Sparxpres WebSale

Installation guide for Magento 2

Rev 1.6
05-09-2023

1   TABLE OF CONTENTS

2   INTRODUCTION

This document contains a brief overview of the installation and setup of Sparxpres WebSale module for Magento 2. It is assumed that you have knowledge of how modules are installed in Magento 2.

Sparxpres WebSale, adds a loan calculator on the product pages and on the cart page. In the configuration, you can decide whether the loan calculator should be displayed with a slider or as a dropdown for the period selection. You can also decide if the loan calculator should be embedded directly on the page, or instead insert a button "Show financing", that shows the loan calculator in a modal window.

In addition, the module installs a payment method that appears on the checkout page and enables your customers to choose Sparxpres part payment and XpresPay as payment methods.

3 INSTALLATION

As with all Magento 2 modules, it is recommended to backup your website, before installation. It is also recommended to install the module on a test- or staging environment before installing it in production. Sparxpres disclaims any responsibility for installation and use of the module.

The module is installed with composer, as follows:

1. Login into your server with SSH, and change directory to Magento 2 root folder
2. Run the following commands:
   ```
   composer require sparxpres/module-websale
   php bin/magento module:enable Sparxpres_Websale
   php bin/magento setup:upgrade
   php bin/magento setup:di:compile
   ```

4 UPDATING

As with all Magento 2 modules, it is recommended to backup your website, before updating. It is also recommended to update the module on a test- or stagging environment before updating it in production. Sparxpres disclaims any responsibility for updating and using the module.
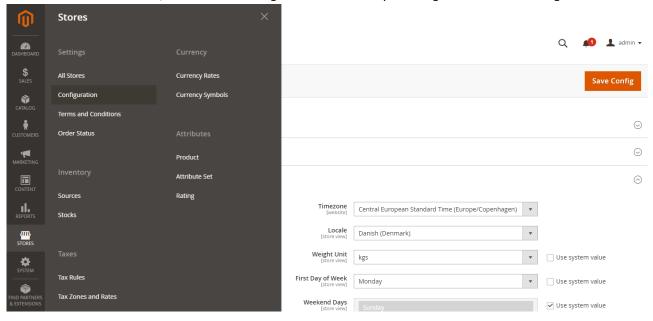
If the module is installed on your website, you can update it to the newest version, this is done with composer, as follows:

1. Login into your server with SSH, and change directory to Magento 2 root folder
2. Run the following commands:
   ```
   composer update sparxpres/module-websale
   php bin/magento setup:upgrade
   php bin/magento setup:di:compile
   ```
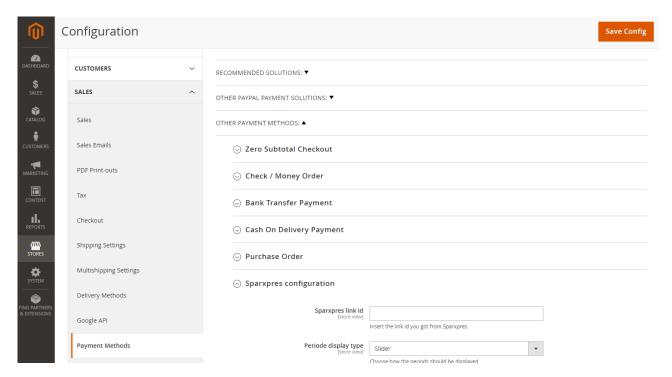
5 CONFIGURATION

Once the module is installed, it needs to be configured. This is done by selecting "STORES" and "Configuration".
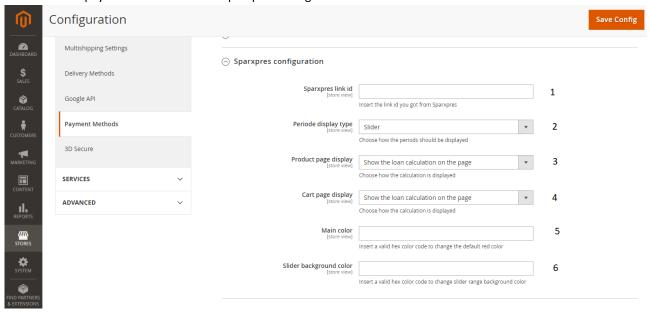

In the configuration menu you must select "SALES" and "Payment Methods".
On the payment methods page, expand the section "OTHER PAYMENT METHODS":

Find the other payment method called "Sparxpres configuration".

The configuration items is:

1. Sparxpres link id: insert the linkId you get from Sparxpres (36-character long UUID) *Mandatory field.
2. Period display type: choose how the periods is displayed (slider is default).
3. Product page display: choose how the loan calculation is displayed on the product page.
4. Cart page display: choose how the loan calculation is displayed on the cart page.
5. Main colour: you can override Sparxpres' default red colour here.
6. Slider background colour: if you use the slider, and the sliders grey background is too close to your sites background colour, you can change the sliders grey background colour here.
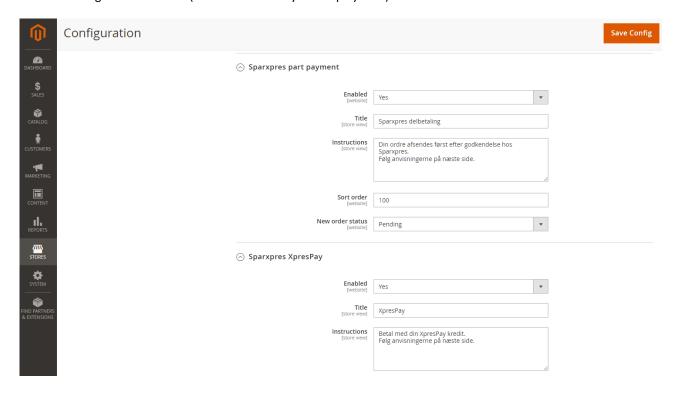
When you have inserted a linkId in field number 1, and saved the configuration, the loan calculator is displayed on the product pages and in the shopping cart (if the amount is in a legal price range and is in Danish kroner (DKK)).

If you scroll further down on the payment methods page, you see two addition sections:

1. Sparxpres part payment
2. Sparxpres XpresPay

For each of these two payment methods, you can:

1. Enable/disable them.
2. Change the payment title that is display to the user.
3. Change the payment instructions displayed to the user.
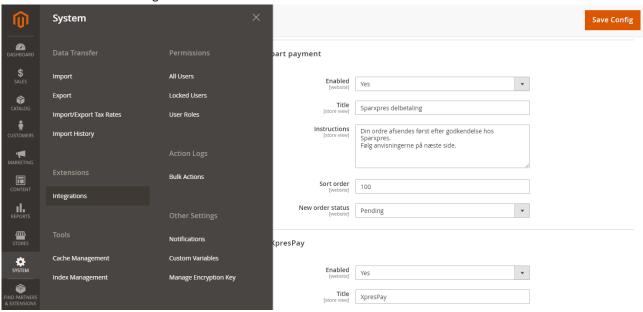4. Change the sort order (the order that they are displayed in)

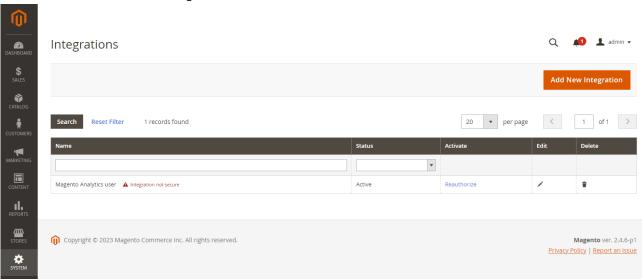The module offers a standard way to handle callbacks. If you would like to make a custom callback handler, you must create an API that can handle the callbacks sent by Sparxpres.

If you want to use the standard callback handler, you must provide Sparxpres the needed access to your system.
This is done in the following way:

1. Choose "SYSTEM" and "Integrations"

2. Click on the button "Add New Integration"

3. In the name field, type "Sparxpres callback" or similar.

4. In the current user identity verification password field, you must type the password for the admin user that you are logged in with.

5. The click on API in the left menu.
6. Select "Comment" as show below (under Sales/Operations/Orders/Actions)

7. Save the integration.
8. Click Active on Sparxpres callback integration, allow the changes, copy the Access Token key, and sent it to: support@sparxpres.dk

7 ORDER PROCESS WITH AND WITHOUT CALLBACK

The following happens when Sparxpres part payment is selected:

1. The order is created in your system, with the order status set to "Processing", and the user is automatically forwarded to Sparxpres' loan application form.
2. When the customer has completed and sent the loan application form to Sparxpres, the customer is automatically sent to your confirmation page (checkout/onepage/success/), Magento does not change the order status at this point, and the order status is therefore still "Processing".

If you choose to use "standard callback", an order gets the following callbacks:

1. NEW: The application is received in Sparxpres backend.
   a. The order status is set to Pending Payment.
   b. The text "Sparxpres har modtaget låneansøgningen" is added to the orders log
2. DECLINE: The application is declined by Sparxpres.
   a. The order status is set to Cancelled.
   b. The text "Sparxpres har givet afslag på låneansøgningen" is added to the orders log
3. REGRETTED: The application is regretted by the applicant.
   a. The order status is set to Cancelled.
   b. The text "Sparxpres har annulleret lånet" is added to the orders log
4. CANCELED: The application is canceled.
   a. The order status is set to Cancelled.
   b. The text "Sparxpres har annulleret lånet" is added to the orders log
5. WAITING_FOR_SIGNATURE: The application is approved and is awaiting the applicant's signature.
   a. The order status is set to Pending Payment.
   b. The text "Sparxpres afventer kundens underskrift" is added to the orders log
6. RESERVED: The loan is reserved and is waiting for you to release the money.
   a. The order status is set to Processing.
   b. The text "Lånet er klar til frigivelse hos Sparxpres" is added to the orders log
7. CAPTURED: The loan is set for payment.
   a. The order status is set to Processing.
   b. The text "Lånet er sat til udbetaling hos Sparxpres" is added to the orders log

8 TEST CALLBACK

When Sparxpres has set up your callback, it is possible to test the flow, including callback. This is done by placing an order on your website where Sparxpres part payment is selected at checkout. The order is then created in your system and you (the user) are forwarded to Sparxpres' loan application form.

It is possible to test two outcomes of the application flow:

1. Decline: sends the following callbacks.
   a. NEW
   b. DECLINE
2. Approved: sends the following callbacks.
   a. NEW
   b. WAITING_FOR_SIGNATURE
   c. RESERVED
   d. CAPTURED

To test the two outcomes, you must use the CPR numbers below in the loan application form (test mode must be activated in Sparxpres' system, otherwise you get the message that the CPR number is invalid):

1. Decline = 1401251163
2. Approved = 3010820537

9 TEST CALLBACK (ADVANCED)

You can test the callback functionality with a third-party application, for example cURL og Postman.

The callback service can send 7 different callbacks statuses per order:
NEW, DECLINE, REGRETTED, CANCELED, WAITING_FOR_SIGNATURE, RESERVED og CAPTURED.

The following is a json example, that show how you build the callback request:
```
{
"status": "NEW",
"amount": 4999,
"transactionId": "100001"
}
```

- Status: one of the 7 callback statues.
- Amount: the orders amount (this is rounded up to the nearest int in Danish kroner).
- TransactionId: the order number.

Example with cURL:
curl -X PUT -H "Authorization: Bearer ACCESS_TOKEN" -H "Content-Type: application/json" -d '{"status": "NEW","amount": 4999,"transactionId": "100001"}' https://YOUR_DOMAIN.dk/rest/V1/sparxpres/callback

Where the parameters are:
ACCESS_TOKEN = The Access Token you got from section 5, pkt. 9
YOUR_DOMAIN = The domain name

10 PRODUCT/CART PAGE LAYOUT (ADVANCED)

The Sparxpres WebSale module inserts the loan calculator in the following locations:

1. Product page:
   a. For simple product the default location is: product.info.addtocart
      The name is: sparxpres-addtocart

   b. For configurable products the default location is: product.info.addtocart.additional
      The name is: sparxpres-addtocart-additional

2. Cart page:
   a. On the cart page the default location is: checkout.cart.methods
      The name is: sparxpres-cart

If then default locations is not working for your layout, you can move its location.
Here is an example of how you can move simple products on the product page:

1) Login into your server with SSH

2) Create the following directory structure:
   <magento_root>/vendor/<vendor>/<theme>/Sparxpres_Websale/layout/
   Fx: <magento_root>/vendor/magento/theme-frontend-luma/Sparxpres_Websale/layout/

3) In the Sparxpres_Websale/layout/ folder, add the file:
   catalog_product_view.xml

4) Edit the catalog_product_view.xml file, and add the content:
   ```xml
   <?xml version="1.0"?>
   <page xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="urn:magento:framework:View/Layout/etc/page_configuration.xsd">
     <body>
       <move element="sparxpres-addtocart" destination="product.info.main" before="product.info.price"/>
     </body>
   </page>
   ```
   Where "destination" and "before" is the location you want it moved to.

5) Clear the server's cache.