



Valitor Salesforce Commerce Cloud SFRA Module

Integrating with Valitor could not be easier. Choose between Hosted, HTTP POST or XML integration options, or alternatively browse our selection of client libraries and e-commerce platform modules. Imagine having full creative control over payment flow, checkout design, and A/B testing without any card data risk.



Table of Contents

1. Summary	3
2. Component Overview	3
2.1 Actors	3
2.2 Functional Overview	3
2.3 Use Cases	3
2.4 Limitations	5
2.5 Compatibility	5
2.6 Privacy, Payment	5
3. Implementation Guide	5
3.1 Setup	5
3.1.1 Import cartridge	5
3.1.2 Metadata import	6
3.1.3 Web Service Import	7
3.1.4 Create AltaPay Payment Processor	8
3.1.5 Payment Methods Import	8
3.2 Configuration	10
3.2.1 Configure AltaPay Site Preferences	10
3.2.2 Select language	12
3.3 Custom Code	13
3.3.1 AltaPay Cartridge Enabled	13
3.3.2 AltaPay Checkout JS	14
3.3.3 Add AltaPay Processor Hook	15
3.3.4 Insert AltaPay processor	16
3.4 External Interfaces	16
3.5 Credit card tokenization	16
3.6 Reconciliation setup	17
3.7 Testing	17
3.7.1 Successful card payment	17
3.7.2 Failed card payment	18
3.7.3 3D Secure card payment	18
3.7.4 3D Secure failed card payment	18
3.7.5 Successful alternative payment	19
3.7.6 Failed alternative payment (Customer cancel)	19
3.7.7 Failed alternative payment	19

3.7.8 Successful alternative payment notification.....	20
3.7.9 Failed alternative payment notification.....	20
3.7.10 Fraud checking (accepted credit card).....	20
3.7.10 Fraud checking (denied credit card).....	21
4. Operations, Maintenance	21
4.1 Data Storage.....	21
4.2 Availability.....	21
4.3 Support.....	21
5. User Guide	22
5.1 Roles, Responsibilities.....	22
5.2 Business Manager.....	22
5.3 Storefront Functionality.....	22

1. Summary

This cartridge enables AltaPay as the Payment Service Provider (PSP) for storefronts using the reference architecture (SFRA) on the Salesforce Commerce Cloud (SFCC) platform. To enable AltaPay as the PSP, the merchant needs to:

1. Sign a contract with AltaPay
2. Install the int_altapay_sfra cartridge
3. Import 'altapay_sfra_metadata.xml'
4. Conduct tests of the payment flow before going live

Before going live, the merchant needs to conduct tests of the payment flow. AltaPay provides a test setup for accepting card and alternative payments. The merchant can also use the AltaPay cartridge to implement captures and releases of payments.

2. Component Overview

2.1 Actors

- o Customer: The Buyer and payer of items at the web shop
- o Merchant: Provides the web shop and items to be sold
- o AltaPay: Processes payment information and verifies payment information provided by the customer with help from an acquirer.

2.2 Functional Overview

The idea of the payment gateway is to allow your customers to perform secure payments without the feeling that they are leaving your web shop. This is possible because AltaPay proxy the payment page from your website – keeping layout and visual identity. The AltaPay Payment Gateway will inject a payment form which reflects the payment method (Credit Card, Bank Payment, etc.).

When doing the integration, you will typically be working against the test environment/gateway, and once the integration is ready it must be enabled for production. AltaPay is using different subdomains for test and production, and this cartridge is using custom site preferences to define if the integration is pointing to either test or production gateway.

2.3 Use Cases

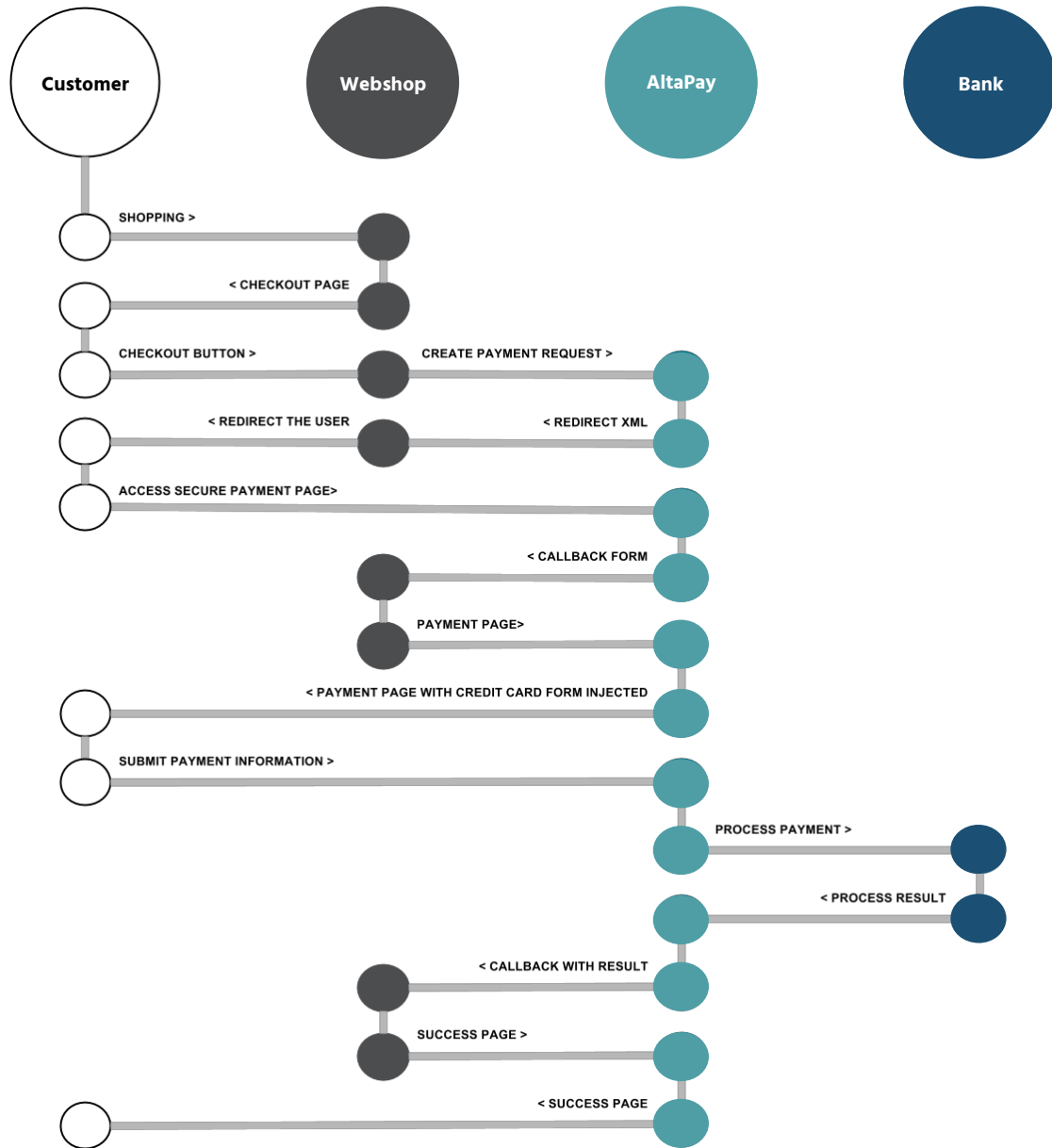
The check-out and payment flow are shown in the below diagram and are briefly described here.

The customer visits the merchant's web shop and add items to the basket. Once the customer has completed the shopping of goods/services and is ready to pay, he or she proceeds to check-out. During the check-out flow details like name, e-mail, shipping- and billing address, voucher-codes is collected, and the customer is ready to pay for the goods/services. The customer selects a preferred payment method and clicks on the place order button to pay. All information about the payment is send to AltaPay which will return a redirect URL to the payment page.

The customer accesses the payment page via AltaPay's secure payment gateway. To ensure that the look'n'feel of the payment page is under your control AltaPay fetch this from your server (using the 'Payment Page' callback URL which is possible to configure in a custom site preference. To ensure that this page is secure AltaPay deliver it using SSL from the gateway and strips all JavaScript and other active content. To limit the work needed designing this page AltaPay proxy all images, CSS, etc. and rewrite any links/forms such that the page works as if it was served directly to the customer.

Depending on the type of payment (credit card, invoice, paypal, etc.) the customer supplies the final payment information. AltaPay will then reach out to the underlying service of the payment-method to validate this information.

Depending on the outcome of the payment AltaPay returns the results to the web shop via a callback (Success/Failure/Open). Most acquirers return the result of the payment immediately so typically "Success" and "Failure" is used. But for some, the acquirer can return the payment in an open state, meaning that the payment is neither success nor failed yet. The result is known later. If a payment is in an open state, AltaPay will use the notification callback to notify when the result of the payment is known. The content returned by these callbacks are displayed to the customer, allowing you to display a message about his or her purchase. If the payment is successful or open the customer is redirected to the confirmation page, and if the payment failed the user is redirected back to check-out flow.



2.4 Limitations

- The merchant will need an AltaPay test/production terminal.
- The merchant must implement their own look and feel on the payment page – otherwise standard page is shown.
- The merchant must implement their own logic to show error messages which are returned on unsuccessful payments.
- Asynchronous payments, which are completed as a successful or declined status at a later stage, the merchant is responsible to notify the customer.

2.5 Compatibility

The int_altapay_sfra cartridge is based on Commerce Cloud version 18.10.

The cartridge is presented as a LINK integration solution for Storefront Reference Architecture (SFRA) v3.2.0, which implies absence of changes to the app_storefront_base cartridge code.

2.6 Privacy, Payment

Information about payment method, debit/credit card data, items, shipping/billing addresses and amount is sent to AltaPay and stored on orders in SFCC.

3. Implementation Guide

3.1 Setup

Download LINK_AltaPay repository from Demandware LINK marketplace.

3.1.1 Import cartridge

1. Import the int_altapay_sfra cartridge into the SFCC Studio Workspace.
2. Open UX Studio.
3. Click File -> Import -> General -> Existing Projects into Workspace.
4. Browse to the directory where you saved the "int_altapay_sfra" cartridge.
5. Click "Finish".
6. Click "OK" when prompted to link the cartridge to the sandbox.
7. Log into the SFCC Business Manager on your sandbox or PIG Instance.
8. Navigate to: Administration -> Manage Sites -> your site -> settings tab.
Add "int_altapay_sfra" cartridge in the cartridge path and click the "Apply" button.

Click **Apply** to save the details. Click **Reset** to revert to the last saved state.

Instance Type: <input type="text" value="Sandbox/Development"/>
Deprecated. The preferred way of configuring HTTP and HTTPS hostnames is by using new features of the site aliases configuration ("Site URLs/Aliases Configuration"). The HTTP/HTTPS hostnames values set in this section will be used if no hostnames are defined by aliases configuration and are intended only to support an older configuration style.
HTTP Hostname: <input type="text"/>
HTTPS Hostname: <input type="text"/>
Instance Type: All
Cartridges: <input type="text" value="int_altapay_sfra:app_storefront_base:modules"/>
Effective Cartridge Path: int_altapay:app_storefront_base:modules:plugin_apple_pay:plugin_facebook:plugin_pinterest_commerce:plugin_web_payments:core
<input type="button" value="Apply"/> <input type="button" value="Reset"/>

3.1.2 Metadata import

1. From the SFCC Business Manager.
2. Navigate to: Administration -> Site Development -> Import & Export.
3. In the "Import & Export Files" section, click the "Upload" link or button.

Import & Export

Meta Data
Import and export your system meta data (i.e., system type extensions, custom object types, custom preference definitions). <input type="button" value="Import"/> <input type="button" value="Export"/>
Geolocations
Import geolocations for a country. <input type="button" value="Import"/>
Import & Export Files
Upload and download your import and export files. <input type="button" value="Upload"/> <input type="button" value="Download"/>

1. Upload the file "altapay_sfra_metadata.xml" from the LINK_AltaPay repository and navigate back.

Upload Import Files

Upload File: <input type="button" value="Choose File"/> altapay_sfra_metadata.xml <input type="button" value="Upload"/>
--

2. In the "Meta Data" section, click the "Import" link or button.

Import & Export

Meta Data

[Import](#) and [export](#) your system meta data (i.e., system type extensions, custom object types, custom preference definitions).

[Import](#) [Export](#)

Geolocations

[Import](#) geolocations for a country.

[Import](#)

Import & Export Files

[Upload](#) and [download](#) your import and export files.

[Upload](#) [Download](#)

3. Select "altapay_sfra_metadata.xml" and click the "Next" button.
4. Wait for validation has completed and click the "Import" button.

3.1.3 Web Service Import

The AltaPay integration is using the web service framework to create web service calls to AltaPay and must therefore be imported to the Sandbox.

1. From the SFCC Business Manager.
2. Navigate to: Administration -> Operations -> Import & Export
3. In the "Import & Export Files" section, click the "Upload" link or button.
4. Upload the file "altapay_sfra_webservice.xml" from the LINK_AltaPay repository.

In the "Services" section, click the "Import" button.

Import & Export

Job Schedules

[Import](#) and [export](#) your job schedules.

[Import](#) [Export](#)

Job Schedules (deprecated)

[Import](#) and [export](#) your deprecated job schedules.

[Import](#) [Export](#)

Services

[Import](#) and [export](#) your services.

[Import](#) [Export](#)

Import & Export Files

[Upload](#) and [download](#) your import and export files.

[Upload](#) [Download](#)

1. Select "altapay_sfra_webservice.xml" and click the "Next" button.
2. Wait for validation and click the "Next" button.
3. Leave "Merge" selected as the import mode and click the "Import" button.

3.1.4 Create AltaPay Payment Processor

1. From the SFCC Business Manager.
2. Select your site from the list in the top navigation bar.
3. Navigate to: Merchant Tools -> Ordering -> Payment Processors.
4. Click the "New" button.
5. For the ID, enter ALTAPAY in all capital letters.
6. For the Description, enter "AltaPay Checkout".
7. Click the "Apply" button.

[Merchant Tools](#) > [Ordering](#) > [Payment Processors](#) > Create Payment Processor

General Settings

New Payment Processor

Click [Back to List](#) to display the list again.

ID:*	<input type="text" value="ALTAPAY"/>
Description:	<input type="text" value="AltaPay Checkout"/>

3.1.5 Payment Methods Import

1. From the SFCC Business Manager.
2. Select your site from the list in the top navigation bar.
3. Navigate to: Merchant Tools -> Ordering -> Import & Export
4. In the "Import & Export Files" section, click the "Upload" link or button.
5. Upload the file "altapay_sfra_paymentmethods.xml" from the LINK_AltaPay repository.
6. In the "Payment Methods" section, click the "Import" button.

Import & Export

Orders (XML)
 Manage order [imports](#) & [exports](#). Export orders into XML files. Import Export

Tax Table
 Manage tax table [imports](#) & [exports](#). Import or export tax classes, tax jurisdictions, and tax rates from or into XML files. Import Export

Shipping Methods
 Manage shipping method [imports](#) & [exports](#). Import or export shipping methods from or into XML files. Import Export

Payment Methods
 Manage payment method [imports](#) & [exports](#). Import or export payment methods from or into XML files. Import Export

Import & Export Files
[Upload](#) import files and [download](#) export files. Upload Download

1. Select "altapay_sfra_paymentmethods.xml" and click the "Next" button.
2. Wait for validation and click the "Next" button.
3. Leave "Merge" selected as the import mode and click the "Import" button.
4. Navigate to: Merchant Tools -> Ordering -> Payment Methods.
5. In the "Payment Methods" section, notice that all AltaPay payment methods is disabled so they not appear as an option in the normal Storefront check-out flow.

Enable the desired payment methods but be aware that some of them is limited to certain countries and currencies. Remember to disable all the normal storefront payment methods.

Payment Methods

Payment methods are managed here. To create a new payment method, click the **New** button. To remove a payment method click the remove icon in the payment method row. The default payment methods cannot be removed, and their IDs cannot be changed. When you select the CREDIT_CARD payment method, credit/debit cards can be reordered through drag and drop.

New Sort Order Credit/Debit Cards Import/Export Language: Default

ID	Name	Enabled	Sort Order	
ALTAPAY_CREDITCARDS	Credit Card	No	7	
ALTAPAY_KLARNA_ACCOUNT	Klarna Account	Yes	12	
ALTAPAY_KLARNA_INVOICE	Klarna Invoice	No	11	
ALTAPAY_MOBILEPAY	MobilePay	No	8	
ALTAPAY_PAYPAL	PayPal	No	10	
ALTAPAY_VIABILL	Viabill	No	9	
BANK_TRANSFER	Bank Transfer	No	3	
BML	Bill Me Later	No	4	
CREDIT_CARD	Credit Card	Yes	2	
DW_ANDROID_PAY	Android Pay	No	6	
DW_APPLE_PAY	Apple Pay	No	5	
GIFT_CERTIFICATE	Gift Certificate	No	1	

3.2 Configuration

3.2.1 Configure AltaPay Site Preferences

1. From the SFCC Business Manager.
2. Select your site from the list in the top navigation bar.
3. Navigate to: Merchant Tools -> Site Preferences -> Custom Preferences -> AltaPay:
4. This is where the merchant can access and configure the AltaPay integration.
5. Fill out the settings as desired. Descriptions of the site preferences are listed in the table below.

Preference	Description
AltaPay Cartridge Enabled	Preference that defines if the AltaPay integration is enabled.
AltaPay Test Mode	Preference that defines if the testing mode should be enabled.
AltaPay base Production URL	URL to the production gateway. E.g. https://yourname.altapaysecure.com/
AltaPay base Test URL	URL to the test gateway. E.g. https://testgateway.pensio.com/
AltaPay Production Username	Username for the production gateway and terminals.
AltaPay Production Password	Password for the production gateway and terminals.
AltaPay Test Username	Username for the test gateway and terminals.
AltaPay Test Password	Password for the test gateway and terminals.
AltaPay Timeout	Timeout (seconds) for communication with AltaPay backend. NB. Don't change!
AltaPay Terminals	Mapping of payment methods in Salesforce and terminals in the AltaPay payment gateway. A terminal can only contain one payment method and one currency, but it is possible to add all the relevant terminals. The setting must be structured as shown below.

	<pre> { "terminals": { "production": { "GBP": [{ "id": "ALTAPAY_CREDITCARDS_GBP", "name": "Shopname CC GBP", "allowedlocales": ["en_GB"] }, { "id": "ALTAPAY_PAYPAL_GBP", "name": "Shopname PayPal Wallet GBP", "allowedlocales": ["en", "en_GB"] }] }, "test": { "GBP": [{ "id": "ALTAPAY_CREDITCARDS_GBP", "name": "Shopname Test Terminal", "allowedlocales": ["en", "en_GB"] }, { "id": "ALTAPAY_PAYPAL_GBP", "name": "Shopname PayPal Wallet Test Terminal", "allowedlocales": ["en", "en_GB"] }] } } } </pre> <p>The attribute 'id' must correspond with the payment method added in: Merchant Tools -> Ordering -> Payment Methods plus the preferred currency. The attribute 'name' is the name and identifier of the AltaPay terminal. The attribute 'allowedlocales' defines which locales that can use the terminal.</p>
AltaPay Payment Page URL	<p>URL for controlling the payment form page which is shown to the customer.</p> <p>It is possible to customize the payment page by changing the callbackform.isml template.</p>
AltaPay Payment Success URL	<p>When a payment is accepted, this callback URL is called, and the data received from AltaPay is validated.</p>

AltaPay Payment Fail URL	In case a payment fails this callback is called. This can be due to incorrect card details, declined by the bank etc. The data received from AltaPay is stored on the order and the customer is redirected back to checkout page.
AltaPay Payment Open URL	To support an asynchronous payment (e.g. wallet payments) where the provider not always accept the payment upfront this callback is called. To indicate this event an open payment contains the confirmation status 'Not confirmed'.
AltaPay Payment Notification URL	In case a payment has not returned an answer (e.g. customer closes window prior to returning to the shop) or when an open payment is accepted/declined. When an answer arrives, this callback is called. This does not apply to card payments.
AltaPay Redirect Page URL	This URL is used when the customer is redirected to a third party (e.g. 3D Secure) to inform the customer about the redirection. A default non-branded page is shown if nothing else is stated.
AltaPay Whitelisted IP's	List of IP addresses that AltaPay is communicating from. Used to secure that only request from AltaPay is handled. You are advised to verify that the following IP addresses is added: 91.199.134.160 91.199.134.161 91.199.134.162 91.199.134.163 91.199.134.164 91.199.134.165 91.199.134.166 91.199.134.167 91.199.134.160/29
AltaPay Error Notifications Enabled	Preference that defines if notification emails should be send when error events occurs in the payment flow
AltaPay Notification Sender	Email used as sender on notification emails, default is 'noreply@dsalesforce.com'
AltaPay Notification Recipients	List of notification recipients

3.2.2 Select language

As part of the setup, the language selection for the check-out process is also on the check list. Navigate to: Merchant Tools -> Site Preferences -> Locales -> select the web shops local language.

AltaPay supports the following languages:

Code	Language
CS	Czech
DA	Danish
DE	German
EN	English
ES	Spanish
FI	Finnish
FR	French
JA	Japanese
LT	Lithuanian
NL	Dutch
NO	Norwegian
NB*	Norwegian (Bokmål) – converted to no
NN	Norwegian (Nynorsk) – converted to no
PL	Polish
SV	Swedish
TH	Thai

TR	Turkish
ZH	Chinese
EE*	Estonian – converted to ET
ET	Estonian
IT	Italian
PT	Portuguese
RU	Russian

If the merchant uses an unsupported language the payment page is shown in English as default.

3.3 Custom Code

We have made some adjustments to the standard Sitegenesis, which you will need to implement to use AltaPay as the Payment Service Provider (PSP) on your storefront.

Four files will have to be modified, and two files must be added as explained in the next sections. The locations of the files are:

File	Location
Checkout.js	app_storefront_base/cartridge/Controllers/Checkout.js
checkout.isml	app_storefront_base/cartridge/templates/default/checkout/checkout.isml
hooks.json	app_storefront_base/hooks.json
ALTAPAY.js	app_storefront_base/scripts/hooks/payment/processor/ALTAPAY.js
altapayCheckout.js	app_storefront_base/client/default/js/altapayCheckout.js
package.json	Package.json

3.3.1 AltaPay Cartridge Enabled

Add the code below to the “Checkout-Begin” controller to activate the AltaPay cartridge.

1. Navigate to: app_storefront_base -> cartridge -> Controllers -> Checkout.js
2. Insert ‘altapayEnabled’ property in the rendering object.
3. Look for an example in the folder ‘Custom Code’

“altapayEnabled: dw.system.Site.current.getCustomPreferenceValue(‘altapayCartridgeEnabled’)”

```
res.render('checkout/checkout', {
  order: orderModel,
  customer: accountModel,
  forms: {
    shippingForm: shippingForm,
    billingForm: billingForm
  },
  expirationYears: creditCardExpirationYears,
  currentStage: currentStage,
  reportingURLs: reportingURLs,
  altapayEnabled:
dw.system.Site.current.getCustomPreferenceValue('altapayCartridgeEnabled')
})
```

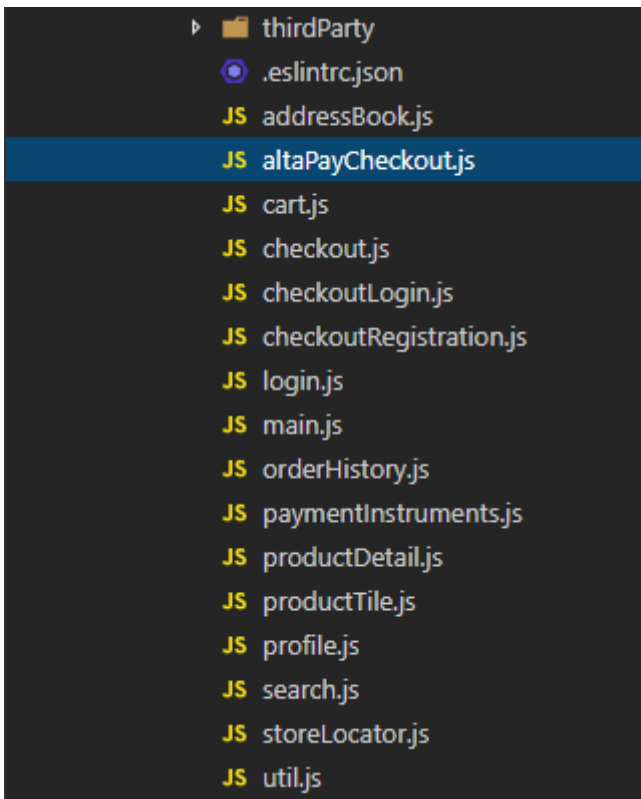
3.3.2 AltaPay Checkout JS

Add the code below to use the AltaPay specific client JS when the AltaPay cartridge is enabled.

1. Navigate to: app_storefront_base -> cartridge -> templates -> default -> checkout -> checkout.isml
2. Insert the AltaPay cartridge enabled check in the top <isscript> tag.
3. Look for an example in the folder 'Custom Code'

```
<!------->
<!-- Load Static Assets -->
<!------->
<isscript>
  var assets = require('*/cartridge/scripts/assets.js');
  if(!pdict.altapayEnabled){
    assets.addJs('/js/checkout.js');
  } else {
    assets.addJs('/js/altaPayCheckout.js');
  }
  assets.addCss('/css/checkout/checkout.css');
</isscript>
```

1. Insert altapayCheckout.js file provided in the "Custom Code" folder to app_storefront_base -> cartridge -> client -> default -> js



- o Navigate to package.json file in the root folder of your project
- o Insert the "path" configs to package.json

```
],  
"packageName": "app_storefront_base",  
"paths": {  
  "base":  
  "../storefront-reference-architecture/cartridges/app_storefront_base/",  
  "altapay":  
  "../storefront-reference-architecture/cartridges/int_altapay_sfra/"  
}
```

2. Remember to compile the client JS with this command:
"npm run compile:js"

3.3.3 Add AltaPay Processor Hook

Add the code below to add AltaPay as processor.

1. Navigate to: app_storefront_base/hooks.json
2. Insert AltaPay below the basic_credit processor

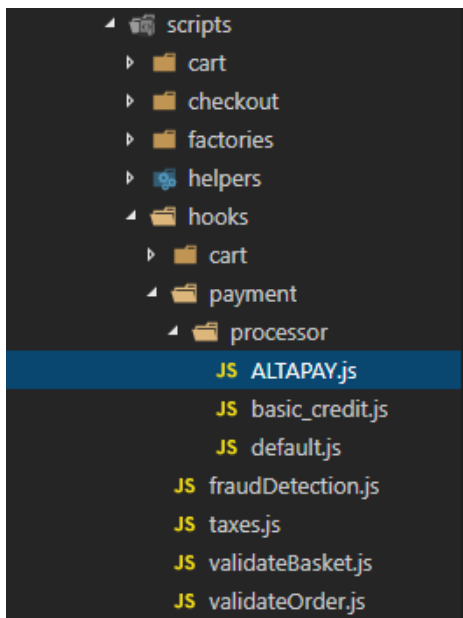

```

},
{
  "name": "app.payment.processor.basic_credit",
  "script": "./cartridge/scripts/hooks/payment/processor/basic_credit"
},
{
  "name": "app.payment.processor.altapay",
  "script": "./cartridge/scripts/hooks/payment/processor/ALTAPAY"
},
}

```

3.3.4 Insert AltaPay processor

Insert the ALTAPAY processor script provided in the 'Custom Code' folder to app_storefront_base -> cartridge -> scripts -> hooks -> processor



3.4 External Interfaces

The AltaPay cartridge communicates with AltaPay's backend where customer data etc. is sent, to verify a transaction. Banks and acquirers make the verification. AltaPay relays the response to the cartridge.

3.5 Credit card tokenization

It's possible to save the customer credit card after a successful transaction. The credit card number is saved securely inside AltaPay payments gateway. To enable this functionality contact AltaPay to enable the credit card token in your terminal.

The credit card terminal must be configured to support credit card tokens. Also, the credit card form template must be set to form_dynamic_div_with_save_cc. This setup is done inside AltaPay payments gateway. Please contact AltaPay to setup your terminal.

If the terminal is configured correctly the customer will have the option to save the credit card information during checkout. Also, a previous saved credit card will appear with a mask in the checkout page.

3.6 Reconciliation setup

Follow the steps below to setup the reconciliation identifier.

1. Navigate to: `int_altapay_sfra/cartridge/scripts/createRequestParameters.js`.
2. Find the Reconciliation Identifier section and remove the comments.

```
/*  
//Reconciliation Identifier  
parameterArr.push(['sale_reconciliation_identifrier', 'Insert reconciliation  
identifier here'].join('='));  
*/
```

3. Replace "Insert the reconciliation identifier here" with the reconciliation identifier that is needed by the ERP system.

3.7 Testing

In general, the merchant can use any card number when testing against the test gateway and they will be accepted. Designated card numbers to trigger different scenarios (3D Secure, failures etc.) can be found [here](#). A Test bank is also available if the merchant needs to test PayPal, iDEAL, or other alternative payment methods.

Preconditions for the following test scenarios:

1. Imported 'altapay_sfra_metadata.xml'
 - a. Updated the 'Custom Site Preferences' with AltaPay user with API access rights
 - b. Added terminals for credit card and/or alternative payment
2. Items available in the storefront
3. That the payment method for cards connects with a terminal that is configured to receive cards. An alternative payment method must be connected to a terminal that accepts that payment method.

3.7.1 Successful card payment

1. Add an item to the cart.
2. Click "View cart".
3. When shopping cart is shown, click on "Checkout" button.
4. Select either Guest checkout or login.
 - a. If guest checkout.
 - i. Fill in the information.
5. Select shipping method.
6. Click on "Next: Payment" button.
7. Fill in remaining information
8. Select "Credit Card" as payment method
9. Click on "Next: Place Order" button.
10. The payment page appears. Ensure it is a payment page for card payments.
11. Enter card details (use random numbers) and click on "Submit" button.
12. Verify that the confirmation page is shown with correct information and without any error message. Take a note of the order number.
13. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly - "Confirmation Status" = confirmed.

14. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is "preauth".

3.7.2 Failed card payment

1. Repeat step 1-10 in Successful card payment.
2. Enter card details – use the following payment information.
 - a. Card number: 4180000000000566
 - b. Expire month: 05
 - c. Expire year: 2019
 - d. CVC: 444
3. Click on "Submit" button.
4. Ensure that the user is redirected back to the Checkout flow. Error messages is returned from the AltaPay controller but as described in the limitations section, the Merchant will have to implement custom functionality to show these messages.
5. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Order Status" = failed. Take a note of the order number.
6. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Ensure that the status of the order is "preauth_failed".

3.7.3 3D Secure card payment

1. Repeat step 1-10 in Successful card payment.
2. Enter card details – use the following payment information.
 - a. Card number: 4170000000000568
 - b. Expire month: 05
 - c. Expire year: 2019
 - d. CVC: 444
3. Click on "Submit" button.
4. The user is redirected to the issuing bank 3D Secure confirmation page. Enter the correct validation information. If you are testing against the test gateway, a mock-up 3D Secure page is shown. Click "Redirect" button.
5. Verify that the confirmation page is shown with correct information and without any error message. Take a note of the order number.
6. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Confirmation Status" = confirmed.
7. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is "preauth" and "3D Secure result" is successful.

3.7.4 3D Secure failed card payment

1. Repeat step 1-10 in Successful card payment.
2. Enter card details – use the following payment information.
 - a. Card number: 4170000000000568
 - b. Expire month: 05
 - c. Expire year: 2019
 - d. CVC: 444

3. The user is redirected to the issuing bank 3D Secure confirmation page. Enter the correct validation information. If you are testing against the test gateway, a mock-up 3D Secure page is shown. Click "Redirect" button.
4. Ensure that the user is redirected back to the Checkout flow. Error messages is returned from the AltaPay controller but as described in the limitations section, the Merchant will have to implement custom functionality to show these messages.
5. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Order Status" = failed. Take a note of the order number.
6. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Ensure that the status of the order is "preauth_failed".

3.7.5 Successful alternative payment

1. Add an item to the cart.
2. Click "View cart".
3. When shopping cart is shown, click on "Checkout" button.
4. Select either Guest checkout or login.
 - a. If guest checkout.
 - i. Fill in the information.
5. Select shipping method.
6. Click on "Next: Payment" button.
7. Fill in remaining information
8. Select preferred "alternative" payment option as payment method
9. Click on "Next: Place Order" button.
10. Verify that the customer is redirected to the alternative payment provider webpage. Verify the pending payment.
11. If you are testing against the test gateway a mock-up for bank and alternative payment solutions will be shown. If that is the case, click "Sign in" (No credentials needed) and "Accept".
12. Verify that the confirmation page is shown with correct information and without any error message. Take a note of the order number.
13. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Confirmation Status" = confirmed.
14. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is "preauth" or "bank_payment_finalized", depending on the acquirer.

3.7.6 Failed alternative payment (Customer cancel)

1. Repeat step 1-10 in Successful alternative payment
2. If you are testing against the test gateway a mock-up for bank and alternative payment solutions will be shown. If that is the case, click "Developer options" and "Cancel".
3. Ensure that the user is redirected back to the Checkout flow. Error messages is returned from the AltaPay controller but as described in the limitations section, the Merchant will have to implement custom functionality to show these messages.
4. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Order Status" = failed.
5. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is "epayment_cancelled" or "preauth_failed", depending on the acquirer.

3.7.7 Failed alternative payment

1. Repeat step 1-10 in Successful alternative payment

2. If you are testing against the test gateway a mock-up for bank and alternative payment solutions will be shown. If that is the case, click "Developer options" and "Declined".
3. Ensure that the user is redirected back to the Checkout flow. Error messages is returned from the AltaPay controller but as described in the limitations section, the Merchant will have to implement custom functionality to show these messages.
4. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Order Status" = failed.
5. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is "epayment_declined" or "preauth_failed", depending on the acquirer.

3.7.8 Successful alternative payment notification

1. Repeat step 1-10 in Successful alternative payment
2. Verify that the customer is redirected to the alternative payment provider webpage. Verify the pending payment.
3. If you are testing against the test gateway a mock-up for bank and alternative payment solutions will be shown. If that is the case, click "Developer options" and "Open" (Opens in a new window). Do not close the test bank page.
4. Verify that the confirmation page is shown with correct information and without any error message. Take a note of the order number.
5. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Confirmation Status" = Not confirmed.
6. Go back to the test bank page and click 'Call success notification now'. It can take a couple of minutes before the actual notification is triggered via the API.
7. Repeat step 5 and verify that the status has changed from "Not confirmed" to "Confirmed".
8. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is "preauth" or "bank_payment_finalized", depending on the acquirer.

3.7.9 Failed alternative payment notification

1. Repeat step 1-10 in Successful alternative payment
2. Verify that the customer is redirected to the alternative payment provider webpage. Verify the pending payment.
3. If you are testing against the test gateway a mock-up for bank and alternative payment solutions will be shown. If that is the case, click "Developer options" and "Open" (Opens in a new window). Do not close the test bank page.
4. Verify that the confirmation page is shown with correct information and without any error message. Take a note of the order number.
5. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – "Confirmation Status" = Not confirmed.
6. Go back to the test bank page and click 'Call declined notification now'. It can take a couple of minutes before the actual notification is triggered via the API.
7. Repeat step 5 and verify that the status has changed from "Not confirmed" to "Cancelled"

3.7.10 Fraud checking (accepted credit card)

1. Repeat step 1-10 in Successful card payment.
2. Use a credit card number enabled for fraud checking and that returns the "Accept" status.
 - a. For example: 4170000000000006
3. Verify that the confirmation page is shown with correct information and without any error message. Take a note of the order number.

4. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – “Confirmation Status” = Confirmed.
5. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is “preauth”.
6. Repeat with a credit card that returns the ‘Challenge’ status.
 - a. For example: 5250000000000121
7. Repeat with a credit card that returns the ‘Unknown’ status
 - a. For example: 5110000000000113

3.7.10 Fraud checking (denied credit card)

1. Repeat step 1-10 in Successful card payment.
2. Use a credit card number enabled for fraud checking and that returns the ‘Deny’ status.
 - a. For example: 4170000000000105
3. Ensure that the user is redirected back to the Checkout flow. Error messages is returned from the AltaPay controller but as described in the limitations section, the Merchant will have to implement custom functionality to show these messages.
4. From the SFCC Business Manager navigate to: Merchant tools -> Ordering -> Orders. Locate and select the order and verify that the order has been handled correctly – “Order Status” = Failed.
5. Login in to <https://testgateway.altapaysecure.com> and locate the order by the order number via the search box in the top right corner. Check that the amount corresponds with the information in Business manager and ensure that the status of the payment is “preauth”.

4. Operations, Maintenance

4.1 Data Storage

Information about payment method, debit/credit card data, items, shipping/billing addresses and amount is sent to AltaPay and stored on orders in SFCC.

4.2 Availability

If you experience any problems with the gateway or payments, please contact AltaPay support. Please supply as much information as possible such as order/Payment ID, payment method, terminal name etc.

4.3 Support

If there is any problem with the payments or integration, please contact AltaPay support on support@altapay.com or +45 70 20 00 56 - option 1 (support).

5. User Guide

5.1 Roles, Responsibilities

The merchant must have access to AltaPay's test terminals before the integration can be completed. The merchant must set up terminals and user credentials correctly on 'Custom preference site' and perform tests on the test environment before going live for each shop.

AltaPay also need to verify that the test setup works prior to go-live. Please send customer access details to AltaPay support when the shop is ready for review.

5.2 Business Manager

No new features.

5.3 Storefront Functionality

Accept payments with different payment methods. For more information about what AltaPay can offer, please contact AltaPay at info@altapay.com.