TECHPOOL

≡   🔍

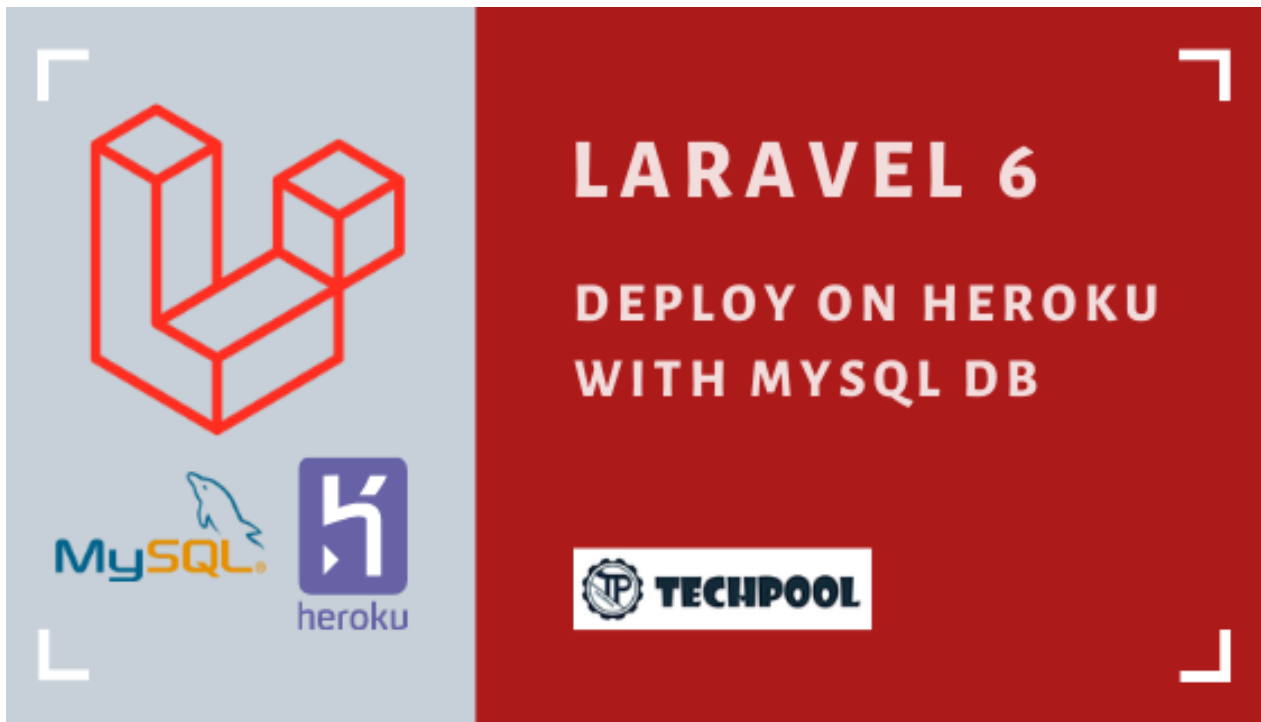# Deploy a Laravel application on Heroku with MySQL database

by TechPool - November 22, 2019 - 2 Comments

Hello guys! What's going on? I hope you are doing well. I'm going to guide you for an interesting tutorial today. Have you ever deployed a **Laravel** application which makes anyone to access on web? If you haven't, this is the article for you! You have only simple things to do! **Just Follow me**...

Step 1 - Create a Heroku Account and Install Heroku CLI

Go to https://heroku.com and create an account for you. Then you have to install Heroku CLI to your machine. It helps to connect with Heroku cloud platform using commands on CLI.

Follow this link:

Message us

https://devcenter.heroku.com/articles/heroku-cli#download-and-install

Step 2 - Create a new Laravel Project

Open terminal and create a project like this..

**composer create-project laravel/laravel herokudep99**

Step 3 - Heroku Login

Open your terminal and Login to Heroku platform. Provide your username and password.
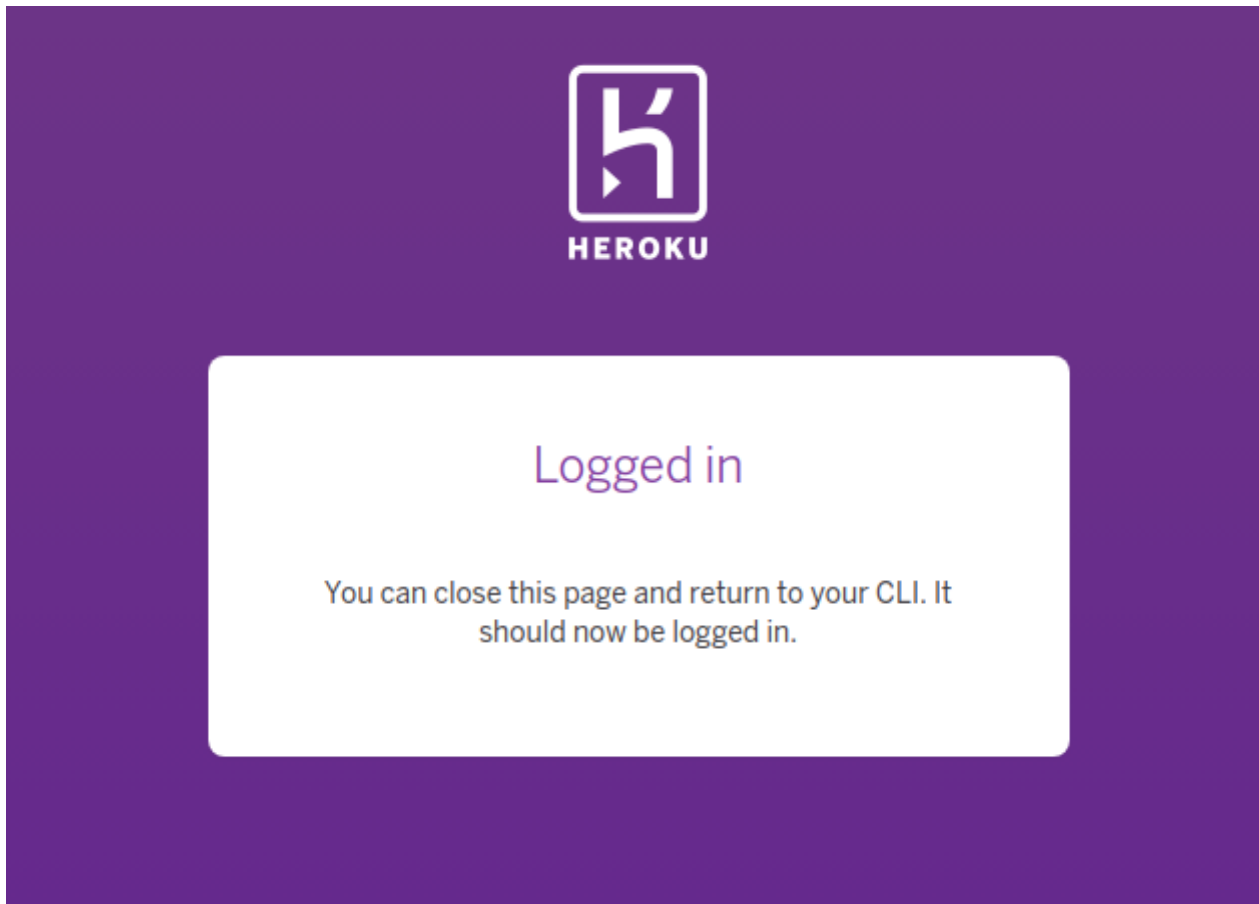
heroku login

Then you will be navigated into a browser window to login.

```
pool@X542UQ     ~/Desktop/testdeploy     heroku login        ✓   584   21:08:22
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/browser/3a51b6a6-d084-44fd-9
c95-aa07f634fbb4
heroku: Waiting for login...
```

Log in to the Heroku CLI

Log in
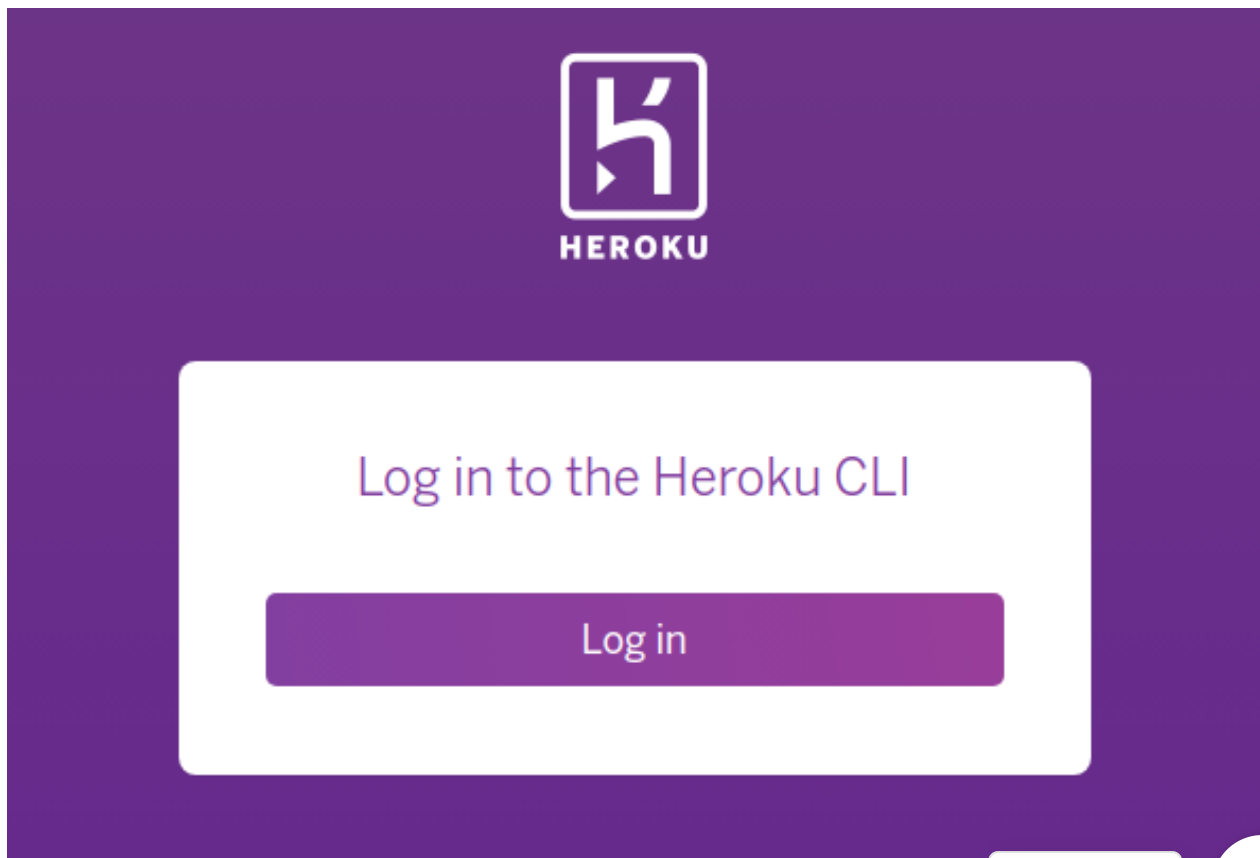
Message us

Now you will be able to communicate with Heroku. Let's configure our app to be able to connect with MYSQL DB when it's deployed to Heroku.

Step 4 - Configure Database

NOTE:

The usual configuration done using .env file in Laravel is not going work on deployed app because that file is not pushed to Heorku when app is deployed.

Open **config/datababe.php** and modify it like this. Insert the below code after this line.
use Illuminate\Support\Str;

```
Code

$url = parse_url(getenv("CLEARDB_DATABASE_URL"));
$host = $url["host"] ?? null;
$username = $url["user"] ?? null;
$password = $url["pass"] ?? null;
$database = substr($url["path"], 1);
```

Message us

Next update the MYSQL array in this file like this...It will take care of DB.

```
Code

    'mysql' => [
        'driver' => 'mysql',
        'host' => $host,
        'port' => env('DB_PORT', '3306'),
        'database' => $database,
        'username' => $username,
        'password' => $password,
        'unix_socket' => env('DB_SOCKET', ''),
        'charset' => 'utf8mb4',
        'collation' => 'utf8mb4_unicode_ci',
        'prefix' => '',
        'strict' => true,
        'engine' => null,
    ]
```

Now create a Model to perform a migration with MYSQL.

php artisan make:model Product -m

Update the migration file created in **database/migrations** folder.

```
Code

    Schema::create('products', function (Blueprint $table) {
        $table->bigIncrements('id');
        $table->string('product_name');
        $table->string('product_description');
        $table->timestamps();
    });
```

Let's create a seeder and prepare some dummy objects.

php artisan make:seed ProductsTableSeeder

Find the relevant seeder with **database/seeds** folder. Modify the seeder file   Message us

```
Code

<?php

use Illuminate\Database\Seeder;
use App\Product;

class ProductsTableSeeder extends Seeder
{
    public function run()
    {
        Product::create([
            'product_name' => 'MacBook Pro',
            'product_description' => 'Apple Laptop'
        ]);

        Product::create([
            'product_name' => 'Huawei GR5 2017',
            'product_description' => 'Andriod Smart Phone'
        ]);
    }
}
```

Now tell the main DatabaseTableSeeder to include this seeder when deploying...

```
Code

    public function run()
    {
        $this->call(ProductsTableSeeder::class);
    }
```

Step 5 - Push the Project to Heroku

Go into the project folder and follow me by typing the below commands. Give a proper unique name with some regex.

Message us

git init

heroku apps:create herokudep99

git init

git add .

echo "web: vendor/bin/heroku-php-apache2 public/" > Procfile

git commit -m "add files to heroku"

Now go to **https://dashboard.heroku.com/apps** and select the app you have already created. Select Resources tab and search for **clearDB MYSQL**. It is the **addon** available in Heorku for MYSQL configuration. Select this addon and click on **Provision** button.
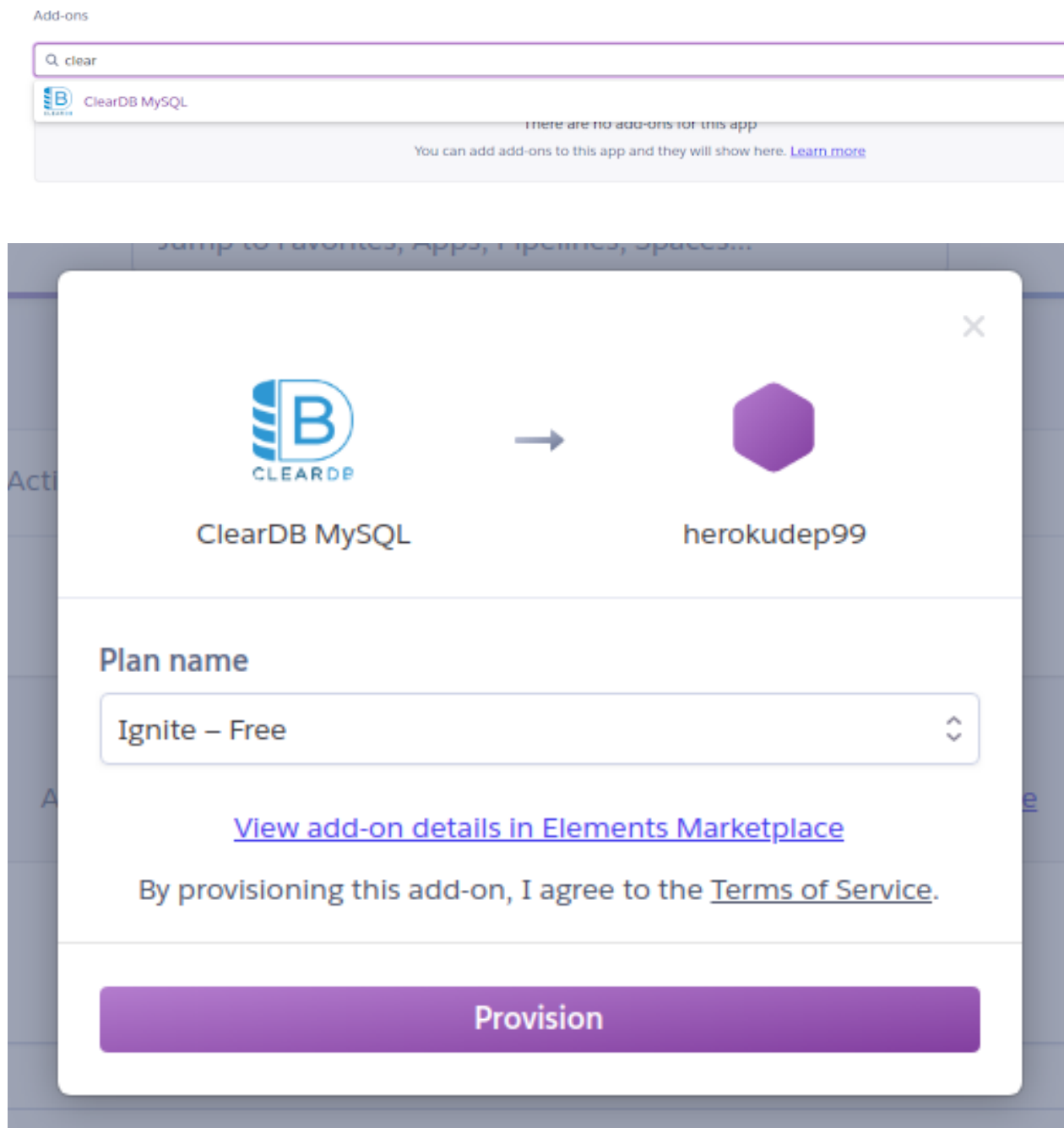
Use the below command on terminal to check whether **clearDB** is configured or not for your app. This command will show you the URL.

Message us

heroku config | grep CLEARDB_DATABASE_URL

Now it's time to set some environment variables. Follow me gyus!!!

1. Generate a APP_KEY

php artisan key:generate --show
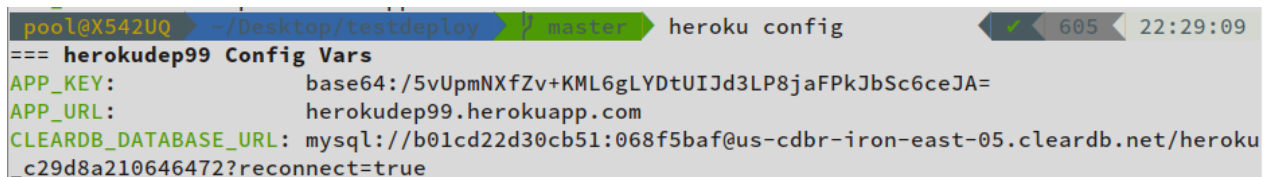
2. Set APP_KEY

heroku config:set APP_KEY=above_key

3. Set URL for app

heroku config:set APP_URL=herokudep99.herokuapp.com

Finally you can check your configs with this command..Check carefully all the variables are there...

heroku config

```
pool@X542UQ    ~/Desktop/testdeploy   master    heroku config          ✔    605    22:29:09
=== herokudep99 Config Vars
APP_KEY:              base64:/5vUpmNXfZv+KML6gLYDtUIJd3LP8jaFPkJbSc6ceJA=
APP_URL:              herokudep99.herokuapp.com
CLEARDB_DATABASE_URL: mysql://b01cd22d30cb51:068f5baf@us-cdbr-iron-east-05.cleardb.net/heroku
_c29d8a210646472?reconnect=true
```

Now we are ready to deploy guys!

We can do it like we do in GIT. we have to use Heroku CLI with GIT commands. Use the following commands.

git push heroku master

The below command will migrate the migrations and feed data into DB.

heroku run php artisan migrate --seed

Message us

```
pool@X542UQ    ~/Desktop/testdeploy     master    heroku run php artisan migrate --seed
Running php artisan migrate --seed on ● testdep99... up, run.2612 (Free)
****************************************
*       Application In Production!      *
****************************************

 Do you really wish to run this command? (yes/no) [no]:
 > yes

Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table (0.04 seconds)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table (0.03 seconds)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated:  2019_08_19_000000_create_failed_jobs_table (0.02 seconds)
Migrating: 2019_11_22_155242_create_products_table
Migrated:  2019_11_22_155242_create_products_table (0.02 seconds)
Seeding: ProductsTableSeeder
Seeded:  ProductsTableSeeder (0.02 seconds)
Database seeding completed successfully.
pool@X542UQ    ~/Desktop/testdeploy     master                          ✓  668   23:10:56
```

Step 6 - Check Database

Select your app in Heroku platform and go into resources tab. Click on **ClearDB MYSQL** addon. Then it will direct you to a new page. There you find a link which is having your DB name(random name). Click on that link. It will direct you to another page. Move into the System Information tab and copy the **username** and **password**.

Open **MYSQL Workbench** software and create a new instance!

NOTE:
Provide HOST name correctly. Use GREP command to see DB URL and from it, extract the host. It should be like the below one starting from us-.

us-cdbr-iron-east-05.cleardb.net

GREP Command:
heroku config | grep CLEARDB_DATABASE_URL

This is my results on workbench..

#	id	product_name	product_description	created_at	updated_at
1	1	MacBook Pro	Apple Laptop	2019-11-22 17:40:53	2019-11-22 17:40:53
2	11	Huawei GR5 2017	Andriod Smart Phone	2019-11-22 17:40:53	2019-11-22 17:40:53
*	NULL	NULL	NULL	NULL	NULL

You got the results guys? If so, you have followed me correctly and nothing to be worried. Ne...

your app deployed on Heoku has been connected to **MYSQL DB** as I promised at the beginning of the article! Now you can work with Laravel to create more **models**, **migrations** and **seeders**. Push them and see again..It will definitely work!!!

Step 7 - More Check On Data

I'm going to modify the project a little bit and check in the changes again. My aim is to configure a view with a **controller** to display the **products array**. Open teminal inside the project and type the command to create a controller..

php artisan make:controller PageController

Let's add a function to the controller to **dump** the products as an array when we go for that view. This function will show the items collection.

```php
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Product;

class PageController extends Controller
{
    public function index(){
        $products = Product::all();
        dd($products);
        return view('welcome')->with(['products' => $products]);
    }
}
```

Now we have to configure the base route to call this function. It's very simple! Add this route to the **routes/web.php** file.

```php
Route::get('/', 'PageController@index');
```

Message us

That's all..Now we can push these changes to Heroku app. Open your terminal and perform these commands.

git add .

git commit -m "add controller"

git push heroku master

heroku run php artisan migrate:refresh

```
pool@X542UQ   ~/Desktop/testdeploy   master   heroku run php artisan migrate:refresh
Running php artisan migrate:refresh on ● testdep99... up, run.9746 (Free)
**********************************************
*      Application In Production!       *
**********************************************

 Do you really wish to run this command? (yes/no) [no]:
 > yes

Rolling back: 2019_11_22_155242_create_products_table
Rolled back:  2019_11_22_155242_create_products_table (0.02 seconds)
Rolling back: 2019_08_19_000000_create_failed_jobs_table
Rolled back:  2019_08_19_000000_create_failed_jobs_table (0.01 seconds)
Rolling back: 2014_10_12_100000_create_password_resets_table
Rolled back:  2014_10_12_100000_create_password_resets_table (0.01 seconds)
Rolling back: 2014_10_12_000000_create_users_table
Rolled back:  2014_10_12_000000_create_users_table (0.01 seconds)
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table (0.04 seconds)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table (0.04 seconds)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated:  2019_08_19_000000_create_failed_jobs_table (0.02 seconds)
Migrating: 2019_11_22_155242_create_products_table
Migrated:  2019_11_22_155242_create_products_table (0.02 seconds)
```

heroku run php artisan db:seed

```
pool@X542UQ   ~/Desktop/testdeploy   master ●   heroku run php artisan db:seed
Running php artisan db:seed on ● testdep99... up, run.1686 (Free)
**********************************************
*      Application In Production!       *
**********************************************

 Do you really wish to run this command? (yes/no) [no]:
 > yes

Seeding: ProductsTableSeeder
Seeded:  ProductsTableSeeder (0.09 seconds)
Database seeding completed successfully.
pool@X542UQ   ~/Desktop/testdeploy   master ●
```

Message us

What will be the result we get when we go and open the appication on browser? Go to **http://herokudep99.herokuapp.com** and check! You must get the this result since we seeded **two products into the table in Heroku**!

```
^ Illuminate\Database\Eloquent\Collection {#186 ▼
  #items: array:2 [▼
    0 => App\Product {#187 ▶}
    1 => App\Product {#188 ▶}
  ]
}
```

<p style="text-align:center">NOW IT'S OVER GUYS!!!</p>

So, this is the end of today's article..I hope you got something interested into your Laravel LIFE! Keep in touch with TechPool to get more articles and more knowledge on web technologies!

GOOD BYE GUYS!

📁 *Laravel, PHP*

(f) (🐦) (𝒫) (G+) (in) (🟢)

You may like these posts

SHOPPING CART

PART 2

Message us

Create a Shopping Cart with Laravel 6 - Part 2

December 29, 2019

Deploy a Laravel application on Heroku with MySQL database

November 22, 2019

Create a Shopping Cart with Laravel 6 - Part 1

November 20, 2019

2 Comments

Raj Sharma   *- December 24, 2019 at 5:04 PM*

Good Post. I like your blog. Thanks for Sharing
PHP Training in Noida

Reply   Delete

TechPool ✔   *- December 25, 2019 at 10:00 AM*

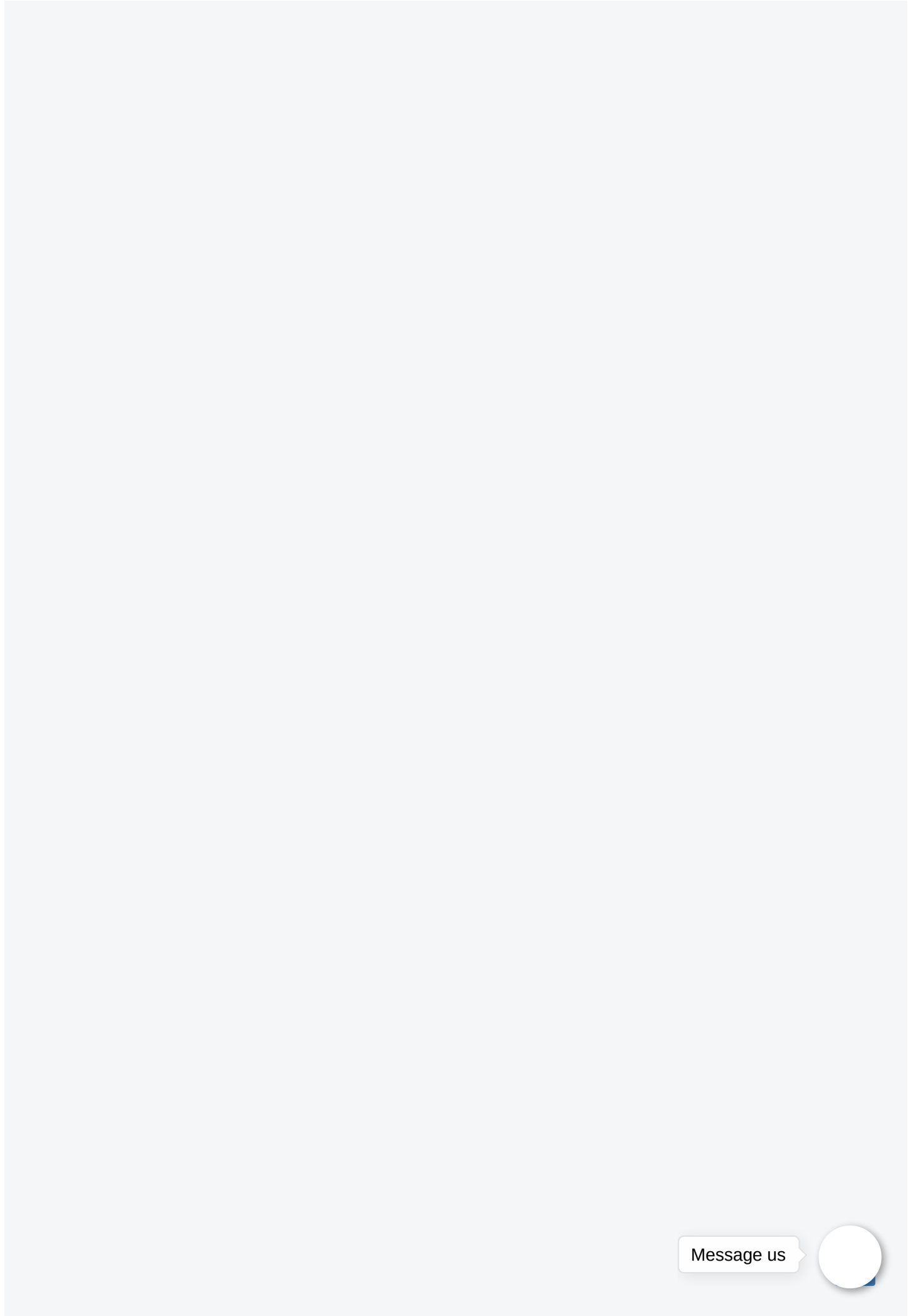Thank you Raj! Keep reading.. :D
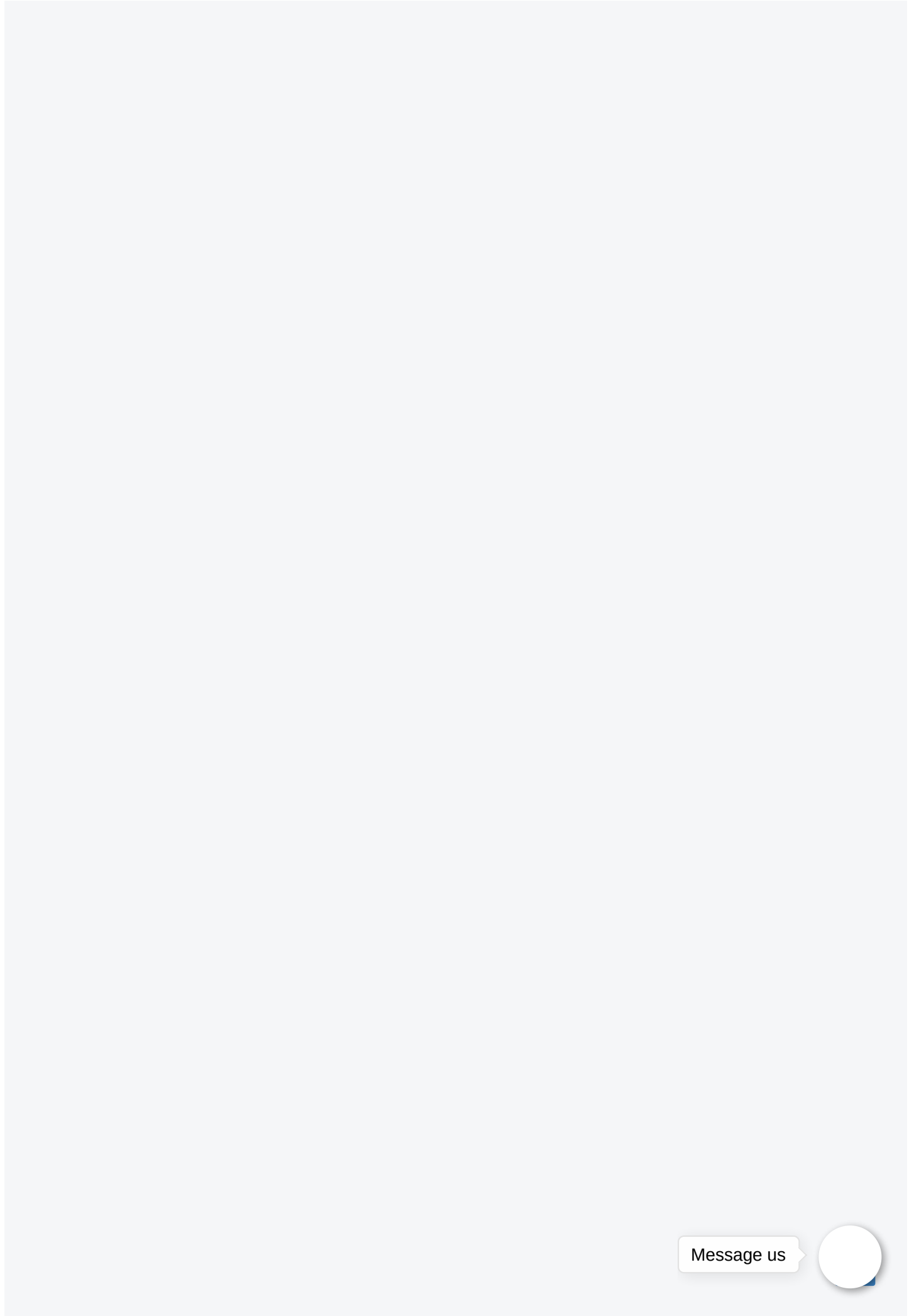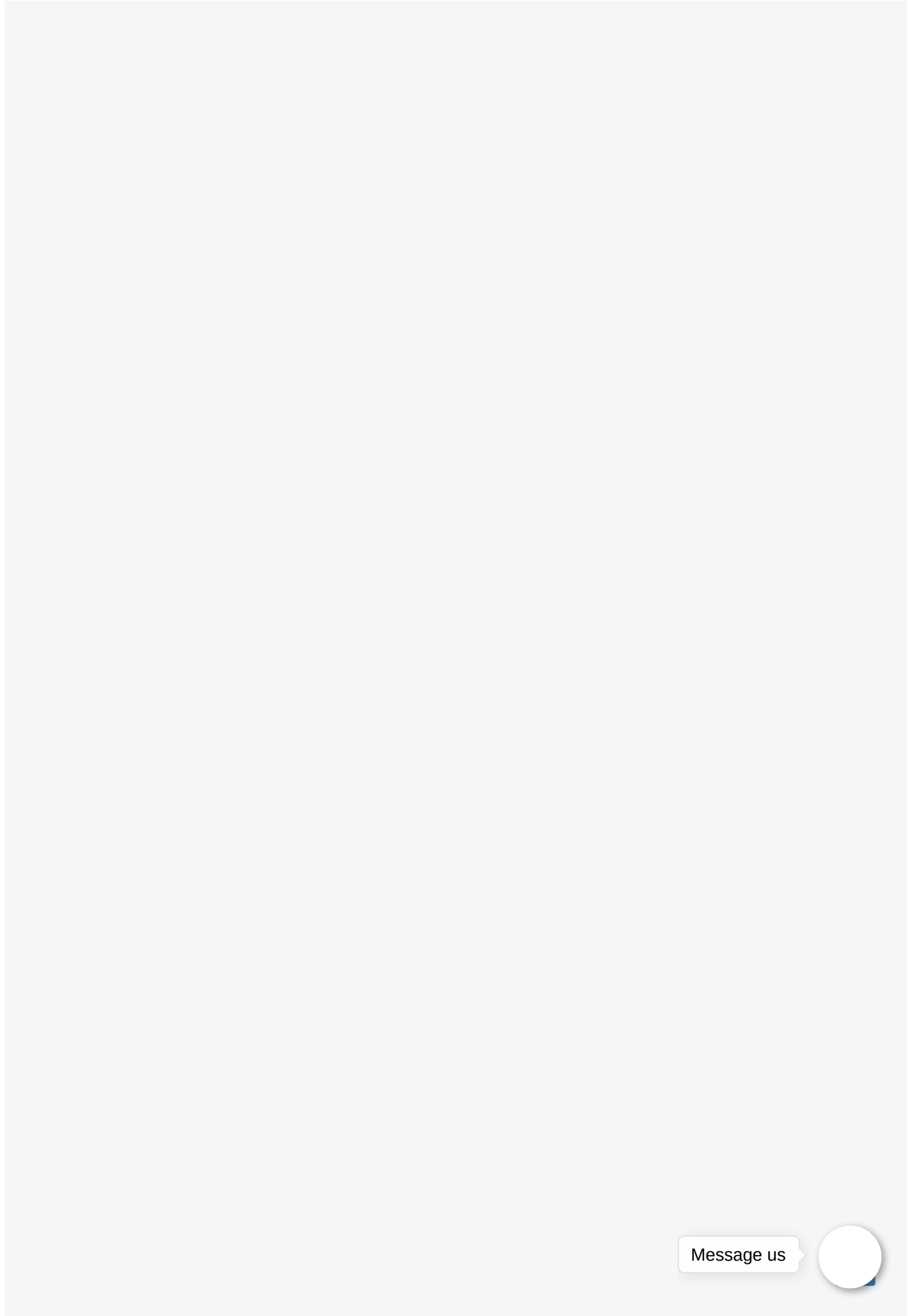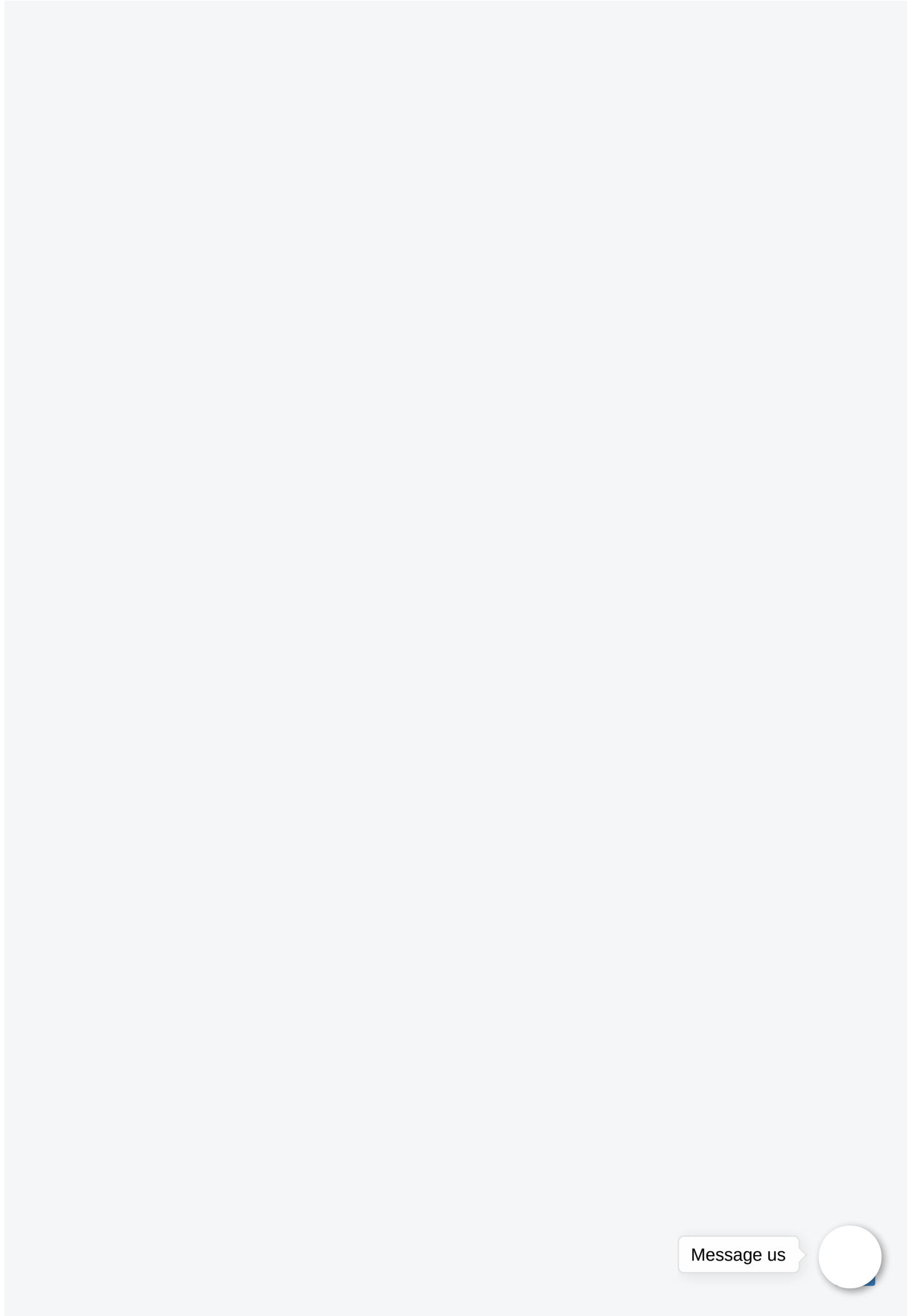
Delete

Enter your comment...

Message us
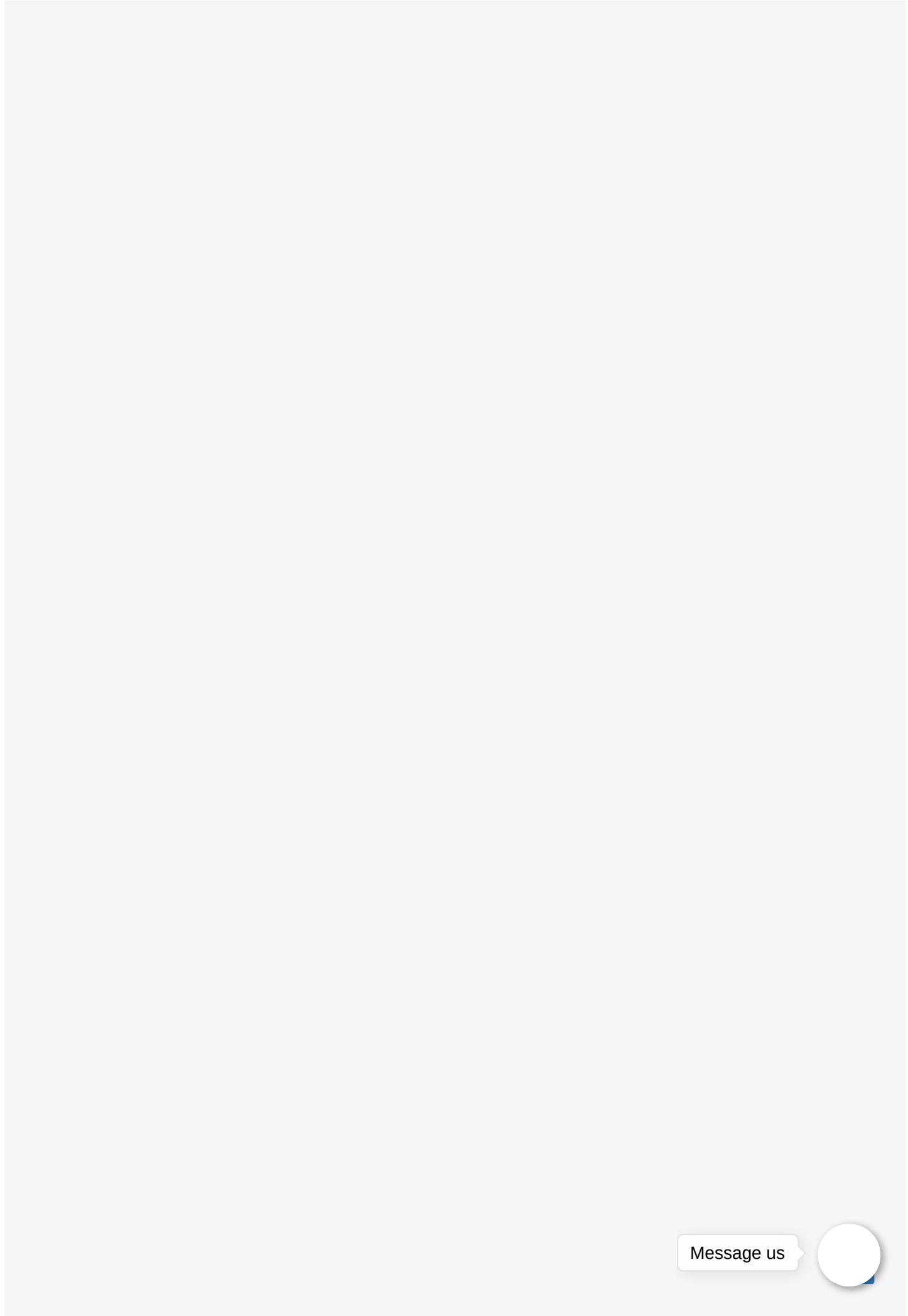
Message us
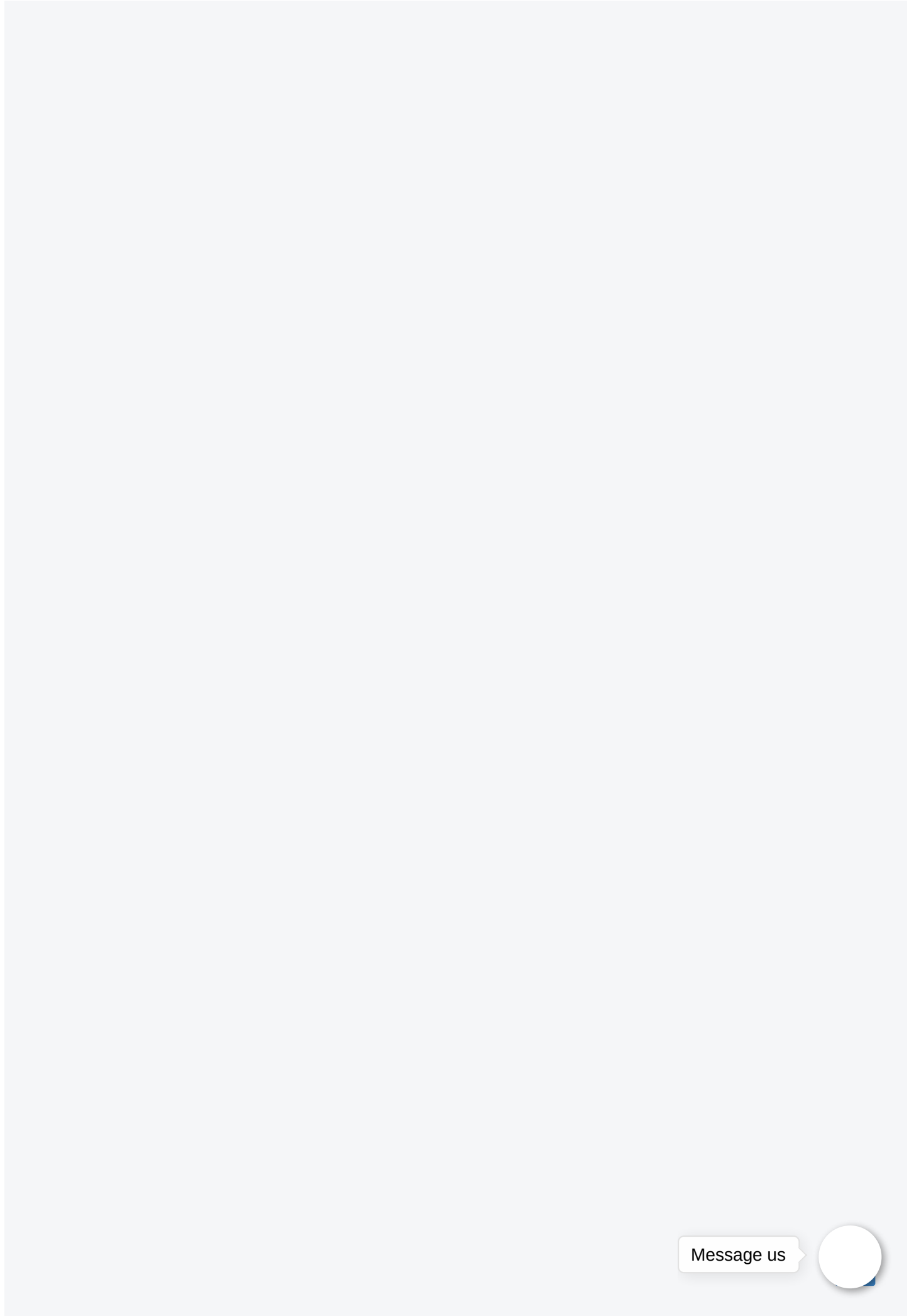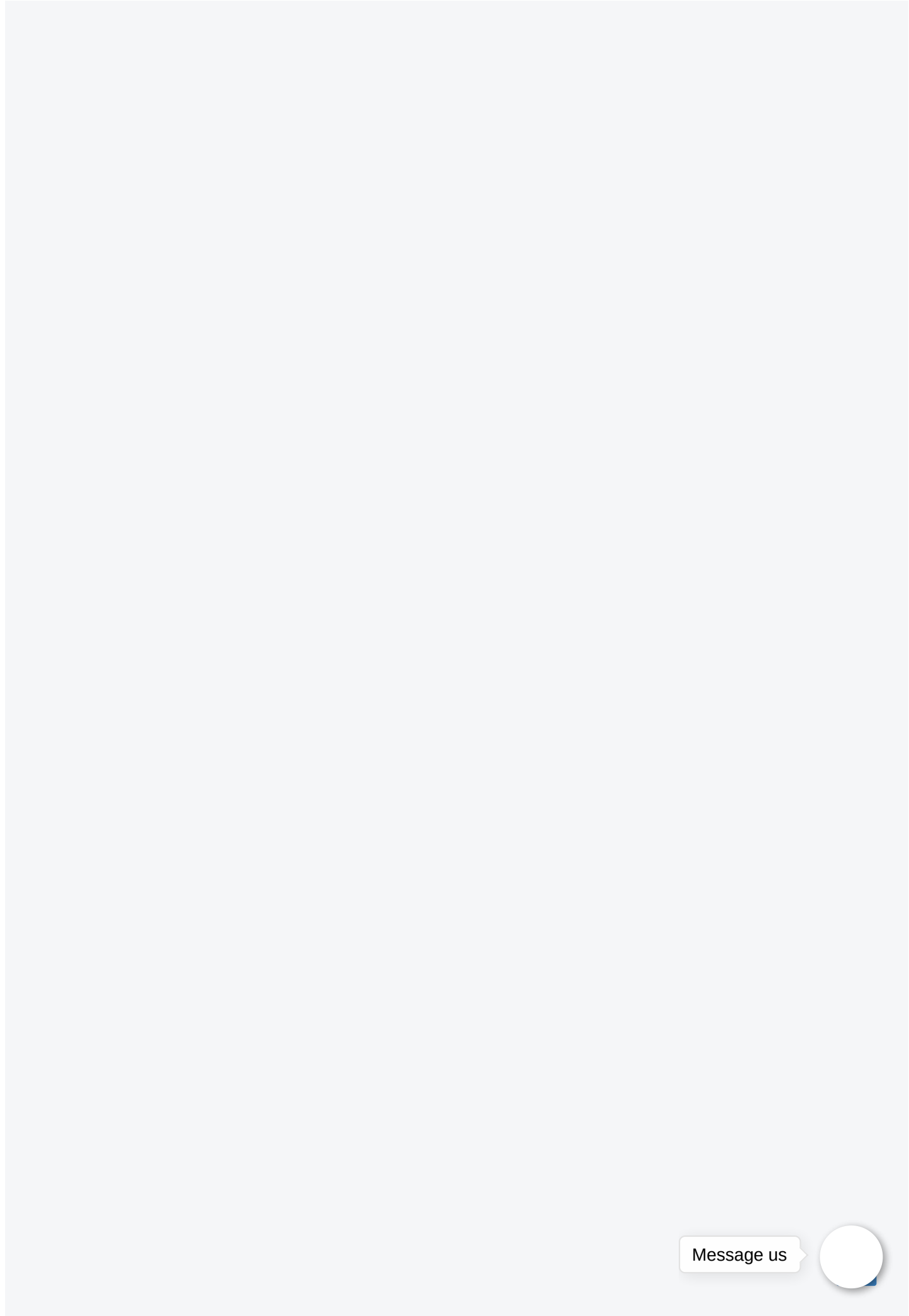
Message us
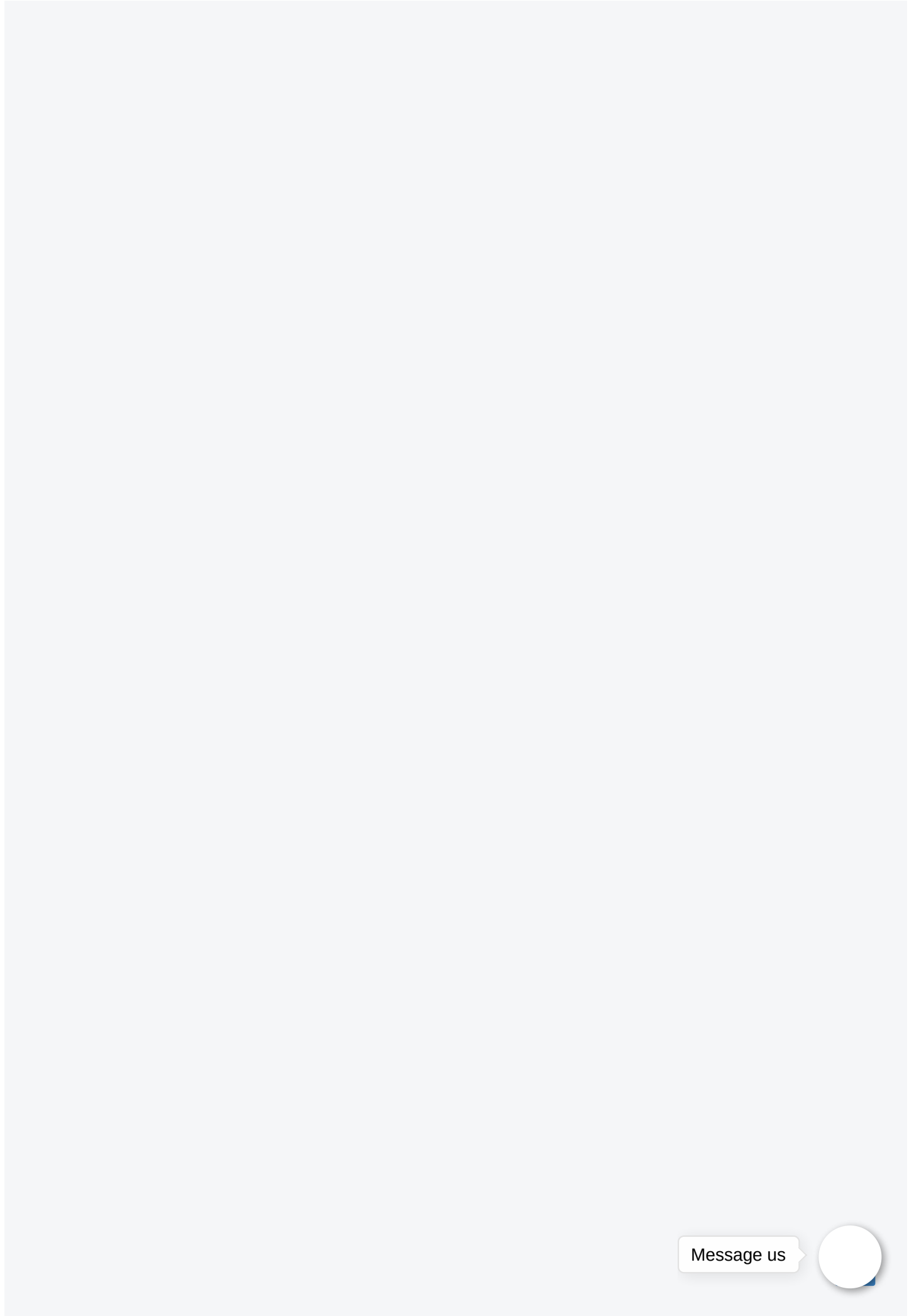
Message us
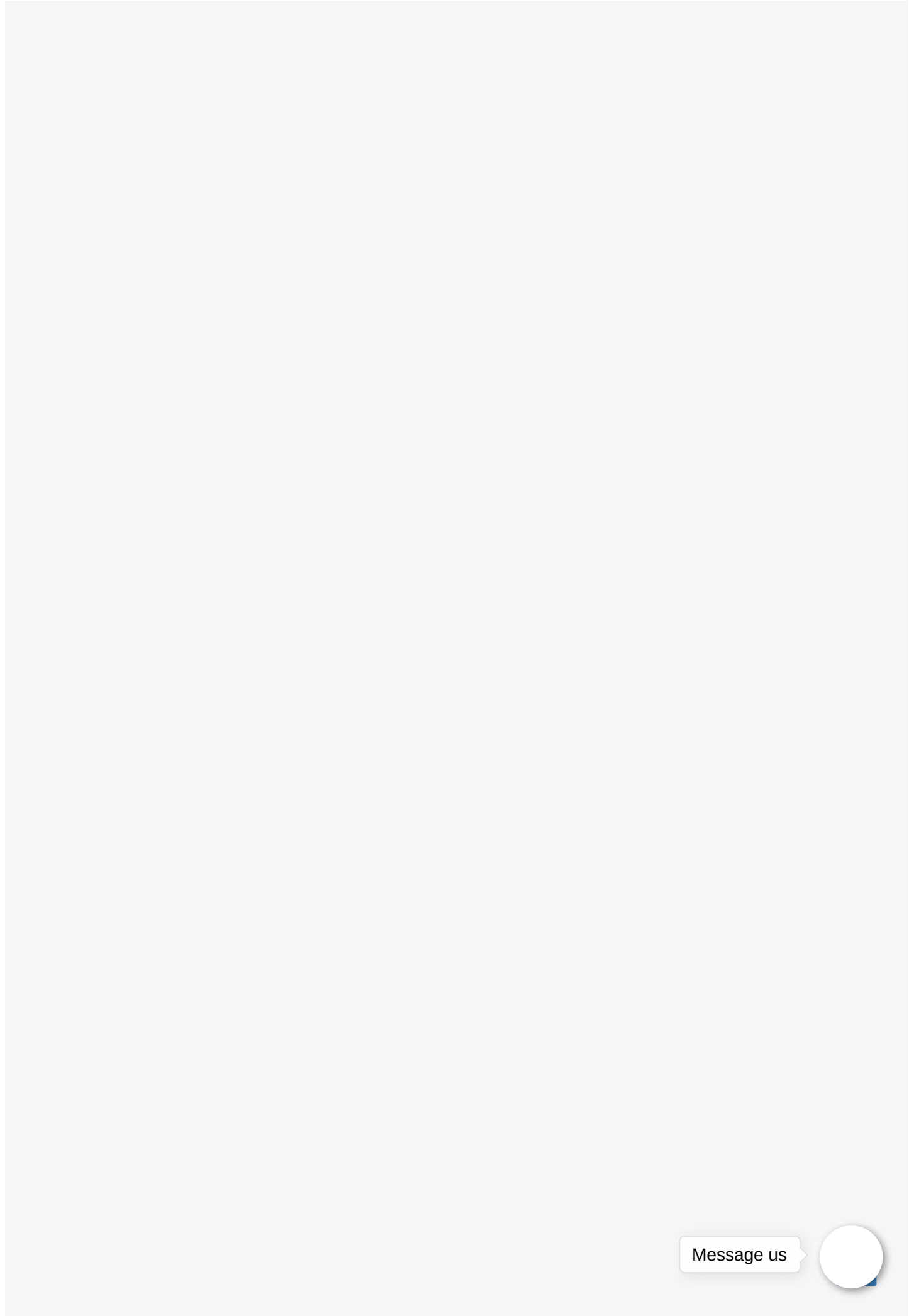
Message us

Message us

Message us

Message us

Message us

Message us

Most Recent

Create a Shopping Cart with Laravel 6 - Part 2

December 29, 2019

Deploy a Laravel application on Heroku with MySQL database

November 22, 2019

Create a Shopping Cart with Laravel 6 - Part 1

November 20, 2019

Follow by Email

Get all latest content delivered straight to your inbox.

Email Address

Subscribe

Most Popular

String Class in Java

October 06, 2019

View data in a Bootstrap modal using Ajax, jQuery and PHP

January 30, 2018

Codeigniter Image Upload

February 02, 2018

Message us