Software Requirement Specifications

Enhanced AML System

Version: [1.0]

Project Code	FYP23-102A
Supervisor	Dr. Zulfiqar Ali Memon
Co Supervisor	
Project Team	Dehya Khurraim Siddiqui 20K-0128 Tayyab Shahzad 20K-1043 Konain Ahmed 20K-1645
Submission Date	10th December 2023

Document History

Version	Name of Person	Date	Description of change
0.0	Dehya Khurraim Siddiqui	24/11/23	*Document Created*
0.1	Dehya Khurraim Siddiqui	24/11/23	*Added INTRODUCTION*
0.2	Tayyab Shahzad	02/12/23	*Added SYSTEM DESCRIPTION*
0.3	Konain Ahmed	05/12/23	*Added FUNCTIONAL REQUIREMENTS*
0.4	Tayyab Shahzad	08/12/23	*Added NON-FUNCTIONAL REQUIREMENTS*
1.0	Dehya Khurraim Siddiqui	10/12/23	*Formatted Document*

Distribution List

Name	Role
Dr. Zulfiqar Ali Memon	*Supervisor*

Document Sign-Off

Version	Sign-off Authority	Sign-off Date

Table of Contents

1. Introduction

1.1. Purpose of Document

This document outlines the Software Requirements Specifications (SRS) for the proposed Anti-Money Laundering (AML) System project. It defines the functional and non-functional requirements for the system, along with its overall architecture, operating environment, and constraints. Providing a comprehensive guide for the project team, stakeholders, and FYP jury.

1.2. Intended Audience

The intended audience for this document includes developers, stakeholders, supervisor and FYP Jury.

1.3 Abbreviations

- *AML: Anti-Money Laundering*
- *SRS: Software Requirements Specification*
- *GCN: Graph Convolutional Network*
- *FP-growth: Frequent Pattern growth*

1.4 Document Convention

- *Acronyms and abbreviations are written in uppercase.*
- *Section headings and subheadings are numbered for easy reference.*

2. Overall System Description

2.1. Project Background

Money laundering is a global threat with significant financial and societal consequences. Traditional methods for detecting and preventing money laundering are often inadequate, highlighting the need for innovative solutions. This project aims to develop an advanced AML system that leverages data mining and machine learning techniques to enhance detection accuracy and efficiency. The Enhanced AML System is developed to combat money laundering using advanced data mining and machine learning techniques.

2.2. Project Scope

The project aims to revolutionize AML strategies, enhance detection capabilities, and contribute to global efforts against money laundering. It encompasses data cleaning, rule mining, classifier construction, reporting, and integration into the financial infrastructure.

2.3. Not In Scope

The project does not cover legal aspects of money laundering, external financial audits, or activities unrelated to the defined AML system layers.

2.4. Project Objectives

The project aims to achieve the following objectives:

- *Enhanced money laundering detection: Improve the accuracy and efficiency of identifying suspicious financial transactions.*
- *Efficient resource utilization: Optimize system performance and minimize computational requirements.*
- *Streamlined investigative processes: Provide investigators with prioritized alerts and tools for effective investigation.*
- *Improved regulatory compliance: Ensure compliance with relevant AML regulations and standards.*
- *Adaptability and scalability: Design a system adaptable to evolving threats and scalable to handle increasing data volumes.*

2.5. Stakeholders

The stakeholders involved in this project include:

- *Financial institutions and regulators*
- *Law enforcement agencies*
- *AML investigators and analysts*
- *Regulatory Bodies*

2.6. *Operating Environment*

The system operates in a financial transaction data environment, requiring seamless integration with existing financial infrastructure.

2.7. *System Constraints*

The project faces the following constraints:

- *Data availability: Access to relevant and up-to-date financial transaction data for training and testing the system.*
- *Computational resources: Managing hardware and software resources required for data processing and machine learning algorithms.*
- *Regulatory compliance: Ensuring adherence to evolving AML regulations and standards.*
- *User acceptance: Training users and gaining acceptance for the system within financial institutions.*

2.8. *Assumptions & Dependencies*

The project assumes the following:

- *Availability of a suitable financial transaction dataset.*
- *Access to necessary hardware and software resources.*
- *Cooperation from financial institutions and regulators.*
- *User willingness to learn and adapt to the new system.*

The project depends on the following:

- *Effectiveness of selected data mining and machine learning algorithms.*
- *Availability of reliable data cleaning and feature engineering techniques.*
- *Robustness of alert generation and prioritization mechanisms.*
- *Efficient investigation workflow and reporting tools.*
- *Continuous system improvement and maintenance.*

3. External Interface Requirements

3.1. Hardware Interfaces

No specific hardware interfaces are required. The system operates in a standard computing environment.

3.2. Software Interfaces

The system interfaces with Python for data preprocessing, Python for backend development, and React for the frontend. Integration with a graph database for efficient data storage is essential.

The system will require interfaces with the following:

- *Financial institution data sources (e.g., APIs)*
- *Regulatory reporting systems*
- *User authentication and authorization systems*
- *Machine learning libraries and tools*

3.3. Communications Interfaces

The system will utilize secure communication protocols for data transfer and alert notification.

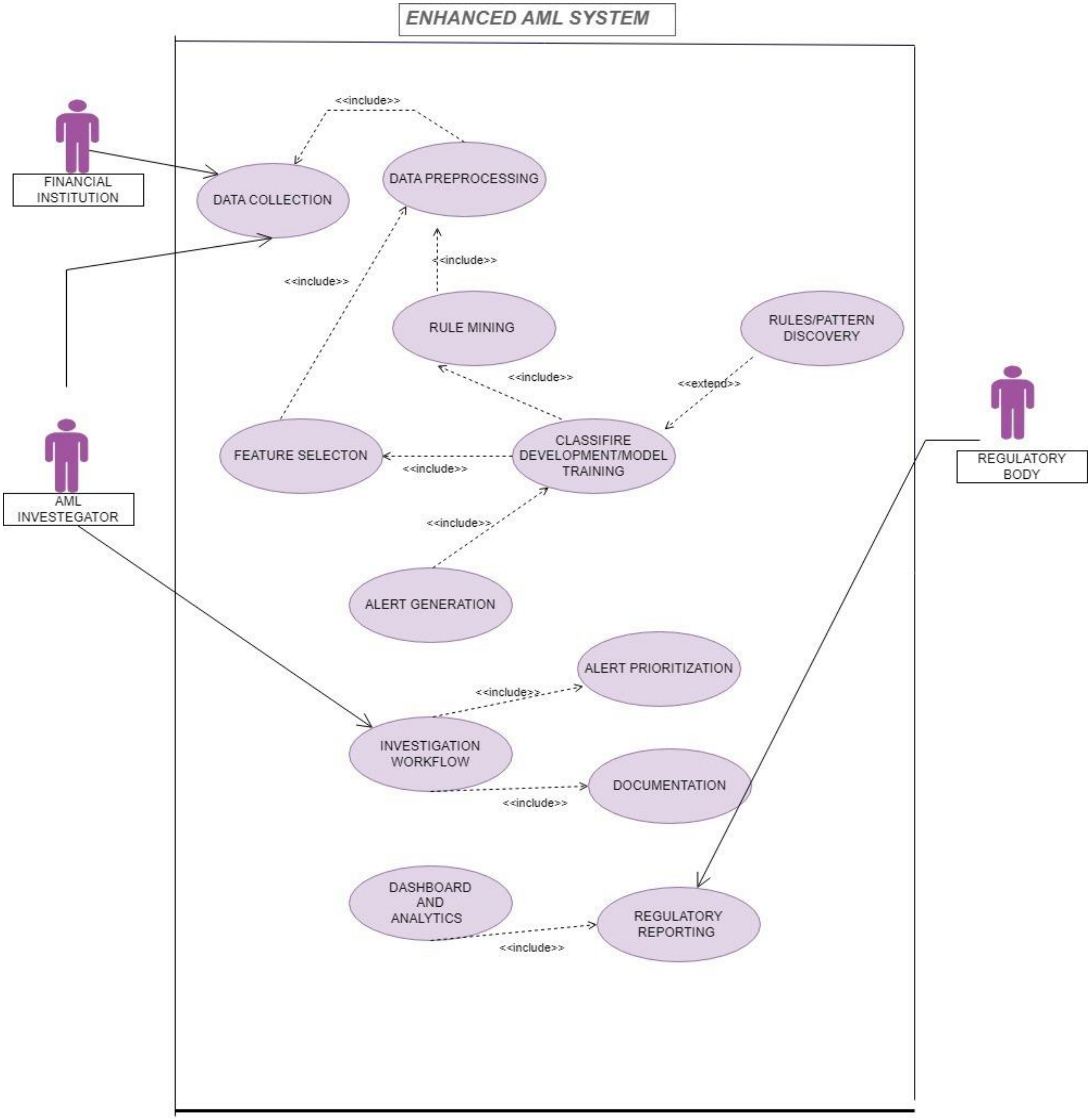
4. Functional Requirements

4.1. Functional Hierarchy

The system will consist of the following modules:

1. ***Data Cleaning Layer***
    - *Data Collection*
    - *Data Preprocessing*
    - *Data Integration*
2. ***Mined Frequent Rules Layer***
    - *Rule Mining Algorithms*
    - *Pattern Discovery*
3. ***Constructing Classifier Layer***
    - *Algorithm Selection*
    - *Classifier Development*
4. ***Reporting Layer***
    - *Alert Generation*
    - *Alert Prioritization*
    - *Investigation Workflow*
    - *Documentation*
    - *Regulatory Reporting*
    - *Dashboard and Analytics*

4.2. Use Cases



4.2.1. Data Collection

Data Collection	
Use case Id:	UC-DC

Actors:	AML Investigator, Financial Institution	
Feature:	Data Collection	
Pre-condition:	Data sources are accessible and available for collection.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	AML Investigator initiates data collection by specifying criteria.	Numbered description of system responses
2.	Financial Institution provides the requested data.	System verifies the received data for completeness and correctness.
Alternate Scenarios:		
1a: *AML Investigator provides invalid criteria.* *System notifies the AML Investigator of the invalid criteria and prompts for correction.*		
Post Conditions		
Step#	***Description***	
	Data relevant to specified criteria is successfully collected.	
Use Case Cross referenced	Investigation Workflow (UC-IW) uses Data Collection for initiating investigations.	

4.2.2. Data Preprocessing

Data Preprocessing		
Use case Id:	UC-DP	
Actors:	System	
Feature:	Data Preprocessing	
Pre-condition:	Collected data is available.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	System initiates data preprocessing by cleaning and handling missing values.	Data is cleaned, and missing values are handled appropriately.
2.	System ensures data quality and consistency through validation.	Processed data is validated for quality and consistency.
Alternate Scenarios:		
1a: *Data preprocessing encounters errors.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		

Step#	***Description***
	Preprocessed data is cleaned, validated, and ready for further processing.
Use Case Cross referenced	Rule Mining (UC-RM) uses Data Preprocessing as input for pattern discovery.

4.2.3. Rule Mining

Rule Mining		
Use case Id:	UC-RM	
Actors:	System	
Feature:	Rule Mining	
Pre-condition:	Preprocessed data is available.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	System performs rule mining on preprocessed data.	Relevant rules and patterns are extracted from the data.
2.	System identifies associations and correlations in financial transactions.	Extracted rules and patterns are analyzed for associations and correlations.
Alternate Scenarios:		
1a: *Rule mining process encounters significant computation errors.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		
Step#	***Description***	
	Rules and patterns in financial transactions are successfully identified.	
Use Case Cross referenced	Pattern Discovery (UC-PD) extends Rule Mining for discovering complex patterns.	

4.2.4. Pattern Discovery

Pattern Discovery		
Use case Id:	UC-PD	
Actors:	System	
Feature:	Pattern Discovery	
Pre-condition:	Rule mining is completed.	
Scenarios		
Step#	***Action***	***Software Reaction***

1.	System extends rule mining to discover complex patterns.	Additional patterns beyond basic rule mining are identified.
Alternate Scenarios:		
1a: *Pattern discovery process encounters significant computation errors.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		
Step#	***Description***	
	Complex patterns in financial transactions are successfully discovered.	
Use Case Cross referenced	- Feature Selection (UC-FS) includes Pattern Discovery for selecting relevant features.	

4.2.5. Feature Selection

Feature Selection		
Use case Id:	UC-FS	
Actors: System		
Feature: Feature Selection		
Pre-condition:	Preprocessed dataset is available.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The system selects relevant features from the preprocessed dataset.	The system selects relevant features from the preprocessed dataset.
2.	The system ensures selected features meet criteria for effective classification.	Selected features are validated for classification criteria.
Alternate Scenarios:		
1a: *Feature selection encounters challenges in identifying relevant features.* *The system provides recommendations for manual intervention by data experts.*		
Post Conditions		
Step#	***Description***	
	Relevant features for effective classification are successfully identified.	
Use Case Cross referenced	- Classifier Development (UC-CD) includes Feature Selection for developing a classifier..	

4.2.6. Classifier Development

Classifier Development		
Use case Id:	UC-CD	
Actors: System		
Feature: Classifier Development		
Pre-condition:	Features are selected.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	System develops a classifier using the selected features and mined rules.	Classifier is constructed based on the identified features and rules.
2.	System ensures the classifier meets performance criteria.	Constructed classifier is validated for performance criteria.
Alternate Scenarios:		
1a: *Classifier development encounters challenges.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		
Step#	***Description***	
	A classifier is successfully developed and ready for alert generation.	
Use Case Cross referenced	- Algorithm Selection (UC-AS) extends Classifier Development for choosing the most suitable algorithm.	

4.2.7. Alert Prioritization

Alert Prioritization		
Use case Id:	UC-AP	
Actors: System		
Feature: Alert Prioritization		
Pre-condition:	Alerts are generated.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	System prioritizes alerts based on risk scores and other criteria.	Generated alerts are prioritized for further investigation.
2.	System ensures prioritized alerts meet predefined criteria.	Prioritized alerts are validated against predefined criteria.

Alternate Scenarios:		
1a: *Alert prioritization encounters challenges.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		
Step#	***Description***	
	Alerts are successfully prioritized for further investigation.	
Use Case Cross referenced	Investigation Workflow (UC-IW) uses Data Collection for initiating investigations.	

4.2.8. Alert Generation

Alert Generation		
Use case Id:	Alert Generation	
Actors:	System	
Feature:	Alert Generation	
Pre-condition:	Classifier is developed.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	System generates alerts for potentially suspicious transactions.	Alerts are generated based on the output of the developed classifier.
2.	System ensures the generated alerts meet predefined criteria.	Generated alerts are validated against predefined criteria.
Alternate Scenarios:		
1a: *Alert generation encounters challenges.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		
Step#	***Description***	
	Alerts for potentially suspicious transactions are successfully generated.	
Use Case Cross referenced	- Alert Prioritization (UC-AP) includes Alert Generation for prioritizing alerts.	

4.2.9. Investigation Workflow

Investigation Workflow		
Use case Id:	UC-IW	
Actors: AML Investigator		
Feature: Investigation Workflow		
Pre-condition:	Alerts are prioritized.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	AML Investigator initiates investigation by reviewing prioritized alerts.	System provides tools and information for investigation.
2.	AML Investigator takes appropriate actions based on investigation findings.	System logs investigation details and outcomes.
Alternate Scenarios:		
1a: *AML Investigator encounters challenges during the investigation.* *System provides recommendations for manual intervention by AML experts.*		
Post Conditions		
Step#	***Description***	
	Investigation details and outcomes are successfully documented.	
Use Case Cross referenced	- Documentation (UC-DO) follows Investigation Workflow for documenting investigation outcomes.	

4.2.10. Documentation

Documentation		
Use case Id:	UC-DO	
Actors: AML Investigator		
Feature: Documentation		
Pre-condition:	Investigation is completed.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	AML Investigator documents investigation details and outcomes.	System provides templates and tools for documentation.
2.	System ensures the documentation meets regulatory and internal standards.	Documented details are validated against standards.

Alternate Scenarios:		
1a: *AML Investigator encounters challenges in documentation.* *System provides recommendations for manual intervention by documentation experts.*		
Post Conditions		
Step#	***Description***	
	Investigation details and outcomes are successfully documented.	
Use Case Cross referenced	- Regulatory Reporting (UC-RR) includes Documentation for regulatory compliance.	

4.2.11. Regulatory Reporting

Regulatory Reporting		
Use case Id:	UC-RR	
Actors:	Regulatory Body	
Feature:	Regulatory Reporting	
Pre-condition:	Relevant data is available.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	Regulatory Body initiates reporting based on the system's findings.	System provides tools and information for generating regulatory reports.
2.	System ensures generated reports meet regulatory standards.	Generated reports are validated against regulatory standards.
Alternate Scenarios:		
1a: *Regulatory reporting encounters challenges.* *System logs errors and notifies the administrator for manual intervention.*		
Post Conditions		
Step#	***Description***	
	Regulatory reports are successfully generated for submission.	
Use Case Cross referenced	- Dashboard and Analytics (UC-DA) includes Regulatory Reporting for monitoring program performance.	

4.2.12. Dashboard and Analytics

Dashboard and Analytics		
Use case Id:	UC-DA	
Actors:	AML Investigator	
Feature:	Dashboard and Analytics	
Pre-condition:	Regulatory reporting is completed.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	AML Investigator accesses dashboards and analytics tools for monitoring AML program performance.	System provides interactive dashboards and analytics features.
2.	System ensures dashboards and analytics tools meet user requirements.	Features are validated against user requirements.
Alternate Scenarios:		
1a: *AML Investigator encounters challenges in accessing dashboards and analytics tools.* *System provides recommendations for manual intervention by system administrators.*		
Post Conditions		
Step#	***Description***	
	AML program performance is successfully monitored through dashboards and analytics tools.	
Use Case Cross referenced	- None	

5. Non-functional Requirements

5.1. Performance Requirements

- ***System response time:*** *The system should be able to process transactions and generate alerts in a timely manner. The average response time for generating an alert should not exceed 10 seconds for transactions considered high-risk.*
- ***Data processing speed:*** *The system should be able to process large datasets efficiently, with a minimum processing speed of 10k transactions per second.*
- ***Scalability:*** *The system should be scalable to handle increasing data volumes and transaction rates. It should be able to adapt to changing data patterns and accommodate future growth.*
- ***Availability:*** *The system should be highly available, with a minimum uptime of 99.5%. This ensures uninterrupted operation and effective detection of suspicious activities.*

5.2. Safety Requirements

- ***Data security:*** *The system should protect sensitive financial data from unauthorized access, disclosure, modification, or destruction. It should implement robust security measures, including encryption, authentication, and access controls.*
- ***System integrity:*** *The system should ensure data integrity and prevent unauthorized modifications or manipulation. This includes implementing data validation and verification mechanisms.*
- ***System resilience:*** *The system should be resilient to cyberattacks and other disruptions. It should have proper disaster recovery and backup procedures in place to ensure continuity of operations.*

5.3. Security Requirements

- ***User authentication and authorization:*** *The system should enforce strong user authentication and authorization procedures to restrict access to sensitive information and functionalities. This may include multi-factor authentication and role-based access control.*
- ***Data encryption:*** *All sensitive data stored and transmitted should be encrypted using industry-standard algorithms. This ensures confidentiality and protects against unauthorized access.*
- ***Audit logging:*** *The system should maintain comprehensive audit logs of all user activities and system events. This allows for monitoring, investigating suspicious activities, and ensuring regulatory compliance.*

5.4. User Documentation

- ***User manual:*** *A comprehensive user manual should be provided to guide users through the system's functionalities and features. It should be written in clear and concise language, with step-by-step instructions and relevant screenshots.*
- ***Training materials:*** *Training materials should be developed to educate users on the system's functionalities and best practices for AML investigations. This can include online modules, video tutorials, and in-person workshops.*

6. References

***[1]** T. E. Senator, “The FinCEN Artificial Intelligence System: Identifying Potential Money Laundering from Reports of Large Cash Transactions,” IAAI, (1995).*

***[2]** P. C. van Duyne and H. de Miranda, “The emperor's clothes of disclosure: Hot money and suspect disclosures,” Crime, Law and Social Change, vol. 31, no. 3, (1999), pp. 245-271.*

***[3]** J. Kingdon, “AI fights money laundering,” Intelligent Systems, IEEE, vol. 19, no. 3, (2004), pp. 87-89*

***[4]** United Nations. Tax abuse, money laundering and corruption plague global finance. https://www.un.org/development/desa/en/news/financing/facti-interim-report.html,September 2020. Accessed: 2021-04-21.*

***[5]** Anders Løland Martin Jullum, Geir ˚Anonsen Ragnar Bang Huseby, and Johannes Lorentzen. Detecting money laundering transactions with machine learning. Journal of Money Laundering Control, 23 No 1, 2020:173–186.*

***[6]** Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I. Weidele, Claudio Bellei, Tom Robinson, and Charles E. Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. CoRR, abs/1908.02591, 2019.*

7. Appendices