicious text detection  
- Rule Engine for compliance policies  
- Alert management and case management  
- Compliance officer dashboard and reporting  
- Admin rule/keyword management

# 3. Specific Requirements

## 3.1 Functional Requirements

### 3.1.1 Customer Module

- FR-C-1: Customer shall register and upload KYC documents.  
- FR-C-2: Customer shall perform secure login with TFA(password + otp).  
- FR-C-3: Customer shall initiate transactions.  
- FR-C-4: Customer shall receive notifications for suspicious or blocked transactions.

### 3.1.2 Compliance Officer Module

- FR-O-1: Officer shall view and filter alerts.  
- FR-O-2: Officer shall investigate alerts with transaction history and NLP analysis.  
- FR-O-3: Officer shall manage cases and add investigation notes.  
- FR-O-4: Officer shall generate regulatory Suspicious Activity Reports (SAR).

### 3.1.3 Admin Module

- FR-A-1: Admin shall manage user roles and access rights.  
- FR-A-2: Admin shall create and modify compliance rules.  
- FR-A-3: Admin shall update suspicious keyword dictionary.  
- FR-A-4: Admin shall monitor system health and maintain audit logs.

### 3.1.4 NLP Module

- FR-NLP-1: System shall preprocess text (tokenization, stopwords, lemmatization).  
- FR-NLP-2: System shall detect suspicious keywords and entities.  
- FR-NLP-3: System shall compute an NLP risk score (0–100).  
- FR-NLP-5: System shall integrate NLP results with the Rule Engine.

## 3.2 Non-Functional Requirements

- NFR-1: Average response time for real-time analysis..  
- NFR-2: All communications shall be secured with HTTPS and JWT authentication.  
- NFR-3: Sensitive data shall be encrypted at rest and in transit.  
- NFR-4: System shall provide audit logs for all access to PII.

# 4. Use Cases

## UC-01: Customer Registration & KYC Upload

Actor: Customer  
Precondition: Customer accesses portal  
Steps:  
 1. Customer fills registration form.  
 2. Customer uploads KYC documents.  
 3. System validates and stores documents.  
Postcondition: Customer account created and pending verification.

## UC-02: Transaction Monitoring with NLP

Actor: Customer, System  
Precondition: Customer initiates transaction  
Steps:  
 1. Customer submits transaction request.  
 2. Rule Engine evaluates transaction.  
 3. NLP microservice analyzes transaction description.  
 4. Combined score computed.  
 5. If threshold exceeded, alert created.  
Postcondition: Transaction processed or blocked, alert stored.

## UC-03: Compliance Officer Investigates Alert

Actor: Compliance Officer  
Precondition: Alert exists  
Steps:  
 1. Officer logs into dashboard.  
 2. Officer reviews alert details with NLP summary.  
 3. Officer adds investigation notes.  
 4. Officer marks alert as Suspicious or Dismissed.  
Postcondition: Case updated with officer’s decision.

## 3.3 Security Enhancements

- SEC-1: System shall implement Google reCAPTCHA during registration and login to prevent automated attacks.  
- SEC-2: All API endpoints shall require JWT (JSON Web Token) authentication for secure communication.  
- SEC-3: JWT tokens shall be short-lived and refresh tokens shall be used for extended sessions.  
- SEC-4: Failed login attempts with invalid reCAPTCHA or expired JWT shall be logged and monitored.

## UC-04: Secure Login with reCAPTCHA and JWT

Actor: Customer  
Precondition: Customer has an active account  
Steps:  
 1. Customer enters username and password.  
 2. System displays reCAPTCHA challenge.  
 3. Customer solves reCAPTCHA.  
 4. System validates credentials and reCAPTCHA.  
 5. If valid, system issues JWT access token and refresh token.  
 6. Customer uses JWT for subsequent API calls.  
Postcondition: Customer securely logged in.