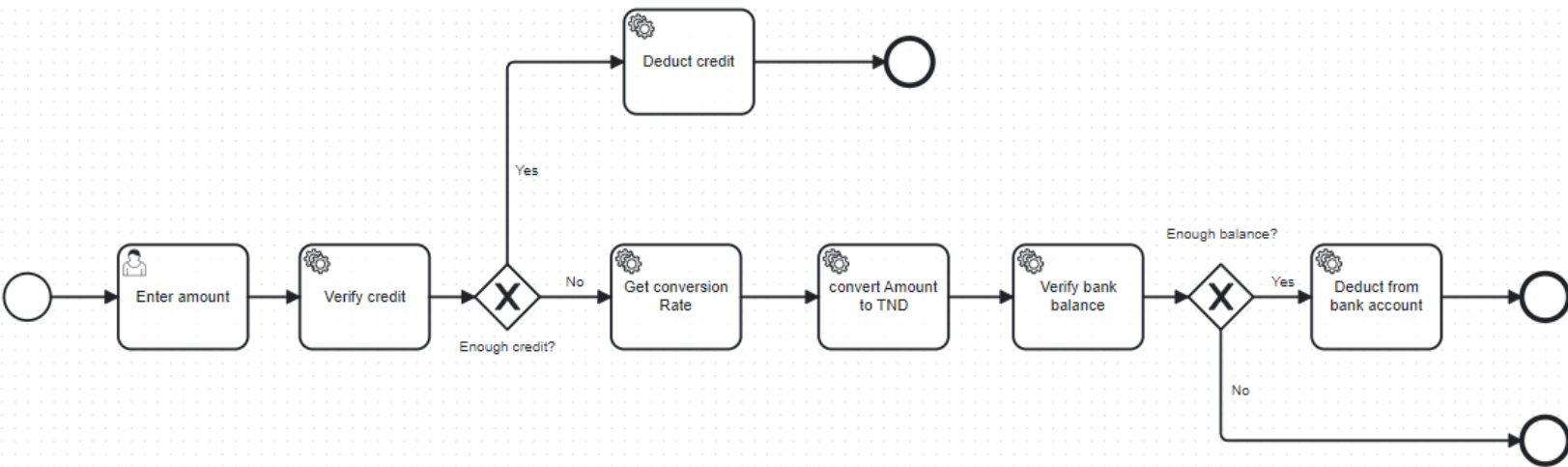


Modélisation du Processus Métier



Verify credit : à travers l'API (GET /credit/verify): Cette API accepte le nom d'utilisateur et le montant en entrée pour vérifier la disponibilité d'un solde suffisant au niveau du crédit amazon.

Deduct credit : à travers l'API (PUT /credit/deduct): Cette API permet de soustraire le montant spécifié du solde crédit.

Get conversion Rate : à travers l'API (GET /rate): En sollicitant cette API, le système extrait le taux de conversion entre deux devises spécifiées (currency to et currency from).

Convert amount to TND : à travers l'API (POST /conversion/convertAmount): Cette API, avec le taux de conversion et le montant à convertir en entrée, calcule le montant converti.

Verify bank balance : à travers l'API (GET /bankwallet/verify): L'utilisation de cette API avec le nom d'utilisateur et le montant converti en entrée permet de vérifier la disponibilité d'un solde bancaire suffisant.

Deduct from bank account : à travers l'API (PUT /bankwallet/deduct): Cette API permet de soustraire le montant spécifié du solde du compte bancaire associé.

Enter amount Configurations

Forms	• ▼
Type	
Generated Task Forms ▼	
Form fields	+ 2 ▼
▶ montant	
▶ user_name	
Task listeners	+
Asynchronous continuations	▶
Inputs	+
Outputs	+

Verify credit Configurations

Connector inputs	+ 2 ▼	Connector outputs	+ 1 ▼
▼ method		▼ enoughCredit	
Local variable name		Process variable name	
method		enoughCredit	
Assignment type		Assignment type	
String or expression ▼		Script ▼	
Value		Format	
GET		JavaScript	
Start typing "\${}" to create an expression.		Type	
▼ url		Inline script ▼	
Local variable name		Script	
url		<pre>\${S(response).prop("enoughCredit").value().toString()}</pre>	
Assignment type			
String or expression ▼			
Value			
<pre>http://localhost:8090/api/credit/verify/\${user_name}/\${montant}</pre>			
Start typing "\${}" to create an expression.			

Deduct credit Configurations

Connector inputs + 2 ▼

▼ method

Local variable name

method

Assignment type

String or expression ▼

Value

PUT

Start typing "\${}" to create an expression.

▼ url

Local variable name

url

Assignment type

String or expression ▼

Value

http://localhost:8090/api/credit/deduct/\${user_name}/\${montant}

Start typing "\${}" to create an expression.

Get Conversion Rate Configurations

Connector inputs	Connector outputs
<div>method</div> <div>Local variable name method</div> <div>Assignment type String or expression</div> <div>Value GET</div> <div>Start typing "{\$}" to create an expression.</div>	<div>rate</div> <div>Process variable name rate</div> <div>Assignment type Script</div> <div>Format JavaScript</div> <div>Type Inline script</div> <div>Script \${S(response).prop("rate").value().toString()}</div>
<div>url</div> <div>Local variable name url</div> <div>Assignment type String or expression</div> <div>Value http://localhost:8090/api/rate/USD/TND</div>	

Convert Amount to TND Configurations

Connector inputs	Connector outputs
<div>method</div> <div>Local variable name method</div> <div>Assignment type String or expression</div> <div>Value POST</div> <div>Start typing "{\$}" to create an expression.</div>	<div>convertedAmount</div> <div>Process variable name convertedAmount</div> <div>Assignment type Script</div> <div>Format JavaScript</div> <div>Type Inline script</div> <div>Script \${S(response).prop("convertedAmount").value().toString()}</div>
<div>url</div> <div>Local variable name url</div> <div>Assignment type String or expression</div> <div>Value http://localhost:8090/api/conversion/convertAmount/USD/TND/\${rate}/\${montant}</div>	

Verify bank balance Configurations

Connector inputs

+ 2 ▼

▼ method

Local variable name

method

Assignment type

String or expression ▼

Value

GET

Start typing "\${}" to create an expression.

▼ url

Local variable name

url

Assignment type

String or expression ▼

Value

`http://localhost:8090/api/bankwallet/verify/${user_name}/${convertedAmount}`

Start typing "\${}" to create an expression.

Connector outputs

+ 1 ▼

▼ enoughMoney

Process variable name

enoughMoney

Assignment type

Script ▼

Format

JavaScript

Type

Inline script ▼

Script

`${S(response).prop("enoughMoney").value().toString()}`

Deduct from bank account Configurations

Connector inputs + 2 ▼

▼ method

Local variable name

method

Assignment type

String or expression ▼

Value

PUT

Start typing "\${}" to create an expression.

▼ url

Local variable name

url

Assignment type


String or expression ▼

Value

http://localhost:8090/api/bankwallet/deduct/

Start typing "\${}" to create an expression.

Déploiement du processus

 **Process definition deployed** ✕

Process definition ID: Amazon:1:ff72
1fa7-b30d-11ee-ab8b-e00af64786
c4

[Open in Camunda Cockpit](#)

4 process definitions deployed					List	Previews
State	Incidents	Running Instances	Name	Tenant ID		
✓	0	1	Amazon			
✓	0	6	Invoice Receipt			
✓	0	11	Process_0o97ztl			
✓	0	2	Review Invoice			

Exécution du processus

✓

Process instance started

✕

Process instance ID: 05eb3a68-b30e-11ee-ab8b-e00af64786c4

[Open in Camunda Cockpit](#)

Camunda Cockpit

Processes

Decisions

Human Tasks

More

Demo De

Dashboard » Processes » Amazon : Runtime

Definition Version: 1

Version Tag: null

Definition ID: Amazon:1:ff721fa7-b30d-11ee-ab8b...

Definition Key: Amazon

Definition Name: Amazon

History Time To Live: null

Tenant ID: null

Deployment ID: #347a65-b30d-11ee-ab8b-e00af64...

Instances Running: 1

Activity Instance Statistics: on

```
graph LR; Start(( )) --> EnterAmount[Enter amount]; EnterAmount --> VerifyCredit[Verify credit]; VerifyCredit --> EnoughCredit{Enough credit?}; EnoughCredit -- Yes --> DeductCredit[Deduct credit]; DeductCredit --> End1(( )); EnoughCredit -- No --> GetConversionRate[Get conversion Rate]; GetConversionRate --> ConvertAmount[convert Amount to TND]; ConvertAmount --> VerifyBankBalance[Verify bank balance]; VerifyBankBalance --> EnoughBalance{Enough balance?}; EnoughBalance -- Yes --> DeductFromBankAccount[Deduct from bank account]; DeductFromBankAccount --> End2(( )); EnoughBalance -- No --> End3(( ))
```

Process Instances

Incidents

Called Process Definitions

Job Definitions

Add criteria

1

State

ID

Start Time

Business Key

✓

50e80f70-b313-11ee-ab8b-e00af64786c4

2024-01-14T20:30:03

Solde crédit et bancaire initiale de l'utilisateur

- Solde crédit

	account_id	credit_balance	last_activity_date	user_id	id
▶	1	200	2024-01-14 17:51:57	sousou123	0

- Solde bancaire

	bank_wallet_id	balance	last_activity_date	user_id
▶	1	2000	2024-01-14 17:21:12	sousou123

Cas où le montant est inférieur au solde crédit

- Formulaire de Saisie d'Informations

Enter amount

Amazon

Set follow-up date
Set due date
Add groups
Demo Demo

Form
History
Diagram
Description

Amount

20

User name

sousou123

Save
Complete

- Résultat d'exécution

	account_id	credit_balance	last_activity_date	user_id	id
▶	1	180	2024-01-14 19:11:33	sousou123	0

Cas où le montant est supérieur au solde crédit et inférieur au solde bancaire

- Formulaire de Saisie d'Informations

Enter amount

Amazon

Set follow-up date

Set due date

Add groups

Demo Demo

Form

History

Diagram

Description

Amount

300

User name

sousou123

Save

Complete

- Résultat d'exécution

	bank_wallet_id	balance	last_activity_date	user_id
▶	1	1071.8000000000002	2024-01-14 19:17:55	sousou123

Cas où le montant est supérieur au solde crédit et au solde bancaire

Lorsque le montant à traiter excède à la fois le solde de crédit et le solde bancaire disponibles, le processus se termine sans effectuer de modifications.