EcoCash Liquidation Interface Specification Low Level Design Document

**EIDIKO SYSTEMS INTEGRATORS**



Version: 1.0

Release Date: 12-08-2020

|  |  |  |
| --- | --- | --- |
| Prepared By | Reviewed By | Approved By |
| EIDIKO |  |  |

Document Classification: CONFIDENTIAL

Contents

[1 Introduction 5](#_Toc48645992)

[1.1 General Description 5](#_Toc48645993)

[2 Purpose 5](#_Toc48645994)

[3 Scope 5](#_Toc48645995)

[3.1 In Scope 5](#_Toc48645996)

[4 Channels Involved 5](#_Toc48645997)

[5 Interface Dependencies 6](#_Toc48645998)

[5.1 External Dependencies 6](#_Toc48645999)

[5.2 Internal Dependencies 6](#_Toc48646000)

[6 Business Process Summary 7](#_Toc48646001)

[6.1 Process Overview 7](#_Toc48646002)

[Figure: 1 Process flow Approach for Salary & Corporate Payments. 7](#_Toc48646003)

[6.1.1 Salary & Corporate Payments Steps 7](#_Toc48646004)

[7 IIB Implementation Process Flow 8](#_Toc48646005)

[7.1 Salary & Corporate Payments Successful Request Response Flows 8](#_Toc48646006)

[7.1.1 Process Diagram 8](#_Toc48646007)

[7.1.2 Process Flow Steps 8](#_Toc48646008)

[7.1.3 ISAVE\_CARD\_MAPPING Details 9](#_Toc48646009)

[7.2 SalaryCorporatePaymentsT24CallFlow Process workflow Diagram 10](#_Toc48646010)

[7.2.1 Process Flow Steps 10](#_Toc48646011)

[7.3 Retry Process workflow 11](#_Toc48646012)

[7.3.1 Process Flow Diagram 11](#_Toc48646013)

[7.3.2 Process Flow steps 11](#_Toc48646014)

[7.4 SalaryCorporatePaymentsZSSCallFlow Process workflow Diagram 12](#_Toc48646015)

[7.4.1 Process Flow steps 12](#_Toc48646016)

[7.5 SB\_ISAVE\_PAYNET\_SCP\_TRACKING Table Details 13](#_Toc48646017)

[7.6 RAW\_AUDIT Process Flow: 14](#_Toc48646018)

[7.6.1 Process Flow Diagram 14](#_Toc48646019)

[7.6.2 Process Flow Steps: 15](#_Toc48646020)

[7.7 ERROR\_AUDIT Process Flow: 15](#_Toc48646021)

[7.7.1 Process Flow Diagram: 15](#_Toc48646022)

[7.7.2 Process Flow Steps: 16](#_Toc48646023)

[8 Interface Definitions 16](#_Toc48646024)

[8.1 Request Message Definition 16](#_Toc48646025)

[8.1.1 CSV Request Message Structure /Schema 16](#_Toc48646026)

[8.1.2 T24 Request Message Structure /Schema 17](#_Toc48646027)

[8.2 Response Message Definition 17](#_Toc48646028)

[8.2.1 Salary Payments T24 Response Messages Details: 17](#_Toc48646029)

[8.2.2 Salary Payments T24 Response Message Details: 18](#_Toc48646030)

[8.2.3 Corporate Payments Internal T24 Response Messages Details: 19](#_Toc48646031)

[8.2.4 Corporate Payments Internal T24 Response Message Details: 20](#_Toc48646032)

[8.2.5 Corporate Payments External T24 Response Messages Details: 21](#_Toc48646033)

[8.2.6 Corporate Payments External T24 Response Messages Details: 22](#_Toc48646034)

[8.3 Salary & Corporate Payments ZSS Request Structure /Schema 23](#_Toc48646035)

[8.3.1 Request Messages Details: 23](#_Toc48646036)

[8.3.2 Response Messages Details: 32](#_Toc48646037)

[9 Logging Mechanism 41](#_Toc48646038)

[9.1 Insert into Raw\_Audit\_Table: 41](#_Toc48646039)

[9.2 Insert into Error\_Audit\_Table: 42](#_Toc48646040)

[9.3 File Based Logging 43](#_Toc48646041)

[9.3.1 Log4j: 43](#_Toc48646042)

**Approval**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Role | Approver | Company | E-mail | Sign-Off Date |
| Delivery Manager |  | Eidiko |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Description | Version | Author |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## General Description

This Interface Design document outlines the integration requirements for the EcoCash Liquidation***(Request and Response)***interface used as banking operations. It summarizes the business processes, which use this interface. It also covers error handling and exception scenarios.

# Purpose

The purpose of this document is to capture events that trigger the interface, main steps within the interface and the integration architecture. This document is intended for use by the developers of the applications identified, the integration development team, and by the test organizations responsible for the testing of these applications.

# Scope

## In Scope

This document focuseson outlining the interface design for the EcoCash Liquidation ***(Request and Response)*** interface. Central to this document are the following:

* Overview of the business process that drives the need for the interface
* Proposed integration approach
* Trigger events and business dependencies on this interface
* Sequence /Flow Chart diagram of the interface
* Data mapping between the source and the target
* Error handling and exception scenarios
* Validation and backup requirements

# Channels Involved

The following table lists EcoCash Liquidation channels

|  |  |
| --- | --- |
| **Item #** | **Channel Name** |
| 1 | Batch - EcoCash |

# Interface Dependencies

## External Dependencies

The following table lists interface specific External requirements.

|  |  |
| --- | --- |
| **Item #** | **External Requirements** |
| 1 | ORADSN Database |
| 2 | Core Banking System(Temenos-T24) |
| 3 | FTP |
| 4 | Mobiquity Call |

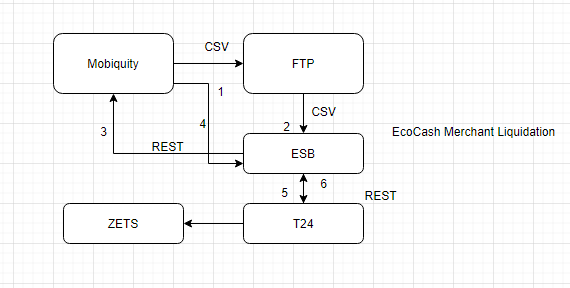
## Internal Dependencies

The following table lists interface specific internal requirements.

|  |  |
| --- | --- |
| **Item #** | **Internal Requirements** |
| 1 | StewardBankCommonEsql (Shared Library) |
| 2 | DB Logging App (Audit Logging) |
| 3 | SB\_EcoCSV\_Lib (DFDL - Shared Library) |

# Business Process Summary

## Process Overview



### Figure: 1 Process flow Approach for EcoCash Liquidation.

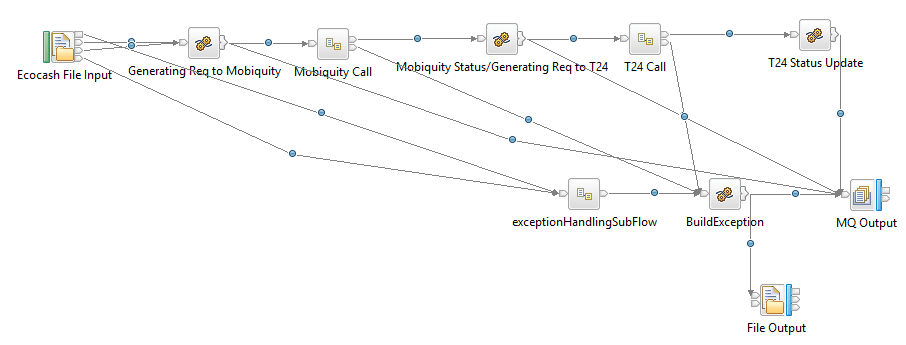
### EcoCash Liquidation Steps

|  |  |
| --- | --- |
| **S. No** | **Activities** |
| 1 | ESB receives the request CSV files from FTP. |
| 2 | After receiving the CSV files from FTP into ESB we validate CSV data in DFDL. |
| 3 | The initial CSV data is logged into RAW\_LOG\_SB\_REQ (RAW\_AUDIT\_LOGGING) and SB\_ECOCASH\_STATUS\_TRACKING table. |
| 4 | According to the MSISDN, Amount & FTXNID in received data, we make a Request body with POST method URL for Mobiquity Call. |
| 5 | The responses from Mobiquity Call, success or failure status are updated in RAW\_AUDIT\_LOGGING & SB\_ECOCASH\_STATUS\_TRACKING table. |
| 6 | On successful call of Mobiquity with success, the further T24 URL request is framed to make a T24 Call according to the BANK field in CSV to frame whether the EcoCash Internal URL or EcoCash External URL. |
| 7 | On unsuccessful call of Mobiquity, with failure, the further T24 URL request is not framed and the SB\_ECOCASH\_STATUS\_TRACKING table status is updated as failure for Mobiquity & T24. |
| 8 | The EcoCash Internal URL or EcoCash External URL is framed according to the accounts belonged to Steward Bank or Other Bank Accounts. |
| 9 | On Successful calls of T24 Call, Request and Response will be placed in the RAW\_LOG\_SB\_REQ (RAW\_AUDIT\_LOGGING) & SB\_ECOCASH\_STATUS\_TRACKING in Database. |
| 10 | On Unsuccessful call of T24 Request and Response will be placed in the ERROR\_LOG\_SB\_REQ (ERR\_AUDIT\_LOGGING) & SB\_ECOCASH\_STATUS\_TRACKING in Database. |

# IIB Implementation Process Flow

## EcoCash Liquidation Successful Request Response Flows

### Process Diagram



### Figure: 2 EcoCash Liquidation Route Flow.

### Process Flow Steps

|  |  |
| --- | --- |
| **S. No** | **Activities** |
| 1. | ESB receives the request CSV files from FTP the SB\_EcoCSV\_Lib DFDL takes the CSV data as DFDL. |
| 2. | The initial CSV request is logged in the RAW\_LOG\_SB\_REQ (RAW\_AUDIT\_TABLE) and SB\_ECOCASH\_STATUS\_TRACKING into Database. |
| 3. | According to the MSISDN, Amount & FTXNID in received data, we make a Request body with POST method URL for Mobiquity Call. |
| 4. | The responses from Mobiquity Call, success or failure status are updated in RAW\_AUDIT\_LOGGING & SB\_ECOCASH\_STATUS\_TRACKING table. |
| 5. | On successful call of Mobiquity with success, the further T24 URL request is framed to make a T24 Call according to the BANK field in CSV to frame whether the EcoCash Internal URL or EcoCash External URL. |
| 6. | On unsuccessful call of Mobiquity, with failure, the further T24 URL request is not framed and the SB\_ECOCASH\_STATUS\_TRACKING table status is updated as failure for Mobiquity & T24. |
| 7. | The EcoCash Internal URL or EcoCash External URL is framed according to the accounts belonged to Steward Bank or Other Bank Accounts. |
| 8. | On Successful calls of T24 Call, Request and Response will be placed in the RAW\_LOG\_SB\_REQ (RAW\_AUDIT\_LOGGING) & SB\_ECOCASH\_STATUS\_TRACKING in Database. |
| 9. | On Unsuccessful call of T24 Request and Response will be placed in the ERROR\_LOG\_SB\_REQ (ERR\_AUDIT\_LOGGING) & SB\_ECOCASH\_STATUS\_TRACKING in Database. |

## Retry Process workflow

### Process Flow Diagram



**Figure: 4 HTTP Retry for EcoCash Liquidation Mobiquity Flow**

### Process Flow steps

|  |  |
| --- | --- |
| **S. No** | **Activities** |
| 1 | HTTP Retry call is used to call the backend service for multiple times |
| 2 | On Successful call of backend service, response will be generated and send back to Business Application For further process |
| 3 | On Unsuccessful call of backend service it Retry for 3 times and sends error message to Business Application for further Process |

## SB\_ ECOCASH\_STATUS\_TRACKING Table Details:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Item #** | **DB Field Name** | **Field Description** | **Data Type** | **Length** | **Mandatory(Yes/No)** | **Comments** |
| 1 | PAYNET\_GUID | Unique ID number. | Varchar | 50 | Y | An Unique identifier. |
| 2 | PAYNET\_CSV\_FILENAME | CSV File Name | Varchar | 50 | Y | The input CSV File Name. |
| 3 | PAYNET\_CSV\_FIELDS | CSV Field Names | Varchar | 2000 | Y | The fields in CSV data. |
| 4 | PAYNET\_CSV\_MSISDN | MSISDN Number | Varchar | 50 | Y | The MSISDN Number. |
| 5 | PAYNET\_CSV\_AMOUNT | Amount in CSV data | Varchar | 50 | Y | Amount field in CSV data. |
| 6 | MOBIQUITY\_REQUEST\_BODY | Mobiquity Request body. | Varchar | 4000 | Y | The Mobiquity Request body for Mobiquity Call. |
| 7 | MOBIQUITY\_RESPONSE\_CODE | Response Code Number | Varchar | 50 | Y | The Response Code from Mobiquity Response |
| 8 | MOBIQUITY\_STATUS | Status of Mobiquity response | Varchar | 20 | Y | The Status of Mobiquity response |
| 9 | MOBIQUITY\_RESPONSE\_MESSAGE | Mobiquity Response body. | Varchar | 4000 | Y | The Mobiquity Response message. |
| 10 | MOBIQUITY\_DATETIME | DateTime from Mobiquity response. | Varchar | 20 | Y | The DateTime from Mobiquity response. |
| 11 | PAYNET\_CSV\_BANK | Bank Account Number | Varchar | 20 | Y | The Bank Account Number. |
| 12 | T24\_REQUEST\_URL | Request URL | Varchar | 500 | Y | The Request URL. |
| 13 | T24\_RESPONSE\_CODE | T24 Response Code | Varchar | 50 | Y | The T24 Response Code. |
| 14 | T24\_STATUS | T24 Status | Varchar | 20 | Y | The T24 Status. |
| 15 | T24\_RESPONSE\_MESSAGE | T24 Response Message | Varchar | 4000 | Y | The T24 Response Message. |
| 16 | PAYNET\_CSV\_STATUS | CSV Status | Varchar | 50 | Y | The CSV Status. |
| 17 | PAYNET\_DATECREATED | Creation Date | TIMESTAMP | (6) | Y | Created date of the record. |
| 18 | PAYNET\_DATEUPDATED | Updated Date | TIMESTAMP | (6) | Y | Updated date of the record. |

## RAW\_AUDIT Process Flow:

### Process Flow Diagram



**Figure: 6 Raw\_Audit\_Logging Flow**

### Process Flow Steps:

|  |  |
| --- | --- |
| **S. No** | **Activities** |
|  | The RAW\_AUDIT locks the input Request and Response |
|  | After the query is executed the record is inserted into the Database. |
|  | Final Response is inserted into the Database and then Response is logged into the RAW\_LOG\_SB\_REQ. |

## ERROR\_AUDIT Process Flow:

### Process Flow Diagram:



**Figure: 7 Exception Subflow**



**Figure: 8 Error\_Audit\_Logging Flow**

### Process Flow Steps:

|  |  |
| --- | --- |
| **S. No** | **Activities** |
|  | The ERROR\_AUDIT\_Q locks the input Request. |
|  | After the query is executed the record is inserted into the Data Base. |
|  | Final Response is inserted into the Database and then Response is logged into the ERR\_LOG\_SB\_REQ. |

# Interface Definitions

## Request Message Definition

### CSV Request Message Structure /Schema

**Request Type:** DFDL

**Request Format:** CSV

**Request CSV Fields:** tran id, date, remitter account number, beneficiary bank name, beneficiary bank code, beneficiary account number, beneficiary name, amount, currency, purpose, remitter name

### Mobiquity Request Message Structure /Schema

**Request METHOD:** ‘POST’

**Request Format:** application/json

**Request URL for Salary Payments T24 URL:** http://IPAddress:Port /stwbc/api/v1.0.0/stwb/

## Mobiquity Request Message Definition

**Request METHOD:** ‘POST’

**Response Type:** JSON

**Request Format:** application/json

### Request Message Details:

|  |
| --- |
| {  "header": {  "transactionStatus": "Live",  "audit": {  "T24\_time": 461,  "parse\_time": 1,  "responseParse\_time": 3,  "requestParse\_time": 659,  "versionNumber": "1"  },  "id": "PI2022807XNKCMYN",  "status": "success"  },  "body": {  "orderingReference": "123456",  "chargeBearer": "SHA",  "debitCurrency": "ZWL",  "invoiceReferences": [  {  "invoiceReference": "SALARYTEST"  }  ],  "transactionAmount": 324.87,  "endToEndReference": "100121",  "executionDate": "20200815",  "debitAccountId": "ZWL1400800010001",  "orderingCustomerName": "MAZOE INTERNATIONAL",  "creditAccountId": "1000000047",  "paymentCurrency": "ZWL",  "paymentOrderProductId": "SALARY"  }  } |
|  |

### Request Message Table Details:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Tag Name/ Field Name** | **Data Type** | **Mandatory(Yes/No)** | **Comments** |
| 1 | header | String | Y | Header tag |
| 2 | transactionStatus | String | Y | Transaction Status |
| 3 | audit | String | Y | Audit time |
| 4 | T24\_time | String | Y | T24 Time. |
| 5 | parse\_time | String | Y | Parsing time. |
| 6 | responseParse\_time | String | Y | Response Parse Time |
| 7 | requestParse\_time | String | Y | Request Parse Time |
| 8 | versionNumber | String | Y | Version Number |
| 9 | id |  | Y | Id Number |
| 10 | status | String | Y | Status |
| 11 | body | String | Y | Body tag. |
| 12 | orderingReference | String | Y | Ordering Reference Number |
| 13 | chargeBearer | String | Y | Charge Bearer |
| 14 | debitCurrency | String | Y | Currency |
| 15 | invoiceReferences\* | String | Y | Invoice References |
| 16 | invoiceReference | String | Y | Invoice References |
| 17 | transactionAmount |  | Y | Amount |
| 18 | endToEndReference | String | Y | End To End Reference number. |
| 19 | executionDate | String | Y | Date |
| 20 | debitAccountId | String | Y | Debit Account |
| 21 | orderingCustomerName | String | Y | CustomerName |
| 22 | creditAccountId | String | Y | Credit Account |
| 23 | paymentCurrency | String | Y | Currency |
| 24 | paymentOrderProductId | String | Y | Payment Order Product Id |

### Corporate Payments Internal T24 Response Messages Details:

**Response Type:** JSON

|  |
| --- |
| {  "header": {  "transactionStatus": "Live",  "audit": {  "T24\_time": 597,  "parse\_time": 1,  "responseParse\_time": 2,  "requestParse\_time": 731,  "versionNumber": "1"  },  "id": "PI2022807XNK8FSK",  "status": "success"  },  "body": {  "chargeBearer": "SHA",  "narratives": [  {  "narrative": "STEWARD"  }  ],  "debitCurrency": "ZWL",  "invoiceReferences": [  {  "invoiceReference": "HERBERT"  }  ],  "transactionAmount": 55.78,  "endToEndReference": "NOTPROVIDED",  "executionDate": "20200815",  "debitAccountId": "ZWL1400800010001",  "orderingCustomerName": "MAZOE INTERNATIONAL",  "creditAccountId": "1000000047",  "paymentCurrency": "ZWL",  "paymentOrderProductId": "CORPPAYINT"  }  } |
|  |

### Corporate Payments Internal T24 Response Message Table Details:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Tag Name/ Field Name** | **Data Type** | **Mandatory(Yes/No)** | **Comments** |
| 1 | header | String | Y | Header Tag. |
| 2 | transactionStatus | String | Y | Transaction Status |
| 3 | audit | Decimal | Y | Audit Times |
| 4 | T24\_time | String | Y | T24 time |
| 5 | parse\_time | String | Y | Parse time |
| 6 | responseParse\_time | String | Y | Response Parse time |
| 7 | requestParse\_time | String | Y | Request Parse time |
| 8 | versionNumber | String | Y | Version Number |
| 9 | id |  | Y | Id Number |
| 10 | status | String | Y | Status |
| 11 | body | String | Y | Body Tag. |
| 12 | chargeBearer | String | Y | Charge Bearer |
| 13 | narratives | String | Y | Narratives |
| 14 | narrative | String | Y | Narrative |
| 15 | debitCurrency | String | Y | Currency |
| 16 | invoiceReferences | String | Y | Invoice References |
| 17 | invoiceReference | String | Y | Invoice Reference Number |
| 18 | transactionAmount | String | Y | Amount |
| 19 | endToEndReference | String | Y | End To End Reference |
| 20 | executionDate | String | Y | Date |
| 21 | debitAccountId | String | Y | Debit Account |
| 22 | orderingCustomerName | String | Y | Ordering Customer Name |
| 23 | creditAccountId | String | Y | Credit Account |
| 24 | paymentCurrency | String | Y | Currency |
| 25 | paymentOrderProductId | String | Y | Payment Order Product Id |

### Corporate Payments External T24 Response Messages Table Details:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Tag Name/ Field Name** | **Data Type** | **Mandatory(Yes/No)** | **Comments** |
| 1 | header | String | Y | Header Tag. |
| 2 | transactionStatus | String | Y | Transaction Status |
| 3 | audit | Decimal | Y | Audit Times. |
| 4 | T24\_time | String | Y | T24 time |
| 5 | parse\_time | String | Y | Parse time |
| 6 | responseParse\_time | String | Y | Response Parse time |
| 7 | requestParse\_time | String | Y | Request Parse time |
| 8 | versionNumber | String | Y | Version Number |
| 9 | id |  | Y | Id Number |
| 10 | status | String | Y | status |
| 11 | body | String | Y | Body tag. |
| 12 | chargeBearer | String | Y | Charge Bearer |
| 13 | debitCurrency | String | Y | Currency |
| 14 | endToEndReference | String | Y | End To End Reference |
| 15 | executionDate |  | Y | Date |
| 16 | additionalInformations | String | Y | Additional Informations |
| 17 | additionalInformation | String | Y | Additional Information |
| 18 | debitAccountId | String | Y | Debit Account |
| 19 | beneficiaryAccountId | String | Y | Beneficiary Account Number |
| 20 | orderingCustomerName | String | Y | Customer Name |
| 21 | orderingReference | String | Y | Ordering Reference |
| 22 | beneficiaryName | String | Y | Beneficiary Name |
| 23 | transactionAmount | String | Y | Amount |
| 24 | paymentCurrency | String | Y | Currency |
| 25 | paymentOrderProductId | String | Y | Payment Order Product Id |

### Corporate Payments External T24 Response Message Details:

**Response Type:** JSON

|  |
| --- |
| {  "header": {  "transactionStatus": "Live",  "audit": {  "T24\_time": 360,  "parse\_time": 1,  "responseParse\_time": 3,  "requestParse\_time": 262,  "versionNumber": "1"  },  "id": "PI2022807XNKBTS1",  "status": "success"  },  "body": {  "chargeBearer": "SHA",  "debitCurrency": "ZWL",  "endToEndReference": "NOTPROVIDED",  "executionDate": "20200815",  "additionalInformations": [  {  "additionalInformation": "ZW0010001"  }  ],  "debitAccountId": "ZWL1400800010001",  "beneficiaryAccountId": "1000000047",  "orderingCustomerName": "MAZOE INTERNATIONAL",  "orderingReference": "CORPRTGS",  "beneficiaryName": "HERBERT",  "transactionAmount": 55.78,  "paymentCurrency": "ZWL",  "paymentOrderProductId": "CORPPAYEXT"  }  } |

|  |
| --- |
| OTB RESPONSE\*\*\* |

# Logging Mechanism

## Insert into Raw\_Audit\_Table:

**Table Name: RAW\_AUDIT\_TABLE**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Item #** | **DB Field Name** | **Field Description** | **Data Type** | **Length** | **Mandatory(Yes/No)** | **Comments** |
| 1 | MSGID | MessageID | Varchar | 100 | Y | From input request |
| 2 | LOGGING\_TIME | Logging Time | TimeStamp | 6 | Y | Name |
| 3 | MESSAGE | Message(Request/Response) | Clob | - | Y | Request from channel |
| 4 | MESSAGETYPE | Type of Message(Request/Response) | Varchar | 100 | Y | Type of message either Request or Response |
| 5 | APPNAME | Application Name | Varchar | 100 | Y | Name of the Application |
| 6 | BROKER | Broker Name | Varchar | 100 | Y | Broker Name |
| 7 | TIME\_LOCAL\_TRANSACTION | Transaction Time | Varchar | 20 | Y | Time of the Transaction |
| 8 | DATE\_LOCAL\_TRANSACTION | Transaction Date | Varchar | 20 | Y | Date of the Transaction |
| 9 | RETRIEVAL\_REFERENCE\_NUMBER | Retrieval Reference Number | Varchar | 20 | Y |  |

## Insert into Error\_Audit\_Table:

**Table Name: ERROR\_AUDIT\_TABLE**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Item #** | **DB Field Name** | **Field Description** | **Data Type** | **Length** | **Mandatory(Yes/No)** | **Comments** |
| 1 | MSGID | MessageID | Varchar | 50 | Y | From input request |
| 2 | LOGGING\_TIME | Application Name | Timestamp | 50 | Y | Time of the logging |
| 3 | MESSAGE | Message Type(Request/Response) | Clob | 50 | Y | Request of the Message |
| 4 | MESSAGETYPE | Message Type | Varchar | 50 | Y | Type of message either Request or Response |
| 5 | APPNAME | Application Name | Varchar | 4000 | Y | Name of the Application |
| 6 | BROKER | Broker Name | Varchar | - | Y | Name of the Broker |
| 7 | ERRORDESCRIPTION | Exception | Clob | 4000 | Y | Exception Information |
| 8 | TIME\_LOCAL\_TRANSACTION | Transaction Time | Varchar | 20 | Y | Time of the Transaction |
| 9 | DATE\_LOCAL\_TRANSACTION | Transaction Date | Varchar | 20 | Y | Date of the Transaction |
| 9 | RETRIEVAL\_REFERENCE\_NUMBER | Retrieval Reference Number | Varchar | 20 | Y | Retrieval Reference Number |
| 10 | BORKER\_ERROR\_CODE | Broker Error Code | Varchar |  | Y | Error code generate by Broker |
| 11 | USER\_DEFINE\_ERROR\_CODE | User Defined Error Code | Varchar |  | Y | User Defined Code |

## File Based Logging

### Log4j:

As part of auditing the request we have to use Log4j mechanism in the application to log the request in the file.

File Path: To be specified, while deploying the application