Stripe | WeWeb Documentation

|  |  |
| --- | --- |
| URL source | <https://docs.weweb.io/plugins/extensions/stripe.html> |
| Date de scraping | 2025-04-08T15:55:46.990809+00:00 |

## Stripe ​

Let's see how you can setup Stripe in WeWeb so that your web-app users can make payments.

## Configure the plugin ​

To add the Stripe Plugin to your WeWeb project, go to the Plugins > Extensions menu:  
  
Once you have added the Stripe Extension, you will be invited to set it up with your production API keys:

### Images

<https://docs.weweb.io/assets/stripe1.E0N6U_ZV.png>

<https://docs.weweb.io/assets/stripe2.BddO2Mm1.png>

## Add a payment form ​

Once you have set up the Stripe plugin, you will have access to the payment form element.  
To display a Stripe payment form on your web app, you'll need to add a Form Container with a submit button:  
  
Then, you'll want to remove the input so you're only left with the button element and add the Stripe - Payment element inside the form container you added above:  
  
In the Stripe payment element's Settings , you'll need a client secret:

### Images

<https://docs.weweb.io/assets/stripe3.Cf6FygCx.png>

<https://docs.weweb.io/assets/stripe4.wmyhJmIV.png>

<https://docs.weweb.io/assets/stripe5.Ctabq1bx.png>

## Create client intent & client secret workflow ​

To get a client secret, we first need to create a client intent. We do this by creating a workflow that triggers on page load.  
In the first step, we create a payment intent, decide which payment methods the client can use and which Stripe products they can purchase (the Price ID will come from your Stripe account):  
  
We will then test Action 1 because we need to bind its result in Action 2.  
In step two, we setup a Change variable value action to update a payment\_intent\_client\_secret variable (type Query) with the value of client\_secret in the result of the previous action:  
  
Once our workflow triggered on page load is complete, we will go back to our Stripe - Payment element and bind the payment\_intent\_client\_secret variable to the Client secret setting:

### Images

<https://docs.weweb.io/assets/stripe6.CLpKFrU3.png>

<https://docs.weweb.io/assets/stripe7.DfuuWfww.png>

<https://docs.weweb.io/assets/stripe5.Ctabq1bx.png>

## Confirm payment workflow ​

Finally, on the form container, we will trigger a workflow On submit to confirm the payment action and redirect the user to a checkout page when they submit the payment form:

### Images

<https://docs.weweb.io/assets/stripe8.BVblTJsa.png>

## Fake payment ​

In order to move forward with the tutorial, you will need to fake a payment intent with a fake card. Read more about simulating payments in the Stripe documentation .  
Once you've entered fake payment details, you can submit the form to trigger the workflow:  
  
You should be redirected to the checkout page.

### Images

<https://docs.weweb.io/assets/stripe9.Dfo0Fy3x.png>

## Add a checkout page ​

On the checkout page, we'll create a workflow that triggers on page to retrieve the user's payment intent:

## Action 1: Retrieve the payment intent ​

We will add the Retrieve payment intent action and bind this to the payment\_intent\_client\_secret variable we created earlier.  
It will look something like this:  
  
If you've named your variable correctly and faked a payment with a fake Stripe card, you should be able to see the the payment\_intent\_client\_secret query string in the URL, which is retrieved by WeWeb in the payment\_intent\_client\_secret variable:

### Images

<https://docs.weweb.io/assets/stripe10.BlxU1k24.png>

<https://docs.weweb.io/assets/stripe11.DP4yTEzK.png>

## Action 2: Store the payment intent ​

After testing Action 1, we will create an Action 2 to store the payment intent in a variable of type Object.  
We will choose a Change variable value action and bind it to the result of the previous action in the workflow:  
  
Again, we need to test this action if we want to update the variable.  
Then, we could use the information from the variable to style the checkout page.  
For example:

### Images

<https://docs.weweb.io/assets/stripe12._jaXojal.png>

<https://docs.weweb.io/assets/stripe13.CAF0io9j.png>

## Before publishing ​

When you publish your application, the Stripe - payment element will not be rendered on the page by default because it can only be rendered if a payment intent is found and, when the browser first loads the page, the client intent will not have been created yet.  
In order to ensure the Stripe payment element is rendered on the page, you'll need to bind it's Conditional rendering setting to the payment\_intent\_client\_secret variable:  
  
That way, when the workflow is triggered on page load and the payment intent variable updated with the client secret, the Stripe payment element will be rendered on the page.

### Images

<https://docs.weweb.io/assets/stripe14.BAJlQnga.png>

## Video Mini-Course ​

If you want to dig deeper and learn how to develop a full Stripe payment system in WeWeb, you can watch our YouTube series on the topic:  
How to Implement Stripe Checkout in a WeWeb app  
How to Implement Stripe Subscriptions in a WeWeb app  
How to Implement Stripe Elements in a WeWeb app