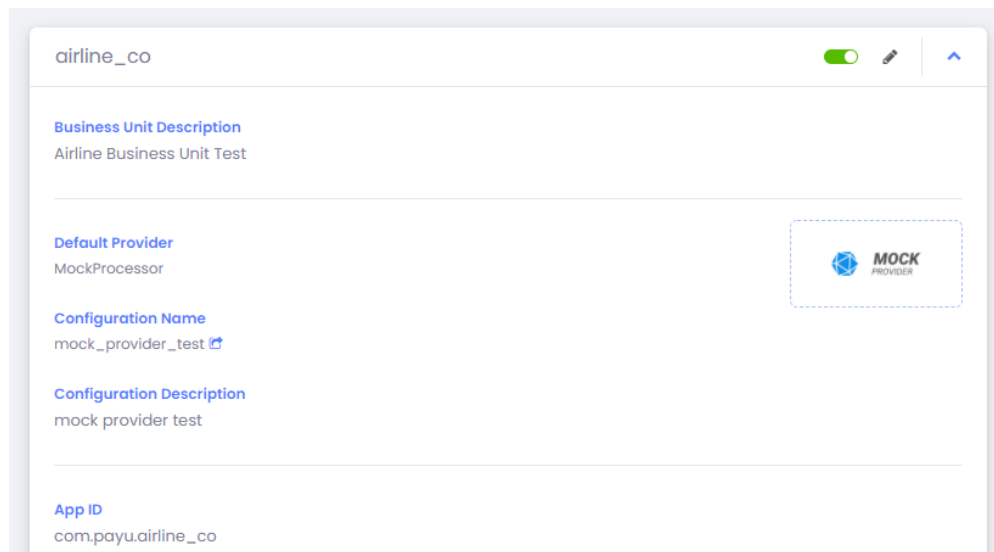


Sales Engineer Technical test

1. CREATE A CONTROL PAYMENTSOS ACCOUNT

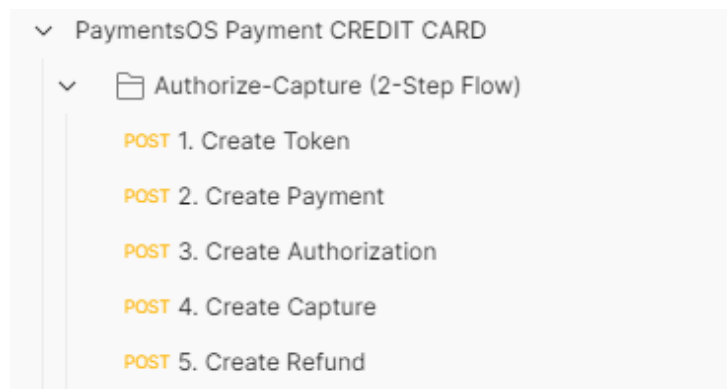
In this section I create a business unit, AirlineCo, and associate a payments provider MockUp.

App Account ID:
98b5ca97-bbd1-46a8-9514-edab7b5e4f1b



2. COMPLETE A PAYMENT FLOW WITH THE FOLLOWING API CALLS (DO NOT USE THE BODY BUILDER TOOL ON THIS POINT):

In this section, I create a payment flow for a credit card payment with a 2 step flow. Request and responses described to:



arjul1989@gmail.com

TEST ON POSTMAN

CREDIT CARD

PaymentsOS Payment CREDIT CARD

No Environment, just now

View Summary

Run Again

New

Export Results

All Tests

Passed (0)

Failed (0)

Iteration 1

POST	1. Create Token	https://api.paymentsos.com/tokens	/ Authorize-Capture (2-Step Flow) / 1...	201 Created	514 ms	1.394 KB
	This request does not have any tests.					
POST	2. Create Payment	https://api.paymentsos.com/payments	/ Authorize-Capture (2-Step Flo...	201 Created	253 ms	1.178 KB
	This request does not have any tests.					
POST	3. Create Authoriz...	https://api.paymentsos.com/payments/{{paymentid}}/auth...	/ Authori...	201 Created	408 ms	1.941 KB
	This request does not have any tests.					
POST	4. Create Capture	https://api.paymentsos.com/payments/{{paymentid}}/captures	/ Authoriz...	201 Created	310 ms	1.541 KB
	This request does not have any tests.					
POST	5. Create Refund	https://api.paymentsos.com/payments/{{paymentid}}/refunds	/ Authorize-...	201 Created	353 ms	1.537 KB
	This request does not have any tests.					

CASH

PaymentsOS Payment CASH OXXO

No Environment, just now

View Summary

Run Again

New

Export Results

All Tests

Passed (0)

Failed (0)

Iteration 1

POST Create Payment

https://api.paymentsos.com/payments

/ Alternative Payment Methods / C...

201 Created

895 ms

1.117 KB

This request does not have any tests.

POST Create Charge

https://api.paymentsos.com/payments/{{paymentid}}/charges

/ Alternative ...

201 Created

355 ms

1.644 KB


This request does not have any tests.




Arley Julian Gutierrez Barrera
arjul1989@gmail.com

- **SEND BACK YOUR SECURE FIELDS CODE, ALL REQUESTS, AND ALL RESPONSES. FOR THIS POINT, YOU HAVE TO USE GITHUB AND SHARE WITH US THE GITHUB LINK TO REVIEW THE SOURCE CODE.**

All request and responses have been submitted to a github public repository.

<https://github.com/JulesSetUp/paymetsOsTest>

 arjul1989 Add files via upload 7275789 2 minutes ago 10 commits

 Airline CO	Add files via upload	5 minutes ago
 PaymentsOS API Flow	Add files via upload	2 minutes ago
 README.md	Update README.md	27 minutes ago

README.md

paymetsOsTest

PaymentsOs Api Call

V1 07032021

This repository describes the activities carried out to test payU's PaymentsOs platform.

The calls to the different APIs in POSTMAN for the flow of payments are described, as well as a small website that models the creation of a token through tokenization through APIs.

AIRLINECO FOLDER:

Contains the generation files of a small website that generates a token by calling APIs

PAYMENTSOS API FLOW:

It contains the .json files exported from POSTMAN, both the calls to the APIs and the generated results.

- **TO TEST THE SECURE FIELDS, FORM FEATURE, PLEASE UPLOAD THE SECURE FIELDS FORM CODE IN A PUBLIC HOSTING AND SHARE THE URL WITH US.**

To test the token, generate function, I created a website that use the API calls for this, instead Secure Field Form feature.

Arley Julian Gutierrez Barrera
arjul1989@gmail.com

← → ↻ 🔒 paymentsostest.000webhostapp.com

Aplicaciones Correo Uniandes GMail Bancolombia Sucur... Aplicaciones Avianca Servicios Pag

Name

Arley Gutierrez

Card Number

4896350049302514

Expiration Date

12/21

Create Token API call

Token

1d055356-bcf6-4f93-aebb-9997685665e7

The site has been deployed at this URL: <https://paymentsostest.000webhostapp.com/>

- **KEEPING IN MIND YOUR EXPERIENCE WITH THIS TECHNICAL ASSESSMENT, PLEASE SHARE WITH US WHAT IS THE DEFINITION OF TOKENIZATION, AUTHORIZE, CAPTURE, CHARGE, VOID, AND REFUND.**

TOKENIZATION: Tokenize, in payment flows, refers to converting payment data into a code that can be safely transferable.

This prevents some companies from having to implement very robust security methods or architecture models for PCI compliance.

Also, by summarizing the data in a single piece of information (token) it reduces errors and improves the integration of payment systems.

AUTHORIZE AND CAPTURE: When a payment is executed it can be of two methods, one is where it is first authorized and then the capture is made.

For example, Uber makes an **AUTHORIZATION** when placing an order, the money is withheld from the credit card and once the service is finished, the **CAPTURE** is made.

CHARGE: Contrary to the authorization and capture model, Charging is making the due directly to the payer's card.

Arley Julian Gutierrez Barrera
arjul1989@gmail.com

For example, Rappi at the time of a purchase does not authorize, but instead makes an immediate charge to the TC. If the service is canceled, Rappi cannot make a **VOID**, but must make a **REFUND**.

VOID: When a transaction is canceled, without having made a charge, a **VOID** is executed. The transaction can be canceled, for example when the bank rejects the payment or when the Uber user cancels the service before it is provided. There must be (in most cases) a prior authorization.

REFUND: When the funds are already debited and leave the payer's accounts towards the merchant (aggregator or gateway model), the money can only be returned through a Refund.

The **REFUND** is a new bank transaction, its movements are registered, and it can be for a different value than the one paid.

Rappi, for example, sometimes makes partial refunds for purchases or the airlines compensate TC for canceled flights through Refund.

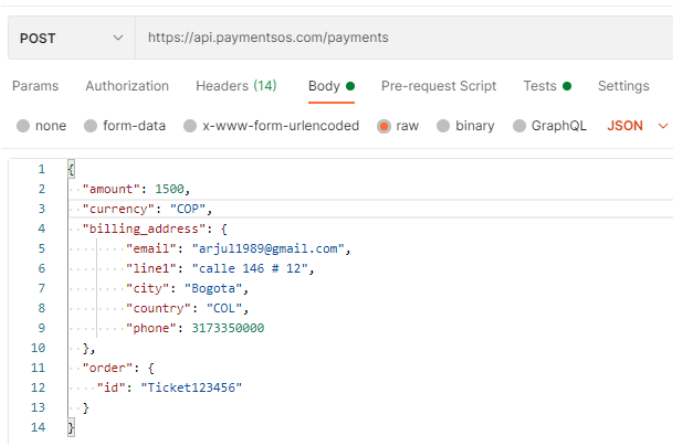
- **AND FINALLY, SHARE WITH US YOUR MERCHANT ONBOARDING, INTEGRATION AND DOCUMENTATION EXPERIENCE NOTES ON A FILE ON THE SAME GITHUB RESOURCE.**

This document will be uploaded at Github site <https://github.com/JulesSetUp/paymentsOsTest>

PLUS POINTS

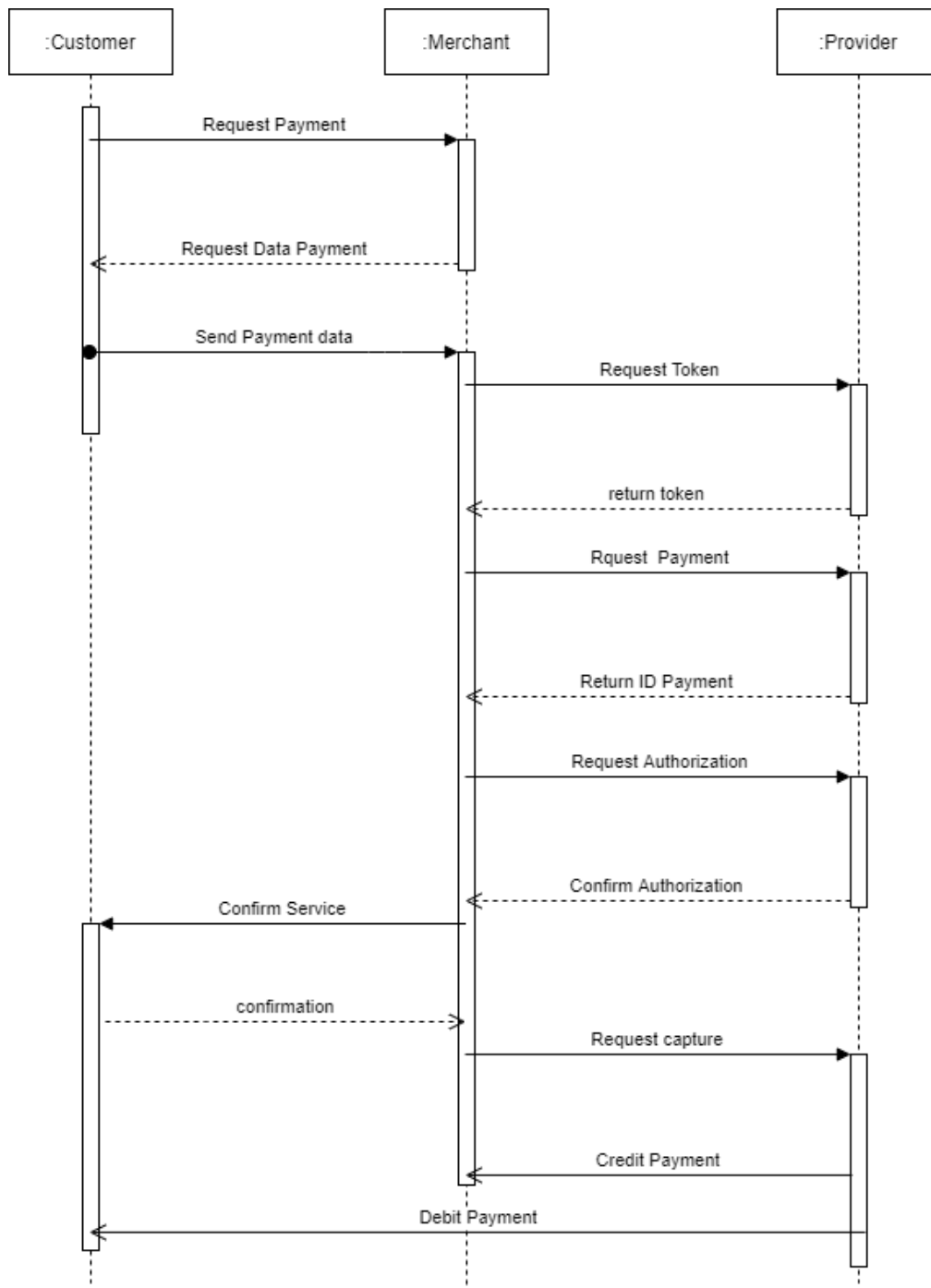
- **ADD THE BILLING INFORMATION (EMAIL, LINE1, CITY, COUNTRY, PHONE) TO THE SECURE FIELDS FORM.**

The billing information has been added to the Creation Payment step with credit card



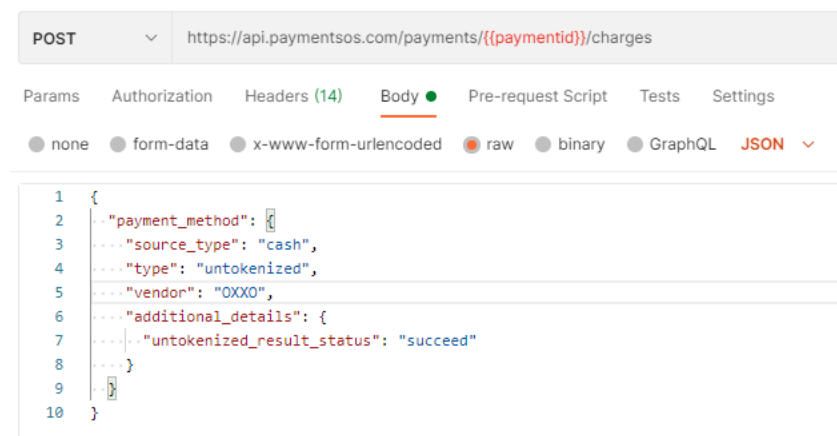
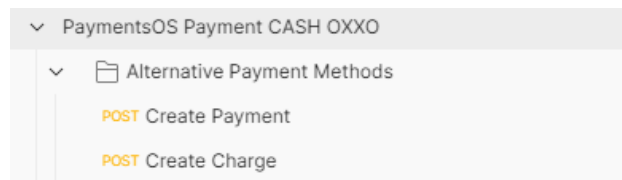
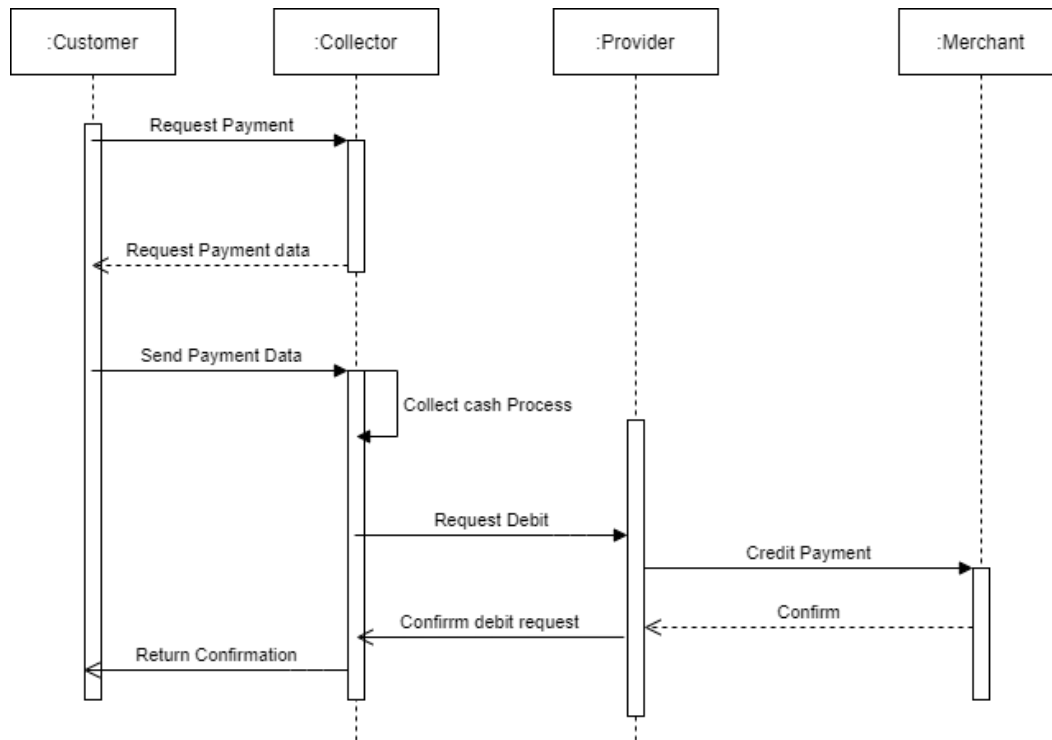
- **CREATE A POSTMAN COLLECTION AND A SEQUENCE DIAGRAM (TWO ACTORS, MERCHANT AND PAYU) EXPLAINING THE FLOWS FOR THE FOLLOWING SCENARIO: PAYU HAS A MERCHANT IN MEXICO THAT WANTS TO INTEGRATE CREDIT CARDS TO PROCESS WITHOUT CVV(SEcurity CODE) AND CASH PAYMENTS(OXXO).**

CREDITCARD PAYMENT 2 STEP FLOW SEQUENCE DIAGRAM (HAPPY PATH WITHOUT VOID OR REFUND PROCESS)



Arley Julian Gutierrez Barrera
arjul1989@gmail.com

CASH PAYMENTS SEQUENCE DIAGRAM (OXXO STORE)



- **SEND THE URL OF THE POSTMAN COLLECTION WITH FOLDERS ONE FOR CASH AND THE OTHER FOR CREDIT CARDS.**

The collection has been charged at <https://github.com/JulesSetUp/paymentsOsTest>