

## Table of Contents

1. Introduction .....	5
2. Business Objectives .....	6
3. Scope .....	6
4. Stakeholders.....	7
5. Current State (As-Is) .....	7
6. Proposed Solution (To-Be).....	7
7. Core Functional Design .....	7
7.1 Preprocessing.....	8
7.2. Post processing .....	10
7.3 Post-Screening Alert Analysis & Efficiency Evaluation .....	14
8. Business Requirements .....	15
BR-1.0: Secure User Authentication .....	15
BR-2.0: Reset Password Functionality .....	15
BR-3.0: Import, Preview and Deletion of OFAC SDN file .....	15
BR-3.1: Import of OFAC Sanctions file .....	15
BR-3.1.1: Import of OFAC Sanctions file .....	15
BR-3.1.2: File Name and Format Validation .....	15
BR-3.1.3: Mandatory Field Validation and Import Failure Notification .....	15
BR-3.2: Address File Handling for OFAC SDN Records.....	16
BR-3.3: File Name and Format Validation of OFAC SDN Address file .....	16
BR-3.4: Preview of Imported OFAC SDN list file .....	16
BR-3.5: Preview of Previous (Historical) OFAC SDN list files.....	17
BR-3.6: Deletion of Imported OFAC SDN list file.....	17
BR-4.0: Import, Preview and Deletion of OFAC NON SDN file .....	17
BR-4.1: Import of OFAC NON SDN list file.....	17
BR-4.1.1: Import of OFAC NON SDN list file .....	17
BR-4.1.2: File Name and Format Validation .....	17
BR-4.1.3: Mandatory Field Validation and Import Failure Notification .....	17
BR-4.2: Address File Handling for OFAC NON SDN Records .....	18
BR-4.3: File Name and Format Validation of OFAC NON SDN Address file .....	18

BR-4.4: Cross-File Validation – NON-SDN file and SDN file .....	18
BR-4.5: Preview of Imported OFAC NON-SDN list file and address file .....	19
BR-4.6: Preview of Previous (Historical) OFAC NON-SDN list files .....	19
BR-4.7: Deletion of Imported OFAC NON SDN list file.....	19
BR-5.0: Import, Preview and Deletion of Miscellaneous files.....	19
BR-5.1: Import of Miscellaneous files.....	19
BR-5.2: File Name and Format Validation .....	19
BR-5.3: System-Generated Unique ID for Miscellaneous file Records.....	20
BR-5.4: Manual Entry of Names into Miscellaneous Lists.....	20
BR-5.5: Preview of Imported Miscellaneous List files.....	20
BR-5.6: Preview of Previous (Historical) Miscellaneous list files .....	20
BR-5.7: Deletion of Imported Miscellaneous list files .....	20
BR-5.8: Duplicate Record Handling.....	21
BR-6.0: Import, Preview and Deletion of UK Sanctions list file .....	21
BR-6.1: Import of UK Sanctions list file .....	21
BR-6.1.1: Import of UK Sanctions list file.....	21
BR-6.1.2: File Name and Format Validation .....	21
BR-6.1.3: Mandatory Field Validation and Import Failure Notification .....	21
BR-6.2: Preview of Imported UK Sanctions list file .....	22
BR-6.3: Preview of Previous (Historical) UK Sanctions list files .....	22
BR-6.4: Deletion of Imported UK Sanctions list files .....	22
BR-7.1: Duplicate Record Handling for OFAC Sanctions.....	22
BR-7.2: Scenario-Based Data Synthesis and Preview for OFAC SDN List files.....	22
BR-7.2.1: Double Letter Transformation Logic .....	23
BR-7.2.2: Nick Names Transformation Logic .....	24
BR-7.2.3: Name Swap Transformation Logic .....	24
BR-7.2.4: Abbreviations Transformation Logic.....	25
BR-7.2.5: Anglicized Word Transformation Logic .....	26
BR-7.2.6: Initials Transformation Logic .....	28
BR-7.2.7: Run-Together Transformation Logic .....	29
BR-7.2.8: Bad Data Transformation Logic.....	29
BR-7.2.9: Missing Words Transformation Logic.....	30

BR-7.2.10: Intervening Words Transformation Logic .....	31
BR-7.2.11: Symbolic Replacement Transformation Logic .....	32
BR-7.2.12: Phonetic Substitution Transformation Logic .....	33
BR-7.2.13: Name Alias Transformation Logic .....	34
BR-7.2.14: Country-Based Address Data Matching Logic .....	35
BR-7.3: Duplicate Record Handling for UK Sanctions .....	35
BR-7.4: Scenario-Based Data Synthesis and Preview for UK Sanctions List files .....	35
BR-7.5: Duplicate Record Handling for OFAC NON SDN List File.....	36
BR-7.6: Scenario-Based Data Synthesis and Preview for OFAC NON SDN List files .	36
BR-7.7: Scenario-Based Data Synthesis and Preview for PEP files .....	37
BR-7.8: Scenario-Based Data Synthesis and Preview for Watch List files .....	38
BR-8.1: Synthesized Data Selection for Bank File Generation.....	39
BR-8.2: Preview of Synthesized Data in BANK_A Bank File Format.....	39
BR-8.3: File Structure and Export Format for BANK_A Bank.....	40
BR-8.4: File Size Limitation for BANK_A Bank File Generation.....	40
BR-8.5: Preview of Synthesized Data in BANK_B Bank File Format.....	41
BR-8.6: File Structure and Export Format for BANK_B Bank.....	41
BR-8.7: Synthesized File Summary View with Scenario and Record Count .....	41
BR-9.1: Import of Alert Files received from external Screening System for BANK_A..	42
BR-9.2: Reconciliation of Alert Files with Bank Files .....	42
BR-9.3: Efficiency Report Preview After Alert Comparison .....	42
BR-9.4: Drill-down View of Hit Records from Efficiency Report.....	43
BR-9.5: Display of Internal Violation References on Processed Name Click .....	43
BR-9.6: Drill-down View of No Hit Records from Efficiency Report.....	43
BR-9.7: Consolidated Export of Efficiency Report in PDF Format .....	44
BR-9.8: Preview of Efficiency reports of historical bank files .....	44
BR-10.1: Import of Alert Files from Screening System for BANK_B .....	44
BR-10.2: Reconciliation of Alert Files with Bank Files .....	45
BR-10.3: Efficiency Report Preview After Alert Comparison.....	45
BR-10.4: Drill-down View of Hit Records from Efficiency Report.....	45
BR-10.5: Display of Internal Violation References on Processed Name Click .....	46
BR-10.6: Drill-down View of No Hit Records from Efficiency Report .....	46

BR-10.7: Consolidated Export of Efficiency Report in PDF Format .....	46
BR-10.8: Preview of Efficiency reports of historical bank files.....	47
BR-11: Display and Management of Processed File Metadata .....	47
BR-12.1: Import of Abbreviations Reference Data .....	47
BR-12.2: Preview of Abbreviations Reference Data.....	48
BR-12.3: Abbreviation reference data – Version Control & History Tracking .....	48
BR-13.1: Import of Name Aliases Reference Data .....	50
BR-13.2: Preview of Name Aliases Reference Data.....	50
BR-13.3: Name Aliases reference data – Version Control & History Tracking .....	51
BR-14.1: Import of Anglicized Words Reference Data .....	52
BR-14.2: Preview of Anglicized Words Reference Data .....	53
BR-14.3: Anglicized Words reference data – Version Control & History Tracking .....	53
BR-15.1: Import of Bad Data Reference .....	54
BR-15.2: Preview of Bad Data Reference .....	55
BR-15.3: Bad Data reference – Version Control & History Tracking .....	56
BR-16.1: Import of Double Letters Reference Data .....	57
BR-16.2: Preview of Double Letters Reference Data .....	58
BR-16.3: Double Letters reference data – Version Control & History Tracking.....	58
BR-17.1: Import of Initials Reference Data.....	59
BR-17.2: Preview of Initials Reference Data .....	60
BR-17.3: Initials reference data – Version Control & History Tracking.....	60
BR-18.1: Import of Nickname Reference Data .....	62
BR-18.2: Preview of Nickname Reference Data.....	62
BR-18.3: Nickname reference data – Version Control & History Tracking .....	63
BR-19.1: Import of Intervening Words Reference Data.....	64
BR-19.2: Preview of Intervening Words Reference Data .....	65
BR-19.3: Intervening Words reference data – Version Control & History Tracking.....	65
BR-20.1: Import of MT202 Reference Data.....	66
BR-20.2: Preview of MT202 Reference Data .....	67
BR-21.1: Import of Countries List Reference Data .....	68
BR-21.2: Preview of Countries List Reference Data .....	68
BR-22.1: Delisting Detection via Referential Integrity for OFAC SDN list.....	69

BR-22.2: Delisting Detection via Referential Integrity for OFAC NON SDN list.....	69
BR-22.3: Delisting Detection via Referential Integrity for UK Sanctions list .....	69
9. Functional Requirements.....	70
10. Non-Functional Requirements .....	70
11. Milestones .....	70
12. Technical Overview / Tools and Technologies .....	70
13. Dependencies .....	70
13. Appendices.....	71

## 1. Introduction

This document outlines the business and technical requirements for the product designed to simulate various forms of sanctioned entity name mutations, with the goal

of evaluating and improving the accuracy and performance of sanctions screening systems (e.g., OFAC, UK, White, Watch, and PEP lists).

Sanctions screening is critical for financial institutions, and government agencies to mitigate risks associated with restricted entities. Our product develops Synthesized Sanctions Data to test and evaluate the efficiency of sanctions screening systems used by banks.

The goal is to identify weaknesses in current sanctions screening tools and improve risk detection.

Our Synthesized Sanctions Data product is designed in such a way to test, evaluate, and improve the efficiency of sanctions screening systems used by financial institutions. It generates realistic, artificial sanctions data that simulates real-world sanctioned entities without using any sensitive or actual data, providing a safe and controlled environment for testing and system evaluation.

This product helps businesses identify gaps and weaknesses in their compliance systems, ensuring they can detect potential threats and avoiding legal or financial penalties related to non-compliance with sanctions regulations.

## 2. Business Objectives

- Enable teams to validate sanctions screening tools against known evasion tactics.
- Ensure compliance systems can detect altered versions of sanctioned names.

## 3. Scope

### **In Scope:**

- Integration with OFAC Sanctions, UK Sanctions, White, Watch, and PEP lists.
- Generation of name mutations (typographical, etc.)
- Automated screening test case creation.
- Reports on screening engine performance.

### **Out of Scope:**

- Real-time sanctions screening for live transactions.

## 4. Stakeholders

Role	Name	Responsibility
Lead	[Name]	Define mutation scenarios
Technical analyst	[Name]	Suggest technical requirements and logic for each mutation scenario
QA Team	[Name]	Run and validate test cases
Dev Team	[Name]	Build product

## 5. Current State (As-Is)

- Manual synthetic data creation to validate efficiency.
- Limited to basic or static test cases.

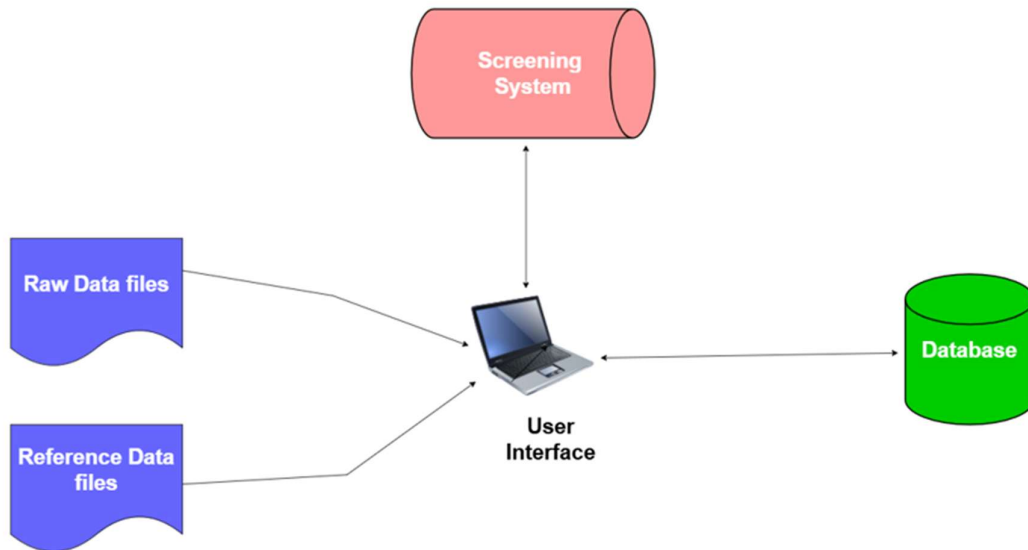
## 6. Proposed Solution (To-Be)

- Automated tool to create realistic name variants.
- Real-time analytics dashboard for coverage and accuracy.

## 7. Core Functional Design

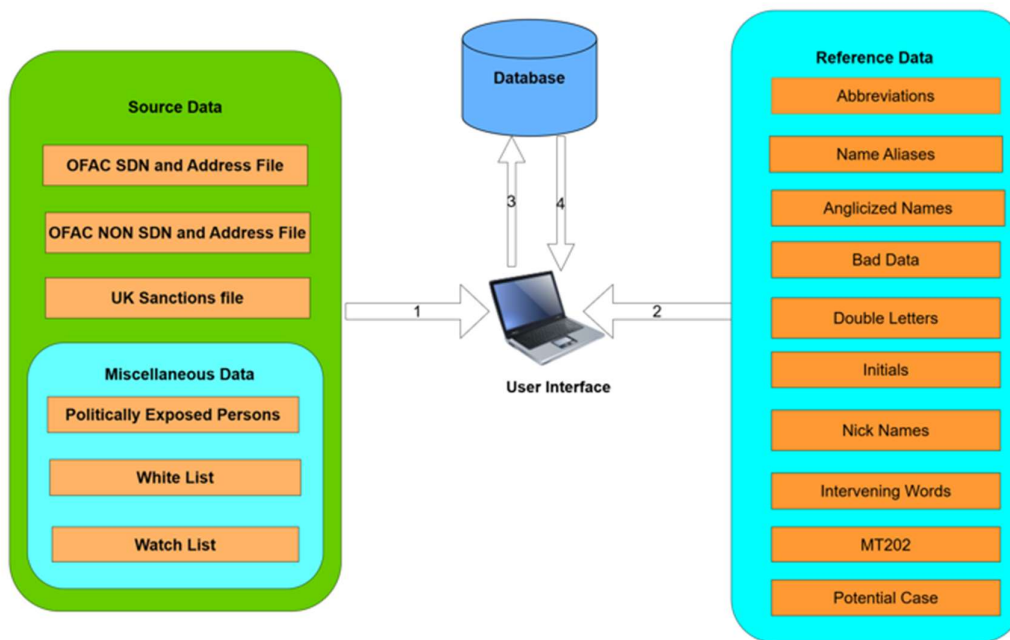
This section outlines the Core Functional Design of the product, encompassing the Sanctions Data Ingestion, Name Mutations, Sourcing and Loading Methodology, and Key Business Scenarios.

### Physical Design



## Logical Design

### 7.1 Preprocessing



- 1 --> User will be able to ingest raw data filesthrough user interface  
 2 --> User will be able to ingest Reference data through user interface  
 3 --> Ingested data through user interface shall be stored into database  
 4 --> Data shall be retrieved from databsed into user interface



## **Inputs to the platform**

The platform ingests a variety of input files that are required to perform comprehensive name screening, mutation generation, matching, and export. These files include:

- Official sanctions lists (OFAC, UK, etc.)
- Politically Exposed Persons (PEPs)
- Institution-specific lists (Whitelists/Watch Lists)
- Reference data used for name mutation logic (abbreviations, nicknames, etc.)
- Known data anomalies for correction (bad data)

Each file type serves a distinct function in the preprocessing, transformation.

## **Data Normalization**

### **Data Cleansing**

All fields imported from the raw files must be automatically cleansed to remove leading and trailing whitespaces.

### **Purpose:**

- Ensure consistent data formatting across the modules.
- Prevent false negatives in sanctions name matching due to whitespace mismatches
- Ensure integrity and alignment in output formats (fixed-width bank files)

## **Data Deduplication**

Deduplication refers to identifying and consolidating duplicate entries in the dataset.

Objectives of deduplication is to ensure the following

- Improve matching accuracy by avoiding duplicate hits
- Avoid redundant processing in mutation and export pipelines
- Simplify results for analysts
- Ensure one unique representation per individual/entity wherever possible

Deduplication logic is flexible and configurable on a per-source basis, as different lists (e.g., OFAC, UK, PEP, Internal Watch Lists) define identity and uniqueness using different sets of fields. The system supports column-level configuration to identify duplicates appropriately for each input format.

## **Data Alignment(Consistency)**

Data alignment refers to the standardization of field structures, formats, and positions across all records and modules involved in the system. This ensures that all entries are processed uniformly and are compatible among the modules such as import, data processing and data selection.

All data must be aligned in terms of:

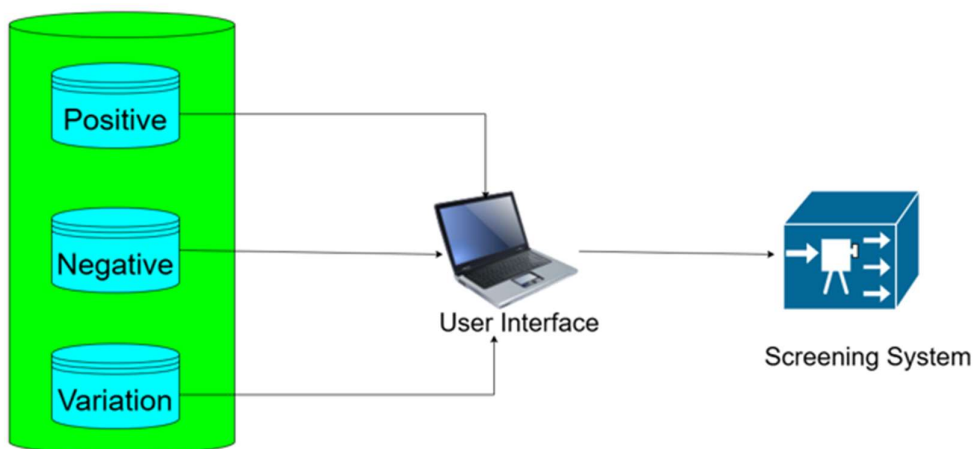
- **Field order and structure**
- **Data types**
- **Fixed widths (if required)**
- **Empty field handling**

**Objective:**

- Ensure interoperability between internal modules (cleansing → mutation → export)
- Maintain format consistency for compliance and auditability
- Enable automated transformation into required bank formats without manual correction
- Reduce false mismatches or processing errors due to misaligned data fields

## 7.2. Post processing

The pre-processed data that is stored in database is post-processed based on three key scenario types selected by the user: Positive, Negative, and Variations. Each scenario determines which processing rules to apply. If applicable, the rules are executed, and the output is structured into bank-specific files for downstream screening systems.



**Positive Scenarios**

The Positive Scenario consists of entities that are direct matches from official or internal high-risk lists. These records are considered screening-ready and do not require any name mutation.

Scenario	Description
OFAC SDN	Baseline file of OFAC SDN list of entities.
OFAC NON SDN	Baseline file of OFAC NON SDN list of entities.
WATCH List	internally curated list of individuals and corporate entities maintained by the organization.
UK SANCTIONS	Baseline file of UK SANCTIONS list of entities.
MT202	MT202 messages were screened for sanctioned entity names.
PEP	Politically Exposed Persons list

#### **Processing Logic:**

##### **1. No Mutations Applied**

- Entity names are taken exactly as they appear in the source file (e.g., OFAC SDN, PEP) and no data transformation is performed.

##### **2. Bank File Generation**

- These entities are exported as-is in the required bank file format.
- The output is compliant with the external screening system's expected format.

##### **3. Screening Alert Expectation**

- Upon sending these records to the external screening engine (e.g., Safe Watch), the system is expected to return an alert for each entry.
- These alerts serve as a validation checkpoint, ensuring the platform successfully detected all known sanctions list names without mutation.

#### **Negative Scenarios**

The Negative Scenario includes entities that are considered low risk or non-actionable, or belong to whitelist, incomplete, or irrelevant data sets. These entries are not intended to result in alerts during external screening and are used to test suppression behaviour or validate configuration.

#### **Processing Logic:**

##### **1. No Name Mutations**

- Names are kept exactly as-is from the source or input files.

- No mutation logic (aliases, abbreviations, etc.) is applied.

## 2. Bank File Generation

- Negative Scenario entries are included in the bank file for control purposes.
- Entries are used to confirm that alerts are not falsely triggered.

## 3. Screening Outcome Expectation

- These entries are not expected to generate any alerts in the external screening engine.
- If alerts are triggered, they may indicate false positives or misconfiguration.

Scenario	Description
Negative Case	Negative scenario to evaluate the robustness of the matching algorithm.
PC-LFM	Another negative scenario to evaluate the robustness of the matching algorithm.
OFAC_SDN_DELISTED	list of entities, removed from OFAC SDN list of sanctions
OFAC_NON_SDN_DELISTED	list of entities, removed from OFAC NON SDN list of sanctions
UK_SANCTIONS_DELISTED	list of Entities, removed from UK Sanctions list
WHITE LIST	internally curated list of individuals and corporate entities maintained by the organization
MT202	MT202 messages were screened for unsanctioned entity names.

## Variation Scenarios

The Variation Scenarios includes entities that require name mutations to increase match coverage in external screening engines. The goal is to simulate real-world name distortions and ensure the screening engine can detect them correctly.

### Processing Logic:

#### 1. Name Mutation Required

- Multiple name variations are generated using transformation rules:
  - Abbreviations (e.g., "Mohamed" → "Mohd")
  - Initials (e.g., "Mohamed Ahmed" → "M A")

- Anglicized versions (e.g., "Giuseppe" → "Joseph")
- Aliases and nicknames
- Letter replacements (e.g., double-letter variations)

## 2. Bank File Generation

- All generated mutations are included in the bank file.
- Each variation is treated as a distinct entry to be screened.
- Export format aligns with external engine requirements.

## 3. Alert Expectation

- All mutations are expected to trigger alerts when screened.
- If no alert is generated, it is flagged as a false negative and included in the Efficiency Report.

Scenario	Description
Abbreviations	Abbreviations are introduced into entity names.
Anglicized Words	Various English spellings of foreign words are inserted into entity name
Bad Data	Introducing bad data (numbers, characters, special characters, combinations) at random positions within the name.
Double Letters	Letters are removed or duplicated to create the appearance of a double letter word in the entity name.
Initials	Initials are introduced into entity names.
Intervening Words	Random words are inserted into the entity name.
Missing Words	Word from the entity is removed for matching.
Nicknames	Nicknames are introduced into entity names.
Run Together	Words of entity names are glued together by removing space.
Symbolic Replacement	"Symbolic Replacement in Text", as it involves replacing a specific character ('a') with a symbol ('@')
Name Swap	First name and last name are swapped in individual entity name
Phonetic Substitution	letters or sounds are replaced by others to mimic or modify pronunciation
Name Aliases	internally curated list of individuals and corporate entities maintained by the organization
Country-Based Address Data	The system will identify and extract address data that corresponds to the listed countries in the

	reference file and return the matched addresses along with their country names.
--	---

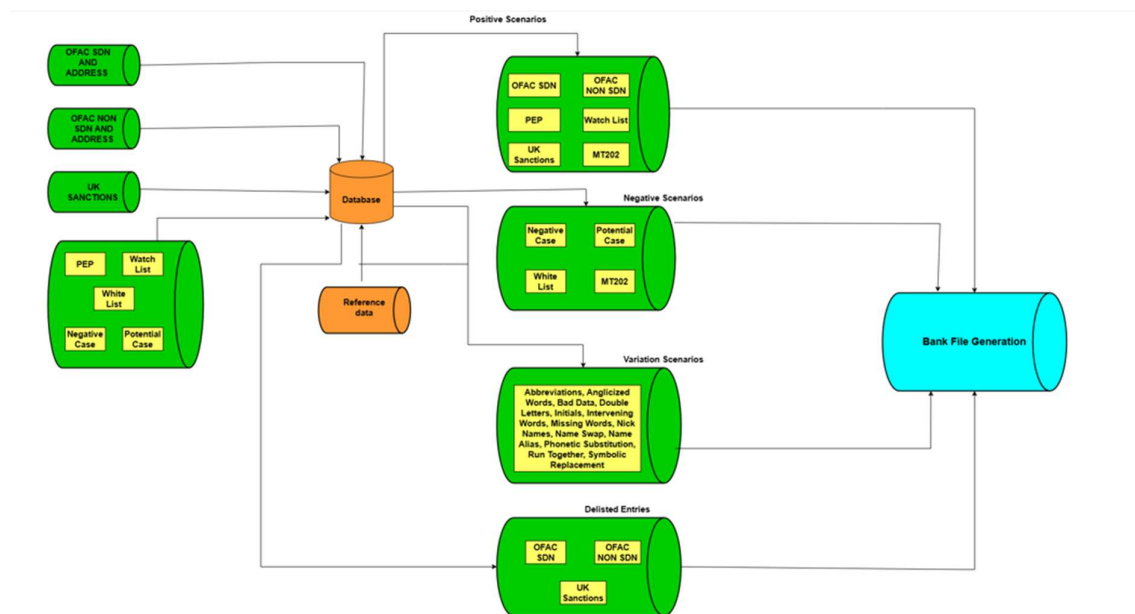
### 7.3 Post-Screening Alert Analysis & Efficiency Evaluation

After generating and sending bank-specific files to the sanctions screening system, the system returns individual alert reports for each bank. These alert files contain potential hits identified during the screening process.

To evaluate the effectiveness of the name mutation and transformation logic applied during file generation, the received alert reports are compared against the original bank files. This comparison enables the calculation of screening efficiency metrics, such as:

- Total Records processed for each scenario,
- Hit Count: number of records that matched with alerts,
- Hit Percentage = (Hit Count / Total Records) × 100,
- No hit Count = number of records that are not matching with alerts,
- No hit Percentage = (No hit Count / Total Records) × 100.

#### Process Data Flow Diagram



## 8. Business Requirements

### BR-1.0: Secure User Authentication

The system must provide a secure, user-friendly login functionality that enables registered users to access their accounts using valid credentials (e.g., username and password). This functionality is essential for protecting user data and ensuring that only authorized users can access the platform.

### BR-2.0: Reset Password Functionality

The system must provide a secure, user-friendly reset password feature that enables registered users to recover access to their accounts.

### BR-3.0: Import, Preview and Deletion of OFAC SDN file

The system must allow users to import raw data file containing sanctions information. Upon upload, the preview of the contents must be presented in the user interface to enable verification for further processing.

#### BR-3.1: Import of OFAC Sanctions file


##### BR-3.1.1: Import of OFAC Sanctions file

The system must allow users to import the official OFAC Sanctions List file exactly as provided by the U.S. Department of the Treasury's Sanctions List Search tool, without requiring any modifications to its structure or format. This ensures seamless integration with official government-provided data and supports regulatory compliance.

[OFAC - Sanctions List Site](#)

##### BR-3.1.2: File Name and Format Validation

The system must validate that the OFAC file being uploaded is in the supported format (i.e., CSV) and file name. If the format is invalid, the system must display an appropriate error message and prevent uploading.

File Name  sdn.csv

##### BR-3.1.3: Mandatory Field Validation and Import Failure Notification

The system must validate that all mandatory fields in the OFAC sanctions file that is being uploaded are present and populated. If any required field is missing or contains null/blank values, the import process must be terminated, and the information of these

records needs to be maintained in separate log tables. The system must then display a clear, user-friendly alert indicating the failure reason.

**Mandatory Fields:**

Attribute	Description
ENT_NUM	Entity Number – A unique numeric identifier assigned by OFAC to each SDN record. This ID helps link records across associated files (e.g., add.txt, aka.txt).
SDN_Name	Sanctioned Party Name – The primary name of the individual, entity, vessel, or aircraft as it appears on the SDN list. For individuals, this is usually the full name; for entities, it is the organization name.
SDN_TYPE	Sanctions Entity Type – Indicates the category of the sanctioned party. Common types include Individual, Entity, Vessel, or Aircraft. This helps define how the entity is treated by the screening engine.

### BR-3.2: Address File Handling for OFAC SDN Records

The system must support the upload of an Address File that contains addresses associated with SDN records, linked via **ENT\_NUM**. The Address File is used to manage and update address data for individuals or entities listed in the SDN file.

**Processing Rules:**

- If the **ID** in the Address File already exists in the system:
  - The existing address records for that ID must be replaced with the new records from the file.
- If the **ID** is new (not previously present in the system):
  - The address records must be inserted as new entries.
- The system must support multiple address entries for the same **ENT\_NUM**.
- Data shall be available in any one of the address fields along with ID

### BR-3.3: File Name and Format Validation of OFAC SDN Address file

- The system must validate that the address file being uploaded is in the supported format (i.e., CSV) and file name. If the format is invalid, the system must display an appropriate error message and prevent uploading.
- File Name = add.csv

### BR-3.4: Preview of Imported OFAC SDN list file and address file

The system must provide a preview feature for the OFAC SDN file and address file once after they are uploaded and committed to the database. This preview allows users to



verify the structure and contents of the file. Display a tabular view of the uploaded data, including all fields

### **BR-3.5: Preview of Previous (Historical) OFAC SDN list files**

The system must provide a preview feature for the OFAC NON SDN files from the database. Display a tabular view of the uploaded data, including all fields.

### **BR-3.6: Deletion of Imported OFAC SDN list file**

The system must provide users with the ability to delete an imported OFAC SDN file from the platform through the user interface (UI), along with all associated processed records.

## **BR-4.0: Import, Preview and Deletion of OFAC NON SDN file**

The system must allow users to import raw data file containing sanctions information. Upon upload, the preview of the contents must be presented in the user interface to enable verification for further processing.

### **BR-4.1: Import of OFAC NON SDN list file**

#### **BR-4.1.1: Import of OFAC NON SDN list file**

The system must allow users to import the official OFAC NON SDN List file exactly as provided by the U.S. Department of the Treasury's Sanctions List Search tool, without requiring any modifications to its structure or format. This ensures seamless integration with official government-provided data and supports regulatory compliance.

[OFAC - Sanctions List Site](#)

#### **BR-4.1.2: File Name and Format Validation**

The system must validate that the OFAC NON SDN list file being uploaded is in the supported format (i.e., CSV) and file name. If the format is invalid, the system must display an appropriate error message and prevent uploading.

File Name  cons\_prim.csv

#### **BR-4.1.3: Mandatory Field Validation and Import Failure Notification**

The system must validate that all mandatory fields in the OFAC NON SDN list file that is being uploaded are present and populated. If any required field is missing or contains null/blank values, the import process must be terminated, and the information of these records needs to be maintained in separate log tables. The system must then display a clear, user-friendly alert indicating the failure reason.

### Mandatory Fields:

Attribute	Description
ENT_NUM	Entity Number – A unique numeric identifier assigned by OFAC to each SDN record. This ID helps link records across associated files (e.g., add.txt, aka.txt).
SDN_Name	Sanctioned Party Name – The primary name of the individual, entity, vessel, or aircraft as it appears on the SDN list. For individuals, this is usually the full name; for entities, it is the organization name.
SDN_TYPE	Sanctions Entity Type – Indicates the category of the sanctioned party. Common types include Individual, Entity, Vessel, or Aircraft. This helps define how the entity is treated by the screening engine.

### BR-4.2: Address File Handling for OFAC NON SDN Records

The system must support the upload of an Address File that contains addresses associated with NON SDN records, linked via **ENT\_NUM**. The Address File is used to manage and update address data for individuals or entities listed in the NON SDN file.

#### Processing Rules:

- If the **ID** in the Address File already exists in the system:
  - The existing address records for that ID must be replaced with the new records from the file.
- If the **ID** is new (not previously present in the system):
  - The address records must be inserted as new entries.
- The system must support multiple address entries for the same **ENT\_NUM**.
- Data shall be available in any one of the address fields along with ID

### BR-4.3: File Name and Format Validation of OFAC NON SDN Address file

- The system must validate that the address file being uploaded is in the supported format (i.e., CSV) and file name. If the format is invalid, the system must display an appropriate error message and prevent uploading.
- File Name = cons\_add.csv

### BR-4.4: Cross-File Validation – NON-SDN file and SDN file

The system must perform a cross-file validation to ensure the presence of latest NON-SDN records in the latest SDN file, based on a defined set of matching attributes

(ENT\_NUM, SDN\_NAME, SDN\_TYPE). If any of the records in the NON-SDN file do not have an exact match in the SDN file, the system must identify and extract these as pure NON-SDN records.

#### BR-4.5: Preview of Imported OFAC NON-SDN list file and address file

The system must provide a preview feature for the OFAC NON SDN file and address file once after they are uploaded and committed to the database. This preview allows users to verify the structure and contents of the file. Display a tabular view of the uploaded data, including all fields.

#### BR-4.6: Preview of Previous (Historical) OFAC NON-SDN list files

The system must provide a preview feature for the OFAC NON SDN files from the database. Display a tabular view of the uploaded data, including all fields.

#### BR-4.7: Deletion of Imported OFAC NON SDN list file

The system must provide users with the ability to delete an imported OFAC NON SDN file from the platform through the user interface (UI), along with all associated processed records.

### BR-5.0: Import, Preview and Deletion of Miscellaneous files

The system must allow users to upload, preview, and delete raw data file containing entities information. Upon upload, the preview of the contents must be presented in the user interface to enable verification for further processing.

#### BR-5.1: Import of Miscellaneous files

The system must support the upload functionality of internally maintained watchlist files, including **PEP (Politically Exposed Persons), Whitelist, Watch List, Negative and Potential Case** data. These lists are defined and managed by the bank to support internal compliance policies and risk-based customer screening.

#### BR-5.2: File Name and Format Validation

The system must validate that the Miscellaneous file being uploaded is in the supported format (i.e., CSV) and file name. If the format is invalid, the system must display an appropriate error message and prevent uploading.

Misc\_DDMMYY

File Extension ? .csv

File Attributes ? Name, Type

### BR-5.3: System-Generated Unique ID for Miscellaneous file Records

For files of type **PEP (Politically Exposed Persons)**, **White**, **Watch**, **Negative** and **Potential Case** lists the system must automatically generate a **Unique ID** for each record during the import process. Since these files typically contain only the **Name** and **Type** fields.

#### Processing Rules:

- A unique alphanumeric ID (e.g., PEP\_123456, WL\_123456, GL\_123456, NC\_123456, PC\_123456 etc.) must be generated sequentially.
- The generated Unique ID must be:
  - Consistently stored alongside the Name field,
  - Displayed in UI previews.

### BR-5.4: Manual Entry of Names into Miscellaneous Lists

The system must allow users to manually enter new names into internal watchlists (e.g., PEP, Whitelist, or Watch List) directly through the user interface. Upon submission, the system must:

- Automatically generate a **Unique ID** for each new entry
- Store the new record in the appropriate list repository

This functionality allows users to maintain up-to-date internal lists dynamically and ensures all records are traceable through system-assigned identifiers.

### BR-5.5: Preview of Imported Miscellaneous List files

The system must provide a preview feature for the Miscellaneous List files once after they are uploaded and committed to the database. This preview allows users to verify the structure and contents of the file. Display a tabular view of the uploaded data, including all fields.

### BR-5.6: Preview of Previous (Historical) Miscellaneous list files

The system must provide a preview feature for the Miscellaneous files from the database. Display a tabular view of the uploaded data, including all fields.

### BR-5.7: Deletion of Imported Miscellaneous list files

The system must provide users with the ability to delete an imported Miscellaneous list files file from the platform through the user interface (UI), along with all associated processed records.

## BR-5.8: Duplicate Record Handling

The system must identify and handle duplicate records during the Miscellaneous list files import process based on a defined deduplication rule. The logic must operate as follows:

- If a record has the same NAME, and TYPE as another record uploaded, must treat these as duplicates and retain only one instance of the record.

## BR-6.0: Import, Preview and Deletion of UK Sanctions list file

The system must allow users to import raw data file containing sanctions information. Upon upload, the preview of the contents must be presented in the user interface to enable verification for further processing.

### BR-6.1: Import of UK Sanctions list file

#### BR-6.1.1: Import of UK Sanctions list file

The system must support the import, preview, and deletion of the UK Sanctions List file to facilitate compliance with United Kingdom financial sanctions regulations. This list typically includes designated individuals, entities, and vessels subject to UK-imposed restrictions.

[The UK Sanctions List - GOV.UK](#)

#### BR-6.1.2: File Name and Format Validation

The system must validate that the file being uploaded is in the supported format (i.e., .ods) and file name. If the format is invalid, the system must display an appropriate error message and prevent uploading.,

File Name  FCDO\_SL\_Tue\_May 13 2025.ods

#### BR-6.1.3: Mandatory Field Validation and Import Failure Notification

The system must validate that all mandatory fields in the UK Sanctions list file that is being uploaded are present and populated. If any required field is missing or contains null/blank values, the import process must be terminated. The system must then display a clear, user-friendly alert indicating the failure reason.

#### Mandatory Fields:

Attribute	Description
Unique_id	A globally unique identifier assigned to each entry in the UK Sanctions List. It ensures each designation is uniquely tracked across systems and versions.

Name	The primary name of the sanctioned individual or entity. This can include full names for persons or legal names for companies, vessels, or other organizations.
Type	The type of listed party. Common values include Individual, Entity, Vessel, Aircraft. This helps systems categorize and process listings appropriately.

## BR-6.2: Preview of Imported UK Sanctions list file

The system must provide a preview feature for the UK Sanctions list file once after they are uploaded and committed to the database. This preview allows users to verify the structure and contents of the file. Display a tabular view of the uploaded data, including all fields.

## BR-6.3: Preview of Previous (Historical) UK Sanctions list files

The system must provide a preview feature for the UK Sanctions list files from the database. Display a tabular view of the uploaded data, including all fields.

## BR-6.4: Deletion of Imported UK Sanctions list files

The system must provide users with the ability to delete an imported Miscellaneous list files file from the platform through the user interface (UI), along with all associated processed records.

## BR-7.1: Duplicate Record Handling for OFAC Sanctions

The system must identify and handle duplicate records while synthesizing the OFAC sanctions data based on a defined deduplication rule. The logic must operate as follows:

- If a record has the same ENT\_NUM, SDN\_NAME, and SDN\_TYPE as another record in the uploaded file, the system must treat these as duplicates and retain only one instance of the record.
- If multiple records have the same SDN\_NAME and SDN\_TYPE, but different ENT\_NUM, the system must treat each record as unique and include all in the import.

## BR-7.2: Scenario-Based Data Synthesis and Preview for OFAC SDN List files

The system must allow users to select specific data processing scenarios (i.e., *Phonetic Substitution*, *Abbreviation*, etc) for each selected file (current or historical). Based on the selected file (current or historical) and then selected scenario, the system

must dynamically process and generate synthesized data and present a preview within the UI.

- For each file listed (Current or History), the user must be able to:
  - Choose single scenario or multiple scenarios from a predefined dropdown or selection list,
  - View the resulting synthesized dataset in a preview table.
- Each scenario must trigger its corresponding logic

#### **Preview Features:**

- Once processed, the UI must display following attributes in tabular format pertaining to the selected file for preview
  - Unique\_id
  - Original entity name
  - Synthesized name
  - Scenario name

#### **Note:**

Current File Selection ☑ User must be able to select only the latest imported file.

Historical File Selection ☑ User must be able to select only the top three historical files.

### **BR-7.2.1: Double Letter Transformation Logic**

The Double Letter transformation logic operates in two distinct modes to support both inflation (single → double letter) and reduction (double → single letter) patterns. These transformations are applied to names from raw input files (e.g., SDN, Non-SDN, UK sanctions) to generate potential variants for screening purposes.

#### **Case 1: Single to Double Letter Substitution (Inflation)**

If the entity name contains any character defined in the reference mapping, replace a single occurrence of that character with its double-letter variant (e.g., J → JJ).

**Example: John is transformed as JJohn**

#### **Rules:**

1. If a name contains multiple matching letters from the reference map, only one replacement occurs.
2. If the same letter appears more than once, only one occurrence is replaced.

3. Case-sensitive substitution: Uppercase and lowercase letters are treated differently.

### **Case 2: Double to Single Letter Substitution (Reduction)**

If the entity name contains any double-letter pattern from the reference mapping (e.g., ll → l), replace one occurrence with its single-letter equivalent.

**Examples: William is transformed as Wiliam**

#### **Rules:**

1. If the name has no matching double-letter pattern, no transformation occurs.
2. If the name contains multiple matching patterns, only one replacement occurs.
3. Even if the same pattern appears multiple times, only a single substitution is performed.

### **BR-7.2.2: Nick Names Transformation Logic**

This transformation replaces formal names within entity records (from SDN and Non-SDN raw files) with their corresponding nicknames, based on a predefined reference mapping (e.g., Aamir → Ami).

- The logic checks the entire entity name to detect any word that exactly matches a key in the reference mapping.
- Upon a match, only one replacement is performed per name.
- The substitution is case-sensitive and word-bound (not part of other alphanumeric sequences).

Example: **CHAUDHRY, Aamir Ali** is transformed as **CHAUDHRY, Ami Ali**  
(*Aamir is replaced with its nickname Ami*)

#### **Key Rules**

1. Single Replacement Rule:  
If the name contains multiple matches from the nickname reference map, only one of them will be replaced.
2. Word Boundary Rule:  
Nickname replacements do not occur if the formal name is embedded inside numbers and other letters (e.g., 123Aamir123).

### **BR-7.2.3: Name Swap Transformation Logic**

This logic handles entity names with commas by swapping the first and last name positions to normalize naming conventions and support alternative match patterns in sanctions screening.



- If an entity name contains a comma (,), it is assumed to separate the last name (before the comma) and the first/middle name(s) (after the comma).
- The transformation removes the comma and places the first/middle name(s) before the last name, separated by a single space.

Example: **JABRIL, Ahmad Ali** is transformed as **Ahmad Ali JABRIL**  
*(Swapped around the first comma)*

### Key Rules

1. Comma Required:  
The transformation is applied only if the name contains at least one comma.
2. First Comma Usage:  
If there are multiple commas, the first comma is used for the swap. The rest of the name remains unchanged.
3. Minimum Word Count:  
The name must contain more than one word (at least one word before and after the comma) to be processed.
4. Formatting Post-Swap:
  - The comma is removed.
  - A single space is added between the name parts.

### BR-7.2.4: Abbreviations Transformation Logic

This transformation expands abbreviated terms in entity names using a predefined abbreviation-to-full-form mapping. It ensures broader match coverage by normalizing commonly used company suffixes and organizational terms.

- If the entity name contains one or more abbreviations defined in the reference list, each abbreviation is replaced individually to generate separate variants.
- If an abbreviation appears multiple times within a name, all occurrences are replaced in a single variant.
- The process generates multiple transformed records if a name includes multiple distinct abbreviations.

### Examples

- Original Name: ANGLO-CARIBBEAN CO., LTD.
- Reference Mapping:
  - CO. → COMPANY

- LTD. → LIMITED
- Transformed Variants:
  0. ANGLO-CARIBBEAN CO., LIMITED (*LTD. → LIMITED*)
  1. ANGLO-CARIBBEAN COMPANY LIMITED (*CO. → COMPANY, LTD. → LIMITED*)
  2. ANGLO-CARIBBEAN COMPANY LTD. (*CO. → COMPANY*)

## Key Rules

1. Multiple Variant Output:
  - If a name includes 3–4 different abbreviations, generate 3–4 separate variants, each applying one abbreviation replacement.
  - If an abbreviation appears multiple times in one name, replace all its instances within that variant.
2. Space-Boundary Rule:
  - Replacements occur only if the abbreviation is bounded by space or punctuation (e.g., " CO. " or "CO., ").
  - No substitution is done if the abbreviation is embedded inside other words (e.g., ANGLOCO.).
3. Case-Insensitive Matching:
  - Uppercase and lowercase are treated equally (e.g., Ltd. and LTD. will both be replaced by Limited).
4. Start-of-Name Rule:
  - If the abbreviation is at the beginning of the name and followed by a space, it will be replaced.

## BR-7.2.5: Anglicized Word Transformation Logic

This logic replaces non-English, non-standard, or culturally specific spellings in names with their more commonly recognized anglicized versions, based on a reference table. This improves consistency and match ability in the sanctions screening process.

- Applies to names from individuals or organizations in the sanction's dataset.
- Uses a reference table mapping:

## Example

- Original: Mohammed Ali Syed

- Reference List Includes:
  - Mohammed → Muhammad
  - Ali → Aly
  - Syed → Syad
- Transformed Variants:
  0. Muhammad Ali Syed
  1. Mohammad Aly Syed
  2. Mohammad Aly Syad

### Key Rules

1. One-at-a-Time Replacements:
 

Each record is transformed once per reference word, resulting in multiple output variants.
2. Full Word Match Only:
 

Replacements occur only for exact word matches, not partial substrings.

  - Ali → Aly
  - Aliya or 123Ali → no change
3. All Occurrences Replaced in a Variant:
 

If the same word appears multiple times, all instances are replaced within that transformed variant.

  - E.g., 'Ali Hasan Ali → 'Aly Hasan Aly
4. Non-Matching Patterns (No Replacement):
  - Words split with spaces or punctuation:
    - A L I, A-L-I, A/L/I → Not replaced
  - Words with internal special characters:
    - Moh@ammed → No match, no replacement
5. Reference Table Dependency:
 

This logic relies entirely on the accuracy and completeness of the anglicized reference mapping. Any missing or incorrect entries will result in missed or incorrect substitutions.

### BR-7.2.6: Initials Transformation Logic

This logic replaces full names with their abbreviated initials using a predefined initials reference mapping, helping to generate alternate name patterns that may appear in sanctions or identity databases.

#### Example

Original: Mohammed Ali Syed

Reference Mapping:

- Mohammed → Md.
- Ali → A
- Syed → S

Generated Variants:

1. Md. Ali Syed
  2. Mohammed A Syed
  3. Mohammed Aly S
- Final Output: 3 transformed records

#### Key Rules

1. Multiple Occurrence Handling:  
If a word appears more than once, replace all its occurrences in a single transformed variant.
  - E.g., 'Ali Hasan Ali → 'A. Hasan A.
2. Multi-Match Variants:  
If multiple reference words exist in a name, generate a record per unique replacement.
3. Exact Word Match Only:
  - Partial matches are not replaced.
  - Ali → A
  - Aliya, 123Ali → no replacement
4. Internal Special Characters Block Match:
  - E.g., Moh@ammed → no match → no replacement
5. Split Characters Disqualify Match:

- E.g., A L I, A-L-I, A/L/I → no match

#### 6. Reference-Driven Logic:

- Transformation is entirely dependent on the accuracy of the initial reference list.

### BR-7.2.7: Run-Together Transformation Logic

This transformation removes all spaces from individual entity names to create a run-together version, improving coverage for names that may appear without spacing in various systems or documents.

- Applies only to individual entities.
- Does not modify special characters or punctuation — only spaces are removed.

#### Key Rules

##### 1. All Spaces Removed:

- Every space in the name is removed, regardless of position or count.
- Example: A L I → ALI

##### 2. Special Characters Not Removed:

- Only spaces are stripped.
- Punctuation and special characters are preserved as-is.
  - E.g.:
    - John-Jani → John-Jani (unchanged)
    - Abc, Co., Ltd → Abc,Co.,Ltd (spaces removed, punctuation preserved)

##### 3. Handles Multiple Spaces:

- Extra spaces between words are also removed completely.
- Example: Bal chander k. → Balchanderk.

### BR-7.2.8: Bad Data Transformation Logic

This transformation introduces random special characters at random positions within the name to simulate or handle corrupted or noisy data scenarios.

#### Transformation Steps

1. **Calculate Character Count:**  
Count all characters in the name excluding spaces.
2. **Generate Random Position:**  
Generate a random integer between 1 and the character count. This will be the insertion point for the special character.
3. **Select Special Character:**  
Pick a random special character from a reference table (e.g., @, #, \$, !).
4. **Insert Special Character:**  
Insert the chosen special character at the randomly generated position within the original name.

### Example

#### ID Original Name Random Number Special Char Bad Data Name Flag

1	Mohammed Ali	3	@	Mo@hammed Ali	no
2	John Smitha	4	!	Joh!n Smitha	no

### Notes

- The transformation relies entirely on the completeness and accuracy of the special characters reference table.
- If the reference table is incomplete or inaccurate, inserted bad data may be missed or incorrectly applied.
- Only one special character insertion occurs per transformation.
- Spaces in the original name are not counted for the insertion position but are preserved in the output.

### BR-7.2.9: Missing Words Transformation Logic

This transformation simulates the random omission of one word in multi-word names ( $\geq 3$  words), imitating scenarios such as data entry mistakes, or partial data loss.

#### Transformation Rules:

1. **Eligible Names:**  
Names with more than 2 spaces (i.e.,  **$\geq 3$  words**) are eligible.
2. **Determine Word to Drop:**
  - Count the number of spaces in the entity name.
  - Generate a random word position between 1 and space\_count

### 3. Name Transformation Logic:

Based on random\_position, and space\_count remove the selected word:

- **random\_position <> space\_count and random\_position <> 1:**  
Drop the word appearing at random\_position and retain the rest.
- **random\_position = space\_count:**  
Drop the word appearing at random\_position and retain the rest.
- **random\_position = 1 :**  
Drop the second word and retain the rest.

#### Notes:

1. Only one word is removed in each transformation.
2. Randomness ensures variation across records.
3. If the word to be removed is the first, middle, or last, logic branches accordingly.
4. No action is performed on names with fewer than 3 words.
5. Transformation ensures no double spaces or leading/trailing whitespace in output.

### BR-7.2.10: Intervening Words Transformation Logic

This transformation introduces random words at random positions within the sanctioned entity names to evade detection

#### Transformation Steps

1. Calculate Space Count:  
Count all spaces in the entity name.
2. Generate Random Position:  
Generate a random integer between 1 and the space count. This will be the insertion point for the random word.
3. Select Random Word:  
Pick a random word from a reference table.
4. Insert Random Word:  
Insert the chosen word at the randomly generated position within the original name.

#### Example:

Original name	space count	random position	Transformed name
---------------	-------------	-----------------	------------------

DELVEST HOLDING, S.A.	2	2	DELVEST HOLDING, UNKNOWN S.A.
EDICIONES CUBANAS	1	1	EDICIONES REVOCABLE CUBANAS

### Notes

- The transformation relies entirely on the completeness and accuracy of the words reference table.
- If the reference table is incomplete or inaccurate, inserted data may be missed or incorrectly applied.
- Only one random word insertion occurs per transformation.

### BR-7.2.11: Symbolic Replacement Transformation Logic

This scenario introduces symbolic transformations by replacing characters in entity names using predefined substitution patterns. The process includes multiple symbolic conversion rules depending on the context (e.g., numerical symbols, hacker-style obfuscation).

#### Symbolic Substitution – Type 1: Numeric to Alphabet Conversion

##### Steps:

1. For each name, replace two digits/letters with corresponding letters.
2. Replacement Mapping:
  - 0 → O, 1 → I, 2 → Z, 4 → A, 5 → S, 7 → T, 8 → B

##### Example:

Name	Transformed Name
ALI5 Corporation	ALIS Corporation
BOND007	BOND00O

#### Symbolic Substitution – Type 2: Alphabet to Numeric Substitution



**Steps:**

1. For each name, replace only two letters with corresponding digits randomly.
2. Replacement Mapping:
  - $Z \rightarrow 2, S \rightarrow 5, O \rightarrow 0, I \rightarrow 1, T \rightarrow 2, A \rightarrow 4, G \rightarrow 6, E \rightarrow 3, D \rightarrow 6, B \rightarrow 8$

**Example:**

Name	Transformed Name
BONDAG Ltd.	B06DAG Ltd.
SMITH, John	5M1TH, John

**Symbolic Substitution – Type 3: Hacker Style (Obfuscated Symbols)****Steps:**

1. Replace only two characters with hacker-style symbols.
2. Replacement Mapping:
  - $Z \rightarrow >_, X \rightarrow ><, V \rightarrow \vee, S \rightarrow \$, I \rightarrow !, H \rightarrow \#, A \rightarrow @, G \rightarrow \&, C \rightarrow ($

**Example:**

Name	Transformed Name
HACKER Zone	#@CKER Zone
SAMSON Corp	\$AM\$ON Corp
VICTOR Ltd.	V!(TOR Ltd.

**BR-7.2.12: Phonetic Substitution Transformation Logic**

This transformation simulates common phonetic spelling variations by replacing specific letters or letter patterns that sound similar. The objective is to generate name variants that account for pronunciation-based discrepancies.

**Steps:**

1. Target all records in the master dataset.
2. Apply phonetic replacement rules on the name column.
3. For each rule:
  - Perform a single substitution across the name.

- Create a new column showing the transformed name.
4. Use UNION to stack results from different substitution rules into one unified dataset.

**Example:**

<b>original name</b>	<b>transformed name</b>
'ABBAS, Yasir	'ABBAS, lasir
'ABBAS, Yasir	'ABBAS, Yasyr
'ABBAS, Yasir	'ABBAS, Yazir
'ABBAS, Yasir	'ABBAS, YAsir

**Note:**

- This logic does not loop or apply multiple substitutions per name in one step.
- Each substitution generates a new row via UNION.
- This is case-sensitive, and the logic explicitly handles both uppercase and lowercase variants.
- Useful in generating mutations to catch phonetically similar matches during screening.

### BR-7.2.13: Name Alias Transformation Logic

This transformation replaces the full name with a known alias based on an external reference table. This is useful to detect names that may appear under a different but known alias (e.g., "William" → "Bill"). Applies to names from individuals or organizations in the dataset.

**Steps:**

1. Match records from master data with alias reference using exact name match.
2. Replace the matched name with its corresponding alias.

**Example:**

<b>original name</b>	<b>Alias</b>
RODRIGUEZ OREJUELA, Gilberto Jose	LUCAS
RODRIGUEZ OREJUELA, Gilberto Jose	THE CHESS PLAYER

### BR-7.2.14: Country-Based Address Data Matching Logic

The system must develop a solution that can match address data with country information by performing a join operation based on a country field. This will use three key files:

- **Country Reference File:** A reference list containing valid country names.
- **Address Data File:** OFAC SDN address file that contains four address attributes, out of which any one attribute includes a country name that will be matched with the country name in the reference file.
- **Entity Name reference file:** Once address is derived from matching logic, entity names of type NC and PC-LFM must be used while generating a Bank preferred file.

### BR-7.3: Duplicate Record Handling for UK Sanctions

The system must identify and handle duplicate records while synthesizing the UK sanctions data based on a defined deduplication rule. The logic must operate as follows:

- If a record has the same UNIQUE\_ID, NAME, and TYPE as another record in the uploaded file, the system must treat these as duplicates and retain only one instance of the record.
- If multiple records have the same NAME, and TYPE, but different UNIQUE\_ID, the system must treat each record as unique and include all in the import.

### BR-7.4: Scenario-Based Data Synthesis and Preview for UK Sanctions List files

The system must allow users to select specific data processing scenarios (i.e., *Phonetic Substitution, Abbreviation, etc*) for each selected file (current or historical). Based on the selected file (current or historical) and then selected scenario, the system must dynamically process and generate synthesized data and present a preview within the UI.

- For each file listed (Current or History), the user must be able to:
  - Choose single scenario or multiple scenarios from a predefined dropdown or selection list,
  - View the resulting synthesized dataset in a preview table.
- Each scenario must trigger its corresponding logic:

- **Phonetic Substitution** → letters or sounds are replaced by others to mimic or modify pronunciation in names
- **Abbreviations** → Abbreviations are introduced into entity names
- **Bad Data** → Bad Data (characters, numbers, combinations) is inserted at the random position.

#### Refer BR-7.2.1 to BR-7.2.13

#### Preview Features:

- Once processed, the UI must display following attributes in tabular format pertaining to the selected file for preview
  - Unique\_id
  - Original entity name
  - Synthesized name
  - Scenario name

#### Note:

- Current File Selection □ User must be able to select only the latest imported file.
- Historical File Selection □ User must be able to select only the top three historical files.

### BR-7.5: Duplicate Record Handling for OFAC NON SDN List File

The system must identify and handle duplicate records while synthesizing the OFAC NON SDN sanctions data based on a defined deduplication rule. The logic must operate as follows:

- If a record has the same ENT\_NUM, SDN\_NAME, and SDN\_TYPE as another record in the uploaded file, the system must treat these as duplicates and retain only one instance of the record.
- If multiple records have the same SDN\_NAME and SDN\_TYPE, but different ENT\_NUM, the system must treat each record as unique and include all in the import.

### BR-7.6: Scenario-Based Data Synthesis and Preview for OFAC NON SDN List files

The system must allow users to select specific data processing scenarios (i.e., *Phonetic Substitution*, *Abbreviation*, etc) for each selected file (current or historical).

Based on the selected file (current or historical) and then selected scenario, the system must dynamically process and generate synthesized data and present a preview within the UI.

- For each file listed (Current or History), the user must be able to:
  - Choose single scenario or multiple scenarios from a predefined dropdown or selection list,
  - View the resulting synthesized dataset in a preview table.
- Each scenario must trigger its corresponding logic:
  - **Phonetic Substitution** → letters or sounds are replaced by others to mimic or modify pronunciation in names
  - **Abbreviations** → Abbreviations are introduced into entity names
  - **Bad Data** → introduces random special characters at random positions within the name

**Refer BR-7.2.1 to BR-7.2.13**

**Preview Features:**

- Once processed, the UI must display following attributes in tabular format pertaining to the selected file for preview
  - Unique\_id
  - Original entity name
  - Synthesized name
  - Scenario name

**Note:**

- Current File Selection □ User must be able to select only the latest imported file.
- Historical File Selection □ User must be able to select only the top three historical files.

## BR-7.7: Scenario-Based Data Synthesis and Preview for PEP files

The system must allow users to select specific data processing scenarios (i.e., *Phonetic Substitution*, *Abbreviation*, etc) for each selected file (current or historical). Based on the selected file (current or historical) and then selected scenario, the system must dynamically process and generate synthesized data and present a preview within the UI.

- For each file listed (Current or History), the user must be able to:

- Choose single scenario or multiple scenarios from a predefined dropdown or selection list,
- View the resulting synthesized dataset in a preview table.
- Each scenario must trigger its corresponding logic:
  - **Phonetic Substitution** → letters or sounds are replaced by others to mimic or modify pronunciation in names
  - **Abbreviations** → Abbreviations are introduced into entity names
  - **Bad Data** → Bad Data (characters, numbers, combinations) is inserted at the end or at the random position.

**Refer BR-7.2.1, BR-7.2.2, BR-7.2.3, BR-7.2.5, BR-7.2.6, BR-7.2.7, BR-7.2.8, BR-7.2.9, BR-7.2.10, BR-7.2.11, BR-7.2.12, BR-7.2.13**

**Preview Features:**

- Once processed, the UI must display following attributes in tabular format pertaining to the selected file for preview
  - Unique\_id
  - Original entity name
  - Synthesized name
  - Scenario name

**Note:**

- Current File Selection □ User must be able to select only the latest imported file.
- Historical File Selection □ User must be able to select only the top three historical files.

## BR-7.8: Scenario-Based Data Synthesis and Preview for Watch List files

The system must allow users to select specific data processing scenarios (i.e., *Phonetic Substitution, Abbreviation, etc*) for each selected file (current or historical). Based on the selected file (current or historical) and then selected scenario, the system must dynamically process and generate synthesized data and present a preview within the UI.

- For each file listed (Current or History), the user must be able to:
  - Choose single scenario or multiple scenarios from a predefined dropdown or selection list,

- View the resulting synthesized dataset in a preview table.
- Each scenario must trigger its corresponding logic:
  - **Phonetic Substitution** → letters or sounds are replaced by others to mimic or modify pronunciation in names
  - **Abbreviations** → Abbreviations are introduced into entity names
  - **Bad Data** → Bad Data (characters, numbers, combinations) is inserted at the end or at the random position.

#### **Refer BR-7.2.1 to BR-7.2.13**

#### **Preview Features:**

- Once processed, the UI must display following attributes in tabular format pertaining to the selected file for preview
  - Unique\_id
  - Original entity name
  - Synthesized name
  - Scenario name

#### **Note:**

- Current File Selection □ User must be able to select only the latest imported file.
- Historical File Selection □ User must be able to select only the top three historical files.

### **BR-8.1: Synthesized Data Selection for Bank File Generation**

The system must allow users to select synthesized datasets based on scenarios that were selected at the time of processing and specify the number of records to include before generating a bank file.

Users must be able to specify a record count (e.g., 100, 500).

### **BR-8.2: Preview of Synthesized Data in BANK\_A Bank File Format**

The system must allow users to preview the synthesized data in the exact structure and format it will appear in the BANK\_A file before initiating the export. After selecting the processing scenario and desired record count, the system must display a bank file preview in a tabular layout.

#### **Preview Features:**

- Reflect the column headers, field order, and formatting as defined by the target bank file schema.
- Show up to a configurable number of records (e.g., first 100 or user-specified).
- Provide a "Download " option in the same format (.txt).

**Note:** Before the user selects processing scenario and desired record count in Pop Up Window, file name that is currently processed must be defaulted.

### BR-8.3: File Structure and Export Format for BANK\_A Bank

The system must export the fetched records into a bank-ready file in the required format (.txt).

File Generation Rules:

- File name must follow a consistent naming convention (YYYYMMDD\_ Nonghyup-Sanctions-Testing-ScenarioName-records.txt).

Example □ 20240807-Nonghyup-Sanctions-Testing-DL-records.txt

- Columns must follow the defined schema required by the bank.

### BR-8.4: File Size Limitation for BANK\_A Bank File Generation

The system must enforce a limit of 5,000 entities per file when generating a bank file from synthesized data.

**Functional Rules:**

- A single bank file shall not contain more than 5,000 entities data.
- If the selected record count exceeds 5,000:
  - The system must:
    - Prevent the generation and alert the user:  
*“Maximum limit exceeded. Please select up to 5,000 records.”*  
 and if user requires desired count of records, then
    - Automatically split the output into multiple files (e.g., File\_1.txt, File\_2.txt)
- Example File Names
  - 20240807-Nonghyup-Sanctions-Testing-DL1-records.txt
  - 20240807-Nonghyup-Sanctions-Testing-DL2-records.txt



### BR-8.5: Preview of Synthesized Data in BANK\_B Bank File Format


The system must allow users to preview the synthesized data in the exact structure and format it will appear in the BANK\_B file before initiating the export. After selecting the processing scenario and desired record count, the system must display a bank file preview in a tabular layout.


Preview Features:

- Reflect the column headers, field order, and formatting as defined by the target bank file schema.
- Show up to a configurable number of records (e.g., first 100 or user-specified).
- Provide a "Download " option in the same format (.txt).

### BR-8.6: File Structure and Export Format for BANK\_B Bank

The system must export the fetched records into a bank-ready file in the required format (.txt).

File Naming Convention  VENDOR.TXT

File Format  Tab delimited text file

### BR-8.7: Synthesized File Summary View with Scenario and Record Count

The system must present a tabular summary of all synthesized files, displaying the file name, processing scenario applied, and the number of synthesized records.

Column Name	Description
File Name	Name of the processed file including timestamp (e.g., ofac-2024-03-15-120000_processed)
Scenario	The specific data processing scenario applied to the file (e.g., Bad data, Aliases)
Count	Number of records produced after synthesis

**Business Rules:**

- Count must reflect final synthesized records, not raw input size.
- File names must follow the conventions while generating accordingly.

## BR-9.1: Import of Alert Files received from external Screening System for BANK\_A

The system must allow users to import alert files generated from external screening systems into the platform for reconciliation with previously generated bank files.

### Key Factors:

- Number of alert files: Two
- Files format: .xlsx
- File Naming Convention:  
NY\_safewatch\_scan\_result\_tdetetection.xlsx  
NY\_safewatch\_scan\_result\_tviolation.xlsx
- Invalid files must be rejected with a clear error message.

## BR-9.2: Reconciliation of Alert Files with Bank Files

After importing alert files, the system must be able to compare the imported alert files with the relevant bank file (previously generated from synthesized data).

- The system must highlight detect:
  - Matched records: IDs/names present in both alert and bank files (expected to be alerted).
  - Unmatched records in bank file (expected but not alerted).

## BR-9.3: Efficiency Report Preview After Alert Comparison

After completion of alert-to-bank file comparison, the system must generate and display an efficiency report summarizing the performance of each scenario in terms of match effectiveness.

- **The system must calculate and display:**
  - Total Records processed for each scenario,
  - Hit Count: number of records that matched with alerts,
  - Hit Percentage =  $(\text{Hit Count} / \text{Total Records}) \times 100$ ,
  - No hit Count = number of records that are not matching with alerts,
  - No hit Percentage =  $(\text{No hit Count} / \text{Total Records}) \times 100$ .

- **All percentages must be rounded to 2 decimal places.**

#### BR-9.4: Drill-down View of Hit Records from Efficiency Report

The system must allow users to click on the "Hit Count" value in the efficiency report to view the list of matched (hit) records for the corresponding scenario.

##### **Key data to be displayed in UI:**

Name: The original name as it appears in the input data.

Processed Name: The version of the name after scenario-specific transformation (e.g., removing spaces, abbreviating).

Display a label like Hits (counts) above the table to indicate total matched records.

#### BR-9.5: Display of Internal Violation References on Processed Name Click

When a user clicks on a processed name in the hit records table, the system must display a list of internal violation references i.e., list maintained by the bank that were matched against the processed name in the screening system. This supports transparency into why a name was flagged as a match, by showing the exact reference patterns it triggered in the internal screening logic.

##### **Key data to be displayed in UI:**

- Display the original name,
- Display the processed name,
- and display a list of all matched reference values.

#### BR-9.6: Drill-down View of No Hit Records from Efficiency Report

The system must allow users to click on the "No Hit Count" value in the efficiency report to view the list of records that were not alerted for the corresponding scenario.

##### **Key data to be displayed in UI:**

Name: The original name as it appears in the input data.

Processed Name: The version of the name after scenario-specific transformation (e.g., removing spaces, abbreviating).

Display a label like No Hits (counts) above the table to indicate total unmatched records.

## BR-9.7: Consolidated Export of Efficiency Report in PDF Format

The system must provide a feature to export the full efficiency report, including summary data and detailed breakdowns, into a single consolidated PDF file with clearly separated sections.

### **PDF Structure and Content:**

1. Sheet/Section 1: Efficiency Summary
2. Sheet/Section 2: Hit Records
3. Sheet/Section 3: Violation Details
4. Sheet/Section 4: Unhit Record

## BR-9.8: Preview of Efficiency reports of historical bank files

The system must allow users to preview efficiency reports for previously generated and processed historical bank files to assess screening effectiveness over time.

### **Contents to be displayed:**

Efficiency Report

Date Generated

Preview button

Download button

## BR-10.1: Import of Alert Files from Screening System for BANK\_B

The system must allow users to import alert files generated from screening systems into the platform for reconciliation with previously generated bank files.

### **Key Factors:**

- Number of alert files: more than one
- File formats: .csv
- File Naming Convention:  
NOLAUS33\_RID 32211\_07162024\_1of2.csv  
NOLAUS33\_RID 32211\_07162024\_2of2.csv
- Invalid files must be rejected with a clear error message.

## BR-10.2: Reconciliation of Alert Files with Bank Files

After importing alert files, the system must be able to compare the imported alert files with the relevant bank file (previously generated from synthesized data).

- The system must highlight detect:
  - Matched records: IDs/names present in both alert and bank files (expected to be alerted).
  - Unmatched records in bank file (expected but not alerted).

## BR-10.3: Efficiency Report Preview After Alert Comparison

After completion of alert-to-bank file comparison, the system must generate and display an efficiency report summarizing the performance of each scenario in terms of match effectiveness.

- **The system must calculate and display:**
  - Total Records processed for each scenario,
  - Hit Count: number of records that matched with alerts,
  - Hit Percentage =  $(\text{Hit Count} / \text{Total Records}) \times 100$ ,
  - Unhit Count = number of records that are not matching with alerts,
  - Unhit Percentage =  $(\text{Unhit Count} / \text{Total Records}) \times 100$ .
- **All percentages must be rounded to 2 decimal places.**

## BR-10.4: Drill-down View of Hit Records from Efficiency Report

The system must allow users to click on the "Hit Count" value in the efficiency report to view the list of matched (hit) records for the corresponding scenario.

### **Key data to be displayed in UI:**

Name: The original name as it appears in the input data.

Processed Name: The version of the name after scenario-specific transformation (e.g., removing spaces, abbreviating).

Display a label like Hits (counts) above the table to indicate total matched records.

## BR-10.5: Display of Internal Violation References on Processed Name Click

When a user clicks on a processed name in the hit records table, the system must display a list of internal violation references i.e., list maintained by the bank that were matched against the processed name in the screening system. This supports transparency into why a name was flagged as a match; by showing the exact reference patterns it triggered in the internal screening logic.

### Key data to be displayed in UI:

- Display the case\_id from alert file,
- Display the original name,
- Display the processed name,
- Display matched\_text and matched\_party\_name from alert file.

## BR-10.6: Drill-down View of No Hit Records from Efficiency Report

The system must allow users to click on the "No Hit Count" value in the efficiency report to view the list of records that were not alerted for the corresponding scenario.

### Key data to be displayed in UI:

Name: The original name as it appears in the input data.

Processed Name: The version of the name after scenario-specific transformation (e.g., removing spaces, abbreviating).

Display a label like No Hits (counts) above the table to indicate total unmatched records.

## BR-10.7: Consolidated Export of Efficiency Report in PDF Format

The system must provide a feature to export the full efficiency report, including summary data and detailed breakdowns, into a single consolidated PDF file with clearly separated sections.

### PDF Structure and Content:

Sheet/Section 1: Efficiency Summary

Sheet/Section 2: Hit Records

Sheet/Section 3: Violation Details

Sheet/Section 4: No hit Record

## BR-10.8: Preview of Efficiency reports of historical bank files

The system must allow users to preview efficiency reports for previously generated and processed historical bank files to assess screening effectiveness over time.

### Contents to be displayed:

Efficiency Report

Date Generated

Preview button

Download button

## BR-11: Display and Management of Processed File Metadata

The system must display a list of all processed files along with their associated metadata—**file name**, **load date**, **end date**, and available **actions**—in a centralized UI table for review and further operations as a separate module.

## BR-12.1: Import of Abbreviations Reference Data

The system must allow users to import a reference file containing abbreviations data used to detect abbreviated forms of sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Abbreviations are commonly introduced into sanctioned entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

### Functional Requirements:

- **File Naming Convention**
  - Abbreviations\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Abbreviations
  - Replace Word
- **File Validation Rules:**

- Mandatory fields: Abbreviation and Replace Word
- If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

#### **Audit & Access Control:**

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify abbreviation reference data.

### **BR-12.2: Preview of Abbreviations Reference Data**

The system shall allow users to preview the Abbreviations Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each abbreviation entry.
Abbreviation	The shortened form or acronym that will be detected for substitution during mutation generation.
Replace Word	The shortened form or acronym that will be substituted during mutation generation.
Created By	Username or system role that created the abbreviation entry
Actions	Available UI operations for the entry, such as Edit and Delete

### **BR-12.3: Abbreviation reference data – Version Control & History Tracking**

The system shall maintain a version history of all changes (edit and delete) made to Abbreviations Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made to abbreviation mappings through the UI.



## Functional Requirements:

### 1. Deletion Handling:

- When a user deletes an abbreviation record from the preview table:
  - The record is **soft-deleted** from the active dataset.
  - The deleted record is **archived** in a **History Table**, with metadata:
    - ID
    - Abbreviation
    - Replace Word
    - Created by
    - Action By
    - Uploaded Date
    - Action At (date)
    - Action Type = Delete

### 2. Edit Handling:

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Abbreviation
    - Replace Word
    - Created by
    - Action By
    - Uploaded date
    - Action At (timestamp)
    - Action Type = Edit

## BR-13.1: Import of Name Aliases Reference Data

The system must allow users to import a reference file containing Name aliases data used to detect aliases of sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Name aliases are commonly replaced with corresponding sanctioned entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

### Functional Requirements:

- **File Naming Convention**
  - NameAliases\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Name
  - Alias Name
- **File Validation Rules:**
  - Mandatory fields: Name and Alias Name
  - If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

### Audit & Access Control:

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify aliases reference data.

## BR-13.2: Preview of Name Aliases Reference Data

The system must allow users to preview the Name Aliases Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Name	Name that is being detected for substitution during mutation generation.
Alias Name	Replace name that will be substituted during mutation generation.
Created By	Username or system role that created the abbreviation entry
Actions	Available UI operations for the entry, such as Edit and Delete

### BR-13.3: Name Aliases reference data – Version Control & History Tracking

The system must maintain a version history of all changes (edit and delete) made to Name Aliases Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made to aliases mappings through the UI.

#### Functional Requirements:

##### 1. Deletion Handling:

- When a user deletes a name alias record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:
    - ID
    - Name
    - Alias Name
    - Created by
    - Action By
    - Uploaded Date
    - Action At (date)
    - Action Type = Delete

##### 2. Edit Handling:

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Name
    - Alias Name
    - Created by
    - Action By
    - Uploaded date
    - Action At (timestamp)
    - Action Type = Edit

### BR-14.1: Import of Anglicized Words Reference Data

The system must allow users to import a reference file containing Anglicized data used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Anglicized words are commonly replaced with corresponding words in sanctioned entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

#### Functional Requirements:

- **File Naming Convention**
  - AnglicizedWords\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Name
  - Anglicized Name
- **File Validation Rules:**
  - Mandatory fields: Name and Anglicized Name

- If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

#### **Audit & Access Control:**

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify aliases reference data.

### **BR-14.2: Preview of Anglicized Words Reference Data**

The system must allow users to preview the Anglicized Words Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Name	Name that is being detected for substitution during mutation generation.
Anglicized Name	Replace name that will be substituted during mutation generation.
Created By	Username or system role that created the abbreviation entry
Actions	Available UI operations for the entry, such as Edit and Delete

### **BR-14.3: Anglicized Words reference data – Version Control & History Tracking**

The system must maintain a version history of all changes (edit and delete) made to Anglicized Words Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made through the UI.

#### **Functional Requirements:**

##### **1. Deletion Handling:**

- When a user deletes an anglicized words record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:
    - ID
    - Name
    - Anglicized Name
    - Created by
    - Action By
    - Uploaded Date
    - Action At (date)
    - Action Type = Delete

## **2. Edit Handling:**

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Name
    - Anglicized Name
    - Created by
    - Action By
    - Uploaded date
    - Action At (timestamp)
    - Action Type = Edit

## **BR-15.1: Import of Bad Data Reference**

The system must allow users to import a reference file containing bad data used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Bad data (numbers, special characters, alphabets and

combinations) is placed at random positions in sanctioned entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

#### **Functional Requirements:**

- **File Naming Convention**
  - **BadData\_DDMMYY**
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Bad\_Data
- **File Validation Rules:**
  - Mandatory field: Bad\_Data
  - If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

#### **Audit & Access Control:**

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify bad data reference.

### **BR-15.2: Preview of Bad Data Reference**

The system must allow users to preview the Bad Data Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Bad Data	Data that must be substituted during mutation generation at random position

Created By	Username or system role that created the abbreviation entry
Actions	Available UI operations for the entry, such as Edit and Delete

## BR-15.3: Bad Data reference – Version Control & History Tracking

The system must maintain a version history of all changes (edit and delete) made to Bad Data Reference via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made through the UI.

### Functional Requirements:

#### 1. Deletion Handling:

- When a user deletes bad data record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:
    - ID
    - Bad data
    - Created by
    - Action By
    - Uploaded Date
    - Action At (date)
    - Action Type = Delete

#### 2. Edit Handling:

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Bad Data
    - Created by
    - Action By



- Uploaded date
- Action At (timestamp)
- Action Type = Edit

## BR-16.1: Import of Double Letters Reference Data

The system must allow users to import a reference file containing **Double Letters** data used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Double Letters are replaced with respective Single Letter and vice versa in sanctioned entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

### Functional Requirements:

- **File Naming Convention**
  - DoubleLetters\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Letter Code
  - Replace Letter Code
- **File Validation Rules:**
  - Mandatory fields: Letter Code and Replace Letter Code
  - If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

### Audit & Access Control:

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify aliases reference data.

## BR-16.2: Preview of Double Letters Reference Data

The system must allow users to preview the Anglicized Words Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Letter Code	Single Letter or Double Letter that is being detected in entity name for substitution during mutation generation.
Replace Letter Code	Single Letter or Double Letter that is being detected in entity name that will be substituted during mutation generation.
Created By	Username or system role that created the entry
Actions	Available UI operations for the entry, such as Edit and Delete

## BR-16.3: Double Letters reference data – Version Control & History Tracking

The system must maintain a version history of all changes (edit and delete) made to Double Letters Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made through the UI.

### Functional Requirements:

#### 1. Deletion Handling:

- When a user deletes a record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:
    - ID
    - Letter Code
    - Replace Letter Code
    - Created by
    - Action By
    - Uploaded Date

- Action At (date)
- Action Type = Delete

## 2. Edit Handling:

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Letter Code
    - Replace Letter Code
    - Created by
    - Action By
    - Uploaded date
    - Action At (timestamp)
    - Action Type = Edit

## BR-17.1: Import of Initials Reference Data

The system must allow users to import a reference file containing **Initials** data used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. **Initials** are replaced with respective part of entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

### Functional Requirements:

- **File Naming Convention**
  - Initials\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Name

- Initials
- **File Validation Rules:**
  - Mandatory fields: Name and Replace Initials
  - If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

#### **Audit & Access Control:**

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify initials reference data.

### **BR-17.2: Preview of Initials Reference Data**

The system must allow users to preview the Initials Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Name	Name that is being detected in entity name for substitution during mutation generation.
Initials	Initial that will be substituted during mutation generation in entity name
Created By	Username or system role that created the entry
Actions	Available UI operations for the entry, such as Edit and Delete

### **BR-17.3: Initials reference data – Version Control & History Tracking**

The system must maintain a version history of all changes (edit and delete) made to Initials Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made through the UI.

## **Functional Requirements:**

### **1. Deletion Handling:**

- When a user deletes a record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:
    - ID
    - Name
    - Initials
    - Created by
    - Action By
    - Uploaded Date
    - Action At (date)
    - Action Type = Delete

### **2. Edit Handling:**

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Name
    - Initials
    - Created by
    - Action By
    - Uploaded date
    - Action At (timestamp)
    - Action Type = Edit

## BR-18.1: Import of Nickname Reference Data

The system must allow users to import a reference file containing **Nickname** data used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Nicknames are replaced with respective part of entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

### Functional Requirements:

- **File Naming Convention**
  - NickName\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Name
  - Nickname
- **File Validation Rules:**
  - Mandatory fields: Name and Nickname
  - If any required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

### Audit & Access Control:

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify nicknames reference data.

## BR-18.2: Preview of Nickname Reference Data

The system must allow users to preview the Nickname Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Name	Name that is being detected in entity name for substitution during mutation generation.
Nickname	Nickname that will be substituted during mutation generation in entity name
Created By	Username or system role that created the entry
Actions	Available UI operations for the entry, such as Edit and Delete

## BR-18.3: Nickname reference data – Version Control & History Tracking

The system must maintain a version history of all changes (edit and delete) made to Nickname Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made through the UI.

### Functional Requirements:

#### 1. Deletion Handling:

- When a user deletes a record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:
    - ID
    - Name
    - Nickname
    - Created by
    - Action By
    - Uploaded Date
    - Action At (date)
    - Action Type = Delete

#### 2. Edit Handling:

- When a user edits a record:

- The new (edited) version is saved and shown in the active preview table.
- The previous (original) version is moved to the History Table, with metadata:
  - ID
  - Name
  - Nickname
  - Created by
  - Action By
  - Uploaded date
  - Action At (timestamp)
  - Action Type = Edit

## BR-19.1: Import of Intervening Words Reference Data

The system must allow users to import a reference file containing Intervening Words data used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. Intervening Words are inserted into entity names to evade detection. This reference data enables the system to identify such patterns during name screening and synthesized data generation.

### Functional Requirements:

- **File Naming Convention**
  - InterveningWords\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Intervening Words
- **File Validation Rules:**
  - Mandatory fields: Intervening Words
  - If required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.



### **Audit & Access Control:**

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify Intervening Words reference data.

### **BR-19.2: Preview of Intervening Words Reference Data**

The system must allow users to preview the Intervening Words Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Intervening Words	Random words that will be inserted during mutation generation in entity name at random position
Created By	Username or system role that created the entry
Actions	Available UI operations for the entry, such as Edit and Delete

### **BR-19.3: Intervening Words reference data – Version Control & History Tracking**

The system must maintain a version history of all changes (edit and delete) made to Intervening Words Reference Data via the user interface to ensure traceability, auditability, and data integrity by preserving a record of all historical changes made through the UI.

#### **Functional Requirements:**

##### **1. Deletion Handling:**

- When a user deletes a record from the preview table:
  - The record is soft-deleted from the active dataset.
  - The deleted record is archived in a History Table, with metadata:

- ID
- Intervening Words
- Created by
- Action By
- Uploaded Date
- Action At (date)
- Action Type = Delete

## **2. Edit Handling:**

- When a user edits a record:
  - The new (edited) version is saved and shown in the active preview table.
  - The previous (original) version is moved to the History Table, with metadata:
    - ID
    - Intervening Words
    - Created by
    - Action By
    - Uploaded date
    - Action At (timestamp)
    - Action Type = Edit

## **BR-20.1: Import of MT202 Reference Data**

The system must allow users to import a reference file containing entity names (both sanctioned and unsanctioned entities) used to detect sanctioned entity names. The imported data must be stored and previewed via the UI for review and validation. This is to ensure that all MT202 messages are screened through the sanction's engine so that any references to sanctioned individuals, entities are flagged appropriately.

### **Functional Requirements:**

- **File Naming Convention**
  - MT202\_DDMMYY
- **File Type**

- .csv (comma separated value)
- **File attributes**
  - Name
  - Type
- **File Validation Rules:**
  - Mandatory fields: Name and Type
  - If required fields are missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

#### **Audit & Access Control:**

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify MT202 reference data.

## **BR-20.2: Preview of MT202 Reference Data**

The system must allow users to preview the MT202 Reference Data via the user interface before applying it during processing or mutation generation.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.
Name	Names of sanctioned and unsanctioned entities
Type	Type of name (for sanctioned entities it is positive and for unsanctioned entities it is negative)
Created By	Username or system role that created the entry
Actions	Available UI operations for the entry, such as Edit and Delete

## BR-21.1: Import of Countries List Reference Data

The system must allow users to import a reference file containing country names used to perform match with SDN Address files. The imported data must be stored and previewed via the UI for review and validation.

### Functional Requirements:

- **File Naming Convention**
  - Countries\_DDMMYY
- **File Type**
  - .csv (comma separated value)
- **File attributes**
  - Country Name
- **File Validation Rules:**
  - Mandatory fields: Country Name
  - If the required field is missing or contains null/blank values, the import process must be terminated for those records. The system must then display a clear, user-friendly alert indicating the failure reason.

### Audit & Access Control:

- All imports must be logged with:
  - User ID
  - Timestamp
  - File name
  - Number of records imported
- Only authorized users can import or modify Countries List reference data.

## BR-21.2: Preview of Countries List Reference Data

The system must allow users to preview the countries list Reference Data via the user interface.

The dataset should be displayed in a tabular format, showing:

Field	Description
ID	Auto-generated unique identifier for each entry.

Country Name	Names of countries
Created By	Username or system role that created the entry
Actions	Available UI operations for the entry, such as Edit and Delete

### BR-22.1: Delisting Detection via Referential Integrity for OFAC SDN list

This business rule detects delisted entities by performing a referential integrity check between the historical OFAC sanctions list and the current OFAC sanctions list, using the unique ID. Any entity present in the historical list but absent in the current list is considered delisted.

The system must identify and extract delisted entries from OFAC sanctions data by comparing the current list with the historical list and must generate the results in the required bank-specific output format (e.g., BANK\_A , BANK\_B).

### BR-22.2: Delisting Detection via Referential Integrity for OFAC NON SDN list

This business rule detects delisted entities by performing a referential integrity check between the historical OFAC NON SDN sanctions list and the current OFAC NON SDN sanctions list, using the unique ID. Any entity present in the historical list but absent in the current list is considered delisted.

The system must identify and extract delisted entries from OFAC NON SDN sanctions data by comparing the current list with the historical list and must generate the results in the required bank-specific output format (e.g., BANK\_A , BANK\_B).

### BR-22.3: Delisting Detection via Referential Integrity for UK Sanctions list

This business rule detects delisted entities by performing a referential integrity check between the historical UK OFAC sanctions list and the current UK sanctions list, using the unique ID. Any entity present in the historical list but absent in the current list is considered delisted.

The system must identify and extract delisted entries from UK sanctions data by comparing the current list with the historical list and must generate the results in the required bank-specific output format (e.g., BANK\_A , BANK\_B).

## 9. Functional Requirements

ID	Requirement	Priority
----	-------------	----------

## 10. Non-Functional Requirements

- Performance
- Security
- Scalability
- Usability

## 11. Milestones

## 12. Technical Overview / Tools and Technologies

Category	Tool/Technology	Purpose
Programming Language		
Frameworks		
Database		
Version Control		
Testing Tools		
Cloud Platform		

## 13. Dependencies

## 13. Appendices