# Gateway Requirements \_v2

# Index

[Gateway Requirements \_v2 1](#_Toc2274)

[Index 2](#_Toc8859)

[ChangeLog 3](#_Toc425)

[1. Introduction 4](#_Toc322)

[1.1. Purpose 4](#_Toc2014)

[1.2. Target Audience 4](#_Toc4294)

[2. Overview 4](#_Toc22791)

[2.1. Communication Protocal 4](#_Toc27589)

[3. Definitions 4](#_Toc29626)

[3.1. PeraHub 4](#_Toc21314)

[3.2. BananaPay(BNNP) 4](#_Toc26272)

[3.3. Message 4](#_Toc3602)

[3.4. Idempotent 5](#_Toc19661)

[3.5. Transaction\_id 6](#_Toc9308)

[4. Message Overview 6](#_Toc2462)

[4.1. Message List 6](#_Toc24695)

[4.1.1. Message Number 6](#_Toc10427)

[4.1.2. Message List 6](#_Toc1908)

[4.2. Data Type 6](#_Toc10989)

[4.3. Data Requirements 7](#_Toc3843)

[4.4. Encoding 7](#_Toc3820)

[4.5. Reserved 7](#_Toc934)

[5. Message Syntax 7](#_Toc7501)

[5.1. Overview 7](#_Toc10566)

[5.2. How to deal with exceptions 7](#_Toc31036)

[6. API Functions 7](#_Toc29097)

[6.1. Authentication And BindCard[Bnnp.101.001.01] 7](#_Toc17318)

[6.1.1. Features 7](#_Toc32600)

[6.1.2. Sequence diagram 8](#_Toc19700)

[6.1.2.1. Authentication 8](#_Toc24822)

[6.1.2.2. Bind-Card 8](#_Toc29083)

[6.1.3. Request Parameters 8](#_Toc20509)

[6.1.4. Response parameters 9](#_Toc24570)

[6.2. Payment after bind[Bnnp.201.001.01] 9](#_Toc22573)

[6.2.1. Features 9](#_Toc14945)

[6.2.2. Sequence diagram 10](#_Toc11462)

[6.2.3. Request parameters 10](#_Toc24132)

[6.2.4. Response parameters 11](#_Toc29877)

[6.3. Refund[Bnnp.301.001.01] 11](#_Toc8658)

[6.3.1. Features 11](#_Toc13723)

[6.3.2. Sequence diagram 11](#_Toc9106)

[6.3.3. Request parameters 11](#_Toc11683)

[6.3.4. Response parameters 12](#_Toc25533)

[6.3.5. Questions 12](#_Toc5112)

[6.4. Payment-to-Bank[Bnnp.402.001.01] 12](#_Toc23901)

[6.4.1. Features 12](#_Toc15838)

[6.4.2. Sequence diagram 13](#_Toc17793)

[6.4.3. Request parameters 13](#_Toc21910)

[6.4.4. Response parameters 14](#_Toc2314)

[6.5. Transaction Query[Bnnp.501.001.01] 15](#_Toc31104)

[6.5.1. Features 15](#_Toc9690)

[6.5.2. Sequence diagram 15](#_Toc3340)

[6.5.3. Request parameters 15](#_Toc8804)

[6.5.4. Response parameters 15](#_Toc4735)

[6.6. Fee Inquiry[Bnnp.502.001.01] 15](#_Toc16122)

[6.6.1. Features 15](#_Toc23446)

[6.6.2. Sequence diagram 15](#_Toc5705)

[6.6.3. Request parameters 15](#_Toc4236)

[6.6.4. Response parameters 15](#_Toc20188)

[7. Exceptions 16](#_Toc18452)

[8. AccountCheckFile 16](#_Toc19875)

[8.1. PaymentCheckFile 16](#_Toc59)

[Append A 16](#_Toc6613)

[A.1 TrxCtgyCd 16](#_Toc1878)

[Append B 16](#_Toc25755)

[B.1 SysRtnCd 16](#_Toc7525)

[B.2 BizStsCd 16](#_Toc31596)

[Append C 16](#_Toc6550)

[C.1 Reserved Words 16](#_Toc31947)

# ChangeLog

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Log |  |  |
| 2017-10-29 | created |  |  |
|  |  |  |  |

# Introduction

## Purpose

This document is a description of the API Functions that BananaPay hope PeraHub can provide.

## Target Audience

PeraHub developers And BananaPay Developers.

# Overview

## Communication Protocal

HTTPS POST

# Definitions

## PeraHub

## BananaPay(BNNP)

BNNP will open a company account in pera-hub .

## Message

## Idempotent

*Idempotent* is a very important features to Transaction System, especially when this system having one than one modules.

To satisfy *Idempotent* , every request between system modules, must have one Unique identity, so we use *transaction\_id* and *mtcn*.

And common sequence of the system may be shown as below.



According the flow shown above, we can ensure that the duplicate request with the same transaction\_id can succeeded ONLY ONCE.

## Transaction\_id

# Message Overview

## Message List

### Message Number

Message Number must follow the rule as “XXXX.YYY.ZZZ.NN”,

* XXXX represents the system id, e.g. ‘bnnp’
* YYY represents Message name, e.g. ‘101’, this will diff message type.
* ZZZ is reserved, now should be filled as ‘001’
* NN represents the version no, now should be filled as ‘01’

### Message List

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Msg No | MsgName | Msg  Direction | Need Signature | Need statement check |
|  | Bnnp.101.001.01 | Authentication And BindCard | BNNP -> pera hub | Y | N |
|  | Bnnp.201.001.01 | Payment after bind | BNNP->pera hub | Y | Y |
|  | Bnnp.301.001.01 | Refund | BNNp->pera hub | Y | Y |
|  | Bnnp.402.001.01 | Payment to bank | Bbnp->pera | Y | Y |
|  | Bnnp.501.001.01 | Transaction Query | BNNP->pera hub | Y | N |
|  | Bnnp.502.001.01 | Fee Inquiry | Bnnp->pera hub | Y | N |

## Data Type

## Data Requirements

## Encoding

UTF-8

## Reserved

# Message Syntax

## Overview

## How to deal with exceptions

# API Functions

## Authentication And BindCard[Bnnp.101.001.01]

### Features

This API includes two parts: Authentication and BindCard.

User who has a pera-hub visa card can initiate an application in BNNP and fill required information, and BNNP post message to pera-hub.

Pera-hub check information is correct or not.

If information is correct, BNNP will request again and pera-hub complete the BindCard operation.

After BindCard operation, BNNP and perahub can indetify this user by *bind\_serialno* .

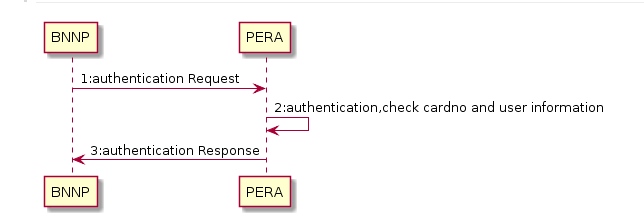
### Sequence diagram

#### Authentication

At first, user fills up his information in BNNP APP, and post to bnnp.

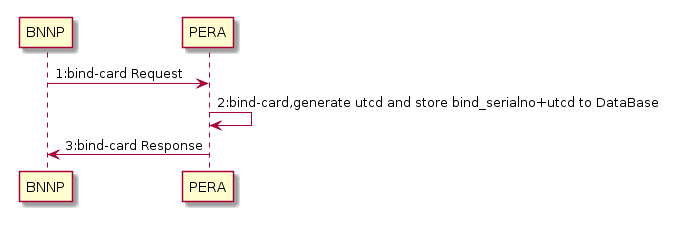
BNNP post user’s form to pera.

Pera check this form is correct or not.



#### Bind-Card

Second, if 6.1.2.1 Authentication is successful, then bnnp will generate a bind\_serialno, and post request data to pera hub again.



### Request Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| bind\_serialno | Alpha:32 | O | This filed is MUST when 6.1.2.2.  In Request 6.1.1.1 leaves empty. |
| Mobile | Alpha:32 | M |  |
| Surname | Alpha:32 |  |  |
| Givenname | Alpha:32 |  |  |
| Middlename | Alpha:32 |  |  |
| Nationality | Alpha:32 |  |  |
| id | Alpha:32 | M | Customer’s national ID CardNo |
| idtype | Alpha:32 | M | Customer’s national ID Type |
| Wucardno | Alpha:32 |  | At first, we support wu-card only.  so pera-hub can validate user information and WU card account. |
| scene | Alpha | M | “authen” : in case 6.1.2.1  “bind” : in case 6.1.2.2 |

### Response parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| bind\_serialno | Alpha:32 | O | This field is MUST when response to 6.1.2.2. |
| Utcn | Alpha:32 | O | This field is MUST provided in the response to 6.1.2.2.  This field should be generated by Perahub, only mapping to bind\_serialno exactly 1:1 . |

### Questions

* What information is submit to Pera-hub when user apply a new Pera-hub Visa Card in pera-hub branch?

## Payment after bind[Bnnp.201.001.01]

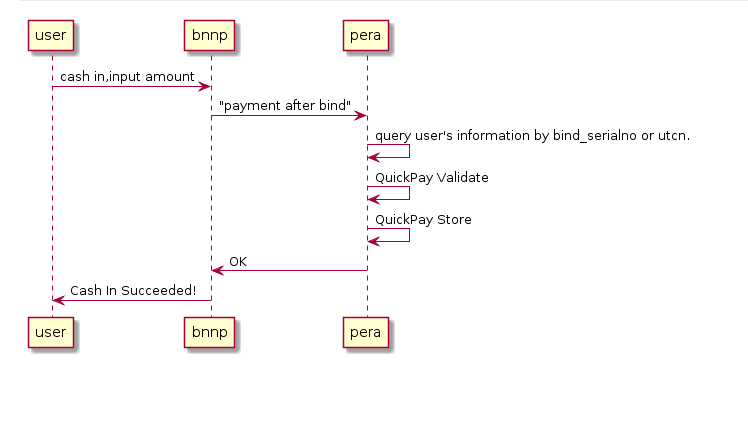
### Features

By this function, customer can transfer their money from their bank account to BNNP Company Account. Customer’s bank account can be identified by the bind\_serialno or utcn.

When user want to pay to BNNP , user submit their request, and BNNP will post request data to perahub, and perahub give respond to BNNP.

This payment transactions record should be listed in T+1’s *8.1 PaymentCheckFile* .

### Sequence diagram



### Request parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| bind\_serialno | Alpha:32 | M |  |
| utcn | Alpha:32 | M |  |
| transaction\_id | Alpha:32 | M |  |
| principal\_amount |  |  | Amount to Send |
| fixed\_amount\_flag |  |  | F if customer wants a fixed  principal amount in the  destination country and will not be subject to currency  fluctuation.  Otherwise N |
|  |  |  |  |
|  |  |  |  |

### Response parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| transaction\_id | Alpha:32 | M |  |
| mtcn |  |  |  |
| new\_mtcn |  |  |  |
| advisory\_text |  |  | Transaction Date and Time |
| filing\_date |  |  |  |
| filing\_time |  |  |  |
| pin\_message |  |  |  |
| other\_message\_1 |  |  |  |
| other\_message\_2 |  |  |  |

## Refund[Bnnp.301.001.01]

### Features

Function 6.3 *Refund* will be the opposite of Function *6.2 Payment after bind*.

In some case, BNNP should return money to one customer, we will call this Function Refund.

One Payment may refunded more than one times, and the sum of the refunded amount SHOULD NOT exceed the amount payed in older payment.

THE REFUND should be listed in the account check file on T+1 Day.

### Sequence diagram

### Request parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| Ori\_transaction\_id | Alpha:32 | M | Transaction\_id of older payment. |
| Ori\_Mtcn |  | M | Mtcn mapping with transaction\_id, returned by perahub in function payment. |
| Ori\_principal\_amount |  |  | Payed by customer in older payment. |
| Transaction\_id | Alpha:32 | M | Refund\_id is generated by BNNP, and must be exactly only one in the system. |
| principal\_amount |  | M | The part of principal\_amount, in this request,this part of money will be return to customer. |

### Response parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| transaction\_id | Alpha:32 | M | Transaction\_id of older payment. |
| mtcn |  | M | Mtcn mapping with transaction\_id, returned by perahub in function payment. |
| principal\_amount |  |  | Payed by customer in older payment. |
| refund\_id | Alpha:32 | M | Refund\_id is generated by BNNP, and must be exactly only one in the system. |
| refund\_amount |  | M | The part of principal\_amount, in this request,this part of money will be return to customer. |
| refund\_mtcn |  | M | Generated by perahub, and mapping with refund\_id by 1:1 . |

### Questions

* How about the fee? Let’s talk about this later.

## Payment-to-Bank[Bnnp.402.001.01]

### Features

This function will transfer money to some bank account shown in request parameters.

And this function seems like FUNCTION *3.2.3.3. SendMoney : Direct to Bank* .

When BNNP call this function, perahub acts *3.2.3.3. SendMoney : Direct to Bank* AND *3.2.3.4. SendMoney : Direct to Bank Store .* But the sender’s information are fixed to BNNP Company bank account .

Receivers’ information will be provided in request parameters.

Besides, BNNP will post parameters transaction\_id, pera-hub should store transaction\_id, when BNNP retry to call this function use the same transaction\_id ,pera-hub MUST transfer money ONLY ONCE.

### Sequence diagram

### Request parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| transaction\_id | Alpha:32 | M |  |
| principal\_amount |  |  | Amount to Send |
| fixed\_amount\_flag |  |  | F if customer wants a fixed  principal amount in the  destination country and will not be subject to currency  fluctuation.  Otherwise N |
| Bank\_name |  | M |  |
| Bank\_code |  | M |  |
| Account\_number |  | M |  |
| Bank\_location |  | M |  |
| Account\_suffix |  | M |  |
| Receiver\_addr\_line |  | C |  |
| Receiver\_addr\_line2 |  | C |  |
| Receiver\_city |  | C |  |
| Receiver\_state |  | C |  |
| Receiver\_postal\_code |  | C |  |

### Response parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Required | Description |
| transaction\_id | Alpha:32 | M |  |
| mtcn |  |  |  |
| new\_mtcn |  |  |  |
| advisory\_text |  |  | Transaction Date and Time |
| filing\_date |  |  |  |
| filing\_time |  |  |  |
| pin\_message |  |  |  |
| other\_message\_1 |  |  |  |
| other\_message\_2 |  |  |  |

## Transaction Query[Bnnp.501.001.01]

### Features

### Sequence diagram

### Request parameters

### Response parameters

## Fee Inquiry[Bnnp.502.001.01]

### Features

### Sequence diagram

### Request parameters

### Response parameters

# Exceptions

# AccountCheckFile

## PaymentCheckFile

# Append A

## TrxCtgyCd

# Append B

## SysRtnCd

## BizStsCd

# Append C

## Reserved Words