

CRedit voucher

Solution Design Document (SDD)

|  |  |
| --- | --- |
| **Process ID** | P-001 |
| **PU** | EPO |
| **Process Group** | EPO |
| **Process Name** | Credit Voucher |
| **Document ID** | 1.0 |

|  |  |
| --- | --- |
| **Version** | 0.1 |
| **Status** | Initial Version |
| **Date** | 22-12-2022 |

# Contents

[Contents 2](#_Toc478342575)

[Document Control 3](#_Toc478342576)

[Template 3](#_Toc478342577)

[Completion stages 3](#_Toc478342578)

[Version history 3](#_Toc478342579)

[1. Introduction 4](#_Toc478342580)

[1.1. Document purpose 4](#_Toc478342581)

[1.2. Process summary 4](#_Toc478342582)

[1.3. Reference artefacts 4](#_Toc478342583)

[2. Solution Overview 5](#_Toc478342584)

[2.1. High level design (HLD) 5](#_Toc478342585)

[2.2. Solution description 6](#_Toc478342586)

[3. Solution Detail 8](#_Toc478342587)

[3.1. Object model 8](#_Toc478342588)

[3.2. Solution components 8](#_Toc478342589)

[4. Operations 9](#_Toc478342590)

[4.1. Business exceptions 9](#_Toc478342591)

[4.2. System exceptions 9](#_Toc478342592)

[4.3. Management information (MI) **Error! Bookmark not defined.**](#_Toc478342593)

[4.4. Scheduling and manual execution 10](#_Toc478342594)

[4.5. Optimization and scaling 10](#_Toc478342595)

[4.6. Alerting 10](#_Toc478342596)

[4.7. Logging 10](#_Toc478342597)

[5. Data Management 11](#_Toc478342598)

[5.1. Storage 11](#_Toc478342599)

[5.2. Privacy 11](#_Toc478342600)

[5.3. Security 11](#_Toc478342601)

[5.4. Preservation 11](#_Toc478342602)

[6. Considerations 12](#_Toc478342603)

[i. Business Glossary 13](#_Toc478342604)

[ii. Attachments 14](#_Toc478342605)

# Document Control

## Template

The following person(s) own the format and information requested in this document template.

|  |  |  |
| --- | --- | --- |
| Team | Name | Relevant Sections |
|  |  |  |

The version history of this template is as follows.

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Change Summary |
| 0.1 | 23/12/2022 | Bakare Sodiq | * Initial Version |

## Completion stages

This document will be completed as per the following sign-off points.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Phase | Relevant Sections | Producer | Sign-off |
| 0.1 | Design | All | Delivery Team | Process Owner ( ) |

## Version history

This document’s change history is as follows.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Author | Approver | Change Summary |
| 0.1 | 23/01/2022 | Bakare Sodiq |  |  |

# 1. Introduction

## 1.1. Document purpose

The purpose of the Solution Design Document (SDD) is to describe the technical solution developed to meet the requirements outlined in the Process Design Document (PDD), including any technical prerequisites and considerations required to deploy, operate, and maintain the process. It is a living document that is incrementally developed as the technical solution is built and is finalized prior to deployment into the production automation infrastructure.

This document will refer to the automation package (“The Solution” or “Solution”) throughout, which represents the Business Objects and Processes, as well as any other peripheral technical components (e.g credentials, templates, databases etc.) used to deliver the automated process.

## 1.2. Process summary

The Credit Voucher process captures the steps for settling all account transactions done on Visa Transactions. This process is aimed at settling different merchants for transactions processed on the Unified Payment platform at our First Bank branches

## 1.3. Reference artefacts

The Process Design Document (PDD) which captures the business-related details of the process being automated and describes how the automated process is intended to work, including risk and data management control has been included as a link in this manual

The following artefacts should be read in conjunction with this document.

|  |  |  |
| --- | --- | --- |
| Title | Version | Location |
| Process Design Document (PDD) –  Credit Voucher | 0.1 |  |

# 2. Solution Overview

This section describes the high-level design of the automated solution.

## 2.1. High level design (HLD)

**Depatcher**

Get AddDayAsset

Bot load local and External config Validate Work folders

Download Report

DetailedTransactionReport

Read DetailedTransactionReport Sheet and Filter By Credit Voucher

Send Report to Card Production Via Email

send Job Report

**Performer**

Get AddDayAsset

Bot load local and External config Validate Work folders

NO

Download Report

From Storage Bucket

Treat Report to get Total Base 2 Amount

Download VSS120 Report

Read Sheet to verified Total Base 2 Amount

Is Account balanced?

Send Account balanced Report to Process Owner

Send Account Not Balanced Report to Process Owner

Build Upload

## 

## 2.2. Solution description (Dispatcher)

* Bot Kill all application
* Bot check the day set on Orchestrator, if it is Monday; Bot will run Saturday, Sunday and Monday. Else the bot only treats the day.
* Sequence: Load all Config Portal
  + Load Internal Config Values: These load all assets in the internal config excel

In the project directory. Output Argument: Config

* + Validate Work Folders: Create folder its does not exist
  + Load External Config workflow: Read system settings output Argument Sysfig
  + Delete work folders and validate again
  + Assign report Date
  + Sequent to check for Monday Logic
* Download Logic workflow
* Assign all WF Variables: all needed variables
* Create File to Processing workflow
* Sequence Download: Logic of Downloading report
* Sequence: Login and Download
* Get Credentials
* Do While: Try to login 3 Times
* Logout from FTP workflow
* Login FTP workflow
* If Login Successfully: Download Report FTP
* Sequence If Can't Login: Exception error throw
* If File Unable to Download: Exception error throw
* READ ALL SHEET
* Delete Download File
* Send Report To Card Production Team
* Send Job Report workflow

## 2.2.1 Solution description (Performer)

* Bot Kill all application
* Bot check the day set on Orchestrator, if it is Monday; Bot will run Saturday, Sunday and Monday. Else the bot only treats the day.
* Sequence: Load all Config Portal
  + Load Internal Config Values: These load all assets in the internal config excel

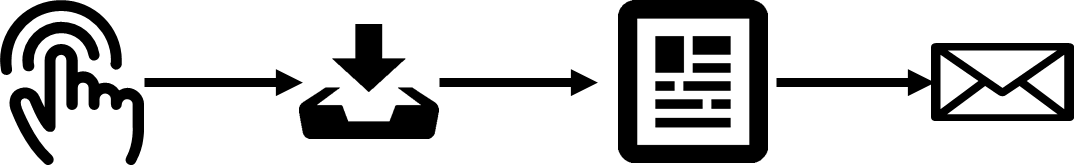
In the project directory. Output Argument: Config

* + Validate Work Folders: Create folder its does not exist
  + Load External Config workflow: Read system settings output Argument Sysfig
  + Delete work folders and validate again
  + Assign report Date
  + Sequent to check for Monday Logic
* Sequence: Get Storage Bucket
* Assign all WF Variables: all needed variables
* Sequence: Download File
* Try and Catch
* Download Storage Bucket File: Credit Voucher Report
* Rename File
* Get TotalAmountBase2
  + Assign TotalAmountBase2: Math.Abs(DT.AsEnumerable().Sum(x => Convert.ToDecimal(x[DestinationColName])))
* Download VSS 120
* Sequence: Read VSS Sheet
* Delete VSS File workflow
* Verify Account Details VSS workflow
* Build Upload File
* Multiple Assign: all needed variables
* Build Upload File Data Table: UploadFileDT
* Get Merchant Details workflow: get details for " VISA MERCHANT FUND
* Assign Credit Account Number
* Sequence: Write all Data
* Read Range Process File: CreditVDT
* For Each in the CreditVDT
  + Invoke code: Get the Transaction Type
  + If Acct type is Unconfirmed
  + Sequence Get Acct Type
* Get Account Type
* Assign Account Type
* If Acct Type is null
  + Sequence Undetermined
  + Multiple Assign Upload details
  + Add Data Row
* Switch: Account Type
* USD
  + Multiple Assign Upload details
  + Add Data Row
* NGN
  + Sequence: Send Mail
  + Multiple Assign Upload details
  + Add Data Row
* GBPEUR
  + Check for GEP or EUR and assign value
  + Multiple Assign Upload details
  + Add Data Row
* Sequence Credit Voucher
  + Multiple Assign Upload details
  + Add Data Row
* END the Upload File
  + Update Row
* Assign debit: Convert.ToDecimal(UploadFileDT.AsEnumerable().Where(x => x.Field<string>("Type") == "D").Sum(z=>z.Field<decimal>("Amount".ToString())))
* Assign Credit : Convert.ToDecimal(UploadFileDT.AsEnumerable().Where(x => x.Field<string>("Type") == "C").Sum(z=>z.Field<decimal>("Amount".ToString())))
* If Account is Balanced: Math.Round(debit, 1) - Math.Round(credit, 1) == 0
  + Assign out\_IsBalanced : True
* Else
  + Assign out\_IsBalanced : False
* Assign out\_UploadFileName
* Copy File: Uploade Template File to Processing folder
* Write Range: UploadFileDT
* Is Account Balanced?
* If True:
* Send Job Report workflow: Account is Balanced
* Else
* Send Job Report workflow: Account not Balanced

# 3. Solution Detail

This section describes the low-level design of the automated solution.

## 3.1. Object model



Open Browser Download report Excel Activities Send Email

## 

## 3.2. Solution components

|  |  |  |  |
| --- | --- | --- | --- |
| Ref. | Type | Name | Purpose |
| C1 | Process | Open Application | Opens Unified Payment portal to download the required reports |
| C2 | Process | Excel Activities | * Read Excel document into datatable * Add rows * Sort document by columns * Perform computations i.e Add, Subtract, Multiply |
| C3 | Process | Send Email | * Attach excel file to email * Send email |

# 4. Operations

This section describes the controls, reporting and alerting required to operate the solution.

## 4.1. Business exceptions

Events classified as Business Exceptions are those that are not expected to be handled by the virtual worker. That is, they are out of scope of what is described in the PDD.

Business Exceptions are marked as follows.

|  |  |  |
| --- | --- | --- |
| No | Exception | Solution |
| 1 | **Change in report Format**   * Change in number of columns. | * Send email alert to process owner |
| 2 | **Inability to find report**   * The bot is unable to download report due to change in naming convention of the required report. | * Send email alert to process owner |
| 2 | **Inability to find report**   * The bot is unable to download report due to change in naming convention of the required report. | * Send email alert to process owner |
|  |  |  |
|  |  |  |

## 4.2. System exceptions

System exceptions can fall in one of two categories:

1. Known system exceptions – which are known problem or risky areas in the process (e.g. to common system unreliability) that have been specifically catered for with extra retries or redundancies, or at least a specific error description.
2. Unknown system exceptions – which are unplanned errors.

### Known system exceptions

System Exceptions with specific catches are marked as follows.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario | Work Queue | Status | Tags | Required Action |
| N/A | N/A | N/A | N/A | N/A |

### Unknown system exceptions

Unknown System Exceptions will be represented as follows.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenario | Work Queue | Status | Tags | Required Action |
| N/A | N/A | N/A | N/A | N/A |

## 4.4. Scheduling and manual execution

The robot will run everyday.

## 4.5. Optimization and scaling

To be determined.

## 4.6. Alerting

Any alerting built into The Solution is described below as per PDD specification.

|  |  |  |  |
| --- | --- | --- | --- |
| Ref. | Scenario | Method | Recipient(s) |
| AL1 | Bot is unable to find Approved/Failed Transaction report | Send Email | Process owner |
| AL2 | Bot has completed its execution | Send Email | Process owner |

## 4.7. Logging

* There is an in-built audit trail which captures actions and timelines of robot’s activities

# 5. Data Management

## 5.1. Storage

The downloaded files i.e Approved/Failed Transactions report and the final report will be stored in a specified folder on the system/server running the bot

## 5.2. Privacy

The bot will not transmit documents/files to external locations (outside Firstbank Bank) and access will be restricted to assigned members of the E-business team

## 5.3. Security

At a specified date (to be determined by members of the COE), the bot will delete all downloaded reports from its download folder.

## 5.4. Preservation

The bot will log onto the applications using credentials supplied by Firstbank bank’s IT

# 6. Considerations

* Stable internet connectivity will be readily available for the bot to function
* Any changes in the naming conventions of documents which are downloaded by the bot may require some updates to robot configuration/process design

# i. Business Glossary

Acronyms and terms used throughout this document are described below.

|  |  |  |
| --- | --- | --- |
| Acronym or Term | Synonym(s) | Full Description |
| VW | PAC, Robot, Bot | Virtual Worker |
| HW |  | Human Worker |
| PDD |  | Process Design Document |
| SDD |  | Solution Design Document |
| VM | VM | Virtual Machine |
| VDI | VDI | Virtual Desktop Interface |
| RPA | RPA | Robotic Process Automation |
| SSO |  | Single Sign-On |

# ii. Attachments

The following attachments relate to this document.

|  |  |
| --- | --- |
| Attachment | Description |
|  |  |
|  |  |