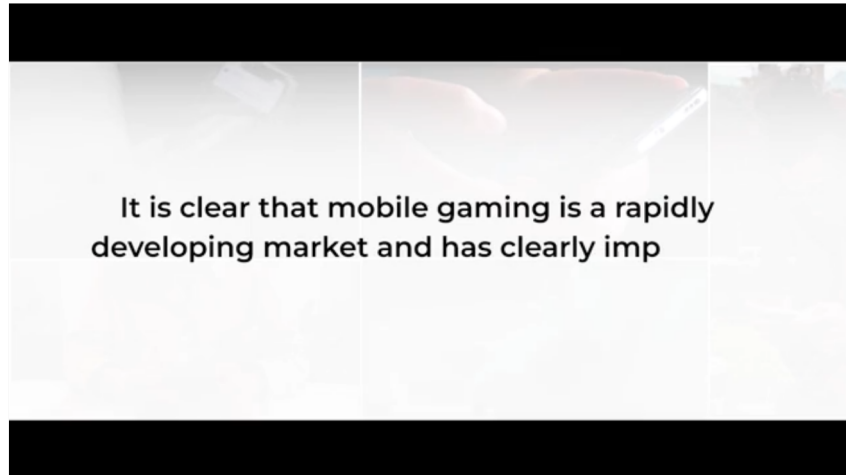# Laravel 9 Stripe Payment Gateway Integration Tutorial

By Hardik Savani   July 1, 2022   Category : Laravel

Powered By **VDO.AI**

Hi,

Here, I will show you laravel 9 stripe integration. you can understand a concept of stripe payment gateway integration in laravel 9. Here you will learn laravel 9 stripe payment gateway example. I explained simply step by step laravel 9 payment integration stripe example. you will do the following things for stripe integration in laravel 9.

You need to create a stripe developer account and need to get API key and secret from there. Then we will use stripe/stripe-php composer library for the stripe payment gateway in laravel 9. I write step by step integration for the stripe payment gateway.

Stripe is a very popular and secure internet payment gateway company that helps to accept payments worldwide. Stripe provides really nice development interface to start and you don't have to pay subscription charges to learn it provides a free developer account, before starting to code in your app.

I will give you an example from scratch to implement a stripe payment gateway in the laravel 9 application. You just need to follow a few steps to get a full example to pay.

X

Step 1: Install Laravel 9

This is optional; however, if you have not created the laravel app, then you may go ahead and execute the below command:

```
composer create-project laravel/laravel example-app
```

Step 2: Install stripe-php Package

In this step we need to install stripe-php via the Composer package manager, so one your terminal and fire bellow command:

```
composer require stripe/stripe-php
```

Step 3: Set Stripe API Key and SECRET

X

Now, we need t   ey and secret
and add bellow

.env

```
STRIPE_KEY=pk_test_reFxwbsm9cdCKASdTfxAR
STRIPE_SECRET=sk_test_oQMFWteJiPd4wj4AtgApY
```

Step 4: Create Controller File

in next step, now we have create new controller as StripePaymentController and write both method on it like as bellow, So let's create both controller:

app/Http/Controllers/StripePaymentController.php

```php
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Session;
use Stripe;

class StripePaymentController extends Controller
{
    /**
     * success response method.
     *
     * @return \Illuminate\Http\Response
     */
    public function stripe()
    {
        return view('stripe');
    }

    /**
     * success response method.
     *
     * @return \Illuminate\Http\Response
     */
    public function stripePost(Request $request)
    {
        Stripe\Stripe::setApiKey(env('STRIPE_SECRET'));

        Stripe\Charge::create ([
                "amount" => 100 * 100,
                "currency" => "usd",
                "source" => $request->stripeToken,
                "description" => "Test payment from itsolutionstuff.com."
        ]);

        Session::flash('success', 'Payment successful!');

        return back();
    }
}
```

If you must need to pass customer name and address with shipping address then you can use bellow method code:

```
/**
 * success response method.
 *
 * @return \Illuminate\Http\Response
 */
public functi
{
    Stripe\St
```
X

```php
        $customer = Stripe\Customer::create(array(
                "address" => [
                        "line1" => "Virani Chowk",
                        "postal_code" => "360001",
                        "city" => "Rajkot",
                        "state" => "GJ",
                        "country" => "IN",
                ],
                "email" => "demo@gmail.com",
                "name" => "Hardik Savani",
                "source" => $request->stripeToken
        ));

    Stripe\Charge::create ([
            "amount" => 100 * 100,
            "currency" => "usd",
            "customer" => $customer->id,
            "description" => "Test payment from itsolutionstuff.com.",
            "shipping" => [
              "name" => "Jenny Rosen",
              "address" => [
                "line1" => "510 Townsend St",
                "postal_code" => "98140",
                "city" => "San Francisco",
                "state" => "CA",
                "country" => "US",
              ],
            ]
    ]);

    Session::flash('success', 'Payment successful!');

    return back();
}
```

Step 5: Create Routes

In this step, we will create two routes for get request and another for post request. So, let's add new route on that file.

routes/web.php

```php
<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\StripePaymentController;

/*
|--------------------------------------------------------------------
| Web Routes                                                      |X|
|-----------
|
| Here is wh
| routes are
```

```
| contains the "web" middleware group. Now create something great!
|
*/

Route::controller(StripePaymentController::class)->group(function(){
    Route::get('stripe', 'stripe');
    Route::post('stripe', 'stripePost')->name('stripe.post');
});
```

Step 6: Create Blade File

In Last step, let's create stripe.blade.php(resources/views/stripe.blade.php) for layout and write code of jquery to get token from stripe here and put following code:

resources/views/stripe.blade.php

X

X

```
                $input.parent().addClass('has-error');
                $errorMessage.removeClass('hide');
                e.preventDefault();
            }
        });

        if (!$form.data('cc-on-file')) {
          e.preventDefault();
          Stripe.setPublishableKey($form.data('stripe-publishable-key'));
          Stripe.createToken({
            number: $('.card-number').val(),
            cvc: $('.card-cvc').val(),
            exp_month: $('.card-expiry-month').val(),
            exp_year: $('.card-expiry-year').val()
          }, stripeResponseHandler);
        }

    });

    /*------------------------------------------
    -------------------------------------------
    Stripe Response Handler
    -------------------------------------------
    -------------------------------------------*/
    function stripeResponseHandler(status, response) {
        if (response.error) {
            $('.error')
                .removeClass('hide')
                .find('.alert')
                .text(response.error.message);
        } else {
            /* token contains id, last4, and card type */
            var token = response['id'];

            $form.find('input[type=text]').empty();
            $form.append("<input type='hidden' name='stripeToken' value='" + token + "'/>");
            $form.get(0).submit();
        }
    }

});
</script>
</html>
```

Run Laravel App:

All the required steps have been done, now you have to type the given below command and hit enter to run the Laravel app:

```
php artisan serve
```

Now, Go to your web browser, type the given URL and view the app output:

```
http://localhost:8000/stripe
```

Output:

X

Laravel 9 - Stripe Payment Gateway Integration Example
ItSolutionStuff.com

Payment Details

Name on Card

Card Number

CVC   **Expiration Month**   **Expiration Year**

ex. 311   MM   YYYY
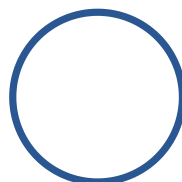
Pay Now ($100)

Now you can check with following card details:

> **Read Also:** [Laravel 9 Google Recaptcha V3 Example Tutorial](#)

```
Name: Test
Number: 4242 4242 4242 4242
CSV: 123
Expiration Month: 12
Expiration Year: 2028
```

I hope it can help you...

Tags :   Laravel   Laravel 9

Hardik Savani

I'm a full-stack developer, entrepreneur and owner of Aatman Infotech. I live in India and I love to write tutorials and tips that can help to other artisan. I am a big fan of PHP, Laravel, Angular, Vue, Node, Javascript, JQuery, Codeigniter and Bootstrap from the early stage. I believe in Hardworking and Consistency.   X

We are Recommending you

› Laravel 9 Fullcalendar Ajax Tutorial Example

› Laravel 9 One to Many Eloquent Relationship Tutorial

› Laravel 9 QR Code Generator Example

› Laravel 9 Google Charts Tutorial Example

› Laravel 9 Socialite Login with Twitter Account Example

› Laravel 9 Resize Image Before Upload Example

› Laravel 9 Scout Full Text Search Tutorial

› Laravel 9 Drag and Drop File Upload with Dropzone JS

› Laravel 9 Autocomplete Search from Database Example

› Laravel 9 Yajra Datatables Example Tutorial

› Laravel 9 Markdown | Laravel 9 Send Email using Markdown Mailables

› Laravel 9 Eloquent Mutators and Accessors Example

X

Popular Posts

› Laravel 7/6 Queue Tutorial | Laravel 7/6 Send Mail with Queue Example

› Laravel 9 Generate Test or Dummy Data using Factory Tinker

› Laravel 9 Auth with Inertia JS Jetstream Example

› How to resolve - tokenmismatchexception in verifycsrftoken.php Laravel 5 Ajax ?

› Laravel 9 Change Date Format Examples

› How to use Google Charts in Laravel 8?

X

› Laravel Cal

› How to crea

> Laravel 8 Ajax Image Upload Example

> Laravel - Implement Flash Messages with example

> Laravel - Line Chart Example using Google Charts API

Categories

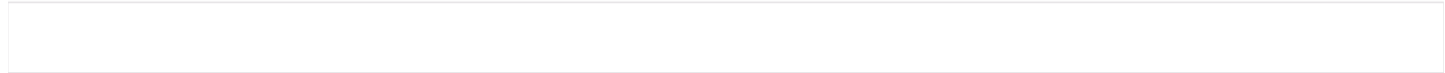|		
Laravel	Angular	PHP
jQuery	Python	Bootstrap
Javascript	MySql	Ajax
Node JS	Ubuntu	HTML
Codeigniter	Guest Post	Vue.JS
React JS	Git	Server
Installation	JSON	CSS
Google Map	.htaccess	SQL
Google API	JQuery UI	Typeahead JS
Socket.io	Axios	Elasticsearch

Latest Posts

> How to Install PHP imap Extension in Ubuntu?

> How to Install PHP Curl Extension in Ubuntu?

> How to Increase memory_limit in PHP Ubuntu?

> How to Increase max_input_vars in PHP Ubuntu?

> How to Increase upload_max_filesize in PHP Ubuntu?

X

X

X

X

X

X

Subscribe to the Email Newsletter

Enter Your Email Address....

SHARES

f Facebook   Twitter   in LinkedIn   Github

X

Home   List Of Categories   List of Tags   Disclaimer   Latest Posts   Contact US   About US   Advertisement

© 2016-2023 All Rights Reserved • www.itsolutionstuff.com

f Facebook   Twitter   in LinkedIn   Github

X

SHARES