Discovery Report

Online Interface

LFVN Finance Vietnam

Version 0.8

Status: Final

Date: 28.04.2021

Prepared by: OpenWay

Author: Le Thi Ngoc Hieu

Approved By: Tu B. Nguyen

1. Table of Contents

[1. Table of Contents 2](#_Toc70498702)

[2. History of changes 8](#_Toc70498703)

[3. Introduction 10](#_Toc70498704)

[3.1. Notations 12](#_Toc70498705)

[3.2. Service Overview 12](#_Toc70498706)

[4. General Details and Methodology 14](#_Toc70498707)

[4.1. Security Check Requirement 14](#_Toc70498708)

[4.2. Request/Response Communication Method 14](#_Toc70498709)

[4.3. Message Authentication 14](#_Toc70498710)

[4.4. Application Layer 14](#_Toc70498711)

[4.5. Request Message ID 14](#_Toc70498712)

[4.6. RRN 14](#_Toc70498713)

[4.7. Error Handling 15](#_Toc70498714)

[4.8. Institution ID – Access to Data 19](#_Toc70498715)

[4.9. User Authentication 20](#_Toc70498716)

[4.10. Items marked as “To Be Determined” (TBD) 20](#_Toc70498717)

[5. Requirement Description – Interface with Online Channel 21](#_Toc70498718)

[5.1. REQWS001. Get credit limit and available amount 21](#_Toc70498719)

[5.1.1. Business Requirements 21](#_Toc70498720)

[5.1.2. Technical Details 21](#_Toc70498721)

[5.1.3. Message Specifications 21](#_Toc70498722)

[5.1.4. Mapping Fields 22](#_Toc70498723)

[5.1.5. Limitations 22](#_Toc70498724)

[5.2. REQWS002. Lock Issuing Contract 22](#_Toc70498725)

[5.2.1. Business Requirements 22](#_Toc70498726)

[5.2.2. Technical Details 22](#_Toc70498727)

[5.2.3. Message Specifications 23](#_Toc70498728)

[5.2.4. Mapping Fields 23](#_Toc70498729)

[5.2.5. Limitations 23](#_Toc70498730)

[5.3. REQWS003. Transaction History 23](#_Toc70498731)

[5.3.1. Business Requirements 23](#_Toc70498732)

[5.3.2. Technical Details 24](#_Toc70498733)

[5.3.3. Message Specifications 24](#_Toc70498734)

[5.3.4. Mapping Fields 24](#_Toc70498735)

[5.3.5. Limitations 25](#_Toc70498736)

[5.4. REQWS004. Get Instalment Portion by PlanID 25](#_Toc70498737)

[5.4.1. Business Requirements 25](#_Toc70498738)

[5.4.2. Technical Details 26](#_Toc70498739)

[5.4.3. Message Specifications 26](#_Toc70498740)

[5.4.4. Mapping Fields 26](#_Toc70498741)

[5.4.5. Limitations 27](#_Toc70498742)

[5.5. REQWS005. Payment History 27](#_Toc70498743)

[5.5.1. Business Requirements 27](#_Toc70498744)

[5.5.2. Technical Details 27](#_Toc70498745)

[5.5.3. Message Specifications 28](#_Toc70498746)

[5.5.4. Mapping Fields 28](#_Toc70498747)

[5.5.5. Limitations 29](#_Toc70498748)

[5.6. REQWS006. Manage Statement 29](#_Toc70498749)

[5.6.1. Business Requirements 29](#_Toc70498750)

[5.6.2. Technical Details 29](#_Toc70498751)

[5.6.3. Message Specifications 29](#_Toc70498752)

[5.6.4. Mapping Fields 30](#_Toc70498753)

[5.6.5. Limitations 30](#_Toc70498754)

[5.7. REQWS007. Get Merchant Lists 30](#_Toc70498755)

[5.7.1. Business Requirements 30](#_Toc70498756)

[5.7.2. Technical Details 31](#_Toc70498757)

[5.7.3. Message Specifications 31](#_Toc70498758)

[5.7.4. Mapping Fields 31](#_Toc70498759)

[5.7.5. Limitations 32](#_Toc70498760)

[5.8. REQWS008. Get Client Contract Hierarchy 32](#_Toc70498761)

[5.8.1. Business Requirements 32](#_Toc70498762)

[5.8.2. Technical Details 32](#_Toc70498763)

[5.8.3. Message Specifications 32](#_Toc70498764)

[5.8.4. Mapping Fields 33](#_Toc70498765)

[5.8.5. Limitations 33](#_Toc70498766)

[5.9. REQWS009. Get BNPL Issuing contract status 33](#_Toc70498767)

[5.9.1. Business Requirements 33](#_Toc70498768)

[5.9.2. Technical Details 33](#_Toc70498769)

[5.9.3. Message Specifications 34](#_Toc70498770)

[5.9.4. Mapping Fields 34](#_Toc70498771)

[5.9.5. Limitations 35](#_Toc70498772)

[5.10. REQWS010. Get Pay Later Loyalty Point 35](#_Toc70498773)

[5.10.1. Business Requirements 35](#_Toc70498774)

[5.10.2. Technical Details 35](#_Toc70498775)

[5.10.3. Message Specifications 35](#_Toc70498776)

[5.10.4. Mapping Fields 36](#_Toc70498777)

[5.10.5. Limitations 36](#_Toc70498778)

[5.11. REQWS011. Convert Pay Later Loyalty Point to Cash 36](#_Toc70498779)

[5.11.1. Business Requirements 36](#_Toc70498780)

[5.11.2. Technical Details 36](#_Toc70498781)

[5.11.3. Message Specifications 36](#_Toc70498782)

[5.11.4. Mapping Fields 36](#_Toc70498783)

[5.11.5. Limitations 36](#_Toc70498784)

[6. Requirement Description – Payment Online Channel 37](#_Toc70498785)

[6.1. Payment to BNPL’s debt by making fund transfer via virtual account number 37](#_Toc70498786)

[6.1.1. REQWS012. WooriBank Interface 37](#_Toc70498787)

[6.1.2. REQWS013. BIDV Interface 39](#_Toc70498788)

[6.2. Payment to BNPL’s debt by cash at Vnpost, Payoo, MoMo 39](#_Toc70498789)

[6.2.1. REQWS014 – VNPost Interface 39](#_Toc70498790)

[6.2.2. REQWS015 – Payoo Interface 41](#_Toc70498791)

[6.2.3. REQWS016 – Momo Interface 43](#_Toc70498792)

[6.3. Online payment to BNPL’s debt via Payoo, MoMo 43](#_Toc70498793)

[6.3.1. REQWS017. Online payment via Payoo Interface 43](#_Toc70498794)

[6.3.2. REQWS018. Online payment via MoMo Interface 45](#_Toc70498795)

[7. Requirement Description – Interface with Pay Later System 46](#_Toc70498796)

[7.1. Virtual Account Creation for BNPL Issuing Contract 46](#_Toc70498797)

[7.2. REQWS019. New customer 46](#_Toc70498798)

[7.2.1. Business Requirements 46](#_Toc70498799)

[7.2.2. Technical Details 46](#_Toc70498800)

[7.2.3. Message Specifications 47](#_Toc70498801)

[7.2.4. Mapping Fields 47](#_Toc70498802)

[7.2.5. Limitations 48](#_Toc70498803)

[7.3. REQWS020. Customer has Loan, no Credit card 48](#_Toc70498804)

[7.3.1. Business Requirements 48](#_Toc70498805)

[7.3.2. Technical Details 48](#_Toc70498806)

[7.3.3. Message Specifications 48](#_Toc70498807)

[7.3.4. Mapping Fields 48](#_Toc70498808)

[7.3.5. Limitations 49](#_Toc70498809)

[7.4. REQWS021. Customer has CIF, no Loan, no Credit card 49](#_Toc70498810)

[7.4.1. Business Requirements 49](#_Toc70498811)

[7.4.2. Technical Details 49](#_Toc70498812)

[7.4.3. Message Specifications 49](#_Toc70498813)

[7.4.4. Mapping Fields 50](#_Toc70498814)

[7.4.5. Limitations 50](#_Toc70498815)

[7.5. REQWS022. Customer has Credit card, no Loan 50](#_Toc70498816)

[7.5.1. Business Requirements 50](#_Toc70498817)

[7.5.2. Technical Details 50](#_Toc70498818)

[7.5.3. Message Specifications 51](#_Toc70498819)

[7.5.4. Mapping Fields 51](#_Toc70498820)

[7.5.5. Limitations 51](#_Toc70498821)

[7.6. REQWS023. Customer has Loan and Credit card 51](#_Toc70498822)

[7.6.1. Business Requirements 51](#_Toc70498823)

[7.6.2. Technical Details 51](#_Toc70498824)

[7.6.3. Message Specifications 52](#_Toc70498825)

[7.6.4. Mapping Fields 52](#_Toc70498826)

[7.6.5. Limitations 53](#_Toc70498827)

[7.7. REQWS024. Customer has closed BNPL account 53](#_Toc70498828)

[7.7.1. Business Requirements 53](#_Toc70498829)

[7.7.2. Technical Details 53](#_Toc70498830)

[7.7.3. Message Specifications 53](#_Toc70498831)

[7.7.4. Mapping Fields 54](#_Toc70498832)

[7.7.5. Limitations 55](#_Toc70498833)

[7.8. REQWS025. Create Authorization Transaction 55](#_Toc70498834)

[7.8.1. Business Requirement 55](#_Toc70498835)

[7.8.2. Technical Details 56](#_Toc70498836)

[7.8.3. Message Specification 56](#_Toc70498837)

[7.8.4. Limitations 57](#_Toc70498838)

[7.9. REQWS026. Create Financial Transaction 57](#_Toc70498839)

[7.9.1. Business Requirement 57](#_Toc70498840)

[7.9.2. Technical Details 57](#_Toc70498841)

[7.9.3. Message Specification 57](#_Toc70498842)

[7.9.4. Limitations 58](#_Toc70498843)

[7.10. REQWS027. Reversal Authorization Transaction 58](#_Toc70498844)

[7.10.1. Business Requirement 58](#_Toc70498845)

[7.10.2. Technical Details 58](#_Toc70498846)

[7.10.3. Message Specification 59](#_Toc70498847)

[7.10.4. Limitations 59](#_Toc70498848)

[7.11. REQWS028. Credit Transaction 59](#_Toc70498849)

[7.11.1. Business Requirement 59](#_Toc70498850)

[7.11.2. Technical Details 60](#_Toc70498851)

[7.11.3. Message Specification 60](#_Toc70498852)

[7.11.4. Limitations 60](#_Toc70498853)

[8. Requirement Description – Interface with Agent Desktop 61](#_Toc70498854)

[8.1. REQWS029. Get BNPL Issuing contract info from CIF 61](#_Toc70498855)

[8.1.1. Business Requirements 61](#_Toc70498856)

[8.1.2. Technical Details 61](#_Toc70498857)

[8.1.3. Message Specifications 61](#_Toc70498858)

[8.1.4. Mapping Fields 62](#_Toc70498859)

[8.1.5. Limitations 62](#_Toc70498860)

[8.2. REQWS030. Get client information from BNPL Issuing contract number 62](#_Toc70498861)

[8.2.1. Business Requirements 62](#_Toc70498862)

[8.2.2. Technical Details 63](#_Toc70498863)

[8.2.3. Message Specifications 63](#_Toc70498864)

[8.2.4. Mapping Fields 63](#_Toc70498865)

[8.2.5. Limitations 64](#_Toc70498866)

[8.3. REQWS031. Get BNPL Issuing contract information 64](#_Toc70498867)

[8.3.1. Business Requirements 64](#_Toc70498868)

[8.3.2. Technical Details 64](#_Toc70498869)

[8.3.3. Message Specifications 65](#_Toc70498870)

[8.3.4. Mapping Fields 65](#_Toc70498871)

[8.3.5. Limitations 66](#_Toc70498872)

[8.4. REQWS032. Get BNPL Issuing contract outstanding information 66](#_Toc70498873)

[8.4.1. Business Requirements 66](#_Toc70498874)

[8.4.2. Technical Details 66](#_Toc70498875)

[8.4.3. Message Specifications 67](#_Toc70498876)

[8.4.4. Mapping Fields 67](#_Toc70498877)

[8.4.5. Limitations 68](#_Toc70498878)

[8.5. REQWS033. Get BNPL Issuing contract payment information 68](#_Toc70498879)

[8.5.1. Business Requirements 68](#_Toc70498880)

[8.5.2. Technical Details 68](#_Toc70498881)

[8.5.3. Message Specifications 68](#_Toc70498882)

[8.5.4. Mapping Fields 69](#_Toc70498883)

[8.5.5. Limitations 69](#_Toc70498884)

[8.6. REQWS034. Get BNPL Issuing contract transaction history 69](#_Toc70498885)

[8.6.1. Business Requirements 69](#_Toc70498886)

[8.6.2. Technical Details 69](#_Toc70498887)

[8.6.3. Message Specifications 69](#_Toc70498888)

[8.6.4. Mapping Fields 70](#_Toc70498889)

[8.6.5. Limitations 71](#_Toc70498890)

[8.7. REQWS035. Get BNPL Issuing contract instalment transaction plan 71](#_Toc70498891)

[8.7.1. Business Requirements 71](#_Toc70498892)

[8.7.2. Technical Details 71](#_Toc70498893)

[8.7.3. Message Specifications 72](#_Toc70498894)

[8.7.4. Mapping Fields 72](#_Toc70498895)

[8.7.5. Limitations 73](#_Toc70498896)

[8.8. REQWS036. Get BNPL Issuing contract instalment portion 73](#_Toc70498897)

[8.8.1. Business Requirements 73](#_Toc70498898)

[8.8.2. Technical Details 73](#_Toc70498899)

[8.8.3. Message Specifications 73](#_Toc70498900)

[8.8.4. Mapping Fields 74](#_Toc70498901)

[8.8.5. Limitations 74](#_Toc70498902)

[8.9. REQWS037. Temporary lock/unlock BNPL Issuing contract 74](#_Toc70498903)

[8.9.1. Technical Details 74](#_Toc70498904)

[8.9.2. Message Specifications 75](#_Toc70498905)

[8.9.3. Mapping Fields 75](#_Toc70498906)

[8.9.4. Limitations 75](#_Toc70498907)

[9. Requirement Description – Interface with LOS 76](#_Toc70498908)

[9.1. REQWS038. Create Credit Card Contract Hierarchy (Liability Contract, Issuing Contract and Card Contract) for existing client 76](#_Toc70498909)

[9.1.1. Business Requirements 76](#_Toc70498910)

[9.1.2. Technical Details 76](#_Toc70498911)

[9.1.3. Message Specifications 76](#_Toc70498912)

[9.1.4. Limitations 76](#_Toc70498913)

[9.2. REQWS039. Update Client Information 76](#_Toc70498914)

[9.2.1. Business Requirements 76](#_Toc70498915)

[9.2.2. Technical Details 79](#_Toc70498916)

[9.2.3. Message Specifications 79](#_Toc70498917)

[9.2.4. Limitations 80](#_Toc70498918)

1. History of changes

|  |  |  |  |
| --- | --- | --- | --- |
| version | date | Description | author |
| 0.1 | 26.03.2021 | Initial Version | Le Thi Ngoc Hieu |
| 0.2 | 05.04.2021 | Update new requirements  For Online Channel Interface | Le Thi Ngoc Hieu |
| 0.3 | 14.04.2021 | Update requirement for Agent Desktop Interface  Update requirement for Payment API | Le Thi Ngoc Hieu |
| 0.4 | 15.04.2021 | Update requirement for Agent Desktop | Le Thi Ngoc Hieu |
| 0.5 | 16.04.2021 | Update requirement for Online Channel | Le Thi Ngoc Hieu |
| 0.6 | 26.04.2021 | Update requirement for Payment API and LOS | Le Thi Ngoc Hieu |
| 0.7 | 27.04.2021 | Update requirement from Online Channel and Payment API for Payoo Interface | Le Thi Ngoc Hieu |
| 0.8 | 28.04.2021 | Update API for onboarding for Pay Later  And mapping fields for MerchantID for Online Channel | Le Thi Ngoc Hieu |

**Copyright**

© OpenWay Asia Limited 2021. All rights reserved.

The Copyright of this complete document and every part it belongs to OpenWay Asia Limited. Proprietary material, brand or product names of other parties or trademarks remain with their respective owners. You may not, except with our express written permission, distribute or commercially exploit the content. Nor may you transmit it or store it in any other website or other form of electronic retrieval system. Any sample data used in examples below are completely fictitious unless otherwise noted.

**Disclaimer**

This document and the OpenWay Asia software it describes are furnished by OpenWay Asia Limited under a Software Licensing Agreement, Consultancy Agreement, Variation Request or Confidentiality Agreement, and may be used or copied only in accordance with the terms of such Agreement. Neither this document nor the OpenWay Asia software it describes may be used, sold, transferred, copied, translated, reproduced, or transmitted in any form or by any means, electronic or mechanical, for any purpose, in whole or in part, other than in accordance with the terms of such Agreement, or otherwise without prior written consent of OpenWay Asia Limited.

This document describes a generic product or service and should be read in conjunction with other documents relevant to the configuration of any specific system. The licensee of OpenWay software or user of OpenWay Asia services is responsible for ensuring that the product or service described herein meets its own requirements.

**Confidentiality**

The information contained in this Document is the property of OpenWay Asia Ltd and contains CONFIDENTIAL information that is produced solely for the benefit of the receiving party named on the front page of this document. The recipient should keep this document and all its information confidential. On no account should this document, in whole or in part, be used, sold, transferred, copied, translated, reproduced, or transmitted in any form or by any means, electronic or mechanical, or disclosed or disseminated to any third party, without the express written permission of OpenWay Asia Ltd.

1. Introduction

LFVN requests support for online XML services currently used to allow access to the WAY4 in-order-to manage data in the system to perform various activities related to card management.

In WAY4 the XML online service implementation is called as UFX or Webservice API services and is referred to as the same in the rest of this document.

The following UFX/WS API services will be supported:

Table . UFX/WS API requirements

| Ref | UFX/WS Service Name | Description |
| --- | --- | --- |
| 1 | REQWS001. Get credit limit and available amount | To retrieve credit limit, available amount, contract status and expiry date of BNPL Issuing contract via online channel. |
| 2 | REQWS002. Lock issuing contract | To temporarily lock BNPL Issuing contract via online channel. |
| 3 | REQWS003. Transaction History | To retrieve transaction history of BNPL Issuing contract via online channel. |
| 4 | REQWS004. Get Instalment Portion by PlanID | To retrieve of get instalment portion by PlanID for BNPL Issuing contract via online channel. |
| 5 | REQWS005. Payment history | To retrieve payment history transaction for BNPL Issuing contract via online channel. |
| 6 | REQWS006. Manage statement | To retrieve statement information of BNPL Issuing contract via online channel. |
| 7 | REQWS007. Get merchant list | To provide list of merchant ID which are collaborated with LFVN to provide BNPL product via online channel. |
| 8 | REQWS008. Get client contract hierarchy | To provide contract hierarchy of a client via online channel. |
| 9 | REQWS009. Get BNPL Issuing contract status | To provide BNPL Issuing contract status via online channel. |
| 10 | REQWS010. Get Pay Later Loyalty Point | To retrieve Pay Later Loyalty Point via online channel. |
| 11 | REQWS011. Convert Pay Later Loyalty Point to Cash | To convert Pay Later Loyalty point to cash via online channel. |
| 12 | REQWS012. Payment API – Fund Transfer via Wooribank interface | To make payment to BNPL Issuing contract debt via Wooribank. |
| 13 | REQWS013. Payment API – Fund Transfer via BIDV | To make payment to BNPL Issuing contract debt via BIDV. |
| 14 | REQWS014. Payment API – Payment by Cash at VNPost | To make payment to BNPL Issuing contract debt by cash at VNPost. |
| 15 | REQWS015. Payment API – Payment by Cash at Payoo | To make payment to BNPL Issuing contract debt by cash at Payoo. |
| 16 | REQWS016. Payment API – Payment by Cash at MoMo | To make payment to BNPL Issuing contract debt by cash at MoMo. |
| 17 | REQWS017. Payment API – Online Payment via Payoo | To make online payment to BNPL Issuing contract via Payoo interface. |
| 18 | REQWS018. Payment API – Online Payment via MoMo | To make online payment to BNPL Issuing contract via MoMo interface. |
| 19 | REQWS019. New Customer | To create new Client, Liability contract, and Issuing contract in WAY4 via Pay Later. |
| 20 | REQWS020. Customer has Loan, no Credit card | To create new Client, Liability contract, Issuing contract in WAY4 via Pay Later. |
| 21 | REQWS021. Customer has CIF, no Loan, no Credit card | To create new Client, Liability contract, and Issuing contract in WAY4 via Pay Later. |
| 22 | REQWS022. Customer has Credit card, no Loan | To create new Liability contract and Issuing contract in WAY4 via Pay Later. |
| 23 | REQWS023. Customer has Loan and Credit card | To create new Liability contract and Issuing contract in WAY4 via Pay Later. |
| 24 | REQWS024. Customer has closed BNPL account | To create new Issuing contract for existing Liability contract in WAY4 via Pay Later. |
| 25 | REQWS025. Create Authorization Transaction | To create authorization transaction for BNPL Issuing contract in WAY4 via Pay Later. |
| 26 | REQWS026. Create Financial Transaction | To create financial transaction for BNPL Issuing contract in WAY4 via Pay Later. |
| 27 | REQWS027. Reversal Authorization Transaction | To create reversal authorization transaction in WAY4 via Pay Later. |
| 28 | REQWS028. Reversal Financial Transaction | To create financial transaction in WAY4 via Pay Later. |
| 29 | REQWS029. Get BNPL Issuing contract from CIF | To get BNPL Issuing contract from Client Number via Agent Desktop. |
| 30 | REQWS030. Get Client Information from BNPL Issuing contract number | To get client information from BNPL Issuing contract number via Agent Desktop. |
| 31 | REQWS031. Get BNPL Issuing contract information | To get BNPL Issuing contract information via Agent Desktop. |
| 32 | REQWS032. Get BNPL Issuing contract outstanding information | To get BNPL Issuing contract outstanding information via Agent Desktop. |
| 33 | REQWS033. Get BNPL Issuing contract payment information | To get BNPL Issuing contract payment information via Agent Desktop. |
| 34 | REQWS034. Get BNPL Issuing contract history | To get BNPL Issuing contract transaction history via Agent Desktop. |
| 35 | REQWS035. Get BNPL Issuing contract instalment transaction plan | To get BNPL Issuing contract instalment transaction plan via Agent Desktop. |
| 36 | REQWS036. Get BNPL Issuing contract instalment portion | To get BNPL Issuing contract instalment portion via Agent Desktop. |
| 37 | REQWS037. Temporary lock/unlock BNPL Issuing contract | To lock/unlock BNPL Issuing contract via Agent Desktop. |
| 38 | REQWS038. Create Credit Card Contract Hierarchy | To create credit card contract hierarchy for existing client via Agent Desktop. |
| 39 | REQWS039. Update client information | To update client information via Agent Desktop. |

* 1. Notations

Requirement notation is used for identifying requirements: REQWSxxx – SOA/WS Service

Notations used in this document are listed in the table below.

Table . Notations

| Notation | Description |
| --- | --- |
| OPW | Openway Vietnam |
| ESI | ESI Vietnam |
| BNPL | Buy Now Pay Later |
| LFVN | LFVN Finance Vietnam |
| LOS | Loan Origination System |
| WS | Webservice |
| HSM | Hardware Security Module |
| RRN | Retrieval Reference Number |
|  |  |
|  |  |
|  |  |

* 1. Service Overview

UFX/WS API services will be used as an interface between the WAY4 system and the outside world. For each UFX/WS API service, a UFX/WS API service request message is constructed by OpenWay and provided to external clients.

The diagram below indicates the global flow in the network:

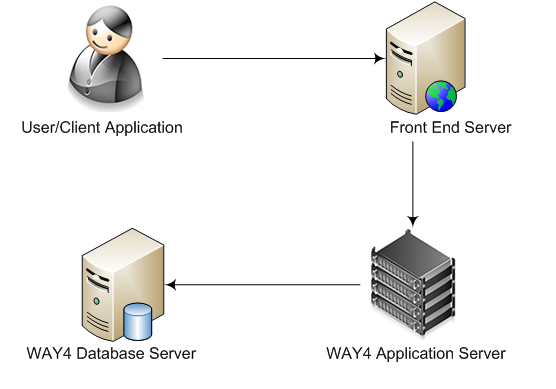


Figure .WAY4 - UFX/WS API service Network

The flow is as follows:

* 1. A message/request is sent by the user/client application to LFVN’s front-end server
  2. LFVN’s front-end server is responsible to authenticate the request and perform all security checks to ensure that the message is valid and is originating from the expected source. Once the Security checks have passed, the call will be passed onto the WAY4 Application Server which will host the WAY4 UFX/WS API services.
  3. The WAY4 Application server verifies the request and transmits the request to the WAY4 database for data retrieval/processing.

The WAY4 Database server process the request and return the requested information back to the WAY4 Application server to be routed to the user/client application

1. General Details and Methodology
   1. Security Check Requirement

LFVN’s security requirements for UFX/WS API services will be managed by LFVN and will not be in the scope for the current project. This includes any validation of the connection, authentication method and user or password validation.

* 1. Request/Response Communication Method

The UFX/WS API messages or services in WAY4 support the HTTPS POST messages for receiving all incoming service requests. The WAY4 application server can process these messages asynchronously. For example, it is possible for a requestor of a UFX service to send a second request without waiting for the response of the first request.

* 1. Message Authentication

The mechanism of authenticating WAY4 UFX/WS API messages is based on a symmetric cryptographic function, the MAC (Message Authentication Code). It is used if both the receiver and the sender of the message have hardware security modules (such as an HSM, a PIN pad, or a smartcard) and have established a key management scheme.

* 1. Application Layer

Every application layer message must have its own format. Currently, formats for representing the following service types have been implemented in the system:

* **Financial:** Allows retrieval of financial information like mini statement, statement (of various types), additional services (example: PIN set and PIN change) and executing standing order requests. The Doc element specifies the structure of a message object.
* **Information:** Allows non-financial information on system objects such as "Contract" or "Client" to be required by their unique identifiers. The Information element specifies the structure of a message object. An object identifier is contained in the ObjectFor sub-element. Information about an object is contained in the DataRs sub-element of a response message.
* **Application:** Allows modification of non-financial objects such as "Contract" or "Client" and so on to be required. The Application element specifies the structure of a message object. Responses may contain either data received as a result of request processing or an indication of the request being accepted for processing. In the latter case, it is necessary to make an additional request to receive the required data. To specify a type of a request, the ResultDtls sub-element is used.
  1. Request Message ID

When a message is sent, a unique id for the message is created by the requestor, identifying the service request and it is recorded in the system of the requestor. This message ID is part of the header information for all UFX/WS API service requests which are received by WAY4. The requestor should put processes in place to ensure that the message ID generated is unique. This is because the same message ID will be sent as an echo field by WAY4 in the response message. The requestor can use this message ID to match the request to the response. It is recommended to generate value for this element basing on a "Universally Unique Identifier (UUID) URN Namespace", RFC 4122 specification.

* 1. RRN

RRN Stand for Retrieval Reference Number, which is unique database number of the original transaction corresponding to a document; when a sequence of documents is created e.g., in a dispute cycle or in case of a decline, WAY4 assigns the same RRN of the original transaction to each document in the sequence.

RRN must be generate base on Openway standard formular as below:

Formula: YDDDSSNNNNNN,

* Y reflects the current year (last digit)
* DDD is the number of the day in the year (sometimes called the Julian date, although this is not quite correct)
* SS is the service identifier of the resource. In SOA this is the ID of WAY4GATE instance.
* NNNNNN is 6 digits of the cycle counter value.
  1. Error Handling

Application errors occur when information included in the submitted request is not valid. In such a case, an error code is set up and error relative information is returned by the service.

An exception is an error occurring independently from application controls, for example a timeout. In this case, there is no return information.

When the UFX response message has been received, the result of the request processing must be analyzed by <UFXMsg> aggregate’s attributes resp\_class and resp\_code. The resp\_code attribute must be analyzed basing on response codes listed below. The list provided below is the complete set of error codes. Some of these error codes might not be used in the implementation of LFVN as they might refer to messages which have not been implemented for LFVN. But it is best for the requestor to implement processes that take care of all the error codes in the response message.

|  |  |  |  |
| --- | --- | --- | --- |
| Code | Type | Message | Comments |
| Common Response Code | | | |
| 0 | Info | Success | The service provider successfully processed the request. |
| 100 | Error | General Error | There was an error that prevented the service provider from processing the transaction. No additional information is provided. |
| 300 | Error | System Not Available | The service provider for this transaction is not available due to a technical problem. Try again later. |
| 400 | Error | Function Not Available | The function selected is not available. Other functions may be available. |
| 500 | Error | Unsupported Service | The service provider does not support the specified service offering. |
| 600 | Error | Unsupported Message | The server does not support the message. |
| 700 | Error | Requested function not supported | Requested function not supported. |
| 1000 | Error | Duplicate Request Identifier | A request with this client message identifier has already been received and processed. |
| 1005 | Error | Incorrect format of parameter: .. | The format of a parameter is not valid |
| 1020 | Error | Required Element Not Included | The request message does not contain one or more required elements. |
| 1050 | Error | Invalid Enum Value | Customer input of an Enum value is not valid. The condition may echo the element name in the error or the value input by the Customer. |
| 1172 | Error | Financial institution not found | Financial institution not found |
| 1200 | Error | General Data Error | One or more of the elements in the request is either invalid or inconsistent with other elements. No additional information is available. |
| 1220 | Error | Invalid Identifier | The reference identifier used is invalid. |
| 1700 | Error | Security Violation | A security violation has occurred. |
| 1740 | Error | Incorrect PIN | Incorrect PIN |
| 1760 | Error | Authorization Failure | The customer is not eligible to perform this function |
| 1770 | Error | Validation Failure | Security values specified by Customer are not correct |
| -500 | Error | Internal Error | System internal error |
| -1000 | Error | Message format error | The format of the received message is inconsistent with the specified version of the WAY4 UFX format. |
| -1005 | Error | Incorrect format of parameter: .. | The format of a parameter is not valid |
| -1010 | Error | Incorrect length(format) of parameter | Format of message is correct formally, but specified element value has incorrect length or format. |
| -1100 | Error | Message sequence error | Unexpected message received in business process workflow |
| -1200 | Error | Message target not found | Abstract message target not found. |
| -1500 | Error | Message Timeout | Message has timed out. |
| -2000 | Error | Policy Error | Policy error |
| Financial Response Code | | | |
| 1 | Error | Refer to card issuer | Refer to card issuer |
| 2 | Error | Refer to card issuer's special condition | Refer to card issuer's special condition |
| 3 | Error | Invalid merchant / source | Invalid merchant / source |
| 4 | Error | PICK UP | PICK UP |
| 5 | Error | Do not Honour | Do not Honour |
| 6 | Error | Error | Error |
| 7 | Error | Pick-up card, special condition | Pick-up card, special condition |
| 8 | Warning | Honour with identification | Honour with identification |
| 9 | Error | Request in progress | Request in progress |
| 10 | Warning | Approved for partial amount | Approved for partial amount |
| 11 | Error | Approved (VIP) | Approved (VIP) |
| 12 | Error | Invalid transaction | Invalid transaction |
| 13 | Error | Invalid amount | Invalid amount |
| 14 | Error | No such card | No such card |
| 15 | Error | No such issuer | No such issuer |
| 16 | Error | Approved, update track 3 | Approved, update track 3 |
| 17 | Error | Customer cancellation | Customer cancellation |
| 18 | Error | Customer dispute | Customer dispute |
| 19 | Error | Re-enter transaction | Re-enter transaction |
| 20 | Error | Invalid response | Invalid response |
| 21 | Error | No action taken | No action taken |
| 22 | Error | Suspected malfunction | Suspected malfunction |
| 23 | Error | Unacceptable transaction fee | Unacceptable transaction fee |
| 24 | Error | File update not supported by receiver | File update not supported by receiver |
| 25 | Error | No such record | No such record |
| 26 | Error | Duplicate record update, old record replaced | Duplicate record update, old record replaced |
| 27 | Error | File update field edit error | File update field edit error |
| 28 | Error | File locked out while update | File locked out while update |
| 29 | Error | File update error, contact acquirer | File update error, contact acquirer |
| 30 | Error | Format error | Format error |
| 31 | Error | Issuer signed-off | Issuer signed-off |
| 32 | Error | Completed partially | Completed partially |
| 33 | Error | Pick-up, expired card | Pick-up, expired card |
| 34 | Error | Pick-up, suspected fraud | Pick-up, suspected fraud |
| 35 | Error | Pick-up, card acceptor contact acquirer | Pick-up, card acceptor contact acquirer |
| 36 | Error | Pick up, card restricted | Pick up, card restricted |
| 37 | Error | Pick up, call acquirer security | Pick up, call acquirer security |
| 38 | Error | Pick up, Allowable PIN tries exceeded | Pick up, Allowable PIN tries exceeded |
| 39 | Error | No credit account | No credit account |
| 40 | Error | Requested function not supported | Requested function not supported |
| 41 | Error | Pick up, lost card | Pick up, lost card |
| 42 | Error | No universal account | No universal account |
| 43 | Error | Pick up, stolen card | Pick up, stolen card |
| 44 | Error | No investment account | No investment account |
| 50 | Warning | Do not renew | Do not renew |
| 51 | Error | Not sufficient funds | Not sufficient funds |
| 52 | Error | No chequing account | No chequing account |
| 53 | Error | No savings account | No savings account |
| 54 | Error | Expired card / target | Expired card / target |
| 55 | Error | Incorrect PIN | Incorrect PIN |
| 56 | Error | No card record | No card record |
| 57 | Error | Transaction not permitted to cardholder | Transaction not permitted to cardholder |
| 58 | Error | Transaction not permitted to terminal | Transaction not permitted to terminal |
| 59 | Error | Suspected fraud | Suspected fraud |
| 60 | Error | Card acceptor contact acquirer | Card acceptor contact acquirer |
| 61 | Error | Exceeds withdrawal amount limit | Exceeds withdrawal amount limit |
| 62 | Error | Restricted card | Restricted card |
| 63 | Error | Security violation | Security violation |
| 64 | Error | Wrong original amount | Wrong original amount |
| 65 | Error | Exceeds withdrawal frequency limit | Exceeds withdrawal frequency limit |
| 66 | Error | Call acquirers security department | Call acquirers security department |
| 67 | Error | Card to be picked up at ATM | Card to be picked up at ATM |
| 68 | Error | Response received too late | Response received too late |
| 75 | Error | Allowable number of PIN tries exceeded | Allowable number of PIN tries exceeded |
| 76 | Error | Wrong PIN, number of PIN tries exceeded | Wrong PIN, number of PIN tries exceeded |
| 77 | Error | Wrong Reference No. | Wrong Reference No. |
| 78 | Error | Record Not Found | Record Not Found |
| 79 | Error | Already reversed | Already reversed |
| 80 | Error | Network error | Network error |
| 81 | Error | Foreign network error / PIN cryptographic error | Foreign network error / PIN cryptographic error |
| 82 | Error | Time-out at issuer system / Bad CVV (VISA) | Time-out at issuer system / Bad CVV (VISA) |
| 83 | Error | Transaction failed | Transaction failed |
| 84 | Error | Pre-authorization timed out | Pre-authorization timed out |
| 85 | Warning | No reason to decline | No reason to decline |
| 86 | Error | ULFVNle to validate PIN | ULFVNle to validate PIN |
| 88 | Error | Cryptographic failure | Cryptographic failure |
| 89 | Error | Authentication failure | Authentication failure |
| 90 | Error | Cutoff is in progress | Cutoff is in progress |
| 91 | Error | Issuer or switch is inoperative | Issuer or switch is inoperative |
| 92 | Error | ULFVNle to route at acquirer module | ULFVNle to route at acquirer module |
| 93 | Error | Cannot be completed, violation of law | Cannot be completed, violation of law |
| 94 | Error | Duplicate Transmission | Duplicate Transmission |
| 95 | Error | Reconcile error / Auth Not found | Reconcile error / Auth Not found |
| 96 | Error | System Malfunction | System Malfunction |
| 101 | Error | Chain not found | Chain not found |
| 102 | Error | Incorrect Chain | Incorrect Chain |
| 111 | Error | Card BIN not on file | Card BIN not on file |
| 112 | Error | Card type not in service for this device | Card type not in service for this device |
| 113 | Error | Invalid operation for this card | Invalid operation for this card |
| 115 | Error | Requested function not supported | Requested function not supported |
| 117 | Error | Suspicious Transaction | Suspicious Transaction |
| 119 | Error | Card BIN not in service for this device | Card BIN not in service for this device |
| 120 | Error | Card not in service for this device | Card not in service for this device |
| 121 | Error | Repeat | Repeat |
| 122 | Error | Previous doc not found | Previous doc not found |
| 123 | Error | Invalid reversal amount | Invalid reversal amount |
| 124 | Error | Capture period expired | Capture period expired |
| 125 | Error | Invalid capture amount | Invalid capture amount |
| 126 | Error | Invalid PIN block format | Invalid PIN block format |
| 128 | Error | No communication keys available for use | No communication keys available for use |
| 129 | Error | Operation key buffer error | Operation key buffer error |
| 130 | Error | Invalid Terminal ID | Invalid Terminal ID |
| 131 | Error | Wrong Transaction Attributes | Wrong Transaction Attributes |
| 132 | Error | Unmatched Transaction Condition | Unmatched Transaction Condition |
| 142 | Error | Destination Channel has deceased | Destination Channel has deceased |
| 143 | Error | Check request declined by Billing Channel | Check request declined by Billing Channel |
| 144 | Error | Payment request declined by Billing Channel | Payment request declined by Billing Channel |
| 160 | Error | Device Hardware/Software Error | Device Hardware/Software Error |
| 161 | Error | Wrong device status | Wrong device status |
| 162 | Error | Unknown status message | Unknown status message |
| 163 | Error | HSM Response error | HSM Response error |
| 164 | Error | Command rejected by device | Command rejected by device |
| 165 | Error | Specific or MAC command rejected by device | Specific or MAC command rejected by device |
| 166 | Error | Command aborted | Command aborted |
| 170 | Error | Message Authentication Key not defined | Message Authentication Key not defined |
| 171 | Error | Message Authentication Field Missing | Message Authentication Field Missing |
| 172 | Error | MAC verification Error | MAC verification Error |
| 173 | Error | MAC Generation error | MAC Generation error |
| 174 | Error | Security Hardware/Software error | Security Hardware/Software error |
| 175 | Error | Security Module Channel Timeout | Security Module Channel Timeout |
| 176 | Error | Link is inactive. Device is not connected | Link is inactive. Device is not connected |
| 177 | Error | Device is not in transaction | Device is not in transaction |
| 178 | Error | Device is already in transaction | Device is already in transaction |
| 179 | Error | Device response timed out | Device response timed out |
| 180 | Error | Amount is too small. Dispense not possible | Amount is too small. Dispense not possible |
| 181 | Error | Amount is too big. Dispense not possible | Amount is too big. Dispense not possible |
| 182 | Error | Amount has cent's | Amount has cent's |
| 183 | Error | Dispense not possible | Dispense not possible |
| 185 | Error | Invalid authorization amount | Invalid authorization amount |
| 188 | Error | The Cardholder has not taken MONEY | The Cardholder has not taken MONEY |
| 189 | Error | Non working time for this device | Non working time for this device |
| 190 | Error | Device not configured or not valid | Device not configured or not valid |
| 191 | Error | Device contract not valid | Device contract not valid |
| 192 | Error | Device not on file | Device not on file |
| 193 | Error | Requested operation not on file for this device | Requested operation not on file for this device |
| 194 | Error | Operation is disabled for this device | Operation is disabled for this device |
| 195 | Error | Currency is not available for device | Currency is not available for device |
| 196 | Error | Internal System Malfunction | Internal System Malfunction |
| 198 | Error | Track 2 Format Error | Track 2 Format Error |
| 199 | Error | PIN Block Conversion Error | PIN Block Conversion Error |
| 200 | Error | The Cardholder has not taken his Card | The Cardholder has not taken his Card |
| 203 | Error | Source contract expired | Source contract expired |
| 254 | Error | Merchant card expired | Merchant card expired |
| 257 | Error | Merchant card contract has not been approved | Merchant card contract has not been approved |
| 258 | Error | Device contract is not on file | Device contract is not on file |
| 261 | Error | Device amount limits exceeded | Device amount limits exceeded |
| 262 | Error | Rejected Some Documents in this Batch | Rejected Some Documents in this Batch |
| 265 | Error | Device frequency limits exceeded | Device frequency limits exceeded |
| 361 | Error | Merchant card amount limits exceeded | Merchant card amount limits exceeded |
| 365 | Error | Merchant card frequency limits exceeded | Merchant card frequency limits exceeded |
| 371 | Warning | Requested period exceeds the limit of 10 years | Requested period exceeds the limit of 10 years. This results in loss of some entries. |

Other error codes from requesting application are available inside document CBS\_XML\_Applications\_R2.pdf Chapter 2 in WAY4 manual.

In cases the UFX request message cannot be processed due to the message format or internal system error, the UFX response message will contain the same body as the request message (the same contents of the aggregate <UFXMsg>). Additionally, errors specific for the transport level, as HTTP errors are out of scope and need to be handled as such by the requestor of the service.

* 1. Institution ID – Access to Data

Based on the Institution ID input parameter the service request must be validated to ensure that the service is requested for a card or merchant belonging to the same institution as the institution ID provided in the message.

The input parameters of all UFX/WS API services will be validated to ensure that they do not try to access entities which do not belong to their own institution.

**Example A**

WAY4: Card 1 in Institution A

UFX service: Update Card Status

UFX service Parameters: Institution B, Card 1

Result: Error, since Card 1 does not belong to Institution B

**Example B**

WAY4: Card 2 in Institution C

UFX service: Update Card Status

UFX service Parameters: Institution C, Card 2

* 1. User Authentication

At the time of processing a UFX/WS API service request the LFVN front-end server must validate the user that is requesting the service. Since these users are not defined on WAY4 (as WAY4 is only a data provider) it is not possible for WAY4 to validate the user.

WAY4 on the other hand uses a fixed username and password (which has been defined as an Oracle user and a WAY4 user) to access the database to perform the function as requested by the incoming UFX/WS API message. This same user is also logged in the change log of WAY4. The integration layer will log the actual client that called the UFX/WS API service and WAY4 will log the change under the database user that it uses to access the WAY4 database.

* 1. Items marked as “To Be Determined” (TBD)

Note that all items in the report marked as TBD will need to be defined during the implementation of the project. Changes provided will be given by following the standard OpenWay variation order procedure. If no updates are given for any items marked as TBD, then WAY4 will be delivered with the standard WAY4 functionality/configuration to satisfy the requirement.

1. Requirement Description – Interface with Online Channel
   1. REQWS001. Get credit limit and available amount

### Business Requirements

This function allows Online Channel sends request to WAY4 to retrieve BNPL Issuing contract information.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Issuing contract number |
| Approved credit limit |
| Available amount |
| Expiry date |
| Contract status |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve Issuing contract number, approved credit limit, available amount, expiry date, contract status of BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Issuing contract number | ContractNumber |
| Credit limit | CreditLimit |
| Available amount | Balance |
| Status of contract | ExternalCode |
| Expired date | AddInfo02 |

The possible BNPL Issuing contract status is described in 5.10.4.

Online Channel does not want to display contract with status is ‘ACC WRITEOFF’.

### Limitations

* 1. REQWS002. Lock Issuing Contract

### Business Requirements

This function allows Online Channel to lock BNPL Issuing contract information. Only Issuing contract with status is ‘ACC OK’ is allowed to be locked. WAY4 does not check this condition, Online channel needs to make sure that BNPL Issuing contract is in ‘ACC OK’ status before deciding to call API ‘SetContractStatus’ to set contract status to locked.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Issuing contract is locked successfully. |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To temporarily lock BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Issuing contract is locked successfully | <RetCode>0</RetCode>  <RetMsg>I Successfully Completed</RetMsg> |

### Limitations

* 1. REQWS003. Transaction History

### Business Requirements

This function allows Online Channel to retrieve transaction history for BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number  Date From  Date To | Transaction datetime |
| Merchant ID |
| RRN |
| Transaction amount |
| Transaction status |
| Payment method |
| Instalment tenor (if payment method is instalment) |
| Instalment PlanID (used to retrieve instalment portion details in the next requirement). |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve transaction history of BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Transaction Date Time | TransDate: YYYY-MM-DD HH:MM:SS (Remove character ‘T’ between Date and Time if necessary). |
| MerchantID | Merchant |
| RRN | RetRefNumber |
| Transaction status | ResponseCode. If ResponseCode is ‘00’ then transaction status is ‘Successful’, else transaction status is ‘Fail’. |
| Transaction amount | TransAmount |
| Payment method | AddInfo tag ‘INST\_TN’. If value of this tag is equal to ‘1’ then payment method is onetime payment; if value is equal to ‘2’ or ‘3’ then payment method is instalment. |
| Instalment tenor | Value of tag ‘INST\_TN’. Only display this field, if payment method is instalment. |
| Instalment PlanID | Value of tag ‘INST\_PLAN\_ID’ in field AddInfo. This tag value only appears incase transaction is settled and converted to instalment successfully. |

Mapping MerchantID with merchant name:

|  |  |
| --- | --- |
| **Mapping MerchantID with merchant name:** | |
| **MerchantID** | **Merchant Name** |
| 000001 | VNTrip |
| 000002 | Sendo |

### Limitations

* 1. REQWS004. Get Instalment Portion by PlanID

### Business Requirements

This function allows Online Channel to retrieve all instalment portion for an Instalment plan (PlanID is retrieved from Instalment PlanID which is retrieved from 5.3.4) for BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Instalment plan ID | Portion Number |
| Portion Amount |
| Fee Amount |
| Paid Amount |
| Due Date |
| Status |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve all instalment portion for an instalment plan for BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Portion Number | PortionNumber |
| Portion Amount | PortionAmount |
| Fee Amount | FeeAmount |
| Paid Amount | PaidAmount |
| Due Date | DueDate |
| Status | Status |

|  |  |
| --- | --- |
| **Portion Instalment Status** | |
| Waiting | The instalment has not yet become effective; all records have this status when an instalment plan is generated. |
| Inactive | Instalment of an inactive loan. |
| Open | The instalment is effective but not paid. |
| Paid/  Partially Paid" | The instalment is paid or partially paid, respectively. |
| Overdue | The instalment is overdue. |
| Closed | Instalment status set when an original transaction is reversed. |
| Written Off | The instalment has been written off. |
| Revised | Status set when restructuring an instalment plan. |
| Waived | Payment of principal waived. |

### Limitations

* 1. REQWS005. Payment History

### Business Requirements

This function allows Online Channel to retrieve all payment transaction history for BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number  Date From  Date To | Transaction date time |
| Transaction amount |
| Payment channel (payment transaction type) |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve payment history of BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components

#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Transaction Date Time | TransDate: YYYY-MM-DD HH:MM:SS (Remove character ‘T’ between Date and Time if necessary). |
| Payment channel | TransType |
| Transaction amount | TransAmount |

|  |  |  |
| --- | --- | --- |
| **BNPL Payment Transaction Mapping** | | |
| **Payment Channel** | **Transaction Type** | **Transaction Type IDT** |
| Payoo | Payment To Client Contract Payoo | PT\_4 |
| VNPost | Payment To Client Contract VNPost | PT\_6 |
| Wooribank | Payment To Client Contract WooriBank | PT\_3 |
| By LFVN Operator | Payment To Client Contract | PT\_1\_1\_1\_1 |

### Limitations

* 1. REQWS006. Manage Statement

### Business Requirements

This function allows Online Channel to retrieve statement information for BNPL Issuing contract and list of transaction history.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number  Date From  Date To | Approved credit limit amount |
| Outstanding amount |
| Min Payment amount |
| Due date |

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number  Date From  Date To | Transaction date time |
| Merchant ID |
| Transaction amount |
| Total transaction amount. |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve statement information and transaction history of BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Credit limit amount | Field ‘BillingDetails’ – tag ‘CR\_LIMIT’ |
| Outstanding amount | StatementBalance |
| Min payment amount | DueAmount |
| Due Date | DueDate |

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Transaction Date | TransDate: YYYY-MM-DD HH:MM:SS (Remove character ‘T’ between Date and Time if necessary). |
| MerchantID | Merchant |
| Transaction amount | TransAmount |
| Total transaction amount | Online Channel makes sum of above transaction amount |

### Limitations

* 1. REQWS007. Get Merchant Lists

### Business Requirements

This function allows to retrieve the list of Merchant Name which are collaborated with LFVN to provide BNPL product. Online channel requests this function so that any update in WAY4 core about merchant information also can be retrieved from Online channel. Online channel will not have to update itself.

To be convenient for Online Channel to get the list of merchants by calling only one API, all the merchants should be configured as an acquiring contract under only one commercial client and input of the API should be the commercial client ID (which is saved in WAY4 core database). Please pay attention to use the correct Client ID of acquirer from database before use.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Client ID | MerchantID; MerchantName (Addinfo01) |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To provide list of merchant information which are collaborated with LFVN to provide BNPL product via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

To get list of merchant from a ClientID, Online channel need to filter API’s response by **‘<ProductCategory>M;Acquiring</ProductCategory>**’ and **‘<ContractCategory>A;Account</ContractCategory>**’.

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| MerchantID;MerchantName | AddInfo01 |

### Limitations

* 1. REQWS008. Get Client Contract Hierarchy

### Business Requirements

This function allows to retrieve the contract hierarchy tree of a client from client number (CIF).

|  |  |
| --- | --- |
| **Input** | **Output** |
| Client Number (CIF) | Issuing contract number |
| Open date |
| Expiry date |
| Approval Credit Limit Amount |
| Contract status |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To provide the contract hierarchy of a client via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

BNPL Issuing contract information can be retrieved from the content of each tag **<IssContractDetailsAPIOutputV2Record> … </IssContractDetailsAPIOutputV2Record>** of the API response by filtering filed ‘Product’ with value ‘**001-BNPL Issuing Contract’**. The mapping fields are described in the below table:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| BNPL Issuing contract number | ContractNumber |
| Open date | OpenDate |
| Expiry date | OpenDate + 12 months |
| Approved credit limit amount | CreditLimit |
| Contract status | Status |

### Limitations

* 1. REQWS009. Get BNPL Issuing contract status

### Business Requirements

This function allows to retrieve BNPL Issuing contract status from Issuing contract number.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Contract status |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve contract status of BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Status of contract | ExternalCode |

|  |  |  |
| --- | --- | --- |
| **Issuing contract status** | | |
| **ExternalCode** | **Name** | **Meaning** |
| A00 | ACC OK | Account is okay |
| A19 | ACC LOCK BY OVD | Block automatically account when customer has late payment in 1-90 days |
| A06 | ACC RESERVE | Block automatically forever when customer has late payment more than 90 days |
| A18 | ACC RETURNED | Customer requests to block the account temporary |
| A04 | ACC CLOSED | Customer request to close the account forever |
| A05 | ACC FRAUD | LFVN automatically block the account due to fraud suspicion base on risk rule |
| A20 | ACC SUSPECTED FRAUD | LFVN manual block the account due to fraud suspicion base on risk rule |
| A22 | ACC EXPIRED | Block account forever when expired |
| 54 | Account to close | Customer request to close the account forever, this is temporary status to check account balance |
| A21 | ACC WRITEOFF | Status is changed when writeoff process is called |

### Limitations

* 1. REQWS010. Get Pay Later Loyalty Point

### Business Requirements

This function allows to retrieve Pay Later Loyalty active point of BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Active point amount |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To query Pay Later Loyalty active point of BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Active point amount | Balance |

### Limitations

* 1. REQWS011. Convert Pay Later Loyalty Point to Cash

### Business Requirements

This function allows to convert Pay Later Loyalty active point to cash. This function is implemented by LFVN IT team (Mr. Nguyen Ngoc Khanh). Openway does not need to do anything.

### Technical Details

### Message Specifications

#### API components

#### Request Message template

#### Response Message template

### Mapping Fields

### Limitations

1. Requirement Description – Payment Online Channel
   1. Payment to BNPL’s debt by making fund transfer via virtual account number

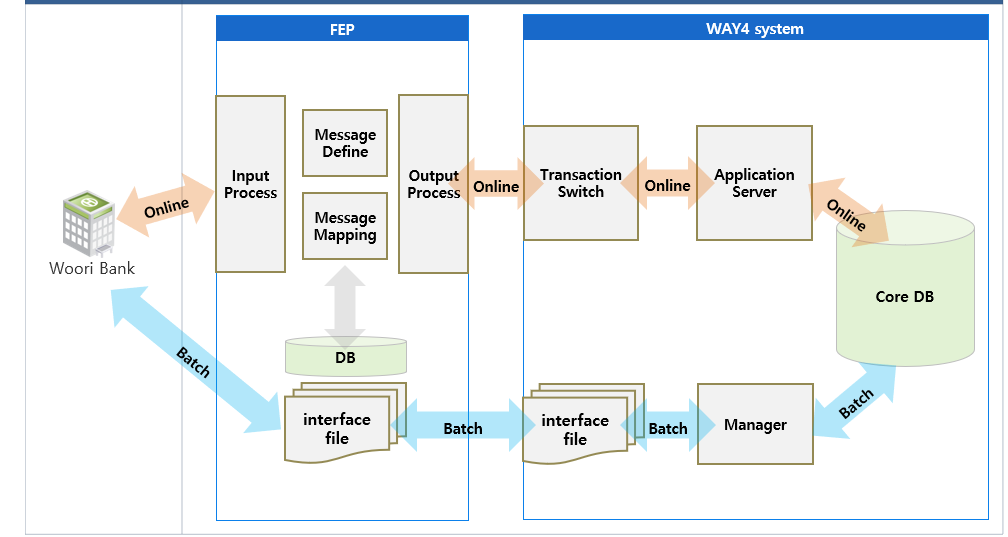
### REQWS012. WooriBank Interface

#### Business Requirements

LFVN needs both batch interface and online interface (web service interface) with Woori Bank for below functions:

|  |  |  |
| --- | --- | --- |
| **ID** | **Function** | **Type of Interface** |
| 1 | Virtual account deposit (HT001) | on-line |
| 2 | Virtual account specification information (MN001) | on-line |
| 3 | Manual Refund (PR001) | on-line |
| 4 | Recipient inquiry (NQ001) | on-line |
| 5 | Inquiry transfer results (TR001) | on-line |
| 6 | Virtual account issue information (BA01) | batch |
| 7 | Changing virtual account conditions (BA02) | batch |
| 8 | Reconcile (BA03) | batch |

Interface Diagram



All above functions are used to support Payment from Virtual Account and Manual Refunds transactions.

**The business requirements of Payment from Virtual Accounts as below:**

1. LFVN opens one virtual account in Woori Bank
2. LFVN has the contract with Woori Bank to open the list of Virtual Account (e.g., 100,000 accounts)
3. Way4 loads the file of above accounts (including Bank\_code, Virtual\_Account). (BA01) which must be saved in a new table. A new screen should display the content of that table.
4. When an Issuing contract is created in WAY4, a WooriBank virtual account will be attached as a contract parameter for this Issuing contract. One WooriBank virtual account is only used for one BNPL issuing contract and cannot be reused.
5. Way4 sends notification (SMS/Email) after an Issuing Contract is created and linked to a WooriBank Account Number. SMS/Email notification template should be provided by LFVN and described in Discovery Report Vol 16.
6. EOD way4 sends the file (BA02) back to Woori bank (Virtual Account, Client Name) (only for the accounts assigned in the day). No need to separate virtual account for credit card and BNPL in BA02 file.
7. Virtual Account is shown in cardholder’s statement.
8. Cardholder can make Credit/Payment transaction to Virtual Account at Woori Bank.
9. Woori Bank sends a real time message to Way4 by information (Virtual Account, Amount) (HT001). In the message, there is a sequence number. Way4 needs to check the sequence number of current transactions with sequence number of previous transaction (Transaction serial number field in HT001) after processing this request message successfully. If previous one + 1 <> current one, then we send the Virtual Account specification information (MN001) to Woori. If current sequence number = 1 then Way4 does not need to check with previous transaction. Because it is beginning number.
10. Way4 will receive the message and map to issuing contract to do the Payment.
11. Way4 sends SMS/Email notification for Payment transaction. SMS/Email notification template should be provided by LFVN and described in Discovery Report Vol 16.
12. When Woori bank runs EOD then Woori will send file (BA03) for reconciliation. The batch file format will be kept the same as current configuration in LFVN production. Openway needs to customize so that with the virtual account given in batch file, WAY4 should find the corresponding Issuing contract number for Credit card and the corresponding Wooribank virtual account for BNPL Issuing contract.

* LFVN needs a screen to monitor daily transaction including the information (Transaction date, Virtual Account, Transaction Amount, Issuing contract).
* LFVN need one transaction type for this Payment (e.g., Woori Bank Payment to Client Contract)

**LFVN also need a customization form for Manual Refund transaction:**

There are fields to show input bank code, input bank account, current balance of cardholder (read only), input amount and input text.

There are 3 buttons: Inquiry; Refund; Result.

Bank staff will key-in an input bank code, input bank account, client name and press the button Inquiry to send the message (NQ001) to Wooribank to validate the input bank account.

If WAY4 receives the validation from Wooribank, then allows cardholder to key-in refund amount; to press the button Refund; and then WAY4 system process as below:

* Calculate Refund amount = current balance of cardholder + input amount
* System check if Refund amount > 0 or not?

WAY4 system generates the Payment/Cash back transaction to cardholder (transaction amount =input amount) and balance of cardholder will be updated to current balance + input amount. It means that we recover the balance before we refund.

WAY4 sends real time message (PR001) to Woori bank with Refund amount, bank account.

If WAY4 system does not get respond of message (PR001) then the system will repeat sending 3 times the message (TR001).

Other, user can press the button Result to manually send the TR001 of the latest transaction.

#### Technical Detail

Openway needs to customize WAY4 system, so that with the virtual account given in online message/batch file, WAY4 should find the corresponding Issuing contract number for Credit card and the corresponding Wooribank virtual account for BNPL Issuing contract.

Openway also need to configure the new SMS/Email notification template as provided by LFVN. The SMS/Email notification template is described in DR Vol 16.

#### Impact Areas

* Customization Web Service interface
* Customization functions and forms in WAY4 to allow sending online message via Web Service interface.
* Customization Java Pipes and menus for batch processing.

### REQWS013. BIDV Interface

#### Business Requirements

Interface between LFVN system and BIDV are already developed by LFVN. Openway does not need to do anything about this function.

As described in 7.1, BIDV virtual account is attached as contract parameter of BNPL Issuing contract. LFVN will then adjust/re-build their developed interfaces and re-test the function themselves.

#### Technical Detail

Interface between LFVN system and BIDV are already developed by LFVN. Openway does not need to do anything about this function.

#### Impact Areas

* 1. Payment to BNPL’s debt by cash at Vnpost, Payoo, MoMo

### REQWS014 – VNPost Interface

#### Business Requirements

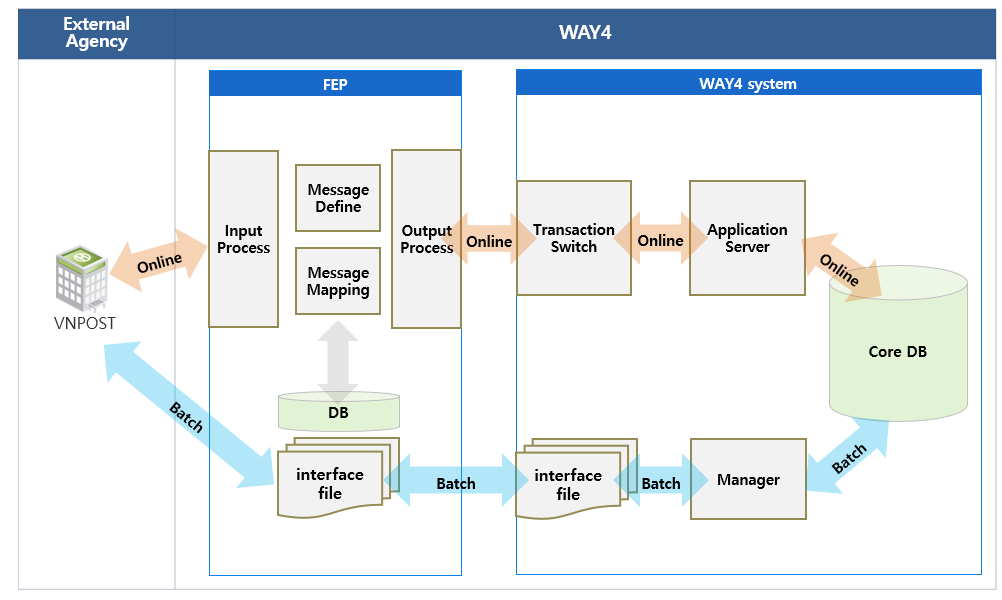
Openway will need to customize the searching algorithm for BNPL Issuing contract from virtual account number (WooriBank/BIDV).

From WooriBank/BIDV virtual account number, WAY4 needs to search for the corresponding BNPL Issuing contract number and then makes the payment transaction to this BNPL Issuing contract number.

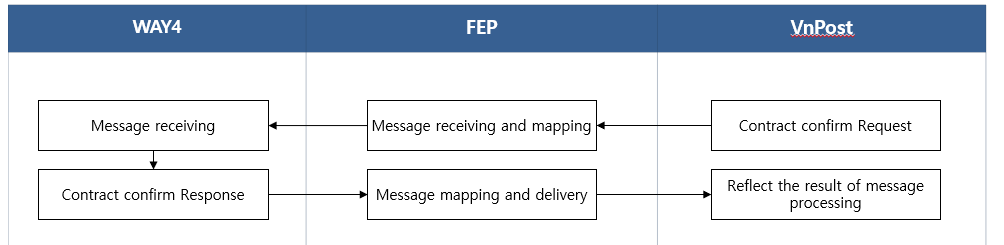
**The other business flow/requirement should be the same as what had been done for Credit card which is re-described below:**

LFVN needs both batch interface and online interface (web service interface) with VNPost for below functions:

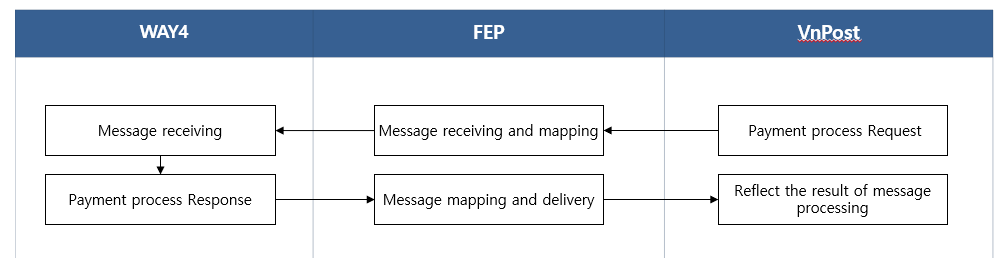
|  |  |  |
| --- | --- | --- |
| **ID** | **Function** | **Type of Interface** |
| 1 | Contract Inquiry | on-line |
| 2 | Payment | on-line |
| 3 | Reconcile | batch |



Contract Function: VNPost will send request to LFVN to get current loan information (MTP = Principal Amount + Interest posted at the time of request) and customer information



Payment Function: LFVN BNPL customers pay by cash at VNPost Branch, after sending Contract request and getting response, VNPost will send Payment request to LFVN.



Reconcile Functions: VNPost uploads reconcile to FTP server and then LFVN can download to process. Openway needs to customize WAY4 system, so that with the virtual account given in online message/batch file, WAY4 should find the corresponding Issuing contract number and the corresponding Wooribank virtual account for BNPL Issuing contract.

#### Technical Detail

Openway needs to customize WAY4 system, so that with the virtual account given in message/batch file, WAY4 should find the corresponding Issuing contract number and the corresponding Wooribank/BIDV virtual account for BNPL Issuing contract.

#### Impact Areas

* Customization Web Service interface
* Customization Java Pipes and menus for batch processing

### REQWS015 – Payoo Interface

#### Business Requirements

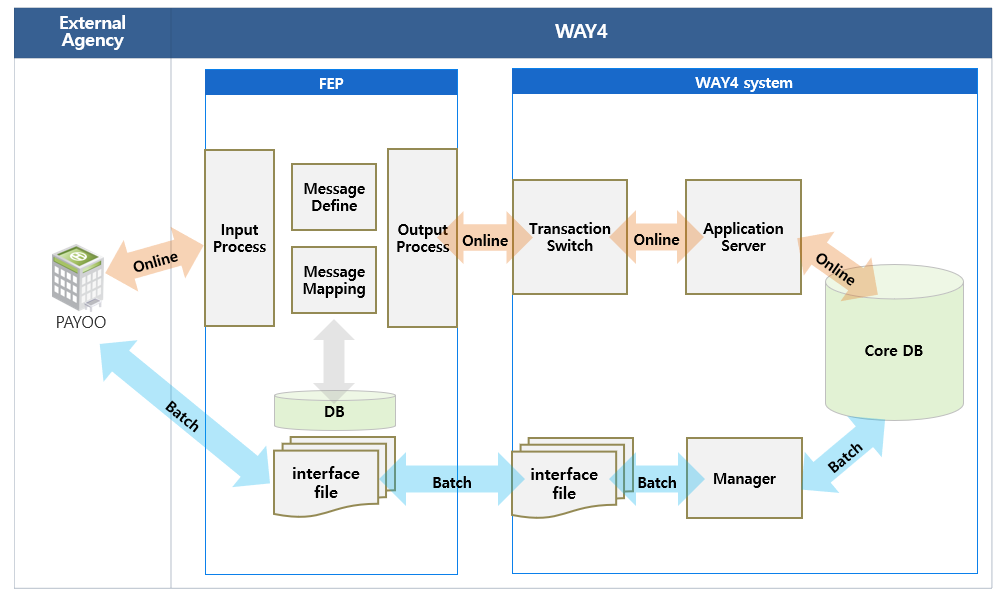
Openway will need to customize the searching algorithm for BNPL Issuing contract from Virtual account number (WooriBank/BIDV).

From WooriBank/BIDV virtual account number, WAY4 needs to search for the BNPL Issuing contract number and then to make payment transaction to this Issuing contract number.

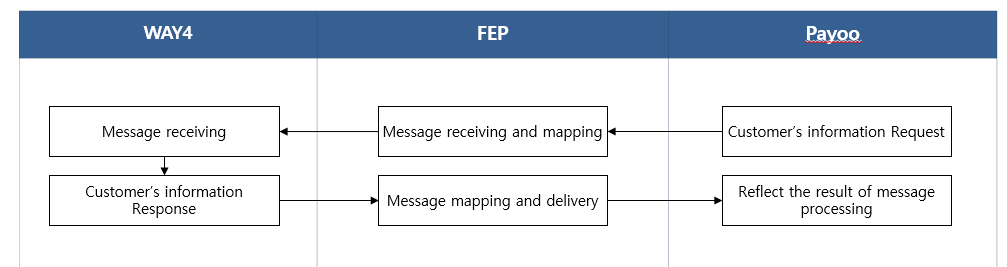
**The other business flow/requirement should be the same as what had been done for Credit card which is re-described below:**

LFVN need both batch interface and web service online interface with Payoo for below functions:

|  |  |  |
| --- | --- | --- |
| **ID** | **Function** | **Type of Interface** |
| 1 | Query | on-line |
| 2 | Pay | on-line |
| 3 | Check Payment | on-line |
| 4 | Reconcile | batch |



Query Function: Payoo will send request to LFVN to get current loan information (MTP = Principal Amount + Interest posted at the time of request) and customer information.

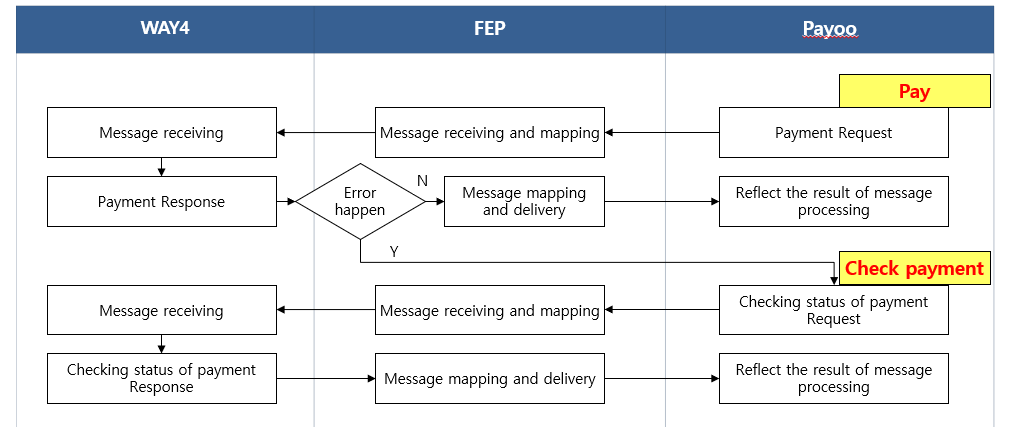


**Transaction Flow**:

LFVN credit cardholders pay by cash at Payoo Branch. They come to Payoo branch. Payoo calls WAY4 API standard to query the cardholder balance. The input parameter can be Wooribank/BIDV virtual account number. WAY4 API will response BNPL Issuing contract number, MTP amount (MTP = Principal Amount + Interest posted at the time of request), total balance…

When Payoo makes the Payment transaction, Payoo will generate the RRN, use virtual account contract number for each transaction.

Daily, Payoo will export the transaction file for reconciliation. Way4 will import transaction file to reconcile in WAY4 system.

~~~~

#### Technical Details

Openway needs to develop mapping rule between virtual account number (Wooribank/BIDV) and BNPL Issuing contract number.

#### Impact Areas

* Customization Web Service interface
* Customization Java Pipes and menus for batch processing
* Payoo needs to update API specification as described in the bellow attachment:

### REQWS016 – Momo Interface

#### Business Requirements

Interface between LFVN and MoMo is already developed by LFVN.

As described in 7.1, BIDV virtual account is attached as contract parameter of BNPL Issuing contract. LFVN will then adjust/re-build their developed interfaces and re-test the function themselves.

#### Technical Details

Interface between LFVN system and BIDV are already developed by LFVN. Openway does not need to do anything about this function.

#### Impact Areas

* 1. Online payment to BNPL’s debt via Payoo, MoMo

### REQWS017. Online payment via Payoo Interface

#### Business Requirements

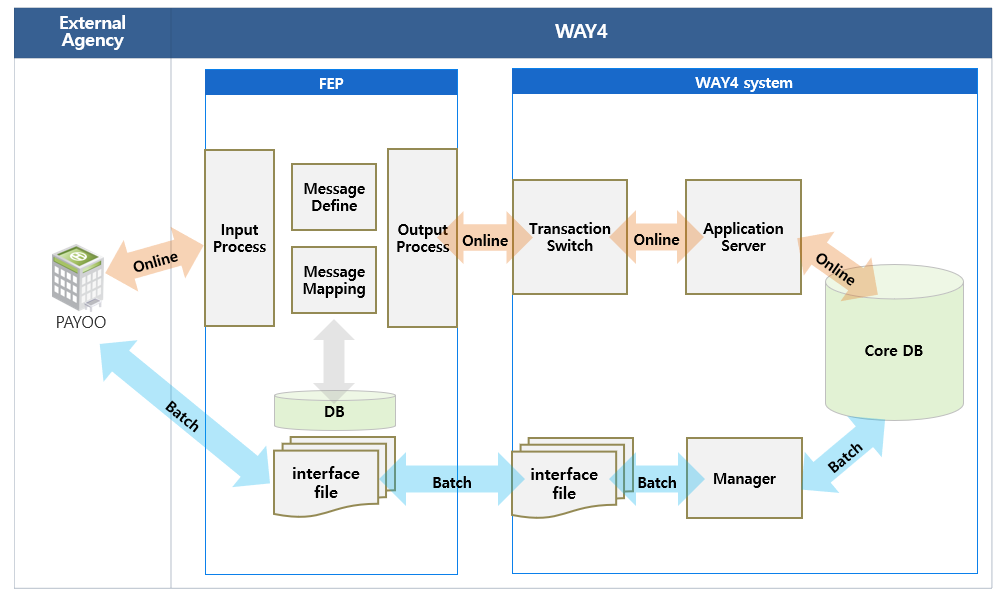
Openway will need to customize the searching algorithm for BNPL Issuing contract from Virtual account number (WooriBank/BIDV).

From WooriBank/BIDV virtual account number, WAY4 needs to search for the BNPL Issuing contract number and then to make payment transaction to this Issuing contract number.

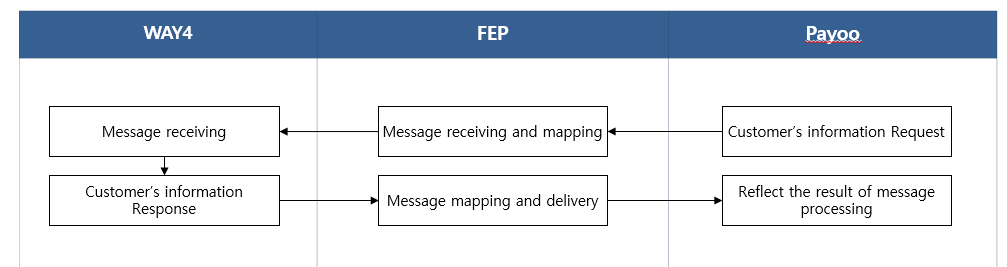
**The other business flow/requirement should be the same as what had been done for Credit card which is re-described below:**

LFVN need both batch interface and web service online interface with Payoo for below functions:

|  |  |  |
| --- | --- | --- |
| **ID** | **Function** | **Type of Interface** |
| 1 | Query | on-line |
| 2 | Pay | on-line |
| 3 | Check Payment | on-line |
| 4 | Reconcile | batch |



Query Function: Payoo will send request to LFVN to get loan information (MTP = Principal Amount + Interest posted at the time of request) and customer information.



**Transaction Flow**:

Step 1: Payoo calls WAY4 API standard to query the BNPL Issuing contract information. The input parameter can be Wooribank/BIDV virtual account number. WAY4 API will response BNPL Issuing contract number, contract status, …

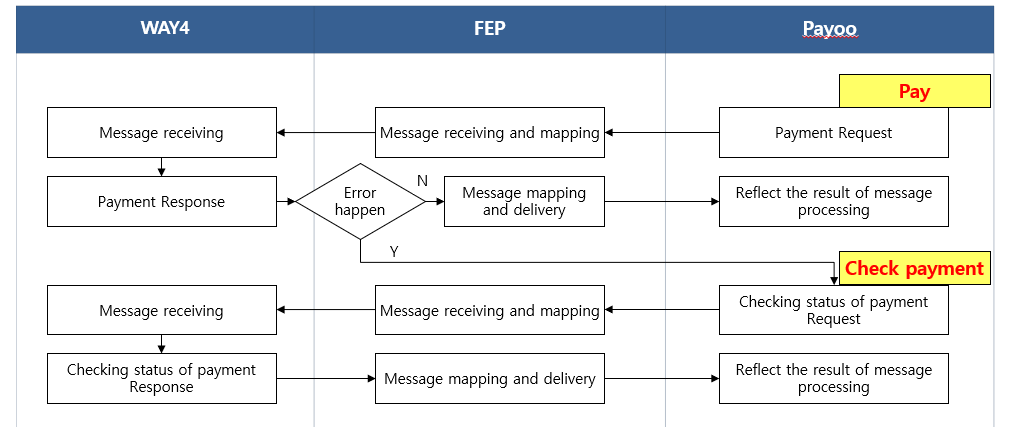
Step 2: Payoo calls WAY4 API standard to query the BNPL Issuing contract balance. The input parameter can be Wooribank/BIDV virtual account number. WAY4 API will response BNPL Issuing contract number, MTP amount (MTP = Principal Amount + Interest posted at the time of request), total balance…

Step 3: When Payoo makes the Payment transaction, Payoo will generate the RRN, use virtual account contract number for each transaction.

Step 4: Payoo calls WAY4 API to check payment transaction. The input parameter is RRN. WAY4 API will response payment transaction status.

Daily, Payoo will export the transaction file for reconciliation. Way4 will import transaction file to reconcile in WAY4 system.

Openway needs to customize WAY4 system, so that with the virtual account given in online message/batch file, WAY4 should find the corresponding Issuing contract number and the corresponding Wooribank virtual account for BNPL Issuing contract.

~~~~

#### Technical Detail

Openway needs to customize WAY4 system, so that with the virtual account given in online message/batch file, WAY4 should find the corresponding Issuing contract number and the corresponding Wooribank/BIDV virtual account for BNPL Issuing contract.

#### Impact Areas

Payoo needs to update API specification as described in the bellow attachment:

### REQWS018. Online payment via MoMo Interface

#### Business Requirements

Interface between LFVN and MoMo is already developed by LFVN.

As described in 7.1, BIDV virtual account is attached as contract parameter of BNPL Issuing contract. LFVN will then adjust/re-build their developed interfaces and re-test the function themselves.

#### Technical Detail

Interface between LFVN system and BIDV are already developed by LFVN. Openway does not need to do anything about this function.

#### Impact Areas

1. Requirement Description – Interface with Pay Later System

## Virtual Account Creation for BNPL Issuing Contract

The flow for creating Virtual account for BNPL Issuing contract is described in the above diagram.

BIDV bank code is set as “BIDV”. LOS will then return to Pay Later BIDV virtual account as value of tag “BIDV”. Pay Later will then send this value to WAY4 as a contract parameter of BNPL Issuing contract in parameter code “BIDV”.

LFVN also requests to save LOS\_APPL to WAY4. Therefore, Pay Later System needs to send this value to WAY4 in “Address Line 1” of BNPL Issuing contract with address\_type code is “LOS\_APPL”.

## REQWS019. New customer

### Business Requirements

Pay Later system calls API to WAY4 to client new Client, Liability contract and Issuing contract and set credit limit for BNPL product.

Customer information is described in Discovery Volume 18.

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To create new client, liability contract and issuing contract and set credit limit for BNPL product from Pay Later system. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Liability contract number | Liability contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-LIBNPL”** |
| Issuing contract number | Issuing contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-IBNPL”** |

### Limitations

## REQWS020. Customer has Loan, no Credit card

### Business Requirements

Pay Later system calls API to WAY4 to client new Client, Liability contract and Issuing contract for BNPL product.

Customer information is described in Discovery Volume 18.

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To create new client, liability contract and issuing contract for BNPL product from Pay Later system. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Liability contract number | Liability contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-LIBNPL”** |
| Issuing contract number | Issuing contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-IBNPL”** |

### Limitations

## REQWS021. Customer has CIF, no Loan, no Credit card

### Business Requirements

Pay Later system calls API to WAY4 to client new Client, Liability contract and Issuing contract for BNPL product.

Customer information is described in Discovery Volume 18.

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To create new client, liability contract and issuing contract for BNPL product from Pay Later system. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Liability contract number | Liability contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-LIBNPL”** |
| Issuing contract number | Issuing contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-IBNPL”** |

### Limitations

## REQWS022. Customer has Credit card, no Loan

### Business Requirements

Pay Later system calls API to WAY4 to create Liability contract and Issuing contract for BNPL product.

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To create liability contract and issuing contract for BNPL product from Pay Later system. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Liability contract number | Liability contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-LIBNPL”** |
| Issuing contract number | Issuing contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-IBNPL”** |

### Limitations

## REQWS023. Customer has Loan and Credit card

### Business Requirements

Pay Later system calls API to WAY4 to Liability contract and Issuing contract for BNPL product.

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To create liability contract and issuing contract for BNPL product from Pay Later system. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Mapping Fields

The below table describes the request fields with the API response field tags:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Liability contract number | Liability contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-LIBNPL”** |
| Issuing contract number | Issuing contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-IBNPL”** |

### Limitations

## REQWS024. Customer has closed BNPL account

### Business Requirements

Pay Later system calls API to WAY4 to retrieve list of Issuing contract under BNPL Liability contract and its outstanding balance and status.

Pay Later system checks Issuing contract outstanding balance and status. If all conditions are satisfied, then Pay Later will call API to WAY4 to update new credit limit for BNPL Liability contract.

Finally, Pay Later system calls API to create new BNPL Issuing contract.

This process is agreed by LFVN, ESI and OPW in Discovery Volume 18.

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To create new BNPL Issuing contract for existing BNPL Liability contract which has BNPL closed Issuing contract.  Step 1: Call API **“GetContractHierarchy\_from\_ClientNumber”** to get all BNPL issuing contract information (status, balance), Liability contract number from client number (CIF). LFVN check BNPL Issuing contract status (ACC CLOSED, ACC EXPIRED, ACC RESERVE) and outstanding amount (>=10000 VND). If all checks are satisfied, then moves to step 2.  As request from Pay Later, OPW provides a new API name “GetIssuingContractInformation\_From\_IssuingContractNumber”. Pay Later can call this API to retrieve status and balance of Issuing contract.  Step 2: Call API **“Update\_LiabilityContract\_CreditLimit”** to update Liability credit limit for Liability contract (with Liability contract number is retrieved from step 1).  Step 3: Call API **“Add\_IssuingContract\_From\_LiabilityContract”** to create new BNPL Issuing contract from existing BNPL Liability contract (with Liability contract number is retrieved from step 1). |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |  |
| --- | --- | --- |
| **Mapping fields** | | |
| **API** | **Business Fields** | **Mapping API Response** |
| GetContractHierarchy\_from\_ClientNumber | BNPL Liability Contract Number | Getting from tag ‘DataRs\ ContractRs’.  If value of tag ‘ProductCode1’ is “PISBNPL”, then BNPL Liability Contract Number can be retrieved from tag ‘ContractNumber’; |
| BNPL Issuing Contract Number | Getting from tag ‘DataRs\ ContractRs’:  If value of tag ‘ProductCode1’ is “00107”, then BNPL Issuing Contract Number can be retrieved from tag ‘ContractNumber’; |
| BNPL Issuing Contract Status | Getting from tag ‘DataRs\ ContractRs’:  If value of tag ‘ProductCode1’ is “00107”, then BNPL Issuing Contract status can be retrieved from tag ‘StatusCode’; |
| BNPL Issuing Contract Outstanding Balance | Getting from tag ‘DataRs\ ContractRs’:  If value of tag ‘ProductCode1’ is “00107” and value of tag ‘Balance\ Type’ is “OUTST\_BALANCE” then BNPL Issuing Contract outstanding balance can be retrieved from tag ‘Balances\ Balance\ Amount’. |
| GetContract\_balance\_status\_From\_IssuingContract | BNPL Issuing Contract Status | Getting value from tag ‘DataRs\ ContractRs\Info\Status\StatusCode’ |
| BNPL Issuing Contract Outstanding Balance | Getting value from tag ‘DataRs\ ContractRs\Info\Balances\Balance\Amount’ if value of tag ‘Balance\Type’ is “OUTST\_BALANCE”. |
| Add\_IssuingContract\_From\_LiabilityContract | BNPL Issuing Contract Number | Issuing contract can be get in tag “<ContractNumber></ContractNumber>” if the application has  **“<ObjectType>Contract</ObjectType>**  **<ActionType>Add</ActionType>”** and value in tag **“ContractNumber”** has pre-fix is **“001-IBNPL”** |

|  |  |  |
| --- | --- | --- |
| **Issuing contract status** | | |
| **Code** | **Name** | **Meaning** |
| A00 | ACC OK | Account is okay |
| A06 | ACC RESERVE | Block automatically forever when customer has late payment more than 90 days |
| A04 | ACC CLOSED | Customer request to close the account forever |
| A22 | ACC EXPIRED | Block account forever when expired |

### Limitations

## REQWS025. Create Authorization Transaction

### Business Requirement

Pay Later system calls API to WAY4 to create Authorization transaction. Transaction should contain the following information:

* Transaction date
* RRN
* Transaction amount
* Transaction currency (default is VND)
* Transaction details (contain good’s category)
* Source contract number
* Target contract number – BNPL Issuing contract number
* Merchant information: merchant name, merchant location, merchant ID
* Instalment tenor (only required when transaction should be converted to instalment transaction).

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | If Pay Later System does not store BNPL Issuing contract number, Pay Later System needs to call API “GetContractHierarchy\_from\_ClientNumber” to get Issuing contract number beforehand.  To create authorization transaction. Authorization document should include the following fields: RRN, transaction amount, transaction details, source contract number, target contract number (BNPL Issuing contract number), Merchant information ,Merchant ID, instalment tenor. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specification

#### API Components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Limitations

## REQWS026. Create Financial Transaction

### Business Requirement

Pay Later system calls API to WAY4 to create Financial transaction. Transaction should contain the following information:

* RRN (RRN of Original Authorization transaction)
* Transaction amount
* Transaction currency (default is VND)
* Transaction details (contain good’s category)
* Source contract number
* Target contract number
* Merchant information
* Instalment tenor (only required when transaction should be converted to instalment transaction).

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | If Pay Later System does not store BNPL Issuing contract number, Pay Later System needs to call API to get Issuing contract number beforehand.  To create financial transaction. Financial document should include the following fields: RRN, transaction amount, transaction details, source contract number, target contract number (BNPL Issuing contract number), Merchant information, Merchant ID, instalment tenor. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specification

#### API Components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Limitations

## REQWS027. Reversal Authorization Transaction

### Business Requirement

Pay Later system calls API to WAY4 to create Reversal Authorization transaction. Transaction should contain the following information:

* RRN (RRN of Original Authorization transaction)
* Transaction amount
* Transaction currency (default is VND)
* Transaction details
* Source contract number
* Target contract number
* Merchant information

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | If Pay Later System does not store BNPL Issuing contract number, Pay Later System needs to call API to get Issuing contract number beforehand.  To create reversal authorization transaction. Reversal Authorization document should include the following fields: RRN, transaction amount, transaction details, source contract number, target contract number (BNPL Issuing contract number), Merchant information, Merchant ID, instalment tenor. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specification

#### API Components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Limitations

## REQWS028. Credit Transaction

### Business Requirement

Pay Later system calls API to WAY4 to create Credit Transaction. Transaction should contain the following information:

* RRN (new RRN, should not be the same as RRN of the original financial transaction).
* RRN of the original financial transaction (for reference to the original transaction)
* Transaction amount
* Transaction currency (default is VND)
* Transaction details
* Source contract number
* Target contract number
* Merchant information
* Instalment tenor (only required when transaction should be converted to instalment transaction).

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | If Pay Later System does not store BNPL Issuing contract number, Pay Later System needs to call API to get Issuing contract number beforehand.  To create reversal financial transaction. Reversal financial document should include the following fields: RRN, transaction amount, transaction details, source contract number, target contract number (BNPL Issuing contract number), Merchant information, Merchant ID, instalment tenor. |
| Process Mode | Online to WAY4 Gate |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Pay Later System |

### Message Specification

#### API Components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Limitations

1. Requirement Description – Interface with Agent Desktop

## REQWS029. Get BNPL Issuing contract info from CIF

### Business Requirements

This function allows Agent Desktop sends request to WAY4 to retrieve BNPL Issuing contract information from client number.

Detail request is defined in the below table:

|  |  |
| --- | --- |
| **Input** | **Output** |
| Client Number (CIF) | BNPL Issuing contract number |
| Open date |
| Expiry date |
| Credit limit amount |
| Contract status |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve BNPL Issuing contract information from client number via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

.

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

BNPL Issuing contract information can be retrieved from the content of each tag **<IssContractDetailsAPIOutputV2Record> … </IssContractDetailsAPIOutputV2Record>** of the API response by filtering filed ‘Product’ with value ‘**001-BNPL Issuing Contract’**. The mapping fields are described in the below table:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| BNPL Issuing contract number | ContractNumber |
| Open date | OpenDate |
| Expiry date | OpenDate + 12 months |
| Approved credit limit amount | CreditLimit |
| Contract status | Status |

### Limitations

## REQWS030. Get client information from BNPL Issuing contract number

### Business Requirements

This function allows Agent Desktop sends request to WAY4 to retrieve client information from BNPL Issuing contract number.

Detail request is defined in the below table:

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Full name |
| Registration number |
| Registration document |
| Registration issued place |
| Date of birth |
| Permanent address |
| Email address |
| Company name |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve client information from BNPL Issuing contract number via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

.

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

BNPL Issuing contract information can be retrieved from the content of each tag **<IssContractDetailsAPIOutputV2Record> … </IssContractDetailsAPIOutputV2Record>** of the API response by filtering filed ‘Product’ with value ‘**001-BNPL Issuing Contract’**. The mapping fields are described in the below table:

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Full name | FullName |
| Registration number | IdentityCardNumber |
| Registration document | IdentityCardType |
| Registration issued place | IdentityCardDetails |
| Date of birth | BirthDate |
| Permanent Address | AddressLine3 |
| Email Address | EMail |
| Company Name | CompanyName |

### Limitations

## REQWS031. Get BNPL Issuing contract information

### Business Requirements

This function allows Agent Desktop sends request to WAY4 to retrieve BNPL Issuing contract information from BNPL Issuing contract number.

Detail request is defined in the below table:

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | BNPL Issuing contract number |
| Contract status |
| ~~Product code~~ |
| Open date |
| Expiry date |
| Credit limit amount |
| Billing date |
| Next billing date |
| Current Billing date |
| Scheme Code |
| Due date |
| Virtual account |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve BNPL Issuing contract information from BNPL Issuing contract number via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

To get the required information, Agent Desktop needs to call the following APIs.

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |  |
| --- | --- | --- |
| **API Name** | **Business Fields** | **API Response Fields** |
| GetcontractByNumberV2 | BNPL Issuing contract number | ContractNumber |
| Contract status | StatusCode |
| ~~Product code~~ | Product |
| Open date | OpenDate |
| Expiry date | OpenDate + 12 months |
| Credit limit amount | CreditLimit |
| Current Billing date | LastBillingDate – 1 day |
| Next Billing date | NextBillingDate |
| Due date | DueDate |
| GetContractClassifiersV3 | Scheme Code | Filter by tag ‘<ClassifierCode>BNPL\_SCHEME</ClassifierCode>’  Scheme Code name in tag ‘ValueName’ |
| Billing Date | Filter by tag ‘<ClassifierCode>BILLING\_DAY</ClassifierCode>’  Billing date value in tag ‘ValueCode’ |
| GetContractParameters | Virtual account  BIDV virtual account and Wooribank virtual account | Filter by tag ‘<Code>BIDV\_VA</Code>’  BIDV virtual account value in tag ‘Value’  Filter by tag ‘<Code>WO\_VA</Code>’  Wooribank virtual account value in tag ‘Value’ |

### Limitations

## REQWS032. Get BNPL Issuing contract outstanding information

### Business Requirements

This function allows Agent Desktop sends request to WAY4 to retrieve BNPL Issuing contract outstanding information from BNPL Issuing contract number.

Detail request is defined in the below table:

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Outstanding amount |
| Statement amount |
| Hold amount |
| Available amount |
| Min Payment amount |
| CIC Level |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve BNPL Issuing contract outstanding information from BNPL Issuing contract number via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

To get the required information, Agent Desktop needs to call the following APIs.

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |  |
| --- | --- | --- |
| **API Name** | **Business Fields** | **API Response Fields** |
| GetcontractByNumberV2 | Outstanding amount | Balance |
| Hold amount | Blocked |
| Available amount | Available |
| Min Payment amount | TotalDue |
| GetContractBalanceV2 | Statement amount | BalanceValue |
| GetContractClassifiersV3 | CIC Level | Filter by tag ‘<ClassifierCode> CIC\_LEVEL</ClassifierCode>’  Scheme Code name in tag ‘ValueName’ |

### Limitations

## REQWS033. Get BNPL Issuing contract payment information

### Business Requirements

This function allows Agent Desktop sends request to WAY4 to retrieve BNPL Issuing contract payment information from BNPL Issuing contract number.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Auto debit mode |
| Auto payment account |
| Auto payment account name |
| Bank name |

### Technical Details

During this project scope, BNPL does not have function auto-debit; therefore, there is no designing database structure for the request fields in WAY4. Openway will provide API to get Issuing contract information, LFVN will need to map required fields with API response later by themselves.

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

### Limitations

## REQWS034. Get BNPL Issuing contract transaction history

### Business Requirements

This function allows Agent Desktop to retrieve transaction history for BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number  Date From  Date To | Transaction code |
| Transaction date |
| Transaction time |
| Posting date |
| Transaction amount |
| Transaction currency |
| Merchant |
| Return code |
| Payment method |
| Instalment tenor |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve transaction history of BNPL Issuing contract via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Transaction code | TransCode |
| Transaction date | TransDate: YYYY-MM-DD (Before character ‘T’) |
| Transaction time | TransDate: HH:MM:SS (After character ‘T’) |
| Posting date | PostingDate |
| Transaction amount | TransAmount |
| Transaction currency | TransCurr |
| MerchantID | SourceNumber |
| Return code | ResponseCode |
| Payment method | AddInfo tag ‘INST\_TN’. If value of this tag is equal to ‘1’ then payment method is onetime payment; if value is equal to ‘2’ or ‘3’ then payment method is instalment. |
| Instalment tenor | Value of tag AddInfo ‘INST\_TN’ |

Mapping MerchantID with merchant name:

|  |  |
| --- | --- |
| **Mapping MerchantID with merchant name:** | |
| **MerchantID** | **Merchant Name** |
| 000001 | VNTrip |
| 000002 | Sendo |

### Limitations

## REQWS035. Get BNPL Issuing contract instalment transaction plan

### Business Requirements

This function allows Agent Desktop retrieve all instalment transaction plan for BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Creation Date |
| First Payment |
| Last Payment |
| Total Amount |
| Principal Amount |
| Fee Amount |
| Paid Amount |
| Due Amount |
| OVD Amount |
| Full Repayment Amount |
| Instalment tenors |
| Number of paid tenors |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve instalment transaction plans for BNPL Issuing contract via online channel. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Creation Date | CreationDate |
| First Payment | FirstPayment |
| Last Payment | LastPayment |
| Total Amount | TotalAmount |
| Principal Amount | PrincipalAmount |
| Fee Amount | FeeAmount |
| Paid Amount | PaidAmount |
| Due Amount | DueAmount |
| OVD Amount | OVDAmount |
| Full Repayment Amount | FullRepaymentAmount |
| Instalment tenors | Tenor |

### Limitations

## REQWS036. Get BNPL Issuing contract instalment portion

### Business Requirements

This function allows Agent Desktop to retrieve all instalment portion for an Instalment plan for BNPL Issuing contract.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Instalment plan ID | Portion Number |
| Invoice Date |
| Due Date |
| Portion Amount |
| Principal Amount |
| Fee Amount |
| Paid Amount |
| Status |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To retrieve all instalment portion for an instalment plan for BNPL Issuing contract via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |
| --- | --- |
| **Mapping fields** | |
| **Business request field** | **API response field** |
| Portion Number | PortionNumber |
| Invoice Date | OpenDate |
| Due Date | DueDate |
| Portion Amount | PortionAmount |
| Principal Amount | PrincipalAmount |
| Fee Amount | FeeAmount |
| Paid Amount | PaidAmount |
| Status | Status |

### Limitations

## REQWS037. Temporary lock/unlock BNPL Issuing contract

This function allows Agent Desktop to lock BNPL Issuing contract information.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Issuing contract number | Issuing contract is locked/unlocked successfully. |

### Technical Details

|  |  |
| --- | --- |
| **General Settings – Way4** | |
| Purpose | To temporarily lock BNPL Issuing contract via Agent Desktop. |
| Process Mode | Online to WAY4 Web Service |
| Process Frequency | N.A. |
| Process Time | N.A. |
| Source |  |
| Destination | Issuer/Cardholder |

### Message Specifications

#### API components



#### Request Message template



#### Response Message template



### Mapping Fields

|  |  |  |
| --- | --- | --- |
| **Mapping fields** | | |
| **Business request field** | **API Request** | **API Response** |
| Issuing contract is locked/unlocked successfully | If value of field ‘NewStatus’ is ‘A00’, then contract will be set to Account OK – unlock account.  If value of field ‘NewStatus’ is ‘A18’ then contract will be set to Account Return – lock account. | <RetCode>0</RetCode>  <RetMsg>I Successfully Completed</RetMsg> |

### Limitations

1. Requirement Description – Interface with LOS

## REQWS038. Create Credit Card Contract Hierarchy (Liability Contract, Issuing Contract and Card Contract) for existing client

### Business Requirements

In case, client information is already existing in WAY4. LOS needs an API to create credit card product contract hierarchy for this client.

### Technical Details

LOS calls API to WAY4 to create Liability contract, Issuing contract and card contract for existing client.

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Limitations

## REQWS039. Update Client Information

### Business Requirements

This function allows LOS sends request to WAY4 to update client information. The detail information is provided by LFVN later. Openway did provide all changeable fields data in API request. LFVN can choose to update or not any field in this API request as their wishes.

| No | LOS | | WAY4 | |
| --- | --- | --- | --- | --- |
|  |  | LOS Field | Update/Not Update | WAY4 Field |
| 1 | Customer Type |  |  | ClientType |
| 2 | CIF Number |  |  | ClientNumber |
| 3 | Salutation |  |  |  |
| 4 | Full Name | Full Name | Update | ShortName |
| 5 | Date of Birth | Date of Birth | Update | BirthDate |
| 6 | Career | Default in LOS: “OTHER” | Update | Position |
| 7 | Qualification | Default in LOS: “KHAC/OTHERS” |  |  |
| 8 | Region | Default in LOS: “KHAC/OTHERS” |  |  |
| 9 | Gender | Gender |  | Gender |
| 10 | Marital Status | Default in LOS: “OTHER” | Update | MaritalStatus |
| 11 | Employment Industry | Career | Update | CompanyName |
| 12 | Employment Position | Position |  |  |
| 13 | Identification Document Type | Default is “IDENTIFY DOCUMENT” | Update | RegNumberType |
| 14 | Identification Document Code | Depend on uploaded identification document type |  |  |
| 15 | Identification Document Number | Registration Card/Passport Number | Update | RegNumber |
| 16 | Country of Issue | Get from Registration Card/Passport Document | Update | RegNumberDetails |
| 17 | Issue Date | Issue Date | Update | ADDDATE01 |
| 18 | ID Issued By | Issue Place | Update | RegNumberDetails |
| 19 | Address Type | Permanent address |  | AddressLine3 |
| 20 | House/Building Number | Get from Registration Card/Passport Document | Update | AddressLine3 |
| 21 | City | Get from Registration Card/Passport Document | Update | AddressLine3 |
| 22 | District | Get from Registration Card/Passport Document | Update | AddressLine3 |
| 23 | Country | Get from Registration Card/Passport Document | Update | AddressLine3 |
| 24 | Postal Code | Default is “100000” | Update | AddressLine3 |
| 25 | Address Valid From | Get from Registration Card/Passport Document | Update | AddressLine3 |
| 26 | Address Type | Mailing Address |  | AddressLine1 |
| 27 | House/Building Number | Detail Address | Update | AddressLine1 |
| 28 | City | Province/City | Update | AddressLine1 |
| 29 | District | District | Update | AddressLine1 |
| 30 | Country | Default is “Việt Nam” | Update | AddressLine1 |
| 31 | Postal Code | Default is “100000” | Update | AddressLine1 |
| 32 | Address Valid From | Register Pay Later date | Update | AddressLine1 |
| 33 | Contact Type – PHONE | PHONE |  | PhoneType = “Mobile” |
| 34 | Contact Category | Default is “Home phone 1” |  |  |
| 35 | Phone Number | Phone Number | Update | PhoneNumber |
| 36 | PhoneType - Home |  |  | PhoneType =”Home” |
| 37 | PhoneNumber |  | Update | PhoneNumber |
| 38 | Contact Type – Email | Email |  |  |
| 39 | Type | Default is “Home” |  |  |
| 40 | Email Address | Email | Update | EMail |
| 41 | Nationality | Nationality | Update | Citizenship |
| 42 | Communication Language | Default is “Việt Nam” | Update | Language |
| 43 | Customer Classification – Update at Liability Contract Level | Base on product scheme |  | Update at Liability Contract Level |
| 44 |  |  | Update | AddInfo01: EXPERIENCE\_YEARS;AVERAGE\_MONTHLY\_INCOME |
| 45 |  |  | Update | AddInfo02: VNAirline\_Member\_Number |
| 46 |  |  | Update | AddInfo03: Lotte\_Card\_Member |
| 47 |  |  | Update | AddInfo04: FirstName;LastName |
| 48 | Client Address |  |  | AddressType - PIN |
| 49 | Client Address |  | Update | City |
| 50 | Client Address |  | Update | AddressLine1 |
| 51 | Client Address |  | Update | AddressLine2 |
| 52 | Client Address |  | Update | AddressLine3 |
| 53 | PlasticInfo |  | Update | FirstName |
| 54 | PlasticInfo |  | Update | LastName |
| 55 | PlasticInfo |  | Update | CompanyName |

### Technical Details

LOS calls API to WAY4 to create Liability contract, Issuing contract and card contract for existing client.

### Message Specifications

#### API components

Refer to document “ESI\_SOA\_Onboarding\_Message\_Specifications&Response\_Code”.

#### Request Message template



#### Response Message template



### Limitations