

Mobile Financial Services

**[FRIENDI PAY]**



Table of Contents

[1 About this Document 3](#_Toc80879283)

[1.1 Purpose 3](#_Toc80879284)

[1.2 References 3](#_Toc80879285)

[1.3 Glossary 3](#_Toc80879286)

[2 Level 1 3](#_Toc80879287)

[2.1 Level 2 **Error! Bookmark not defined.**](#_Toc80879288)

**About** this Document

Purpose

This document aims to provide all the information about the FRIENDI PAY application, the used databases, the problems that users may encounter, and their solutions.

Glossary

|  |  |
| --- | --- |
| Item | Description |
| MFS | **M**obile **F**inancial **S**ervices |
| OTP | **O**ne **T**ime **P**assword |
| API | **A**pplication **P**rogramming **I**nterface |
| dob | **D**ate **O**f **B**irth |
| POS | **P**oint **O**f **S**ales |
| KYC | **K**now **Y**our **C**ustomer |
| GOSI | **G**eneral **O**rganizationfor **S**ocial **I**nsurance |

Introduction

Freindi Pay is an electronic wallet affiliated with Virgin Saudi Arabia and linked to the Saudi Investment Bank. It allows users to open an account, add funds, transfer money, and manage beneficiaries efficiently.

Application Interfaces

This section provides an overview of the main interfaces involved in the customer registration and onboarding journey within the application.

**A white and blue logo

Description automatically generatedA hand holding a phone

Description automatically generated**

## Registration Interface

During the registration process, the customer creates an account by completing a series of steps. This includes providing all the required information and ensuring that each field is accurately filled out. Displays the step-by-step process for creating a new account in the application.

A phone number with a number on it

Description automatically generated with medium confidenceA screenshot of a phone

Description automatically generatedGraphical user interface, text

Description automatically generated

**A screenshot of a cell phone

Description automatically generated**Text, letter

Description automatically generatedA screenshot of a login form

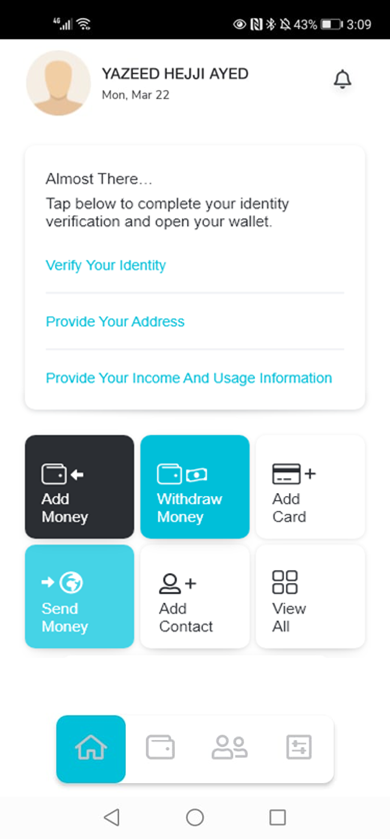
Description automatically generated

**3.2 Identity Verification (Nafath)**

Handles integration with the Nafath system to verify the user's national identity securely.

A screenshot of a login screen

Description automatically generatedScreens screenshot of a phone

Description automatically generated

A qr code on a white background

Description automatically generated

**3.3 Customer National Address (KYC)**

To complete the KYC process, the customer must provide personal and financial details, including their full name, national ID/Iqama number, date of birth, place of residence, income, and intended usage of the account. The bank verifies this information upon submission to issue an IBAN and a wallet number, activating the account.

Text

Description automatically generatedGraphical user interface, text, application

Description automatically generatedMap

Description automatically generated

A screenshot of a phone

AI-generated content may be incorrect.

**3.4 Income and Usage (KYC)**

Captures and validates the customer’s income and service usage data as part of KYC compliance.

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

**3.5 Sign-In Interface**

A screenshot of a login page

Description automatically generatedAllows existing users to securely log in to the application.

A screenshot of a mobile phone

Description automatically generatedA screen shot of a phone

Description automatically generatedA screenshot of a login page

Description automatically generatedA logo for a company

Description automatically generated

Beneficiary

## Add Beneficiary Feature

The Add Beneficiary feature allows customers to securely add and manage recipients for fund transfers within their bank account or wallet. This functionality simplifies transactions by enabling quick and seamless money transfers without repeatedly entering recipient details.

## Beneficiary Registration:

Customers can add a new beneficiary by providing essential details such as the recipient’s name, bank account or wallet number, and bank details (if applicable).

## Verification Process:

To ensure security, beneficiaries must be verified before activation.

Verification methods may include OTP authentication, approval via the banking app, or additional security checks based on regulatory requirements.

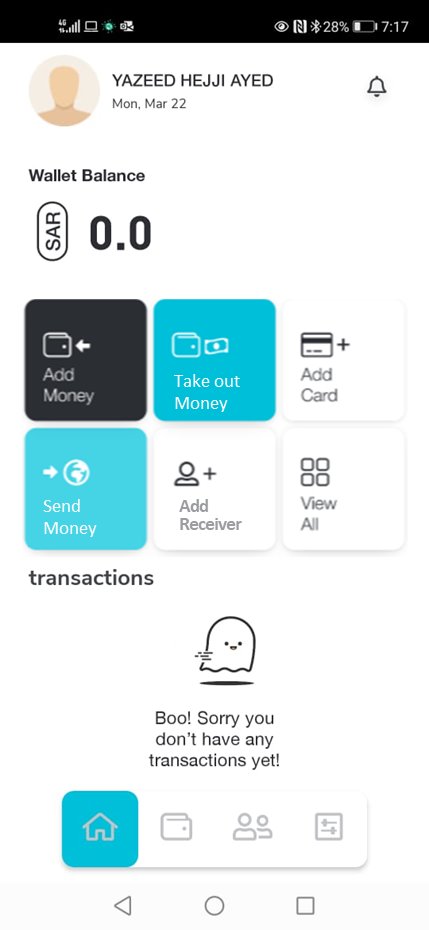
## Beneficiary Management:

Customers can view, modify, or delete saved beneficiaries as needed.

## Transaction Convenience:

Once added and verified, customers can quickly transfer funds to the saved beneficiary without re-entering details.

A screenshot of a phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

A screenshot of a account

Description automatically generated

A screenshot of a phone

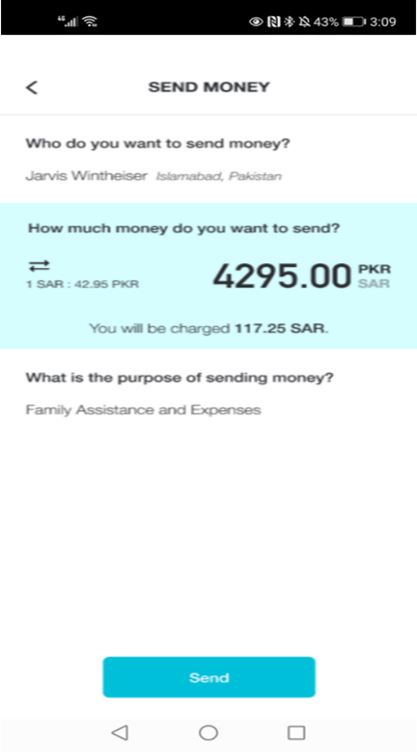
AI-generated content may be incorrect.A screenshot of a phone

Description automatically generated

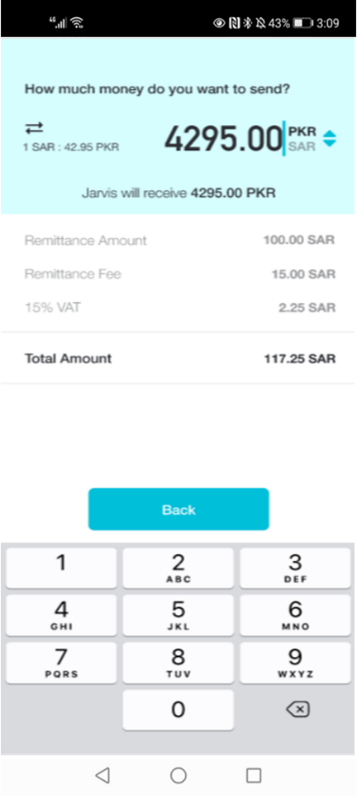
Remittance (Send Money)

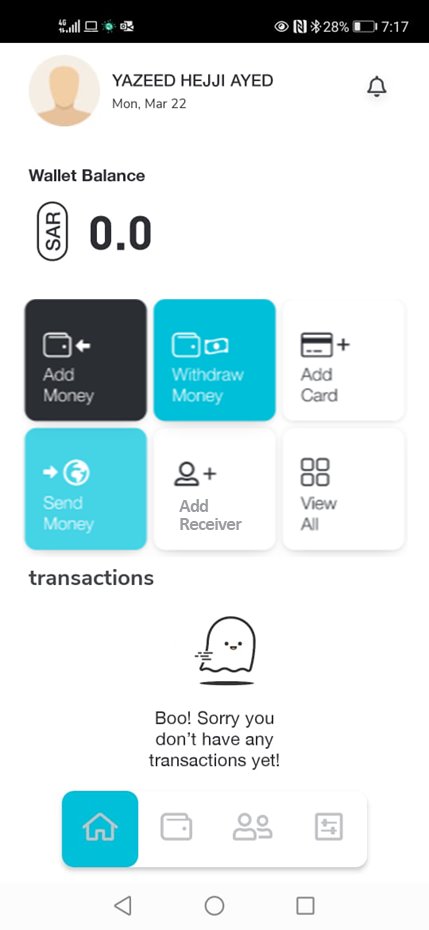
A screenshot of a phone

Description automatically generatedThis service enables customers to perform international money transfers. The maximum transfer limit is determined based on the customer's salary information retrieved from GOSI. According to business rules, the minimum amount that can be transferred is 60 SAR.

A screenshot of a receipt

Description automatically generatedA screenshot of a cell phone

Description automatically generated**A screenshot of a phone

Description automatically generated**

Database DB

## Identity

* **Tables**

1. **AspNetUsers (Login Table)**

* This table stores user login information.
* The primary key used for identifying a user is the ID, which corresponds to the Customer ID.
* When querying this table, always verify the following fields for consistency:
  + **DateOfBirth**
  + **DateOfBirthCalendarType**

**Sample Query**

To retrieve a specific user’s data from the AspNetUsers table, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-identity].dbo.AspNetUsers**

**WHERE id = 'BFD14C61-2E9C-49D9-08EB-08DA49FE1E6D';**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

1. **Identity cards** This table contains customer identity information.

 The key used to retrieve records is the CustomerID.

 When reviewing data, always validate the following fields:

IdentityCardType — Indicates whether the card is **Iqama** or **National ID (Saudi)**.

DateOfBirthCalendarType — Specifies if the date of birth is in the **Hijri** or **Gregorian** calendar format.

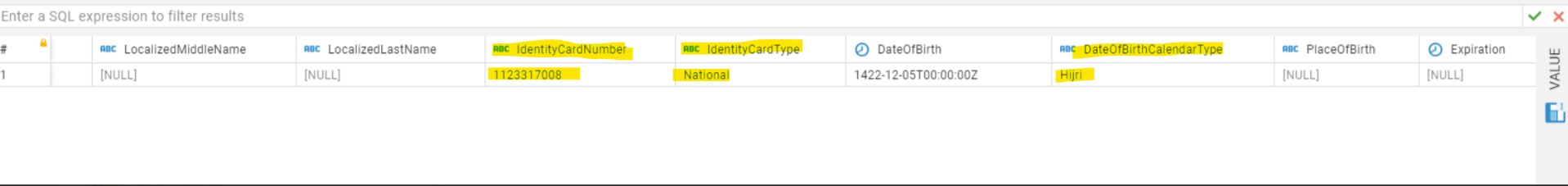
**Sample Query**

To retrieve identity information for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.IdentityCards**

**WHERE customerid = 'BFD14C61-2E9C-49D9-08EB-08DA49FE1E6D';**

A screenshot of a computer

Description automatically generated

1. **Identity Verification Entity**

* **This table is used to track the status of customer identity verification attempts.**
* **The primary identifier is the ID (Customer ID).**
* **When analyzing records, make sure to review the following key fields:**
* **PhoneNumber**
* **IdentityCardNumber**
* **Method — Indicates the verification method used (Virgin or Nafath).**
* **Status — Reflects the result of the verification (Success or Pending).**
* **FailureReason — Provides details in case the verification fails.**

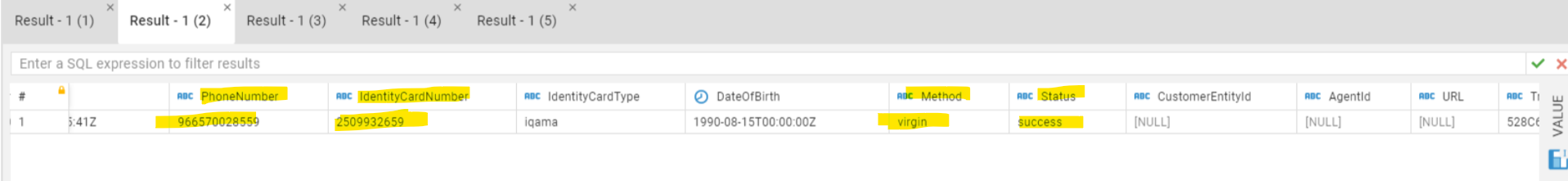
**Sample Query**

To retrieve a specific verification record, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.IdentityVerificationEntity**

**WHERE id = 'C82090B3-082A-4932-0AAB-08DA752BDAAA';**

A screenshot of a computer

Description automatically generated

1. **Customer**

**Nafath Integration**

* This table stores customer profile information, including the status of Nafath identity verification.
* The primary identifier used is the **ID** (Customer ID).
* When reviewing records, ensure the following fields are checked:
  + **BankingCustomerID**
    - A default value of 0 indicates that the callback from Nafath was not received.
  + **Status**
    - Must reflect one of the following for a valid Nafath integration:
      * **Active**
      * **HasRequestedNafath**
      * **HasOpenNafath**
  + **FailureReason**
    - Helps identify issues in the verification process, especially in case of callback errors.

**Common Nafath Errors**

|  |  |
| --- | --- |
| **Error Code** | **Discription** |
| **500** | |  | | --- | | **Rejected – Customer did not approve the request in the Nafath app** |  |  | | --- | |  | |
| **501** | **ID Expired** |
| **502** | **Invalid Legal Status** |
| **503** | **Missing Parameter** |
| **504** | **Age is less than 18 years** |

**Sample Query**

To retrieve customer data using PhoneID, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.Customers**

**WHERE phoneid = '6B4AA523-DAD6-4297-BBA2-08D9E650EF04';**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

1. **Customer Addresses (KYC)**

* This table stores address information associated with each customer.
* The primary identifier used to retrieve records is the CustomerId.

**Sample Query**

To retrieve address data for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.CustomerAddresses**

**WHERE CustomerId = 'FF660EC7-4360-4EB2-AC6C-08D9751EF28E';**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

1. **Customer Incomes (KYC)**

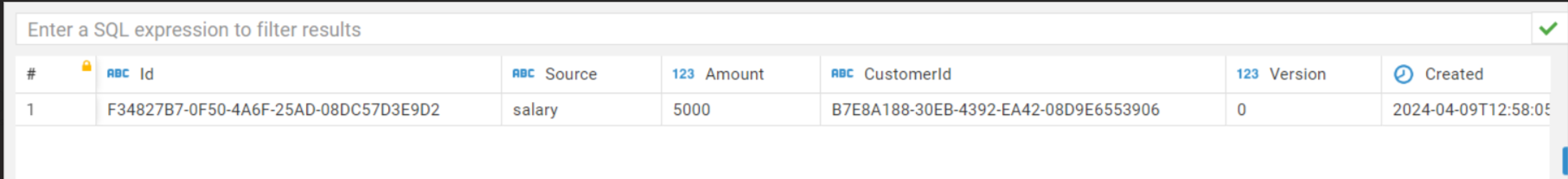
* This table contains income-related details for each customer.
* The data is retrieved using the CustomerId as the key identifier.

**Sample Query**

To fetch income information for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.CustomerIncomes**

**WHERE CustomerId = 'FF660EC7-4360-4EB2-AC6C-08D9751EF28E';**

****

1. **Customer Employments (KYC)**

* This table holds employment information related to each customer.
* Records are retrieved using the CustomerId as the primary reference key.

**Sample Query**

To fetch employment details for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.CustomerEmployments**

**WHERE CustomerId = 'FF660EC7-4360-4EB2-AC6C-08D9751EF28E';**

**A screenshot of a computer

Description automatically generated**

1. **Customer Usage (KYC)**

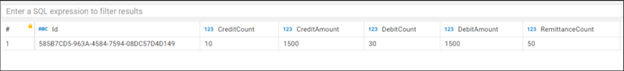
* This table contains information about customer usage behavior and related metrics.
* Data is retrieved using the CustomerId as the key identifier.

**Sample Query**

To retrieve usage data for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.CustomerUsages**

**WHERE CustomerId = 'FF660EC7-4360-4EB2-AC6C-08D9751EF28E';**

A screenshot of a computer

Description automatically generated

1. **Employment Details :**

* This table provides additional employment-related information for each customer.
* The CustomerId is used as the primary key to retrieve specific records.

**Sample Query**

To fetch employment details for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-profiles].dbo.EmploymentDetails**

**WHERE CustomerId = '7334df3b-7c1b-4293-65a3-08dad061224c';**



## Wallets

**Tables:  
- Wallets:**

* This table contains wallet-related data for each customer.
* The primary identifier used to query this table is the ID (Wallet ID).
* When reviewing wallet records, make sure to check the following:
  + **Status —** Typically expected to be Enabled, but may also appear as Pending or BankingUnderReviewed.
  + **Currency —** Should be set to SAR.

**Sample Query**

To retrieve wallet details using the Wallet ID, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-wallets].dbo.Wallets**

**WHERE id = 'BFD14C61-2E9C-49D9-08EB-08DA49FE1E6D';**

A screenshot of a computer

Description automatically generated

**- Consents:**

* This table stores consent records associated with each wallet.
* The primary key used to retrieve data is the ID (Wallet ID).

**Sample Query**

To fetch consent information for a specific wallet, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-wallets].dbo.Consents**

**WHERE id = '0F8F51F8-3E8A-4FA5-AC6D-08D9751EF28E';**

A screenshot of a computer

Description automatically generated **- Accounts:**

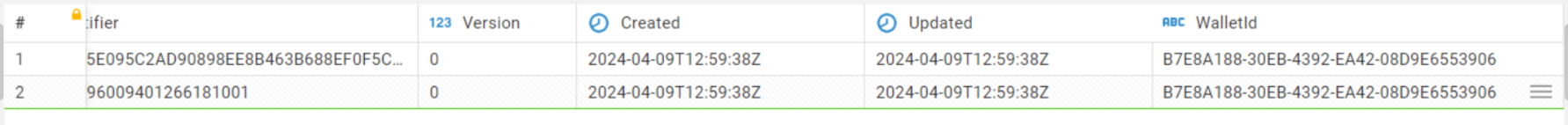
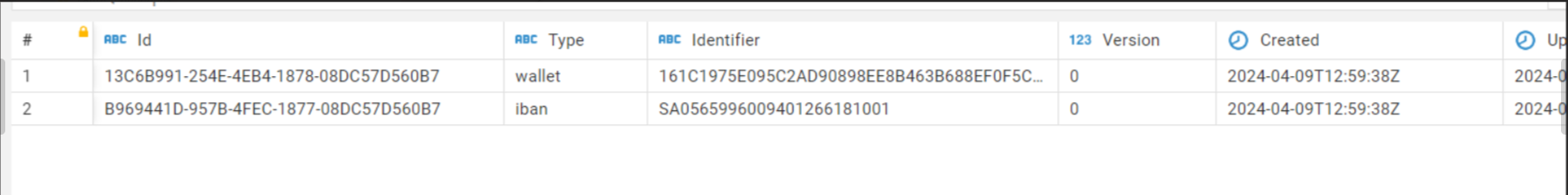
* This table contains account details linked to wallets.
* The primary identifier used to query this table is the Identifier (Wallet ID or unique account identifier).

**Sample Query**

To retrieve account information for a specific identifier, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-wallets].dbo.Accounts**

**WHERE identifier = 'A7E51F0465C3272434F7827FDA201C2CE9096C67';**



## Devices

**Tables:  
- Phone numbers:**

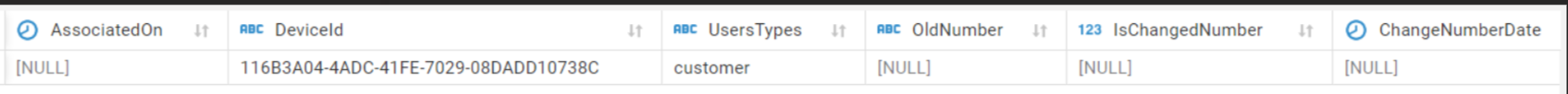
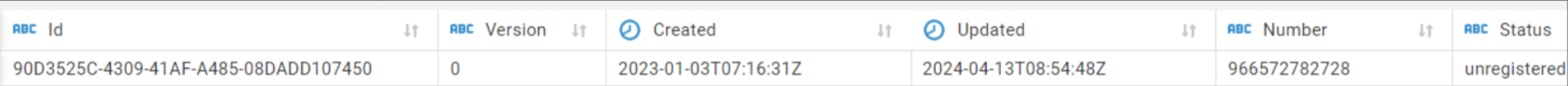
* This table stores phone number records.
* Data is retrieved using the Number field as the key.

**Sample Query**

To fetch details for a specific phone number, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-devices].dbo.PhoneNumbers**

**WHERE number = '966572782728';**



## Banking

**Tables:**

**\_\_HttpClientEventMetadata**

* This table logs HTTP request and response metadata for various customer transactions.
* It is commonly used to trace all transactions performed by a customer on the same day.
* The ExtraData field contains the Customer ID, which is used to retrieve relevant entries.

**Important Fields to Check**

* RequestBody — The payload sent in the request.
* ResponseContent — The content returned from the API or service.

**Sample Query**

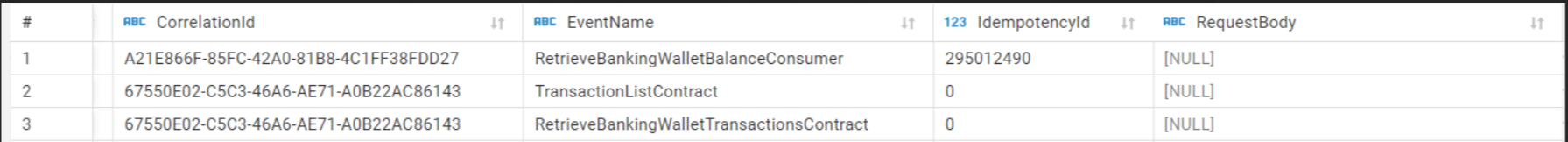
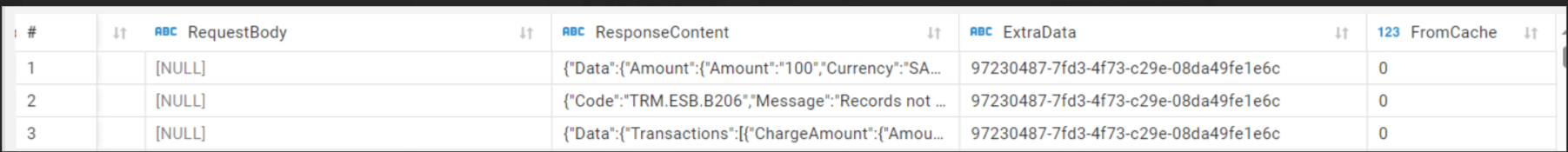
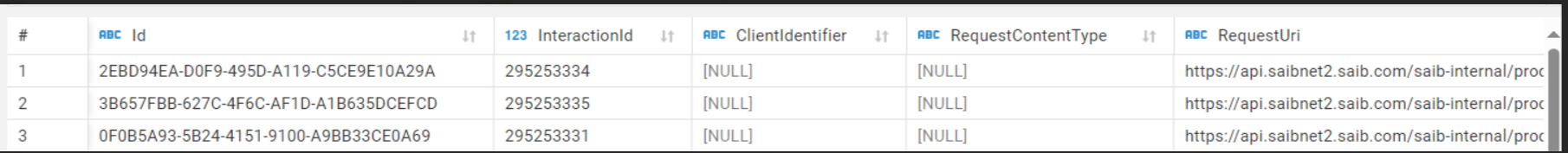
To retrieve the latest 500 transaction metadata entries for a specific customer, use the following SQL query:

**SELECT TOP (500) \***

**FROM [com-mfs-dbs-prd-banking].dbo.[\_\_HttpClientEventMetadata] p**

**WHERE p.ExtraData = '69903dc2-9d5a-42c1-4a55-08db23a9cce1'**

**ORDER BY p.Created DESC;**

 **\_\_HttpClientEventMetadata\_archive**

* This table contains archived metadata for all past customer transactions, excluding those that occurred today.
* The ExtraData field holds the Customer ID, which is used to filter records for a specific user.

**Key Fields to Review**

* **RequestBody —** The content sent in the original request.
* **ResponseContent —** The response returned from the corresponding service.

**Sample Query**

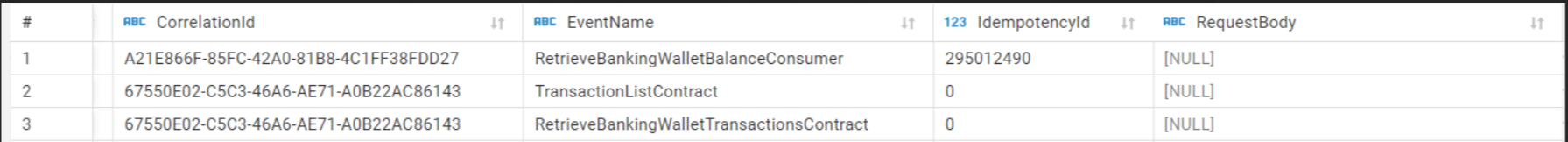
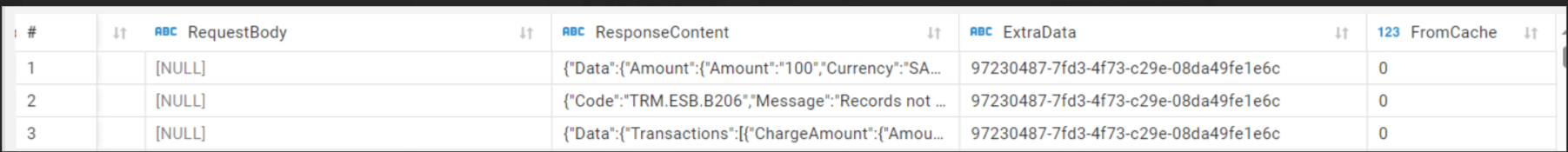
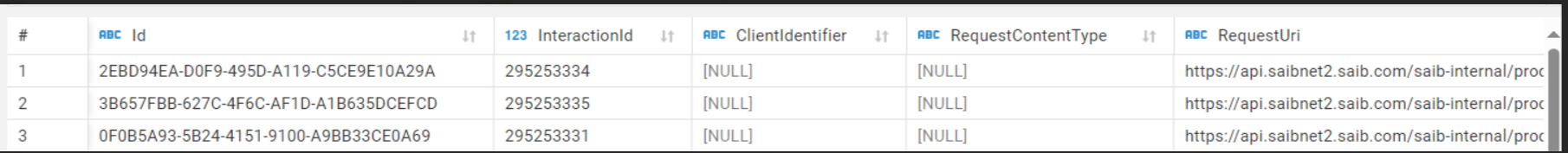
To retrieve the latest 200 archived transaction records for a specific customer, use the following SQL query:

**SELECT TOP (200) \***

**FROM [com-mfs-dbs-prd-banking].dbo.[\_\_HttpClientEventMetadata\_archive] p**

**WHERE p.ExtraData = '69903dc2-9d5a-42c1-4a55-08db23a9cce1'**

**ORDER BY p.Created DESC;**



## Beneficiaries

**Tables:  
- Beneficiaries:**

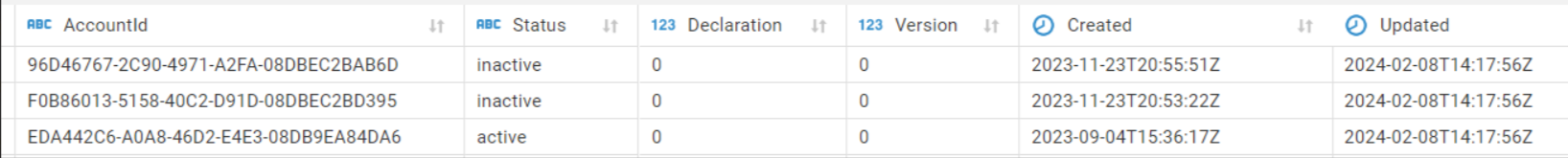
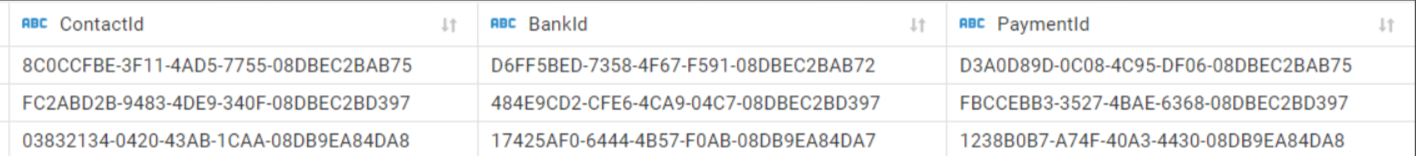
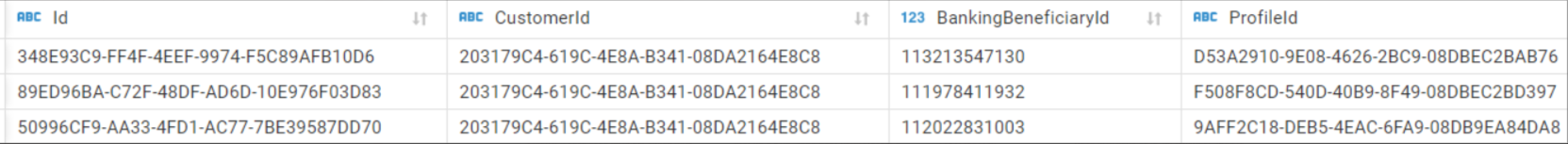
* This table contains information about the beneficiaries linked to a customer.
* The primary key used to retrieve data is the CustomerId.

**Sample Query**

To retrieve beneficiary records for a specific customer, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-beneficiaries].dbo.Beneficiaries**

**WHERE CustomerId = 'c856fdf4-e5ce-43e9-3a16-08dc57d7297b';**

 **- Beneficiary profiles:**

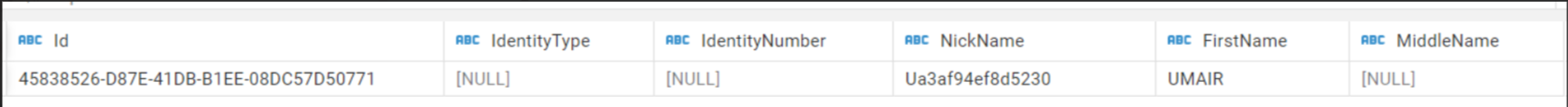
* This table stores detailed profile information for each beneficiary.
* Data is retrieved using the ID field, which represents the Profile ID.

**Sample Query**

To fetch a specific beneficiary profile, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-beneficiaries].dbo.BeneficiaryProfiles**

**WHERE id = '45838526-D87E-41DB-B1EE-08DC57D50771';**



**- Beneficiary Contacts:**

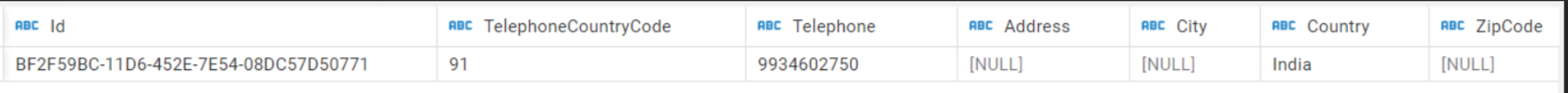
* This table contains contact information associated with beneficiaries.
* The primary identifier used to query this table is the ID (Contact ID).

**Sample Query**

To retrieve contact details for a specific beneficiary, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-beneficiaries].dbo.BeneficiaryContacts**

**WHERE id = 'BF2F59BC-11D6-452E-7E54-08DC57D50771';**



**- Beneficiary Banks:**

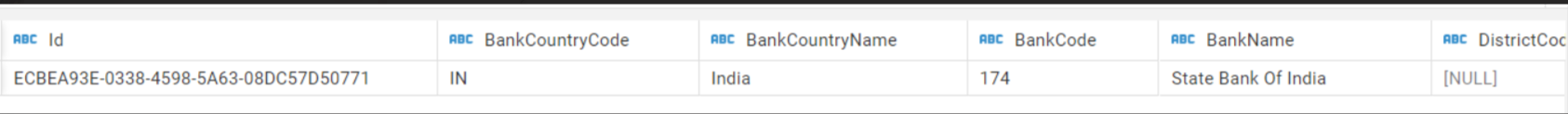
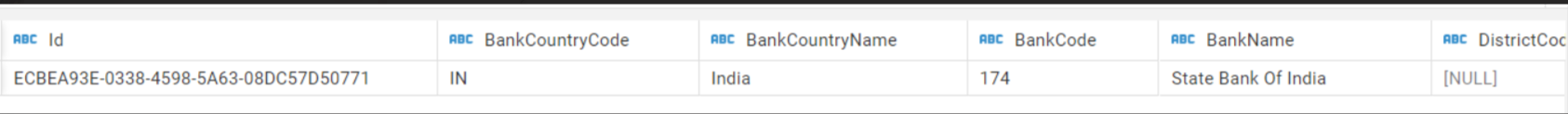
* This table holds bank account details linked to each beneficiary.
* Data is retrieved using the ID field, which represents the Banking Beneficiary ID.

**Sample Query**

To retrieve banking information for a specific beneficiary, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-beneficiaries].dbo.BeneficiaryBanks**

**WHERE id = 'ECBEA93E-0338-4598-5A63-08DC57D50771';**



**- Beneficiary Payments:**

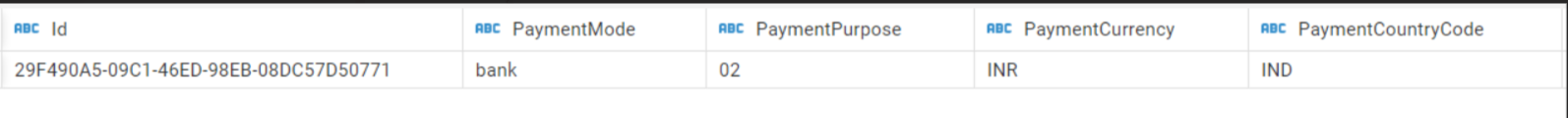
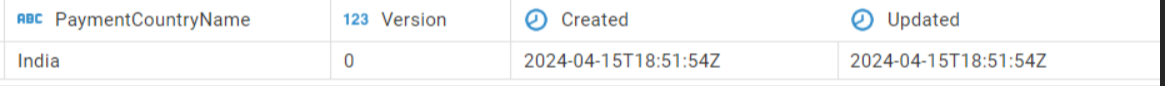
* This table stores payment transaction details made to beneficiaries.
* Records are retrieved using the ID field, which represents the Payment ID.

**Sample Query**

To fetch payment information for a specific transaction, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-beneficiaries].dbo.BeneficiaryPayments**

**WHERE id = '29F490A5-09C1-46ED-98EB-08DC57D50771';**

 ****

**- Beneficiary Accounts:**

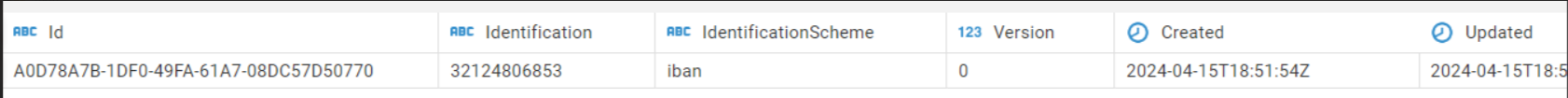
* This table contains account-related information associated with beneficiaries.
* The primary identifier used to retrieve data is the ID, which represents the Account ID.

**Sample Query**

To retrieve account details for a specific beneficiary, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-beneficiaries].dbo.BeneficiaryAccounts**

**WHERE id = 'A0D78A7B-1DF0-49FA-61A7-08DC57D50770';**



## Notification

**Tables:  
- Notification Transactions:**

* This table logs all notifications sent to customers (e.g., SMS, email, push notifications).
* Data is commonly retrieved using the Identifier field, which may include transaction or reference codes.

**Sample Query**

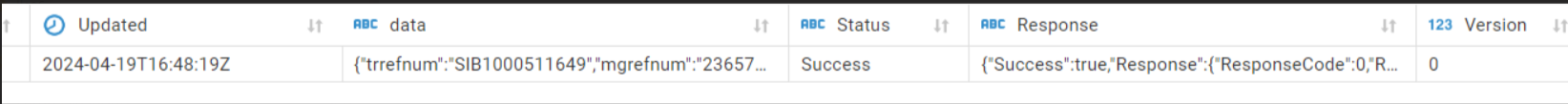
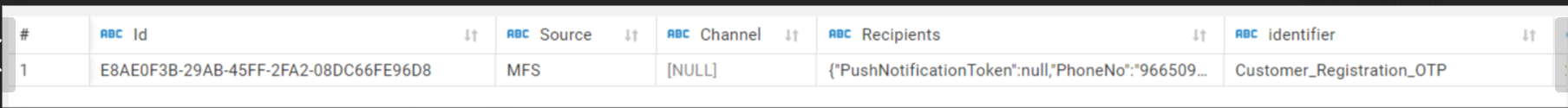
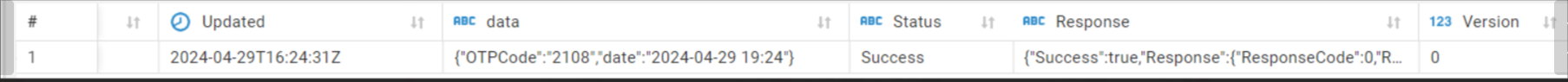
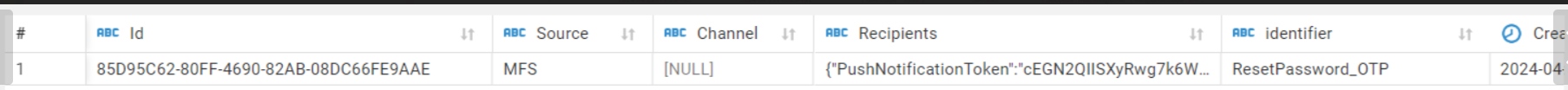
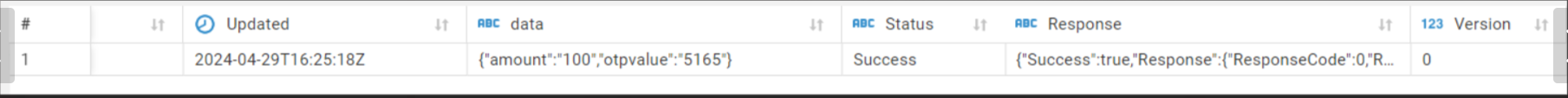
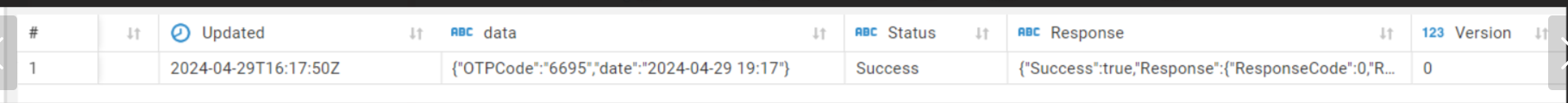
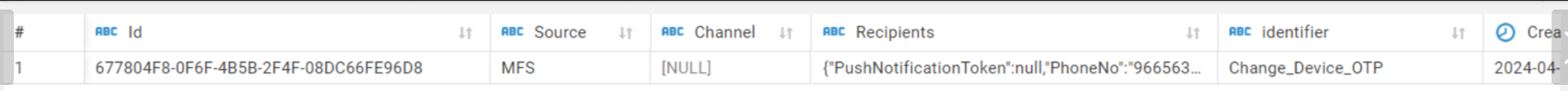
To retrieve the latest notification transaction that contains 'IR' in the identifier, use the following SQL query:

**SELECT TOP (1) \***

**FROM [com-mfs-dbs-prd-notification].dbo.NotificationTransactions**

**WHERE Identifier LIKE '%IR%'**

**ORDER BY Created DESC;**

**Identifier :  
-IR\_Initiated (Remittance OTP)** **-Customer\_Registration\_OTP** **-ResetPassword\_OTP** **-Refund\_OTP** **-Change\_Device\_OTP**

## HyperPay

**Tables:  
- Cards Detail Profile:**

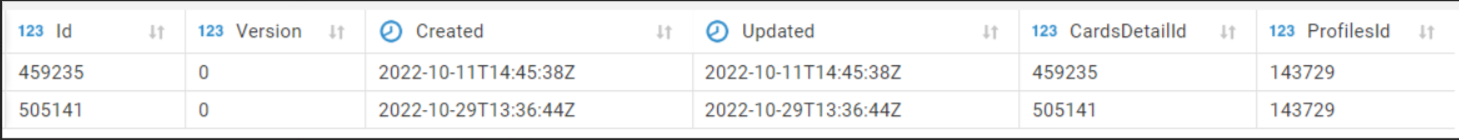
* This table stores detailed information about customer card profiles used for payments.
* Data is retrieved using the ProfilesId, which represents the Profile ID.

**Sample Query**

To retrieve card details for a specific profile, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-hyperpay].dbo.CardsDetailProfile**

**WHERE ProfilesId = '143729';**

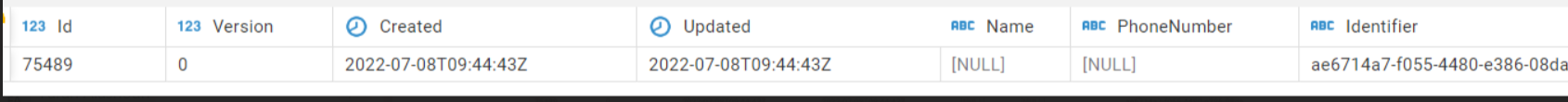
 **- Profiles:**

* This table contains profile information related to payment processing.
* Data is retrieved using the Id field, which represents the unique Profile ID.

**Sample Query**

To fetch profile details for a specific ID, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-HyperPay].dbo.Profiles WHERE id = 75489;**



**- Cards Details:**

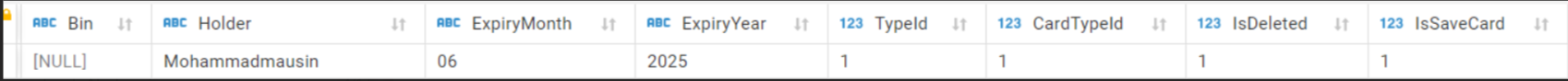
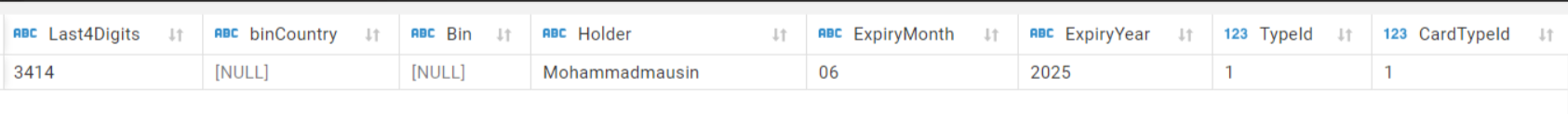
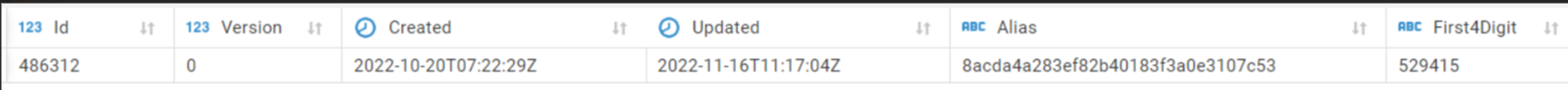
* This table contains detailed records of customer cards.
* Data is retrieved using the ID field, which identifies individual card records.

**Sample Query**

To retrieve card details for specific card IDs, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-hyperpay].dbo.CardsDetails**

**WHERE id IN (364111, 372042, 372043, 464446, 486312);**

 **- Check Out Details:**

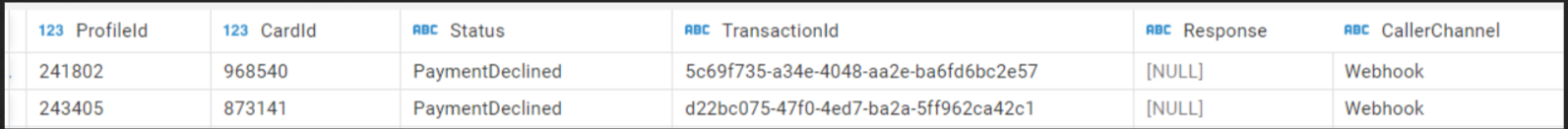
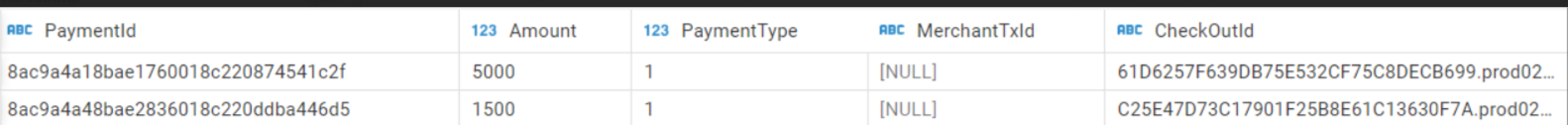
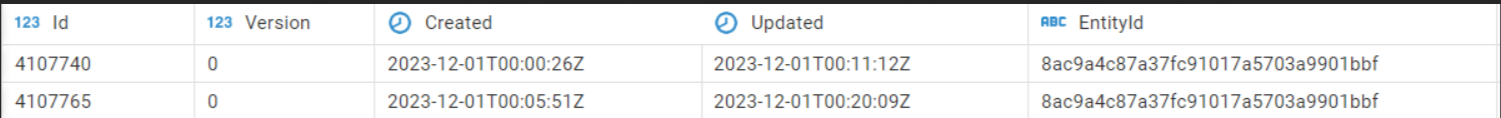
* This table stores details related to payment checkouts.
* Data is retrieved using the WebhookCode to identify specific webhook events.
* Additionally, filtering by the Created date helps narrow down results within a time range.

**Sample Query**

To retrieve checkout details for a specific webhook code from December 1, 2023 onwards, use the following SQL query:

**SELECT \* FROM [com-mfs-dbs-prd-HyperPay].dbo.CheckOutDetails**

**WHERE webhookcode = '100.380.501' AND created >= '2023-12-01';**



## Web API Log

**Tables:  
- Web API logs:** Displays all transactions performed by the customer on the current day (today).

**- Web API logs stage:** Displays all transactions the customer has completed historically, excluding those performed today.

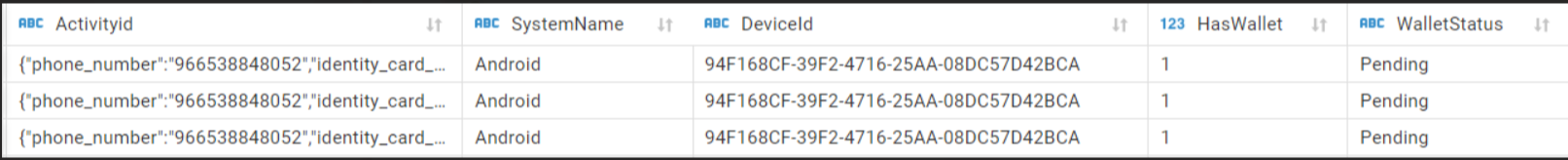
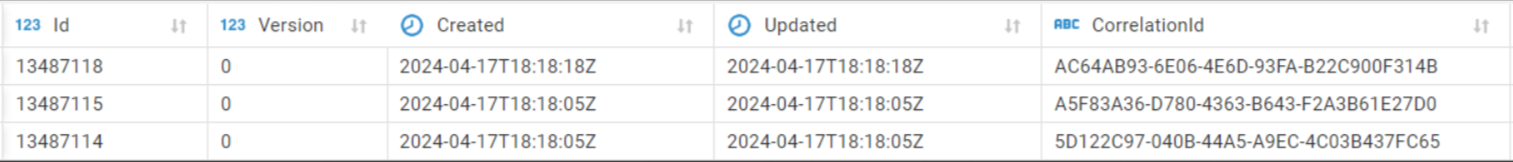
**- Web API back:** Displays all historical transactions (excluding today) related to the customer stored in the profiles database.

**6.9 Logger Controller:**

**Tables:  
- logger controller:**

**select \* from [com-mfs-dbs-loggercontroller]..loggercontroller**

**where phonenumber='966538848052'**

**order by created desc**A screenshot of a computer

Description automatically generated

**Sample of ticket:**

|  |  |
| --- | --- |
| Name Of Ticket | Page Number |
| Log In -Technical Error | **16** |
| Update Id Expiry Date | **21** |
| Technical error while processing the remittance | **14/21** |
| Change Phone Number | **21/27** |
| Account Status | **20** |
| Error code while adding beneficiary | **22/23** |
| Refund Process Failure | **20** |
| SMS not received -remittance confirmation | **21** |
| Top Up process failure | **11/21** |
| Process completed; Amount not added | **21** |