

Installation Guide for Novalnet Gambio Module

Version	Date	Remarks
12.1.0	08.01.2024	[New] Implemented Direct Debit ACH, Blik, Payconiq and MB Way payments [Fix] Callback has been optimized as per the new testcase

🔗 For previous version changelogs, go to <https://github.com/Novalnet-AG/gambio-payment-integration-novalnet/blob/master/changelog.txt>

TABLE OF CONTENTS

1	QUICK SETUP	3
1.1	Module Installation	3
1.2	Upgrading the Novalnet Payment Module	5
1.3	Global Configuration in the Gambio shop system.....	5
1.3.1	Notification/Webhook configuration in the Novalnet Admin Portal	8
1.3.2	Client key configuration in the Novalnet Admin Portal	9
1.4	Payment Activation in the Novalnet Admin Portal	11
1.5	Payment Activation in the Gambio shop system	13
1.6	Readme procedures	14
1.6.1	To display transaction details with proper alignment in order confirmation E-mail	14
1.6.2	To display payment reference in order confirmation E-mail	15
1.6.3	To display follow-up process in shop admin orders.....	16
1.6.4	To display and process Apple Pay and Google Pay button in shopping cart page	17
1.6.5	To display and process the Apple pay and Google pay button in product page.....	19
1.6.6	To display the Barzahlen/viacash payment slip on the shop success page	21
2	TESTING AND GOING LIVE	22
3	ADDITIONAL CONFIGURATION	22
3.1	Additional configuration for all the payment methods.....	22
3.2	Additional configuration for Credit/Debit Cards	28
3.3	Additional configuration for Apple Pay payment.....	28
3.4	Additional configuration for Google Pay payment.....	30
4	MANAGING GAMBIO E-COMMERCE ADMIN PANEL	31
4.1	Order Management.....	31
4.2	Transaction Overview.....	31
4.3	Order details for Instalment payments	32
4.4	Confirming / Canceling a transaction	32
4.5	Refunding an order.....	33
4.5.1	Refunding Instalment orders	34
4.6	Cancelling Instalment orders.....	35
4.6.1	Cancelling Instalment options.....	35
5	UNINSTALLATION	36
6	TECHNICAL SUPPORT THROUGH NOVALNET	37

1 QUICK SETUP

This guide describes the quick installation procedure of Novalnet payment module (this guide comes along with the module), to integrate it with your shop system and to start accepting payments worldwide. For this integration, a Novalnet merchant account is required, so please make sure that you have received your merchant account details from our sales team. If not, drop a mail to sales@novalnet.de

This Novalnet payment module package version (12.1.0) supports the following versions of Gambio:



To get started:

1. Log in to the [Novalnet Admin Portal](#) with your merchant account details (user credentials).
2. Log in to your Gambio shop system.
3. Make sure that you have extracted the payment module package from the zip file you have received. If you have received only the installation guide without the payment module package (zip file), please send us a mail to technic@novalnet.de with your merchant ID.

1.1 Module Installation

To install the Novalnet Payment Module, please go to your Gambio shop system and follow the steps below.

Step 1: Upload the contents of the **NovalnetGateway** folder (which you have extracted from the payment module package/zip file) into the file path: **<Root_Directory>** as shown below,

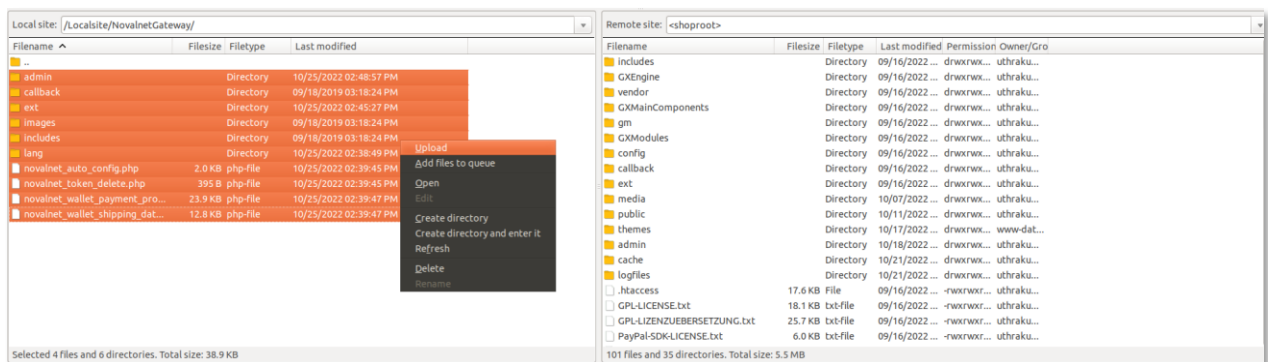


Figure 1

Step 2: Login to shop backend, go to **Modules** → **Payment Systems** as shown below,

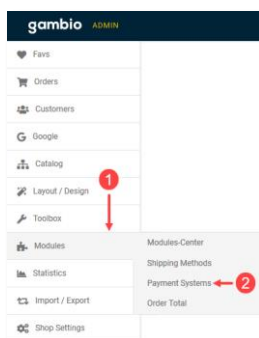


Figure 2

Step 3: Go to **Miscellaneous** → **added modules**, now Novalnet payment methods will be displayed as shown below

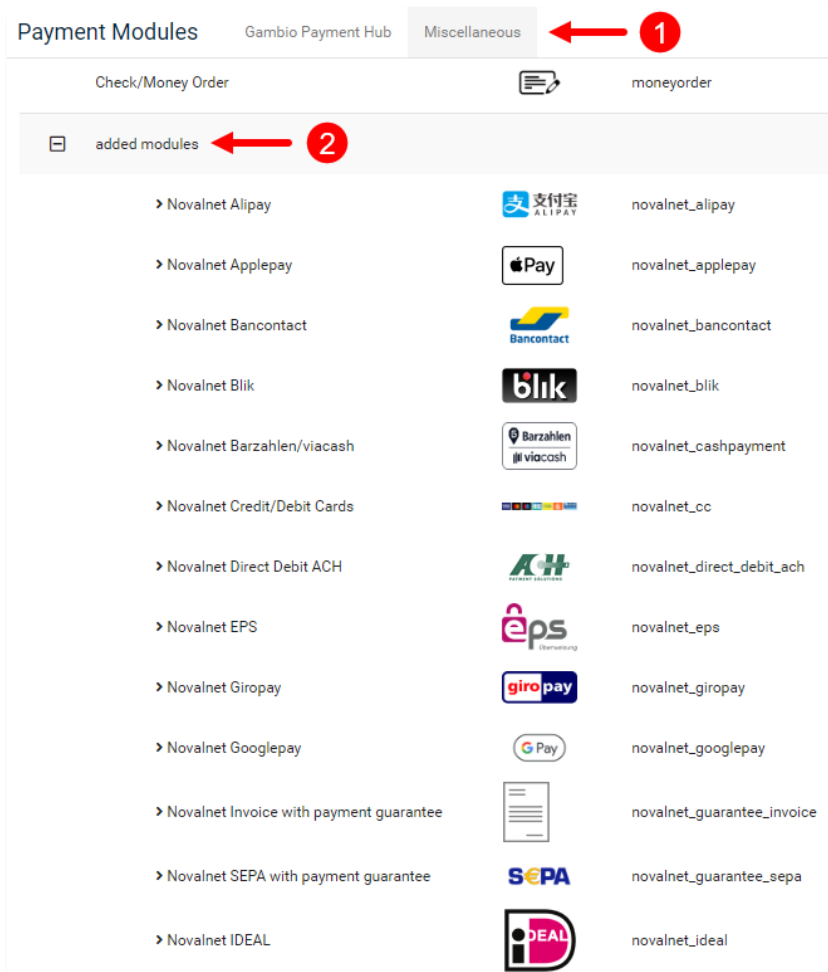


Figure 3

1.2 Upgrading the Novalnet Payment Module

To upgrade the latest Novalnet payment module, please follow the below steps. If you have installed the module via package, upgrade the module only through file server.

Step 1: It is recommended to take a proper backup before deleting the existing Novalnet payment module for disaster recovery.

Step 2: To remove the Novalnet payment module refer to the topic [5 UNINSTALLATION](#) and follow the same procedure to uninstall the Novalnet payment methods.

Step 3: Upload the contents of the **NovalnetGateway** folder into the file path: **<Root_Directory>** as shown below

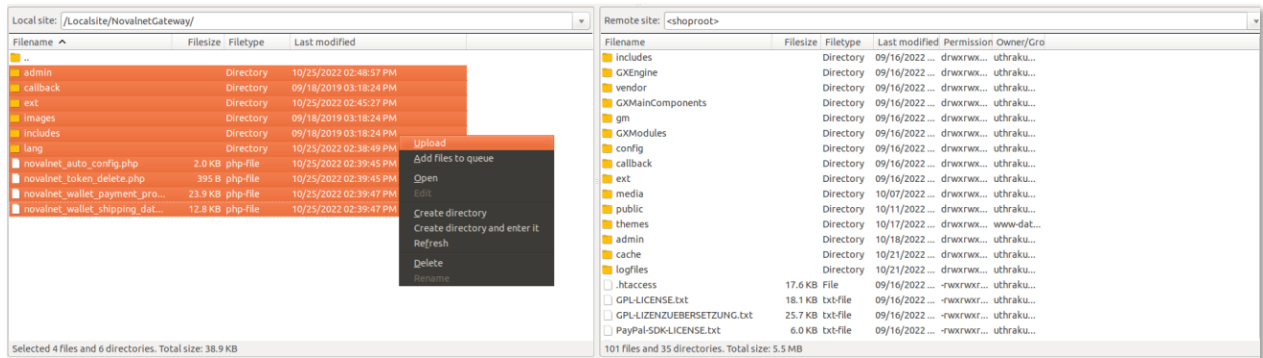


Figure 4

1.3 Global Configuration in the Gambio shop system

After successful installation of the Novalnet module, go to **Modules → Payment Systems → Miscellaneous → added modules → Novalnet API Configuration (V_12.1.0)**. Click **Install** as shown below

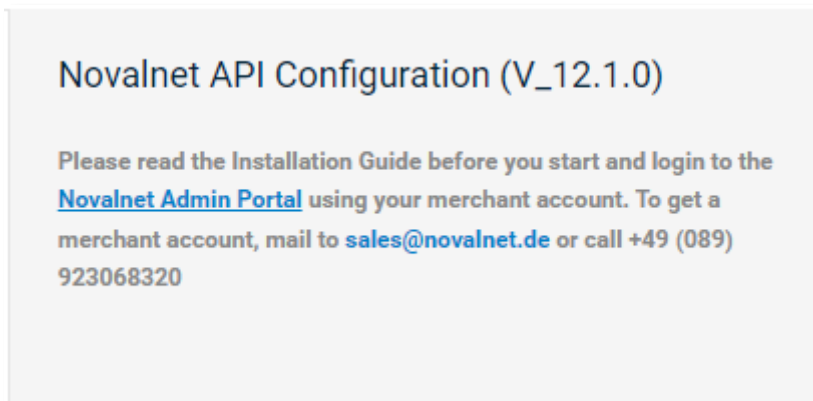


Figure 5

Here you can view the Novalnet API Configuration settings. Enter the following keys as shown below:

- ❗ **Product activation key** - a unique token for merchant authentication and payment processing.
- ❗ **Tariff ID** - a unique identifier which is dynamically created based on the tariff type at Novalnet.
- ❗ **Client key** - a unique key which is linked with your Novalnet merchant account to authenticate your client based requests. You need the client key to render the secure payment form and tokenize (temporarily) the sensitive information.

Novalnet API Configuration (V_12.1.0)

Alias for orders overview

Product activation key

Get your Product activation key from the [Novalnet Admin Portal](#)
Project > Choose your project > API credentials > API Signature
(Product activation key)

Payment access key

Get your Payment access key from the [Novalnet Admin Portal](#)
Project > Choose your project > API credentials > Payment access
key

Select Tariff ID

Select a Tariff ID to match the preferred tariff plan you created at the
Novalnet Admin Portal for this project

Client key

Figure 6

To get your **Product activation key** and **Payment access key**, go to the [Novalnet Admin Portal](#), navigate to **Projects** menu and choose your project as shown below,

The screenshot shows the Novalnet Admin Portal interface. On the left is a dark sidebar with a menu. The 'Projects' menu item is highlighted with a red circle and the number '1'. The main content area shows a table of projects under the 'Own Projects' tab. A 'Create new project' button is located at the top right of the table area, highlighted with a red circle and the number '2'.

S.No	Project ID	Project	URL	On-Hold	Activated	
1	14	GAMBIO DEMO SHOP	https://gambio.novalnet.de	✗	✓	🔗

Figure 7

Click **API credentials**, and copy the **API Signature (Product activation key)** and **Payment access key** as shown below,

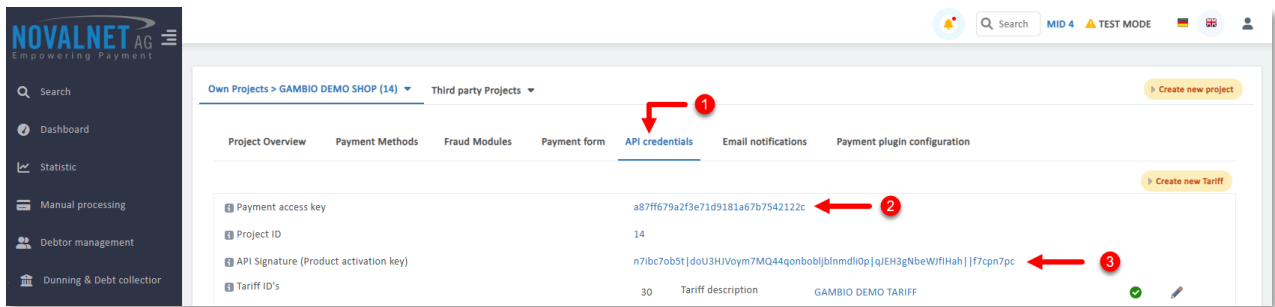


Figure 8

Paste the copied **Product activation key** and **Payment access key** in the respective fields in your shop admin panel. Next, choose the Tariff ID that you have created in the [Novalnet Admin Portal](#) for this Gambio shop system. Select your preferred **Tariff ID** from the drop-down menu and then click **Update** to save the changes as shown below,

Novalnet API Configuration (V_12.1.0)

Alias for orders overview

Product activation key

Get your Product activation key from the [Novalnet Admin Portal](#)
Project > Choose your project > API credentials > API Signature (Product activation key)

Payment access key

Get your Payment access key from the [Novalnet Admin Portal](#)
Project > Choose your project > API credentials > Payment access key

Select Tariff ID

Select a Tariff ID to match the preferred tariff plan you created at the Novalnet Admin Portal for this project

Client key

Figure 9

1.3.1 Notification/Webhook configuration in the Novalnet Admin Portal

In the **Novalnet API Configuration** page on your shop admin panel, you will find your **Notification / Webhook URL** under **Notification / Webhook URL Setup** as shown below,

Figure 10

Click **Configure** to link your **Notification / Webhook URL** in to the **Novalnet Admin Portal** as shown below,

Figure 11

After successful configuration, your shop **Notification / Webhook URL** will be linked with your Novalnet Merchant account which can be seen under **Vendor script URL/ Notification & Webhook URL** field of your project as shown below,

Figure 12

In your shop system, you can additionally test the Webhook URL manually and address notification emails to specific email addresses as shown below,

Notification / Webhook URL Setup

Allow manual testing of the Notification / Webhook URL

Enable this to test the Novalnet Notification / Webhook URL manually. Disable this before setting your shop live to block unauthorized calls from external parties

x

Notification / Webhook URL

You must configure the webhook endpoint in your [Novalnet Admin Portal](#). This will allow you to receive notifications about the transaction

Configure

Send e-mail to

Notification / Webhook URL execution messages will be sent to this e-mail

Figure 13


- i** **Notification / Webhook URL** - Required to keep the merchant’s database/system up to date and always synchronized with Novalnet (for example: up-to-date transaction status delivery).
- i** **Allow manual testing of the Notification / Webhook URL** - Click the toggle to enable and to test the Novalnet Notification / Webhook URL manually. Disable this before setting your shop live to block unauthorized calls from external parties.
- i** **Send e-mail to** - Every execution will be sent as a message to the e-mail address defined in this field.

1.3.2 Client key configuration in the Novalnet Admin Portal

The client key is a unique key which is linked with your Novalnet merchant account to authenticate your client based requests. You need the client key to render the secure payment form and tokenize (temporarily) the sensitive information.

🔑 Novalnet allows client-side request(s) only from the **whitelisted domains under a particular project** in the [Novalnet Admin Portal](#). The domain must be a valid (secured https://) website. Please follow the below steps to whitelist multiple domains under a single client key in the [Novalnet Admin Portal](#).

Step 1: Go to the [Novalnet Admin Portal](#).

Step 2: Navigate to the **Projects** menu and click view icon  on the right to view your project details as shown below

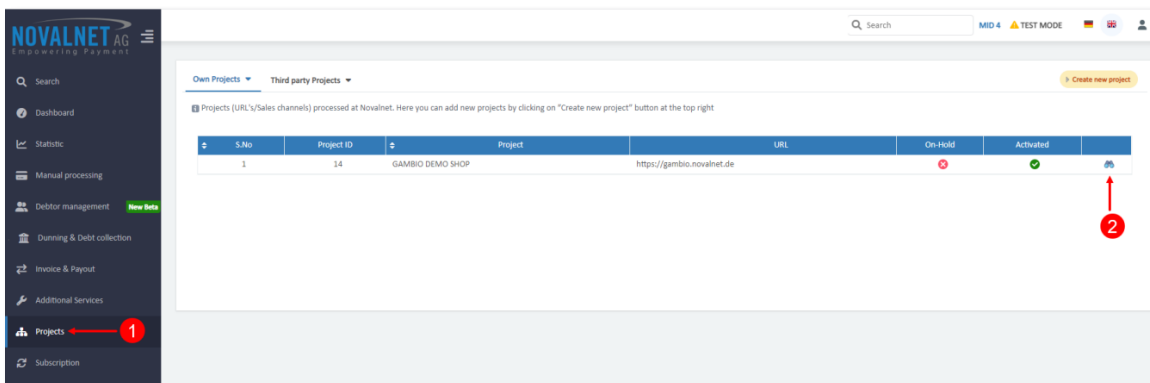


Figure 14

Step 3: On the project details page click **API credentials** and click **Edit** beside **Allowed Domain(s)** as shown below,

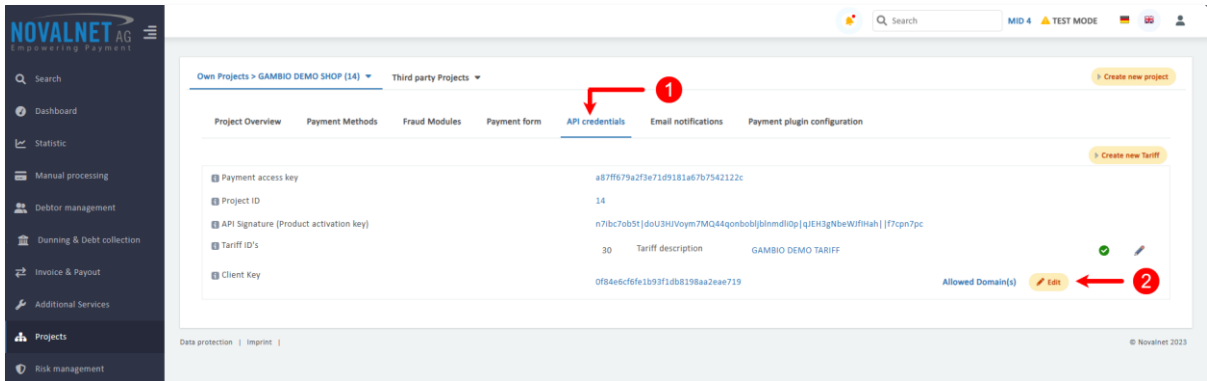


Figure 15

Step 4: Paste your website URL in the **Domain name** field and click **+ Add** as shown below,

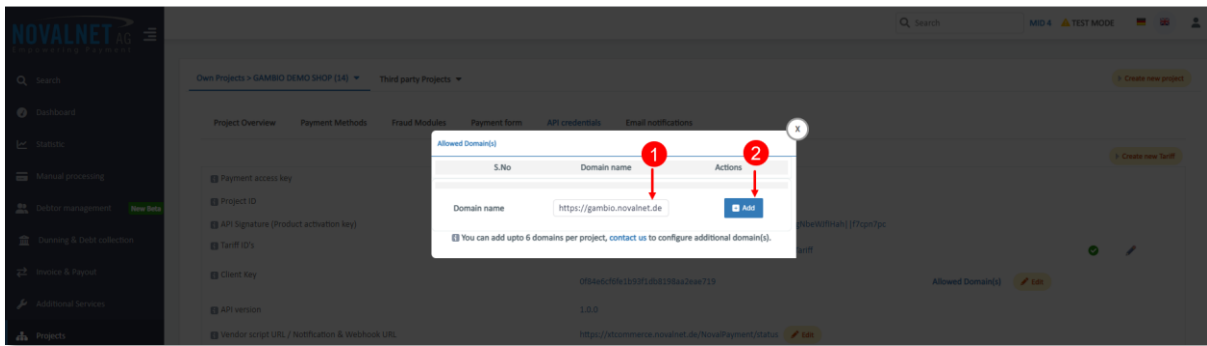


Figure 16

Step 5: After adding the **Domain name**, the web URLs are listed under **Allowed Domain(s)** as shown below,

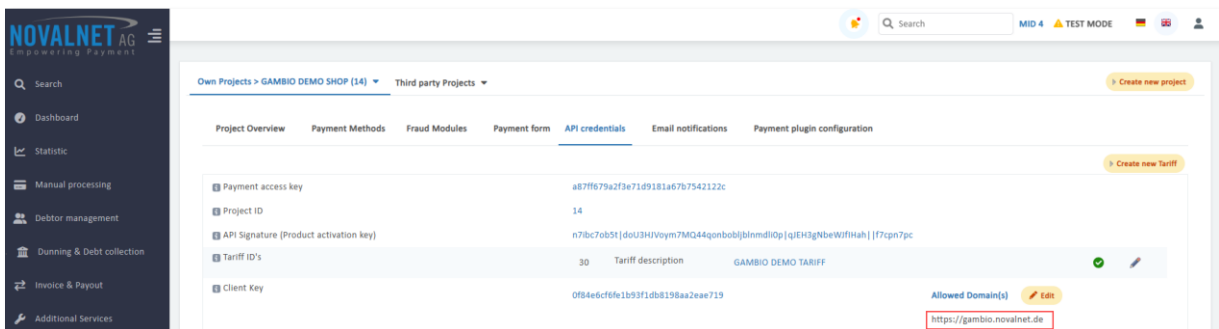


Figure 17

1.4 Payment Activation in the Novalnet Admin Portal

All the Novalnet supported payment methods can be viewed here: www.novalnet.com/payment if you have any questions on the payment methods or have additional payment method requests, please contact sales@novалnet.de

To activate the [preferred payment methods](#) for your website, navigate to [Novalnet Admin Portal](#) → **Projects** → choose your project → **Payment Methods** → click **Edit Payment Methods** in the top right corner as shown below,

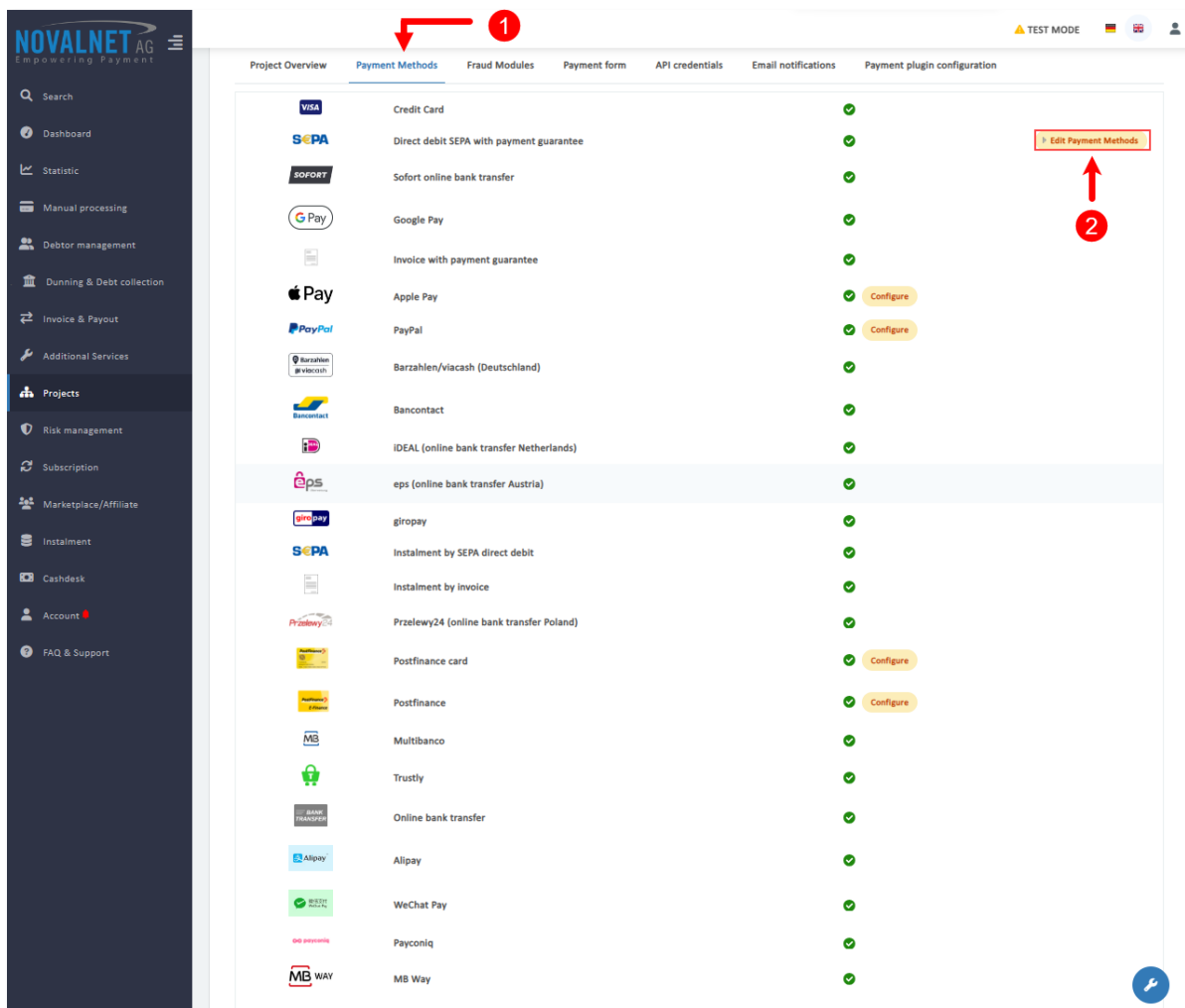


Figure 18

Next, select the preferred payment methods and click **Update** to activate them.

To use the **APPLE PAY** payment method, go to **Apple Pay** → **Configure** → **Add new domain** in the [Novalnet Admin Portal](#) as shown below,

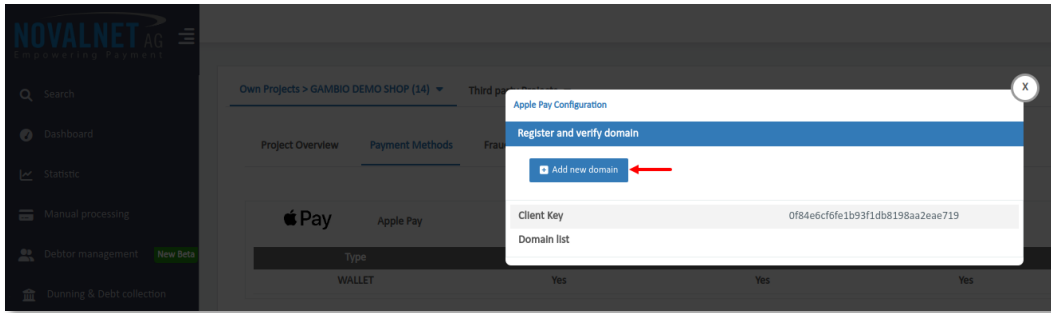


Figure 19

Then, paste your shop domain for which you enable **APPLE PAY**, and click **Download verification file**. You will need to host that verification file in your domain's root directory.

For example: the path should be, <https://gambio.novalnet.de/.well-known/apple-developer-merchantid-domain-association>

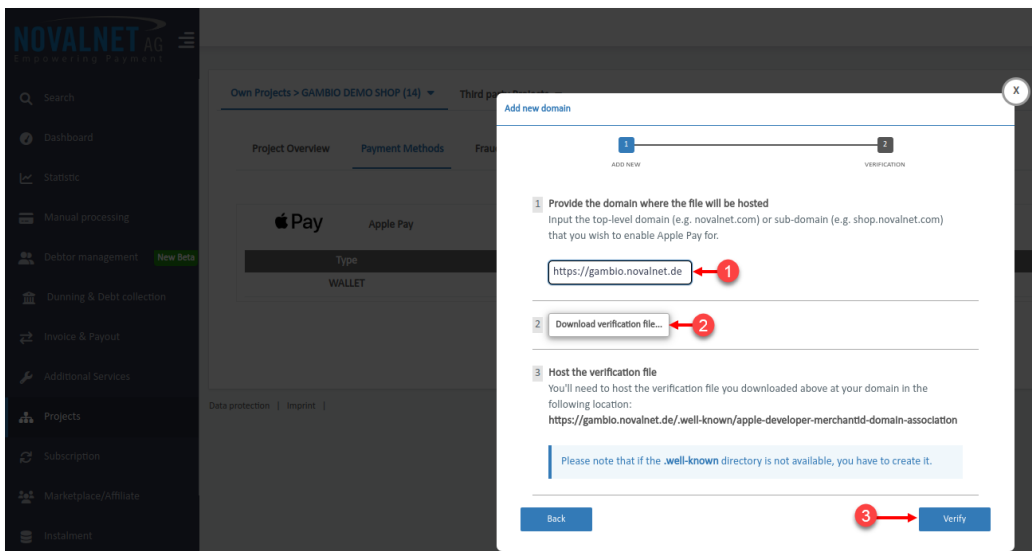


Figure 20

⚠ Please note that if the **.well-known** directory is not available, you have to create it.

To use the **PayPal** payment method, configure the PayPal API details in the [Novalnet Admin Portal](#) as shown below,

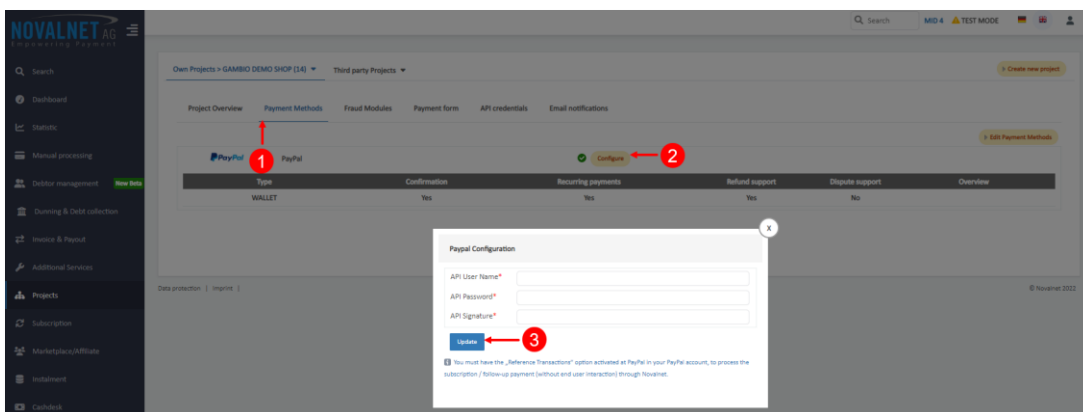


Figure 21

To use the **PostFinance** payment method, configure **Ep2-Merchant ID** details in the [Novalnet Admin Portal](#) as shown below,

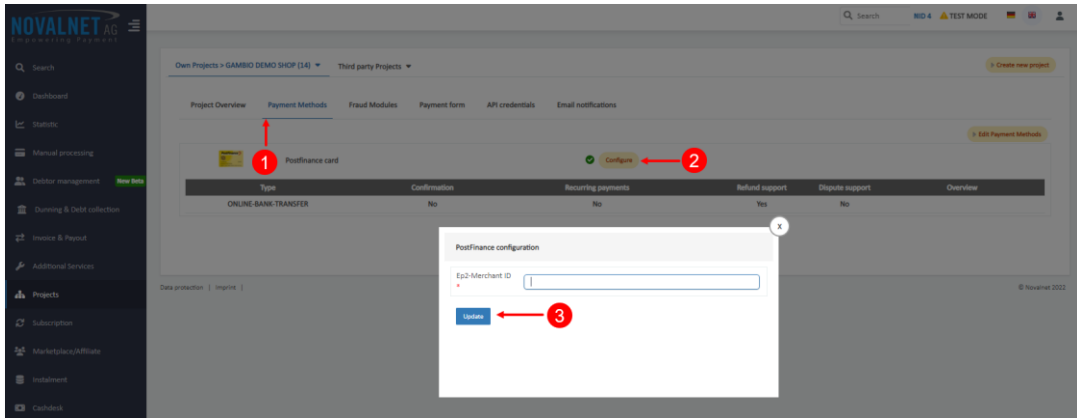


Figure 22

1.5 Payment Activation in the Gambio shop system

After activating the payment methods in the [Novalnet Admin Portal](#), you must enable these payment methods in your shop system to display them on your Gambio checkout page. Navigate to **Modules** → **Payment Systems** → **Miscellaneous** → **added modules** as shown below,

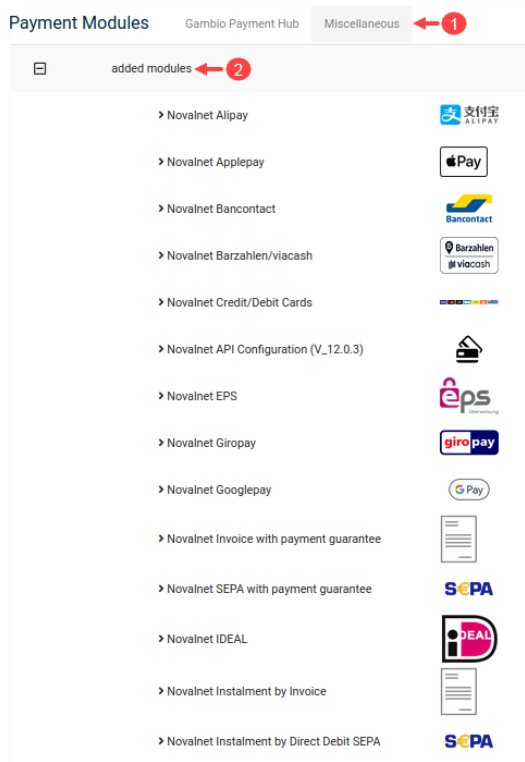


Figure 23

Click **Install** to install the payment method as shown below,

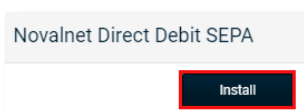


Figure 24

Enable the toggle in **Enable payment method**, to display the payment in the web shop. Click **Update** to save the changes made.

Figure 25

➡ Follow the same procedure to install the remaining Novalnet payment methods.

Refer chapter [3 ADDITIONAL CONFIGURATION](#) for more payment configurations.

📄 If you have any recommendations or suggestions for improvement, kindly share your thoughts with us to develop our payment module further technic@novalnet.de or call us at +49 89 9230683-19.

1.6 Readme procedures

1.6.1 To display transaction details with proper alignment in order confirmation E-mail

To display the transaction details in proper alignment in the order confirmation E-mail for Gambio versions **4.7.1.1 - 4.9.3.0** follow the below procedures

File path: <Root_Directory>/GXMainComponents/View/ThemeContentViews /orders/SendOrderThemeContentView.inc.php

Search the below line

```
$this->set_content_data('NAME', $t_order->customer['name']);
```

and add the following code above the searched line

```
// Novalnet code begins

    if (strpos($t_order->info['payment_method'], 'novalnet') !== false) {
        $t_order->info['comments'] = nl2br($t_order->info['comments']); }

// Novalnet code ends
```

After adding the codes save the file.

1.6.2 To display payment reference in order confirmation E-mail

To display the payment references in the order confirmation E-mail for Gambio versions 4.5.1.0 & 4.6.0.2-4.9.3.0 follow the below procedures

File path: <Root_Directory>/system/classes/orders/SendOrderProcess.inc.php

Search the below line

```
$coo_send_order_content_view = MainFactory::create_object('SendOrderThemeContentView');
```

and add the following code below the searched line

```
// Novalnet code begins
if ((in_array($_SESSION['response']['transaction']['payment_type'], array('INVOICE', 'PREPAYMENT')))
 || $_SESSION['response']['transaction']['payment_type'] == 'GUARANTEED_INVOICE'
 && $_SESSION['response']['transaction']['status'] != 'PENDING') {
$order->info['comments'] .= PHP_EOL. MODULE_PAYMENT_NOVALNET_PAYMENT_REFERENCE_TEXT .PHP_EOL;
$order->info['comments'] .= sprintf(MODULE_PAYMENT_NOVALNET_PAYMENT_REFERENCE, 1, ('TID ' .
$_SESSION['response']['transaction']['tid']) . PHP_EOL;
$order->info['comments'] .= sprintf(MODULE_PAYMENT_NOVALNET_PAYMENT_REFERENCE, 2, ('BNR-' .
MODULE_PAYMENT_NOVALNET_PROJECT_ID . '-' . $order->info['orders_id'])) . PHP_EOL;
} else if ($_SESSION['response']['transaction']['payment_type'] == 'CREDITCARD' &&
($_SESSION['response']['transaction']['payment_data']['cc_3d'] == 1)) {
$order->info['comments'] .= PHP_EOL. MODULE_PAYMENT_NOVALNET_TRANSACTION_ID
.$_SESSION['response']['transaction']['tid'];
$order->info['comments'] .= ($_SESSION['response']['transaction']['test_mode'] == 1) ? PHP_EOL.
MODULE_PAYMENT_NOVALNET_PAYMENT_MODE : "";
} else if($_SESSION['response']['transaction']['payment_type'] == 'INSTALMENT_INVOICE' &&
(in_array($_SESSION['response']['transaction']['status'], array('CONFIRMED', 'ON_HOLD')))) {
$order->info['comments'] = str_replace('###SHOPORDERNUMBER###', $order->info['orders_id'], $order->info['comments']);
}
// Novalnet code ends
```

After adding the codes save the file.

To display the payment references in the order confirmation E-mail for Gambio versions **4.2.0.0 & 4.4.0.3 – 4.9.3.0** follow the below procedures

File path: <Root_Directory>/system/classes/orders/SendOrderProcess.inc.php

Search the below line

```
$coo_send_order_content_view = MainFactory::create_object('SendOrderContentView');
```

and add the following code below the searched line

```
// Novalnet code begins
if ((in_array($_SESSION['response']['transaction']['payment_type'], array('INVOICE', 'PREPAYMENT')))
|| $_SESSION['response']['transaction']['payment_type'] == 'GUARANTEED_INVOICE'
&& $_SESSION['response']['transaction']['status'] != 'PENDING') {
$order->info['comments'] .= PHP_EOL. MODULE_PAYMENT_NOVALNET_PAYMENT_REFERENCE_TEXT .PHP_EOL;
$order->info['comments'] .= sprintf(MODULE_PAYMENT_NOVALNET_PAYMENT_REFERENCE, 1, ('TID ' .
$_SESSION['response']['transaction']['tid'])) . PHP_EOL;
$order->info['comments'] .= sprintf(MODULE_PAYMENT_NOVALNET_PAYMENT_REFERENCE, 2, ('BNR-' .
MODULE_PAYMENT_NOVALNET_PROJECT_ID . '-' . $order->info['orders_id'])) . PHP_EOL;
} else if ($_SESSION['response']['transaction']['payment_type'] == 'CREDITCARD' &&
($_SESSION['response']['transaction']['payment_data']['cc_3d'] == 1)) {
$order->info['comments'] .= PHP_EOL. MODULE_PAYMENT_NOVALNET_TRANSACTION_ID
.$_SESSION['response']['transaction']['tid'];
$order->info['comments'] .= ($_SESSION['response']['transaction']['test_mode'] == 1) ? PHP_EOL.
MODULE_PAYMENT_NOVALNET_PAYMENT_MODE : '';
} else if($_SESSION['response']['transaction']['payment_type'] == 'INSTALMENT_INVOICE' &&
(in_array($_SESSION['response']['transaction']['status'], array('CONFIRMED', 'ON_HOLD')))) {
$order->info['comments'] = str_replace('###SHOPORDERNUMBER###', $order->info['orders_id'], $order->info['comments']);
}
// Novalnet code ends
```

After adding the codes save the file.

1.6.3 To display follow-up process in shop admin orders

To display **Manage transaction & Refund process** options in shop admin orders, follow the below procedures

File path: <Root_Directory>/admin/html/compatibility/order_details.php

Search the below line

```
<div class="content article-table grid">
```

and add the following code above the searched line

```
<?php
// Novalnet code begins

if (preg_match("/novalnet/i", $order->info['payment_method'])) {
include_once(DIR_FS_CATALOG . 'admin/novalnet_extension.php');
appendNovalnetOrderProcess((int)$GLOBALS['oid']);
}
// Novalnet code ends
?>
```

After adding the codes save the file.

1.6.4 To display and process Apple Pay and Google Pay button in shopping cart page

To display and process **Apple Pay** and **Google Pay** buttons on the cart page, follow the below procedures

Step 1: File path: <Root_Directory>/themes/{current-active-theme}/html/system/cart_totals.html

Search the below code

```
{if isset($PAYPAL_EC_BUTTON)}
<div class="paypal-ec-button-container">

</div>
{/if}
</div>
```

and add the following code below the searched line

```
<div id="novalnet_applepay_wallet_div" style="display:none" >
<div id="novalnet_applepay_wallet_button" ></div>
</div>
</br>
<div id="novalnet_googlepay_wallet_div" style="display:none" >
<div id="novalnet_googlepay_wallet_button" ></div>
</div>
<script>
window.onload = function() {
novalnet_wallet_cart_page();}
</script>
```

After adding the codes save the file.

Step 2: File path: <Root_Directory>/shopping_cart.php

Search the below code

```
unset($_SESSION['nvpReqArray']);
```

and add the following code above the searched line

Applepay

```
require_once(DIR_FS_CATALOG.'includes/modules/payment/novalnet_applepay.php');
$applepay_cart = new novalnet_applepay();
$display_page = defined('MODULE_PAYMENT_NOVALNET_APPLEPAY_BUTTON_DISPLAY') ? explode('|',
MODULE_PAYMENT_NOVALNET_APPLEPAY_BUTTON_DISPLAY) : array();
if($applepay_cart->enabled === true && in_array('shopping cart page',$display_page)){
if (defined('MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED') &&
trim(MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED) != '') {
$t_gm_allowed_countries = defined('MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED') ? explode(',',
MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED) : array();
if (!in_array($order->delivery['country']['iso_code_2'], $t_gm_allowed_countries)) {
$this->enabled = false;
}
} else {
$applepay_cart->applepayCart_page_button();
}
}
```

Googlepay

```
require_once(DIR_FS_CATALOG.'includes/modules/payment/novalnet_googlepay.php');
$googlepay_cart = new novalnet_googlepay();
$googlepay_display_page = defined('MODULE_PAYMENT_NOVALNET_GOOGLEPAY_BUTTON_DISPLAY') ? explode('|',
MODULE_PAYMENT_NOVALNET_GOOGLEPAY_BUTTON_DISPLAY) : array();
if($googlepay_cart->enabled === true && in_array('shopping cart page',$googlepay_display_page)){
if (defined('MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED') &&
trim(MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED) != '') {
$t_gm_allowed_countries = defined('MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED') ? explode(',',
MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED) : array();
if (!in_array($order->delivery['country']['iso_code_2'], $t_gm_allowed_countries)) {
$this->enabled = false;
}
} else {
$googlepay_cart->googlepayCart_page_button();
}
}
```

After adding the codes save the file.

1.6.5 To display and process the Apple pay and Google pay button in product page

To display and process **Apple Pay** and **Google Pay** buttons on the product page, follow the below procedures

Step 1: File path: <Root_Directory>

/themes/{current-active-theme}/html/system/product_info_product_box_bottom.html

Search the below code

```
{block name="module_product_info_standard_sticky_box_paypal_ec_button"}
{if isset($PAYPAL_EC_BUTTON)}
<div class="paypal-ec-button-container">

</div>
{/if}
{/block}
```

and add the following code below the searched line

```
<div id="novalnet_applepay_wallet_div" class="button-container" style="display:none" >
<div id="novalnet_applepay_wallet_button" ></div>
</div>
</br>
<div id="novalnet_googlepay_wallet_div" class="button-container" style="display:none" >
<div id="novalnet_googlepay_wallet_button" ></div>
</div>
<script>
window.onload = function() {
novalnet_wallet_product_page();
};
</script>
```

After adding the codes save the file.

Step 2: File path: <Root_Directory>/product_info.php

Search the below code

```
$coo_product_info_view->setLanguageId($language_id->value());
```

and add the following code below the searched line

Applepay

```
require_once(DIR_FS_CATALOG.'includes/modules/payment/novalnet_applepay.php');
$applepay_cart = new novalnet_applepay();
$display_page = defined('MODULE_PAYMENT_NOVALNET_APPLEPAY_BUTTON_DISPLAY') ? explode('|',
MODULE_PAYMENT_NOVALNET_APPLEPAY_BUTTON_DISPLAY) : array();
if($applepay_cart->enabled === true && in_array('product page',$display_page)){
if (defined('MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED') &&
trim(MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED) != '') {
$t_gm_allowed_countries = defined('MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED') ? explode(',',
MODULE_PAYMENT_NOVALNET_APPLEPAY_ALLOWED) : array();
if (!in_array($order->delivery['country']['iso_code_2'],$t_gm_allowed_countries)) {
$applepay_cart->enabled = false;
}
} else {
$applepay_cart->applepayProduct_page_button();}}
```

Googlepay

```
require_once(DIR_FS_CATALOG.'includes/modules/payment/novalnet_googlepay.php');
$googlepay_cart = new novalnet_googlepay();
$display_page = defined('MODULE_PAYMENT_NOVALNET_GOOGLEPAY_BUTTON_DISPLAY') ? explode('|',
MODULE_PAYMENT_NOVALNET_GOOGLEPAY_BUTTON_DISPLAY) : array();
if($googlepay_cart->enabled === true && in_array('product page',$display_page)){
if (defined('MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED') &&
trim(MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED) != '') {
$t_gm_allowed_countries = defined('MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED') ? explode(',',
MODULE_PAYMENT_NOVALNET_GOOGLEPAY_ALLOWED) : array();
if (!in_array($order->delivery['country']['iso_code_2'],$t_gm_allowed_countries)) {
$googlepay_cart->enabled = false;
}
} else {
$googlepay_cart->googlepayProduct_page_button();}
```

After adding the codes save the file.

1.6.6 To display the Barzahlen/viacash payment slip on the shop success page

To display the Barzahlen/viacash payment slip on the shop success page for Gambio versions **4.5.1.0 - 4.8.0.2** – **4.9.3.0** follow the below procedures

Step 1: File path: <Root_Directory>/themes/{current-active-theme}/html/system/checkout_success.html

Search the below line

```
{block name="checkout_success_margin"}
<p>&nbsp;</p>
{/block}
```

and add the following code above the searched line

```
{block name="nn_barzahlen_button"}
{if $NOVALNET_PAYMENT_TYPE}
<script type="text/javascript" data-token="{ $NOVALNET_CHECKOUT_TOKEN}" class="bz-checkout"
src="{ $NOVALNET_CHECKOUT_JS}"></script>
<button id="barzahlen_button" class="bz-checkout-btn">{ $NOVALNET_CASHPAYMENT_SUCCESS_BUTTON}</button>
{/if}
{/block}
<br><br><br>
```

After adding the codes save the file.

Step 2: File path: <Root_Directory>/GXMainComponents/View/ThemeContentViews/checkout/CheckoutSuccessThemeContentView.inc.php

Search the below line

```
$this->set_content_data('LOGOFF_URL', xtc_href_link(FILENAME_LOGOFF, '', 'NONSSL'));
```

and add the following code above the searched line

```
// Novalnet code begins
$nn_get_cashpayment_token = xtc_db_fetch_array(xtc_db_query("select payment_type ,payment_details from
novalnet_transaction_detail where order_no = " . (int)$this->order_id));
if (!empty($nn_get_cashpayment_token)) {
$nn_get_query = json_decode($nn_get_cashpayment_token['payment_details'], true);
if ($nn_get_cashpayment_token['payment_type'] == 'CASHPAYMENT' &&
!empty($nn_get_query['novalnet_checkout_token']) && !empty($nn_get_query['novalnet_checkout_js'])) {
// include language
include_once(DIR_FS_CATALOG ."lang/" . $_SESSION['language'] . "/modules/payment/novalnet.php");
$this->set_content_data('NOVALNET_CASHPAYMENT_SUCCESS_BUTTON',
(defined('MODULE_PAYMENT_NOVALNET_BARZAHLEN_SUCCESS_BUTTON')
MODULE_PAYMENT_NOVALNET_BARZAHLEN_SUCCESS_BUTTON : ''));
$this->set_content_data('NOVALNET_CHECKOUT_TOKEN', $nn_get_query['novalnet_checkout_token']);
$this->set_content_data('NOVALNET_CHECKOUT_JS', $nn_get_query['novalnet_checkout_js']);
$this->set_content_data('NOVALNET_PAYMENT_TYPE', $nn_get_cashpayment_token['payment_type']);
}}
// Novalnet code ends
```

After adding the codes save the file.

2 TESTING AND GOING LIVE

To execute test transactions by navigating to **Modules** → **Payment Systems** → **Miscellaneous** → **added modules** and enable the toggle in **Enable test mode** at each individual payment configuration page. The payment will be processed in the test mode therefore amount for this transaction will not be charged.

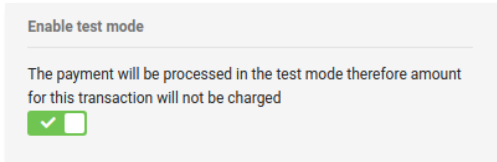


Figure 26

i Refer below URL for the Novalnet test payment data for testing <https://developer.novalnet.com/testing>

Execute orders in LIVE MODE

To proceed with **LIVE** orders, don't forget to disable the toggle in the **Enable test mode** option in the individual payment configuration page.

i If you have any recommendations or suggestions for improvement, kindly share your thoughts with us to develop our payment module further technic@novalnet.de or call us at +49 89 9230683-19.

3 ADDITIONAL CONFIGURATION

3.1 Additional configuration for all the payment methods

For additional payment configuration settings for each payment methods, navigate to **Modules** → **Payment Systems** → **Miscellaneous** → **added modules** in the shop admin panel, choose the respective payment methods and configure the required additional payment settings.

Enable test mode

Enable the toggle in **Enable test mode** to test payments at your checkout page as shown below. In the test mode, the transaction amount will not be charged by Novalnet.

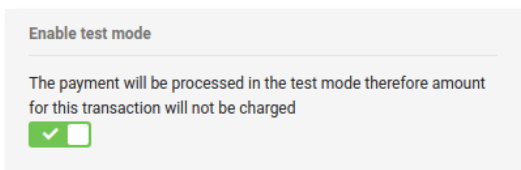


Figure 27

Notification for the buyer

Text entered text in this field will be displayed on the checkout page. The message can be for example: *Thank you for shopping with us.*

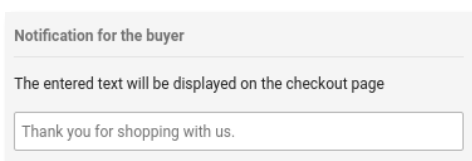


Figure 28

Refer to the below image to view how the notification appears to the buyer.

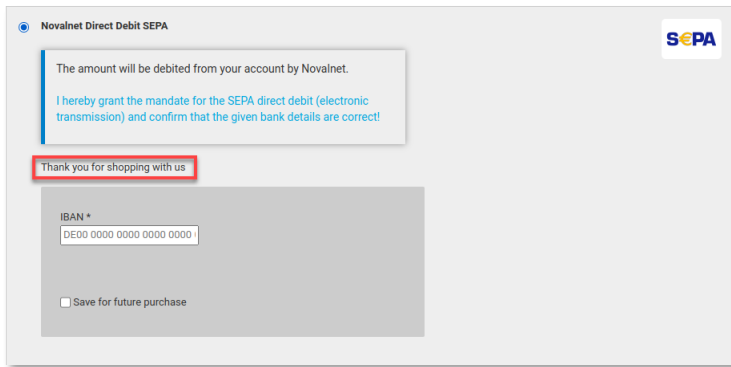


Figure 29

Display Sort Order

The payment methods will be listed on your checkout page (in ascending order) based on your given sorting order.

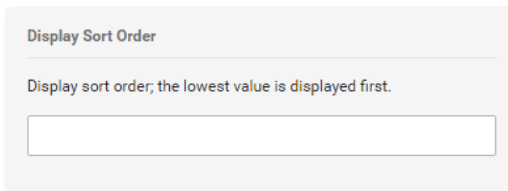


Figure 30

Payment zone

This payment method will be displayed only for the mentioned zone

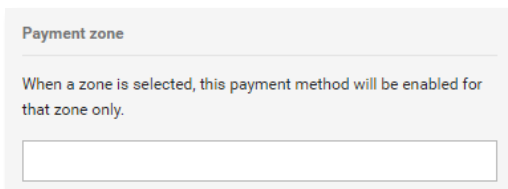


Figure 31

Payment action (Debit immediately / Reserve funds for later/ Authorize with zero amount)

☛ This option is available only for the following payment methods:

Credit/Debit Cards, Direct Debit SEPA, Direct Debit SEPA with payment guarantee, Instalment by Direct Debit SEPA, Invoice, Invoice with payment guarantee, Instalment by invoice, Google pay, Apple pay and PayPal.

You can choose between two options – **Capture** and **Authorize** which are both explained below.

❗ **Capture** - This is the default setting where transactions are directly confirmed, and funds are automatically transferred from buyer's account to the merchant account while placing an order. This can be changed as per your business requirement.

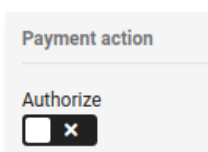


Figure 32

- Authorize** - Payment details are verified while the funds are reserved, which will be captured later. Please note for card payments, if you fail to execute a capture within 14 days the reservation on the card will automatically expire!
- Minimum transaction amount for authorization** - Transactions above this amount will be “authorized” only and captured later. Leave the field blank to authorize all transactions.

Payment action

Choose whether or not the payment should be charged immediately. Authorize verifies payment details and reserves funds to capture it later, giving time for the merchant to decide on the order.

Authorize

Minimum transaction amount for authorization

In case the order amount exceeds the mentioned limit, the transaction will be set on-hold till your confirmation of the transaction. You can leave the field empty if you wish to process all the transactions as on-hold (in minimum unit of currency. E.g. enter 100 which is equal to 1.00).

Figure 33

Authorize with zero amount

This option is available only for the Credit/Debit Cards and Direct Debit SEPA

If the purchase order succeeds, a transaction with the amount 0 is executed. This gives you the advantage of deducting the amount in advance from the buyer (Example: if certain goods have yet to be manufactured or not in stock).

For the desired payment type, select **Authorize with zero amount** and click **Save**.

Payment action

Choose whether or not the payment should be charged immediately.

Authorize with zero amount

Figure 34

To set an amount for each transaction with the amount 0, navigate to **Orders** → **Orders** and Select the respective order. Enter the desired amount for the €0 transaction (enter 1999 = €19.99 in the smallest currency unit) and click **Confirm**.

gambio ADMIN Categories / Products

Order 400242

Order: 400242 | Sum: 19,99 EUR | Date: 12.12.2022 12:29 | Payment Method: Novalnet Direct Debit SEPA | Status: Processing

Book transaction

Novalnet transaction ID: 14660400018307418

Transaction booking amount: 1999 (in minimum unit of currency. E.g. enter 100 which is equal to 1.00)

Confirm Back

Figure 35

The amount will be debited and a new TID will be generated.

Figure 36

Payment due date (in days)

Payment due date (in days) refers to the duration (number of days) given for buyer to complete the payment. The payment process and duration may differ for each payment method.

☛ This option is available only for the following payment methods:

Direct Debit SEPA, Invoice, Prepayment and Barzahlen/ viacash.

Figure 37

- ❗ For **Direct Debit SEPA**, enter the number of days after which the payment is debited (must be between 2 and 14 days).
- ❗ For **Invoice, Prepayment**, enter the number of days given to the buyer to transfer the amount to Novalnet (must be greater than 7 days). If this field is left blank, 14 days will be set by default.
- ❗ For **Barzahlen/viacash**, enter the number of days given to the buyer to pay at a store nearby. If this field is left blank, 14 days will be set by default for slip expiry.

Completed order status

☛ Set the status that will be used for completed orders.

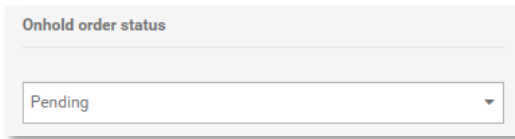
Figure 39

On-hold order status

☛ This option is available only for the following payment methods:

Credit/Debit Cards, Direct Debit SEPA, Direct Debit SEPA with payment guarantee, Instalment by Direct Debit SEPA, Invoice, Invoice with payment guarantee, Instalment by Invoice, PayPal, Applepay and Google pay

Set the status that will be used for on-hold orders until the transaction is confirmed or canceled.

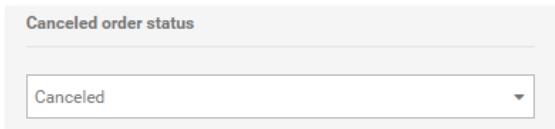


A screenshot of a configuration window titled "Onhold order status". It features a dropdown menu with the word "Pending" selected.

Figure 40

Cancellation order status

☛ Set the status that will be used for cancelled orders.



A screenshot of a configuration window titled "Canceled order status". It features a dropdown menu with the word "Canceled" selected.

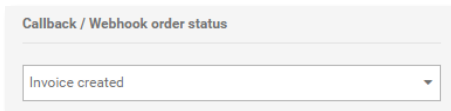
Figure 41

Callback / Webhook order status

☛ This option is available only for the following payment methods:

Invoice, Prepayment, Barzahlen/viacash and Multibanco.

Set the status that will be applied for orders when callback script returns successful payment status.



A screenshot of a configuration window titled "Callback / Webhook order status". It features a dropdown menu with the text "Invoice created" selected.

Figure 42

Payment guarantee configuration

☛ This option is available only for the following payment methods:

Direct Debit SEPA with payment guarantee and Invoice with payment guarantee

When the basic requirements (shown in the image below) are met, Novalnet offers you the option to process payment as guarantee payments. For more information about Novalnet's guarantee payments,

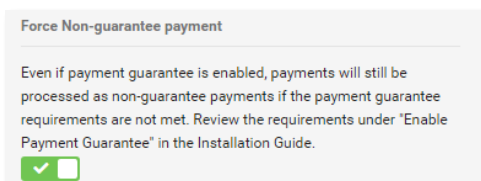
Please visit: <https://developer.novalnet.com/onlinepayments/aboutguarantee#basic-requirements>

Force Non-guarantee payment

☛ This option is available only for the following payment methods:

Direct Debit SEPA with payment guarantee and Invoice with payment guarantee

Enable this option to process payments as non-guarantee payments when the guarantee conditions are not met.



A screenshot of a configuration window titled "Force Non-guarantee payment". It contains a text block explaining that even if payment guarantee is enabled, payments will still be processed as non-guarantee payments if requirements are not met. Below the text is a checked checkbox.

Figure 43

3.2 Additional configuration for Credit/Debit Cards

Enforce 3D secure payment outside EU

We have integrated the Enforce 3D secure fraud module for the Credit Card payment method. It activates 3D secure process based on the filters and settings configured by you. The transaction will be routed to authenticate via 3D secure only if the predefined conditions triggers to do so; otherwise, it will be processed as a non-3D secure transaction.

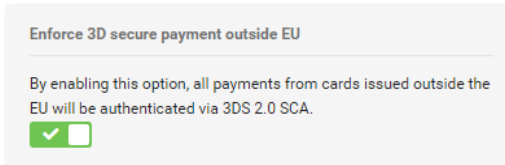


Figure 46

Custom CSS settings

If you wish to change the default design of the Novalnet Credit/Debit Cards form (for example to use your corporate identity), you can do it here. Leave this field unchanged to use the default settings.

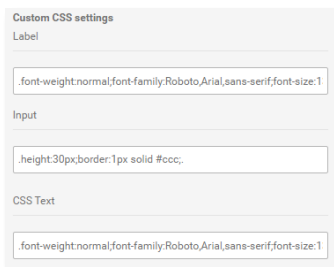


Figure 47

3.3 Additional configuration for Apple Pay payment

Business Name

This is the text that appears as PAY 'BUSINESS NAME' in the Apple Pay modal payment sheet

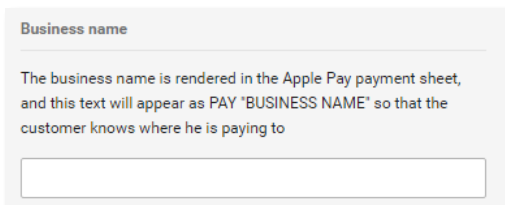


Figure 48

Button Type

Select the button type for Apple Pay from this drop-down.

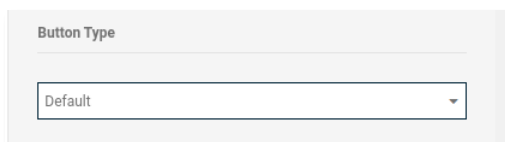


Figure 49

Button Theme

Select the button theme for Apple Pay from this drop-down.



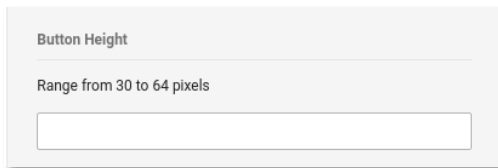
Button Theme

Dark

Figure 50

Button Height

Set the button height for Apple Pay, range from 30 to 64 pixels



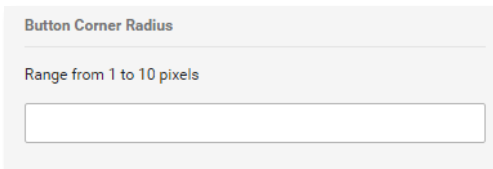
Button Height

Range from 30 to 64 pixels

Figure 51

Button Corner Radius

Set the corner radius of the button for Apple, range from 0 to 10 pixels.



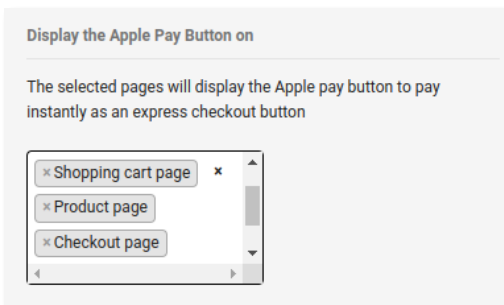
Button Corner Radius

Range from 1 to 10 pixels

Figure 52

Display the Apple Pay Button on

Choose where to display the Apple Pay button in the shop frontend.



Display the Apple Pay Button on

The selected pages will display the Apple pay button to pay instantly as an express checkout button

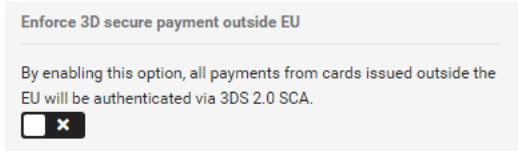
- * Shopping cart page *
- * Product page
- * Checkout page

Figure 53

3.4 Additional configuration for Google Pay payment

Enforce 3D secure payment outside EU

By enabling this option, all payments from cards issued outside the EU will be authenticated via 3DS 2.0 SCA.

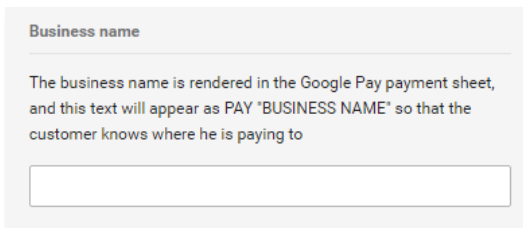


The screenshot shows a configuration panel titled "Enforce 3D secure payment outside EU". Below the title, there is a text box explaining that enabling this option will enforce 3DS 2.0 SCA for payments from cards issued outside the EU. At the bottom of the panel, there is a checkbox that is currently unchecked, with a small 'x' icon to its right.

Figure 54

Business name

This is the text that appears as PAY 'BUSINESS NAME' in the Google Pay modal payment sheet.

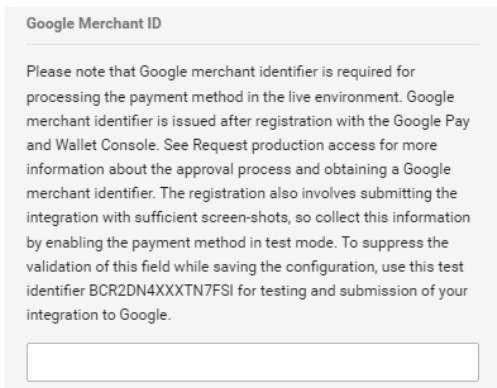


The screenshot shows a configuration panel titled "Business name". Below the title, there is a text box explaining that the business name is rendered in the Google Pay payment sheet and will appear as PAY 'BUSINESS NAME'. Below the text box, there is an empty text input field for entering the business name.

Figure 55

Google Merchant ID

Enter your Google's merchant identifier for processing the payment method in live environment.



The screenshot shows a configuration panel titled "Google Merchant ID". Below the title, there is a text box providing detailed instructions: "Please note that Google merchant identifier is required for processing the payment method in the live environment. Google merchant identifier is issued after registration with the Google Pay and Wallet Console. See Request production access for more information about the approval process and obtaining a Google merchant identifier. The registration also involves submitting the integration with sufficient screen-shots, so collect this information by enabling the payment method in test mode. To suppress the validation of this field while saving the configuration, use this test identifier BCR2DN4XXXTN7FSI for testing and submission of your integration to Google." Below the text box, there is an empty text input field for entering the merchant ID.

Figure 56

Button Type

Select the button type for Google Pay from this drop-down.



The screenshot shows a configuration panel titled "Button Type". Below the title, there is a drop-down menu with "Plain" selected as the current option.

Figure 57

Button Height

Set the button height for Google Pay, range from 30 to 64 pixels.

Figure 58

Display the Google Pay Button on

Choose where to display the Google Pay button in the shop frontend

Figure 59

4 MANAGING GAMBIO E-COMMERCE ADMIN PANEL

4.1 Order Management

Manage your orders and view their details under **Orders** → **Orders** as shown below,

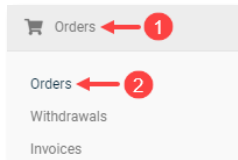


Figure 60

4.2 Transaction Overview

Click **Show** for the particular order number to see an overview of the Novalnet transaction details for that order.

<input type="checkbox"/>	Nr.	Customer	Group	Sum	Payment	Shipping	Country	Date	Status	Kg	1 to 6 (of 6)	<	1 from	>
<input type="checkbox"/>	6	tester tester	Admin	15,00 EUR	NN_SEPA	FREI	DE	24.03.21 - 07:05	Processing	2			20 per page	

Figure 61

Click the **i** icon under **Comments** to view the Novalnet transaction overview.

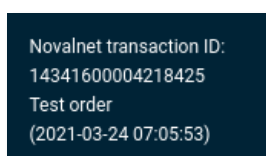


Figure 62

4.3 Order details for Instalment payments

To review the completed and pending payments for a particular instalment order, click the order number and scroll down to the **Instalment Summary** section as shown below,

S.No	Amount	Next instalment date	Paid on	Status	Novalnet Transaction ID
1	10,00 EUR <input type="button" value="Refund"/>	2023-08-06 14:34:03	2023-07-06 11:04:29	Paid	14800600029305500
2	10,00 EUR <input type="button" value="Refund"/>	2023-09-06 14:35:07	2023-07-06 11:05:28	Paid	14894037739662528
3	9,99 EUR <input type="button" value="Refund"/>	-	2023-07-06 11:05:55	Paid	14248569656279471

Figure 63

4.4 Confirming / Canceling a transaction

To confirm or cancel an on-hold payment, go to **Orders** → **Orders** in the shop admin panel to view the order details. Then, click **Show**.

Nr.	Customer	Group	Sum	Payment	Shipping	Country	Date	Status	Kg
6	tester tester	Admin	15,00 EUR	NN_SEPA	FREI	DE	24.03.21 - 07:05	Processing	2

Figure 64

You can use the drop-down list to confirm / cancel the order under **Please select status** field of the **Manage transaction**. Finally, click Confirm to confirm the changes.

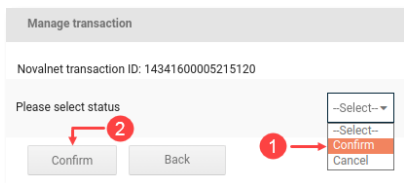


Figure 65

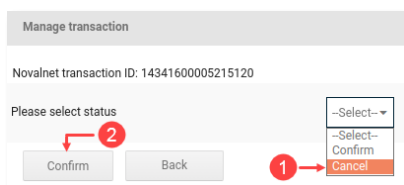


Figure 66

After you have confirmed / cancelled an order, the new transaction status will be shown under the **Comments** section as shown below. Refer to the chapter [4.2 Transaction Overview](#) for more details about the Order Comments.

The transaction has been confirmed on
19.10.2022

Novalnet transaction ID:
14635500017919121
Test order (2022-10-19 15:21:11)

Figure 67

The transaction has been canceled on
26.10.2022

Novalnet transaction ID:
14636200076600224
Test order (2022-10-26 16:10:24)

Figure 68

4.5 Refunding an order

You can refund the buyer either the Full or Partial order amount. Refund can be initiated by navigating to **Orders** → **Orders**. Select the order and then click **Show** to proceed further.

<input type="checkbox"/>	Nr.	Customer	Group	Sum	Payment	Shipping	Country	Date	Status	Kg	1 to 6 (of 6)	<	1 from	>
<input type="checkbox"/>	6	tester tester	Admin	15,00 EUR	NN_SEPA	FREI	DE	24.03.21 - 07:05	Processing	2				

Figure 69

Under **Refund Process** enter the refund amount (in minimum unit of currency. E.g. 100 which is equal to 1.00). Click **Confirm** as shown below

Refund process

Novalnet transaction ID: 14341600005215120

Please enter the refund amount (In minimum unit of currency. E.g. enter 100 which is equal to 1.00)

Figure 70

Once the full or partial order amount has been successfully refunded, the order status will get changed. At this point, a new transaction (TID) will be generated. You will see the new transaction status under **Comments** section of the order. Refer to the chapter [4.2 Transaction Overview](#) for more details about the Order Comments.

Reason for refund

Reason for refund is vital for reference purposes.

Refund process

Novalnet transaction ID: 14635500019902620

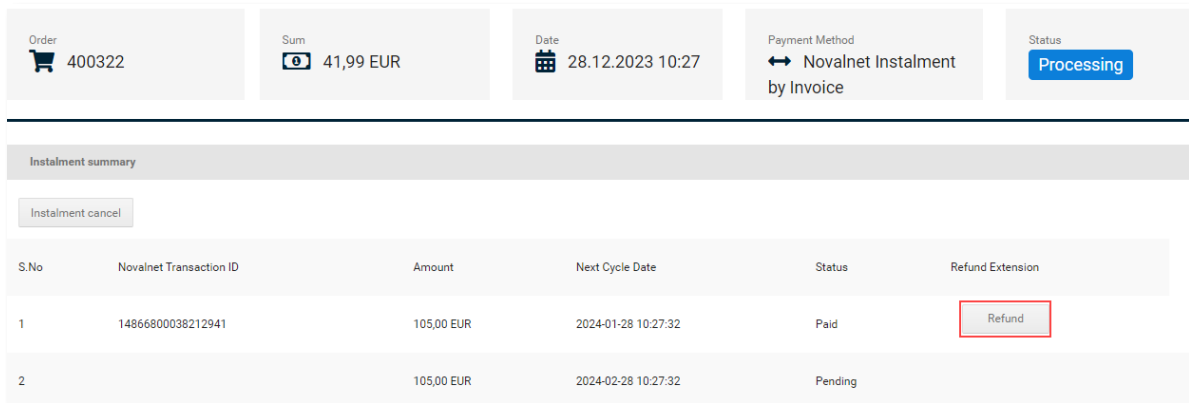
Please enter the refund amount (In minimum unit of currency. E.g. enter 100 which is equal to 1.00)

Reason for refund (optional)

Figure 71

4.5.1 Refunding Instalment orders

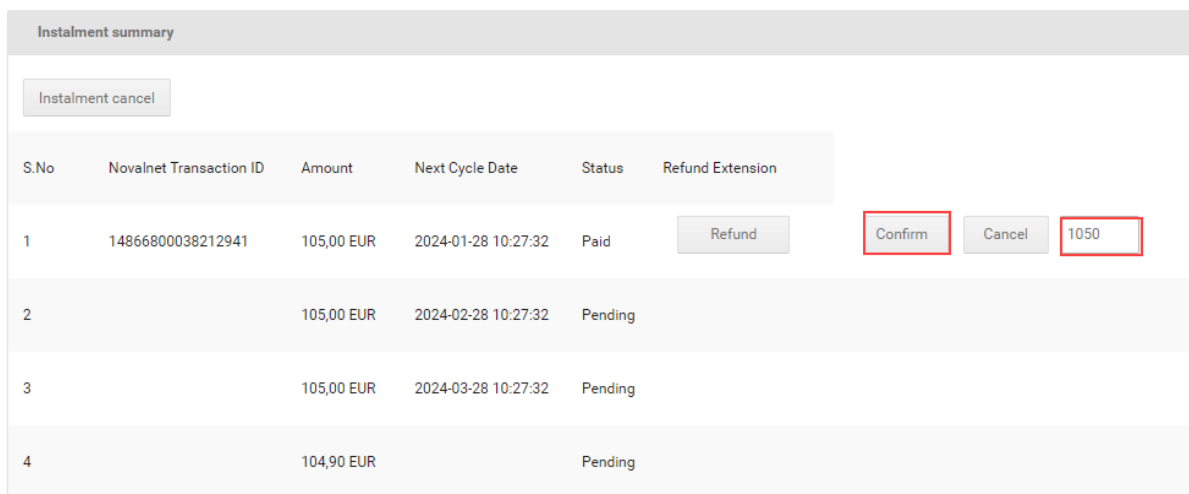
To refund an instalment order, navigate to **Orders** → **Orders** and click a particular order. Scroll down to the **Instalment Summary** section and click **Refund** next to the completed instalment cycle that should be refunded as shown below



S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866800038212941	105,00 EUR	2024-01-28 10:27:32	Paid	<input type="button" value="Refund"/>
2		105,00 EUR	2024-02-28 10:27:32	Pending	

Figure 72

Enter the Refund amount and click **Confirm** as shown below



S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866800038212941	105,00 EUR	2024-01-28 10:27:32	Paid	<input type="button" value="Refund"/> <input type="button" value="Confirm"/> <input type="text" value="1050"/>
2		105,00 EUR	2024-02-28 10:27:32	Pending	
3		105,00 EUR	2024-03-28 10:27:32	Pending	
4		104,90 EUR		Pending	

Figure 73

Once the instalment order amount has been successfully refunded, you will see the new transaction status under Comments section of the order. Refer to the chapter [4.2 Transaction Overview](#) for more details about the Order Comments.

4.6 Cancelling Instalment orders

4.6.1 Cancelling Instalment options

To cancel the instalment orders, navigate to **Orders** → **Orders** and click a particular order. Scroll down to the **Instalment Summary** section and click the **Instalment Cancel** as shown below,

Instalment summary					
Instalment cancel					
S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866800038212941	105,00 EUR	2024-01-28 10:27:32	Paid	<input type="button" value="Refund"/>
2		105,00 EUR	2024-02-28 10:27:32	Pending	

Figure 74

Please follow the below steps to cancel the instalments in two different ways,

1 **Cancel All Instalments** - Clicking this option will cancel the current instalment and all the subsequent instalment orders as shown below,

Order	Sum	Date	Payment Method	Status
400312	23,49 EUR	26.12.2023 14:06	↔ Novalnet Instalment by Direct Debit SEPA	Processing

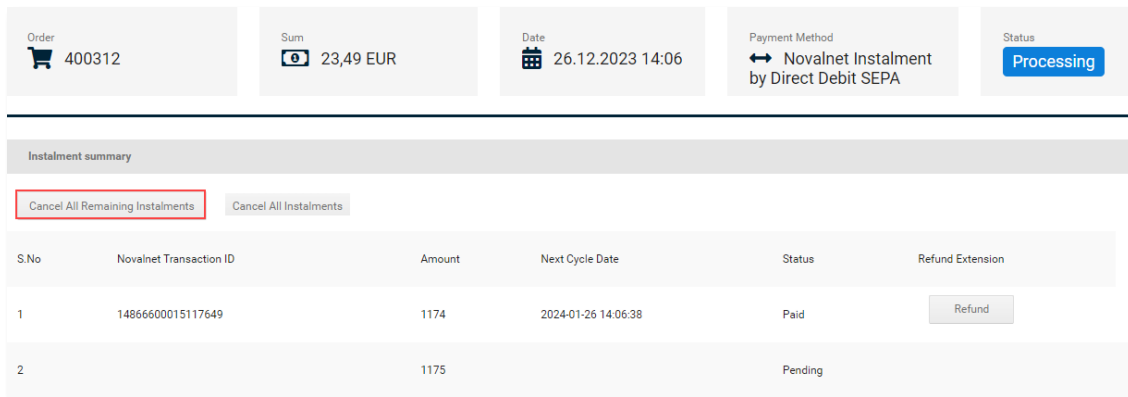
Instalment summary					
<input type="button" value="Cancel All Remaining Instalments"/> <input type="button" value="Cancel All Instalments"/>					
S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866600015117649	1174	2024-01-26 14:06:38	Paid	<input type="button" value="Refund"/>
2		1175		Pending	

Figure 75

Instalment summary					
S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866800017409821	105,00 EUR	2024-01-28 07:18:05	Refunded	
2	14866800017619576	105,00 EUR	2024-02-28 07:18:05	Refunded	

Figure 76

❗ Cancel All Remaining Instalments - Clicking this option will cancel all the upcoming instalment orders excluding the current instalment in progress as shown below,



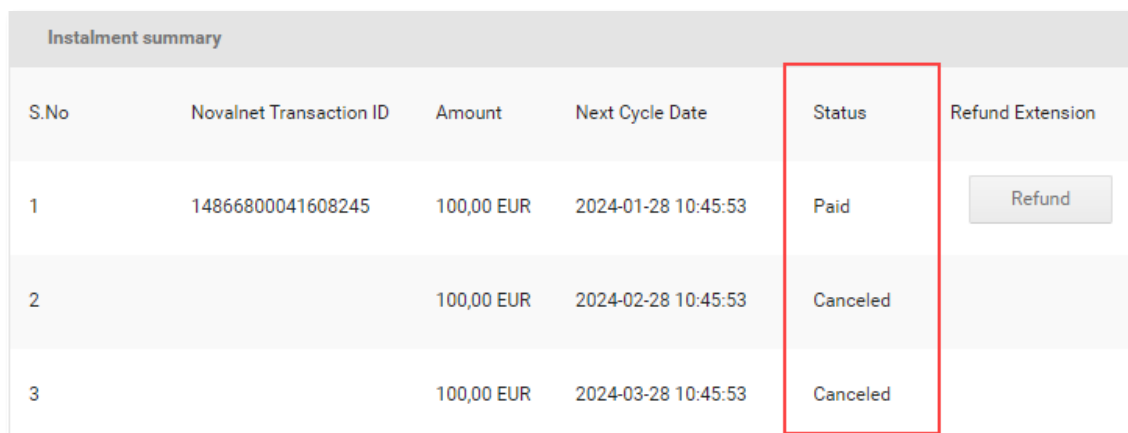
Order: 400312 | Sum: 23,49 EUR | Date: 26.12.2023 14:06 | Payment Method: Novalnet Instalment by Direct Debit SEPA | Status: Processing

Instalment summary

Cancel All Remaining Instalments | Cancel All Instalments

S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866600015117649	1174	2024-01-26 14:06:38	Paid	<input type="button" value="Refund"/>
2		1175		Pending	

Figure 77



Instalment summary

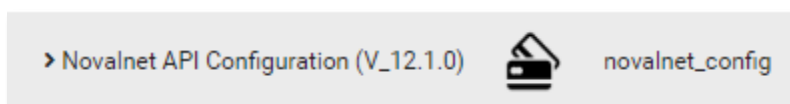
S.No	Novalnet Transaction ID	Amount	Next Cycle Date	Status	Refund Extension
1	14866800041608245	100,00 EUR	2024-01-28 10:45:53	Paid	<input type="button" value="Refund"/>
2		100,00 EUR	2024-02-28 10:45:53	Canceled	
3		100,00 EUR	2024-03-28 10:45:53	Canceled	

Figure 78

5 UNINSTALLATION

To uninstall the Novalnet Payment module, please follow the steps below.

Step 1: Navigate to **Modules** → **Payment Systems** → **Miscellaneous** → **added modules** → **Novalnet API Configuration (V_12.1.0)** as shown below




> Novalnet API Configuration (V_12.1.0)  novalnet_config

Figure 79

Step 2: Select Novalnet API Configuration and click **Uninstall**.



Figure 80

➡ Follow the same procedure to uninstall the remaining Novalnet payment methods.

6 TECHNICAL SUPPORT THROUGH NOVALNET

For any questions or further enquiries please contact one of the following departments as per your requirement. Our in-house experts are ready to assist you in case of queries or issues.

For **installation assistance** contact technic@novalnet.de or call +49 89 9230683-19.

For a **merchant account, new payment module or additional payment methods**, please contact sales@novalnet.de or call +49 89 9230683-20.

If you have any recommendations or suggestions for improvement, kindly share your thoughts with us to develop our payment module further technic@novalnet.de or call us at +49 89 9230683-19.

For our License details, see for [Freeware License Agreement](#).