Anomaly Detection Configuration Guide
Oracle Banking Digital Experience
Patchset Release 22.2.6.0.0

Part No. F72987-01

April 2025

ORACLE

UK Open Banking Anomaly Detection Configuration Guide

April 2025

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway

Goregaon (East)

Mumbai, Maharashtra 400 063

India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax:+91 22 6718 3001

www.oracle.com/financialservices/

Table of Contents

1. Preface

1.1 Purpose

Welcome to the User Guide for Oracle Banking Digital Experience. This guide explains the operations that the user will follow while using the application.

1.2 Audience

This manual is intended for Customers and Partners who setup and use Oracle Banking Digital Experience.

1.3 Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit, http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

1.4 Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at Critical Patches, Security Alerts and Bulletins. All critical patches should be applied in a timely manner to ensure effective security, as strongly recommended by Oracle Software Security Assurance.

1.5 Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1.6 Conventions

The following text conventions are used in this document:

Convention	Meaning

boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
Italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1.7 Screenshot Disclaimer

The images of screens used in this user manual are for illustrative purpose only, to provide improved understanding of the functionality; actual screens that appear in the application may vary based on selected browser, theme, and mobile devices.

1.8 Acronyms and Abbreviations

The list of the acronyms and abbreviations that you are likely to find in the manual are as follows:

Abbreviation	Description
OBDX	Oracle Banking Digital Experience

2. Introduction

2.1 Purpose of the Document

This user manual provides step-by-step instructions for managing and configuring anomaly detection models for the following use cases:

- **Login Data**
- **Payment Data**

The system is designed to detect anomalies in login activities and payment transactions, ensuring security and fraud prevention. By leveraging machine learning techniques, it helps identify unusual patterns that may indicate unauthorized access attempts or fraudulent transactions.

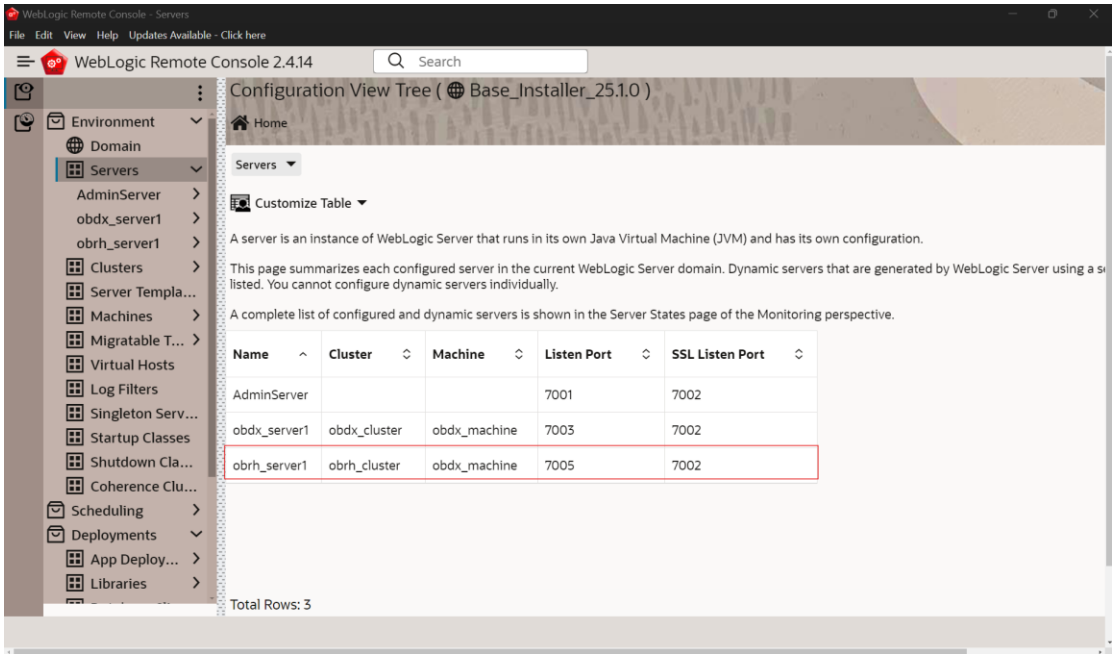
2.2 Key Features of the System

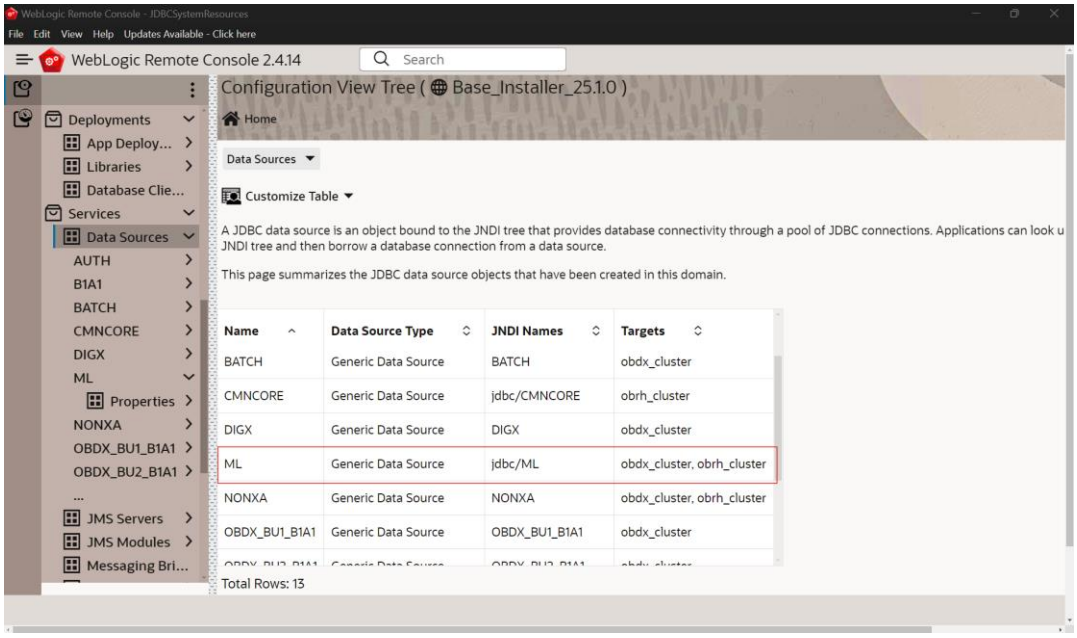
- **Automated Anomaly Detection**: The system automatically flags suspicious login attempts and payment activities.
- **Customizable Model Settings**: Users can define and adjust various model parameters, including sensitivity, error metrics, and data sources.
- **Real-time Monitoring**: The system enables continuous tracking and drift detection to ensure model effectiveness over time.
- **Debugging and Logging**: Provides detailed logs for troubleshooting.
- **User-Friendly Interface**: Simplifies model setup, evaluation, and maintenance through intuitive screens and action buttons.

3. Prerequisites

3.1 Configure Bus Service

Before defining models, configure the Bus Service by inserting the required App ID and Bus Service URL:

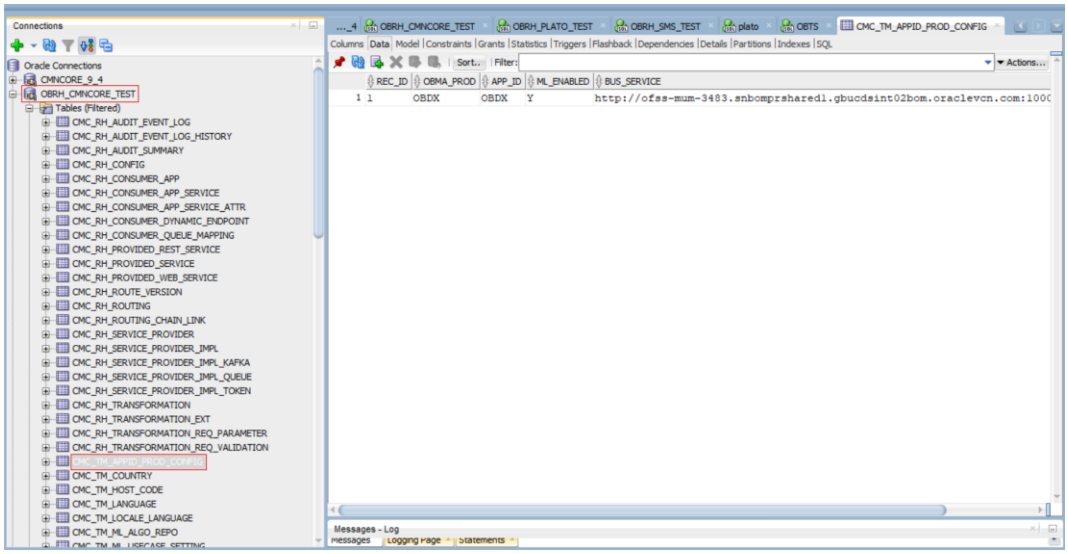




Steps:

1. Make sure **ML** Data source exists with the respective targets displayed above.
2. If ML doesn't exist, create ML schema in database and execute the following SQL queries.

   OBDX_Installer/installables/OBDX/BASE/25.1.0.0.0/obdx_obrh/db/ml/grants.sql
3. Insert your OBMA_PROD, APP_ID & BUS_SERVICE (Add respective to your WebLogic)



Steps:

1. Connect to your database.
2. Navigate to Commoncore (CMNCORE) Schema.
3. Insert your OBMA_PROD, APP_ID & BUS_SERVICE (Add respective to your WebLogic)

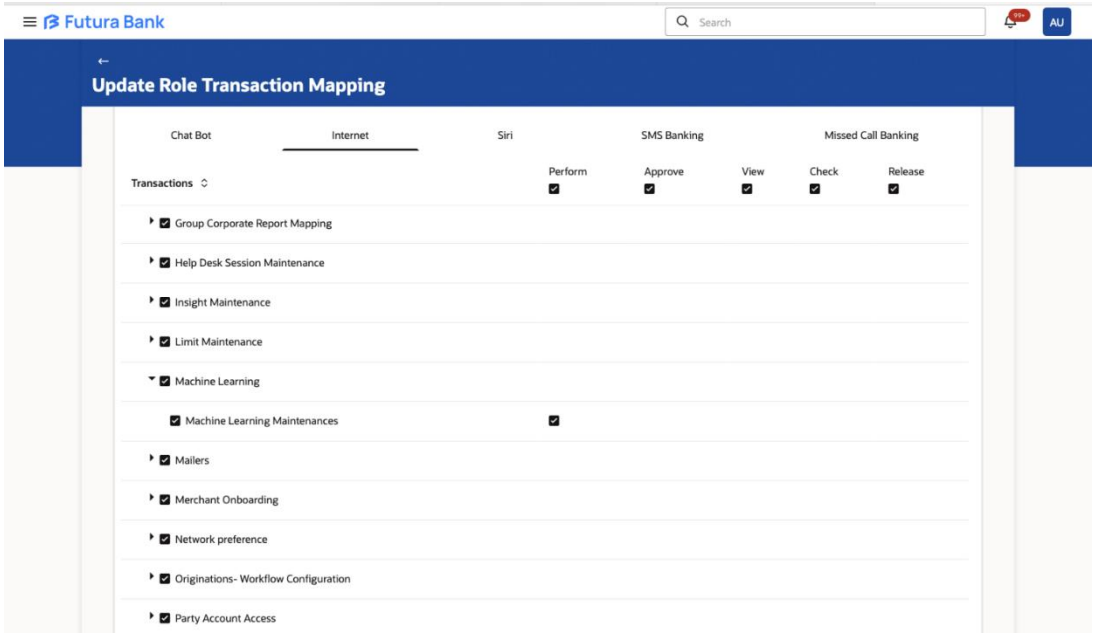
   Example:

   INSERT INTO CMC_TM_APPID_PROD_CONFIG (REC_ID, OBMA_PROD, APP_ID, ML_ENABLED, BUS_SERVICE) VALUES ('1', 'OBDX', 'OBDX', 'Y', 'http://ofss-mum-3483.snbomprshared1.gbucdsint02bom.oraclevcn.com:10002/digx-ml-indb');

3.2 OBDX Configuration Guide

Steps for Role Maintenance and Machine Learning Selection

1. Navigate to Role Maintenance.
2. Select the User Type as admin.
3. Go to Administrator Maintenance.
4. Select Machine Learning.



Steps for Security Authentication in Admin

1. Access the Admin Panel.
2. Navigate to Security Authentication.
3. Select the Enterprise Role.
4. Set up Two-Factor Authentication (2FA) as OTP for the desired transaction:
   - Login
   - Internal Transfer

Steps to Make a Database Entry into DIGX_FW_CONFIG_ALL_B table for the Desired Transaction

1. Identify the Task ID for the transaction.
2. Map the Task ID to the prop_id column based on the transaction type:
   - PC_CM_ME → Login
   - PC_F_CRNSFTV2 → Own Account Transfer
3. Insert the entry into the database with the corresponding task_id and prop_id.
4. You can add other task codes for desired transactions

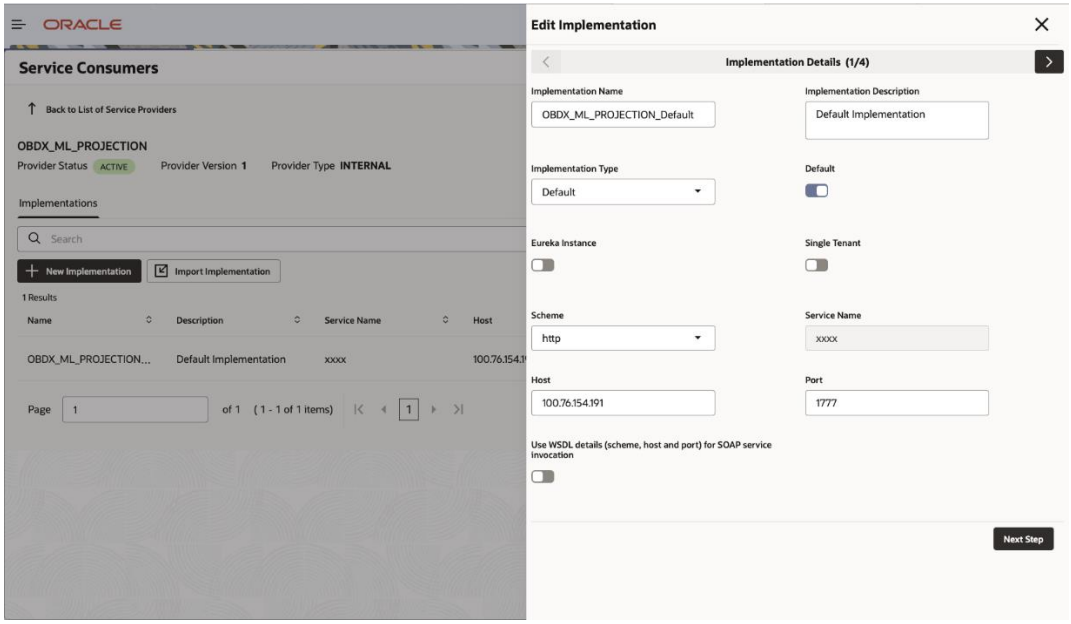
Example query:

Insert into DIGX_FW_CONFIG_ALL_B
(PROP_ID,CATEGORY_ID,PROP_VALUE,FACTORY_SHIPPED_FLAG,PROP_COMMENTS,SUMMARY_TEXT,CREATED_BY,CREATION_DATE,LAST_UPDATED_BY,LAST_UPDATED_DATE,OBJECT_STATUS,OBJECT_VERSION_NUMBER,EDITABLE,CATEGORY_DESCRIPTION)

values ('PC_F_CRNSFTV2','TwoFactorAuthenticationRuleEvaluator','ANOMALY_RULE','N',null,'External transactions repository adapter class','ofssuser',sysdate,'ofssuser',sysdate,'Y',1,'N',null);

Steps to Update OBRH Configuration

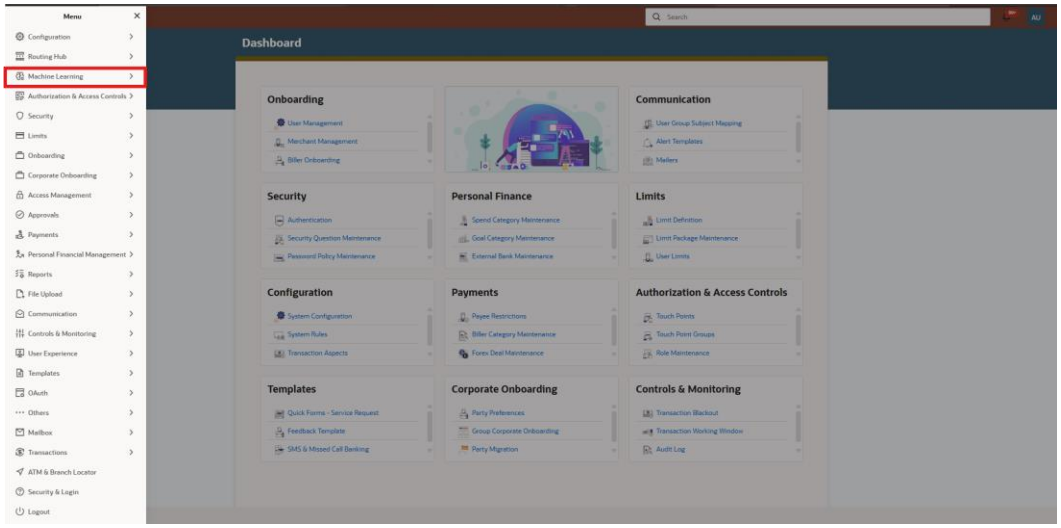
1. Navigate to the Service Consumers Section.
2. Select OBDX_TRUNK.
3. Go to the Service Providers Section.
4. Select OBDX_ML_PROJECTION.
5. Edit the Host and Port to match your required host and port settings.
6. Save the changes and ensure proper connectivity.

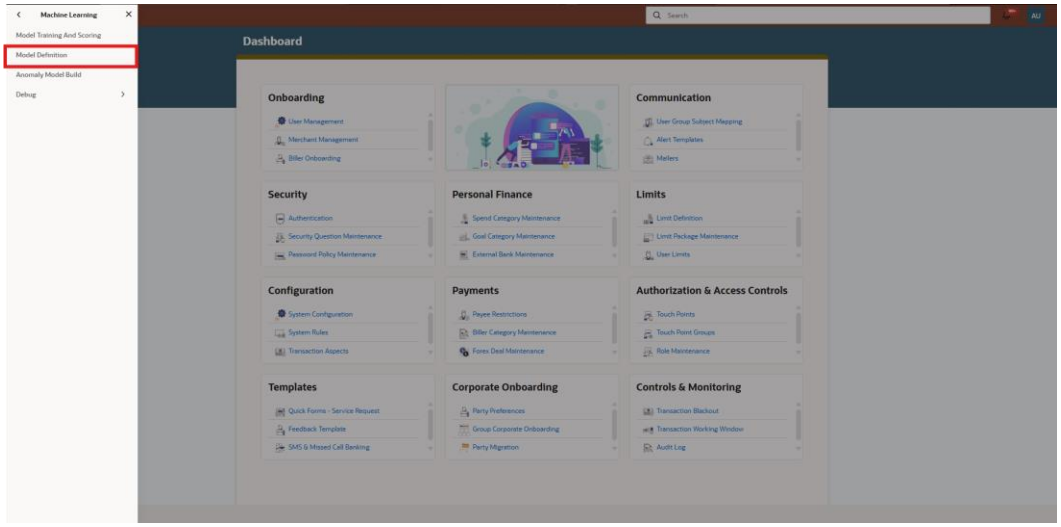


4. Model Definition Overview

4.1 Key Features

The Model Definition screen displays a list of configured anomaly detection models.





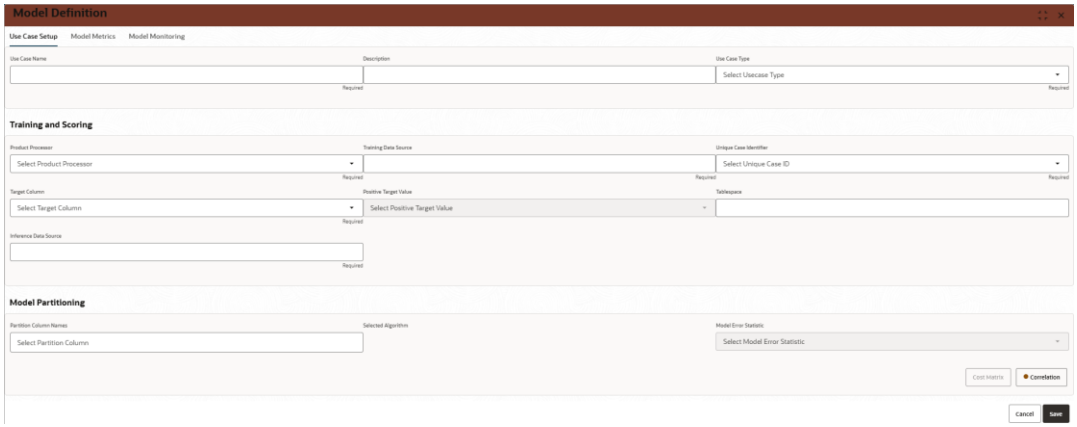
1. **Use Case Cards**

- Each card represents an anomaly detection model.
- Displays:
  - Use Case Name (e.g., OBDX_ANOMALY_PAYMENT, OBDX_ANOMALY_LOGIN)
  - Model Number (Versioning)
  - Correlation Status (Y/N)
  - Authorized / Unauthorized Status

2. **Navigation Controls**
    - Scroll through models using pagination.
3. **Action Buttons**
    - **Add New Model**: Create a new model.
    - **Refresh**: Update the model list.
    - **Settings/Options**: Manage, edit, or delete models.

5.1 Fields

This section allows users to define basic model details.



- **Use Case Name:**
  - Enter a unique name for the model.
  - Example: "Login_Anomaly_Model" or "Payment_Fraud_Detection"
  - **(Required)** – This field must be filled to proceed.
- **Description:**
  - Provide a summary of the model's purpose.
  - Example: "Detects unusual login attempts based on user behaviour patterns."
- **Use Case Type:**
  - Select the type of use case as **Anomaly_Detection**.
  - Options may **Regression & Classification**, or any other specific use cases.**(Required)**
- **Product Processor:**
  - **Select** the system or processor that will handle training.
  - Example: "OBDX"
  - **(Required)**
- **Training Data Source:**
  - Specify the dataset used to train the anomaly detection model.
  - The dataset **must include** the target column (i.e., the column indicating whether an instance is anomalous or normal).
  - Example: A CSV file or database table containing past login records.
  - **(Required)**
- **Inference Data Source:**
  - Specify the dataset used when making predictions.

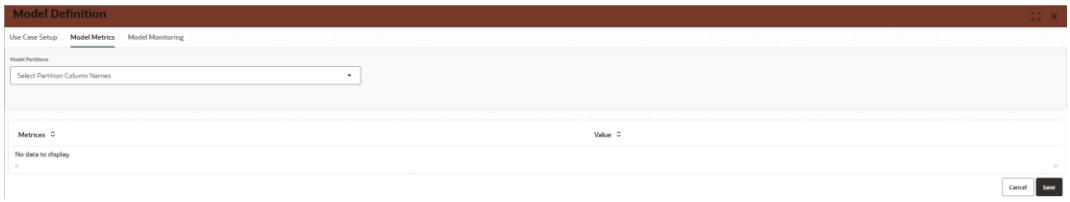
- Unlike the training dataset, this dataset **should not** include the target column.
- Example: "Live payment transaction records without labels."
- **(Required)**

- **Unique Case Identifier:**
  - Select the column in the dataset that uniquely identifies each record.
  - Example: "User_ID" for login data or "Transaction_ID" for payment data.
  - **(Required)**
- **Target Column:**
  - Select the column that defines whether a transaction/login attempt is an anomaly.
  - Example: A column labelled "Anomaly_Flag" where 1 indicates an anomaly and 0 indicates normal behaviour.
  - **(Required)**
- **Positive Target Value:**
  - Specify the value that represents an anomaly.
  - Example: If "1" indicates fraud or an unauthorized login, set "1" as the positive target value.
- **Tablespace:**
  - Define the storage location for the model's data within the system.
- **Partition Column Names:**
  - Select the columns used for partitioning the dataset.
  - Example: "Date" to separate records by time period.
- **Selected Algorithm:**
  - Choose the machine learning algorithm to be used.
  - Example: ALGO_SUPPORT_VECTOR_MACHINES, ALGO_NEURAL_NETWORK etc.
- **Model Error Statistic:**
  - Select an error metric to evaluate the model's accuracy.
  - Example: F1 Score, Precision-Recall, or AUC-ROC.
- **Correlation Button:**
  - Clicking this button will analyse relationships between features and the target variable.
  - Helps in understanding the significance of different input features.
- **Cost Matrix Button:**
  - Allows users to define cost-sensitive learning, useful for reducing false positives or false negatives.
  - **(Optional)**
- **Save Button:**
  - Saves the model configuration.
- **Cancel Button:**
  - Exits without saving any changes.

6. Model Metrics

6.1 Features

This section provides model evaluation metrics.

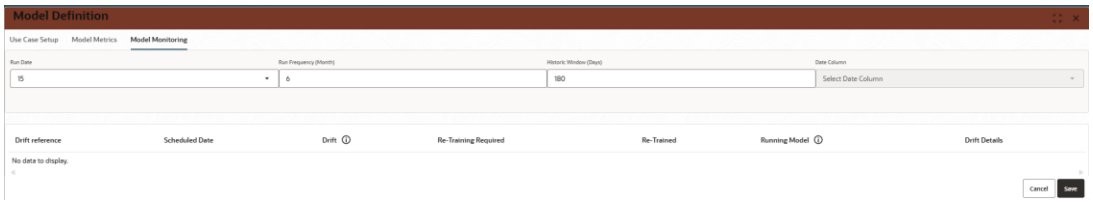


- **Model Partitions:**
  - Select different dataset partitions for viewing metrics.
  - **(Not Required)**
- **Metrics Table:**
  - Displays various performance evaluation metrics once the model is trained.
  - Initially, this table is empty until training is complete.
- **Save Button:**
  - Saves any updates made to the displayed metrics.
- **Cancel Button:**
  - Exits without saving changes.

7. Model Monitoring

7.1 Fields

Allows users to define model monitoring parameters.



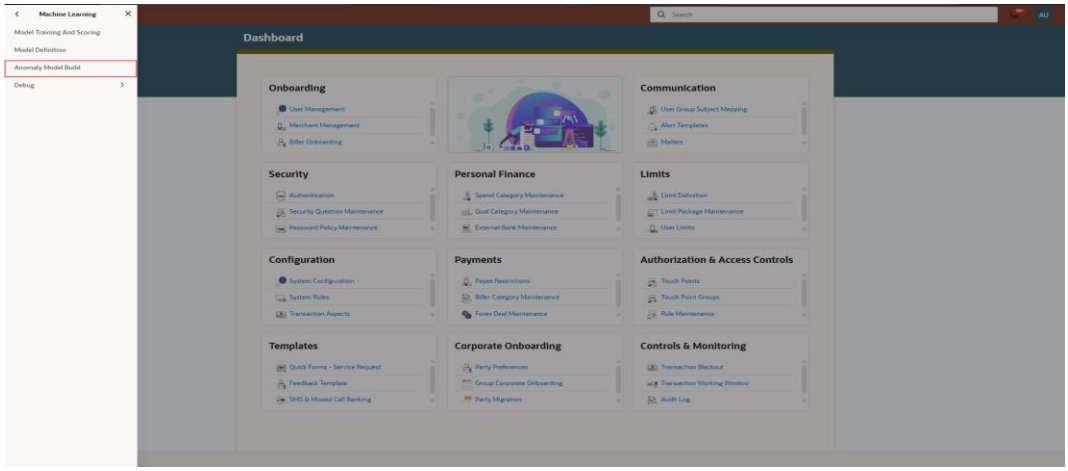
- **Run Date:**
  - A dropdown to select the scheduled monitoring run date.
- **Run Frequency (Months):**
  - Defines how often the model should be monitored.
  - Make sure training data consists data in range of frequency(For an instance if you set 180 days, then the training data should have data ranging in last 180 days)
  - Example: Every **6 months** or **quarterly.**
- **Historic Window (Days):**
  - Specifies how much past data should be considered for anomaly monitoring.
  - Example: "Last 90 days."
- **Date Column:**
  - The column used for time-based tracking of anomalies.
- **Drift Reference:**
  - Displays data drift detection results.
  - **Initially empty** but fills once monitoring is active.
- **Scheduled Date:**
  - Displays the next scheduled model monitoring date.
- **Drift:**
  - Shows whether significant changes in data distribution have been detected.
- **Re-Training Required:**
  - Indicates if the model requires retraining due to data drift or performance decline.
- **Re-Trained:**
  - Displays whether the model has been successfully retrained.
- **Running Model:**
  - Shows the status of the currently active model version.
- **Drift Details:**

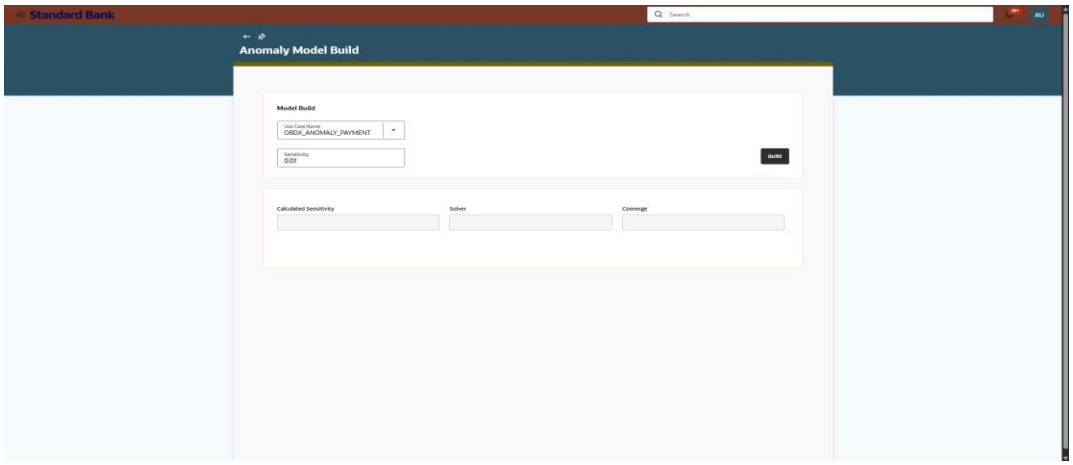
- Provides additional information on detected data drift and its impact on model performance.

- **Save Button:**
  - Saves the monitoring configuration settings.
- **Cancel Button:**
  - Exits without saving any changes.

8. Anomaly Model Build

Defines and builds the anomaly detection model with sensitivity settings.





8.1 Model Build Section

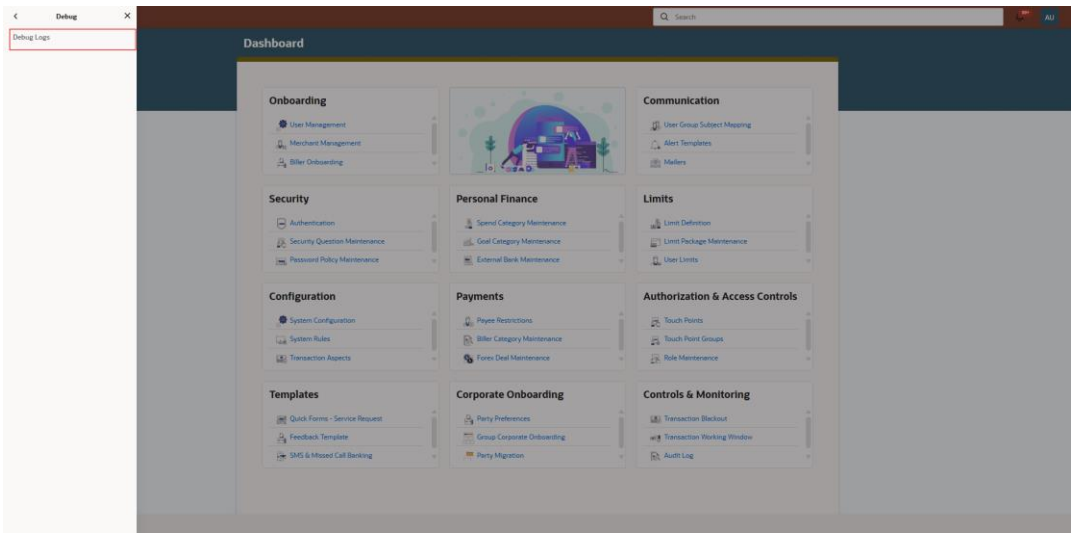
- **Use Case Name**: Select predefined use case (OBDX_ANOMALY_PAYMENT or OBDX_ANOMALY_LOGIN)
- **Sensitivity**: Define anomaly detection sensitivity (default: 0.01)
- **Build Button**: Start model training

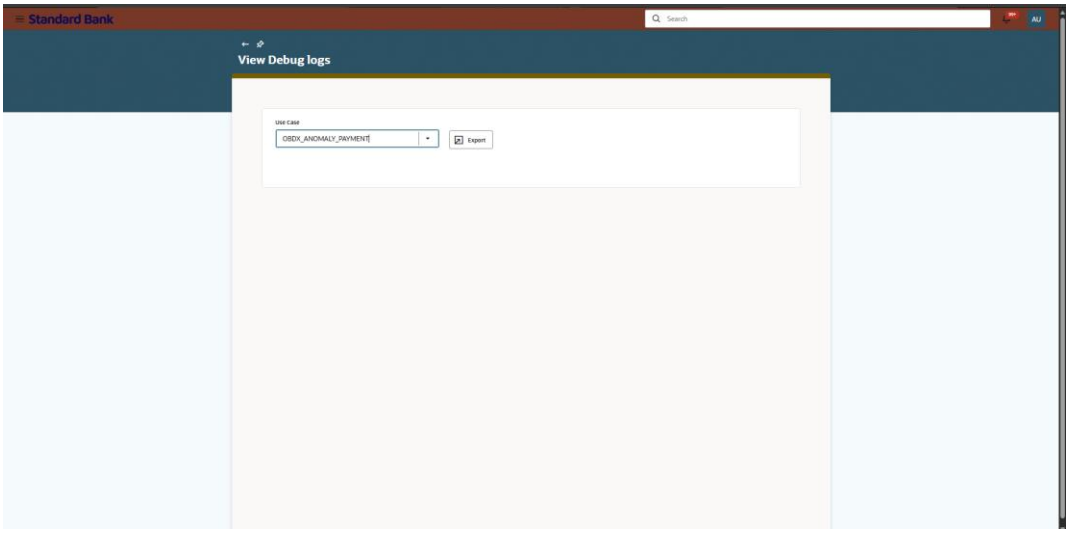
8.2 Model Output Section

- **Calculated Sensitivity**: Display computed sensitivity
- **Solver**: Show optimization method used
- **Converge**: Indicate if model reached an optimal solution

9. View Debug Logs

This section allows users to retrieve debug logs for model diagnostics.





9.1 Steps

- **Select Use Case**: Choose between OBDX_ANOMALY_PAYMENT or OBDX_ANOMALY_LOGIN.
- **Export Logs**: Click the **Export** button to download logs.

10. Conclusion

This user manual provides a detailed guide on setting up, managing, and monitoring anomaly detection models for login and payment data. Follow the outlined steps to ensure accurate anomaly detection and security monitoring.