

Simple File Upload in Laravel 5.5

Boban Joksimoski

December 28, 2018

1 Setup Laravel 5.5

Install Laravel using some of the method described here. Name the project as you wish, here we'll name it `fileupload`.

Go into the project folder and install the frontend dependencies.

```
1 npm install
```

Setup the database inside the `.env` file to correspond to an empty database.

If needed, set the default string length for the database inside `app/Providers/AppServiceProvider.php`.

```
1 //File: app/Providers/AppServiceProvider.php
2
3 public function boot()
4 {
5     Schema::defaultStringLength(191);
6 }
```

Next step is to use the built in authentication using the following command.

```
1 php artisan make:auth
```

Migrate the database using the following command.

```
1 php artisan migrate
```

2 Create Files table

We need to create a **File** model and migrations:

```
1 php artisan make:model File -m
```

Define the schema inside the newly created migration file.

```
1 // database/migrations/YYYY_MM_DD_create_files_table.php
2 <?php
3
4 use Illuminate\Support\Facades\Schema;
5 use Illuminate\Database\Schema\Blueprint;
6 use Illuminate\Database\Migrations\Migration;
7
8 class CreateFilesTable extends Migration
9 {
10     /**
11      * Run the migrations.
12      *
13      * @return void
14      */
15     public function up()
16     {
17         Schema::create('files', function (Blueprint $table) {
18             $table->increments('id');
19             $table->integer('user_id')->unsigned()->index();
20             $table->string('title');
21             $table->text('overview');
22             $table->decimal('price', 6, 2);
```

```

23         $table->timestamps();
24         $table->softDeletes();
25         $table->foreign('user_id')
26         ->references('id')
27         ->on('users')
28         ->onDelete('cascade');
29     });
30 }
31
32 /**
33  * Reverse the migrations.
34  *
35  * @return void
36  */
37 public function down()
38 {
39     Schema::dropIfExists('files');
40 }
41 }

```

Now, migrate using the following command.

```

1 php artisan migrate

```

Next step is to build a relationship between **User** and **File** models. Inside the model for file (**File.php**) write the following code.

```

1 // app/File.php
2 <?php
3
4 namespace App;
5
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Database\Eloquent\SoftDeletes;
8
9 class File extends Model
10 {
11     use SoftDeletes;
12
13     protected $fillable = [
14         'title',
15         'overview',
16         'price'
17     ];
18
19     public function user()
20     {
21         return $this->belongsTo(User::class);
22     }
23 }

```

So, here we have stated that every file belongs to a particular **User**. Now, define the Relationship inside **User.php** model.

```

1 // app/User.php
2 <?php
3
4 namespace App;
5
6 use Illuminate\Notifications\Notifiable;
7 use Illuminate\Foundation\Auth\User as Authenticatable;
8
9 class User extends Authenticatable
10 {
11     use Notifiable;
12
13     /**
14      * The attributes that are mass assignable.
15      *
16      * @var array
17      */
18     protected $fillable = [
19         'name', 'email', 'password',
20     ];
21
22     /**
23      * The attributes that should be hidden for arrays.
24      *
25      * @var array
26      */
27     protected $hidden = [
28         'password', 'remember_token',

```

```

29     ];
30
31     public function files()
32     {
33         return $this->hasMany(File::class);
34     }
35 }

```

3 Create a View and Route for uploading files.

Create a FileController using the following command.

```

1  php artisan make:controller FileController

```

Inside `home.blade.php` file, write the following code inside it.

```

1  // File: resources/views/home.blade.php
2
3  @extends('layouts.app')
4
5  @section('content')
6  <div class="container">
7      <div class="row justify-content-center">
8          <div class="col-md-8">
9              <div class="card">
10                 <div class="card-header">File Upload</div>
11                 <div class="card-body">
12                     <form method="POST" action="{{ route('file.upload') }}" aria-label="{{ __('Upload') }}">
13                         {{ csrf_field() }}
14                         <div class="form-group row">
15                             <label for="title" class="col-sm-4 col-form-label text-md-right">{{ __('Title') }}</label>
16                             <div class="col-md-6">
17                                 <input id="title" type="text" class="form-control{{ $errors->has('title') ? ' is-invalid' : '' }}" name="title" value="{{
18                                     old('title') }}" required autofocus />
19                                 @if ($errors->has('title'))
20                                     <span class="invalid-feedback" role="alert">
21                                         <strong>{{ $errors->first('title') }}</strong>
22                                     </span>
23                                 @endif
24                             </div>
25                         </div>
26                         <div class="form-group row">
27                             <label for="overview" class="col-sm-4 col-form-label text-md-right">{{ __('Overview') }}</label>
28                             <div class="col-md-6">
29                                 <textarea id="overview" cols="10" rows="10" class="form-control{{ $errors->has('overview') ? ' is-invalid' : '' }}"
30                                     name="overview" value="{{ old('overview') }}" required autofocus></textarea>
31                                 @if ($errors->has('overview'))
32                                     <span class="invalid-feedback" role="alert">
33                                         <strong>{{ $errors->first('overview') }}</strong>
34                                     </span>
35                                 @endif
36                             </div>
37                         </div>
38                         <div class="form-group row">
39                             <label for="price" class="col-md-4 col-form-label text-md-right">{{ __('Price') }}</label>
40                             <div class="col-md-6">
41                                 <input id="price" type="text" class="form-control{{ $errors->has('price') ? ' is-invalid' : '' }}" name="price" required>
42                                 @if ($errors->has('price'))
43                                     <span class="invalid-feedback" role="alert">
44                                         <strong>{{ $errors->first('price') }}</strong>
45                                     </span>
46                                 @endif
47                             </div>
48                         </div>
49                         <div class="form-group row mb-0">
50                             <div class="col-md-8 offset-md-4">
51                                 <button type="submit" class="btn btn-primary">
52                                     {{ __('Upload') }}
53                                 </button>
54                             </div>
55                         </div>
56                     </form>
57                 </div>
58             </div>
59         </div>
60     </div>
61 @endsection

```

Also, we need to define the Post route inside `routes/web.php` file.

```

1 // File: routes/web.php
2 <?php
3
4 Route::get('/', function () {
5     return view('welcome');
6 });
7
8 Auth::routes();
9
10 Route::get('/home', 'HomeController@index')->name('home');
11
12 Route::post('file/upload', 'FileController@store')->name('file.upload');

```

Start the server and register a new user to show this page.

Laravel
Boban ▼

File Upload

Title

Overview

Price

Upload

Okay, now from the form you can see that we have not used any file form field yet. We will do it in a minute, but this is our primary form setup.

4 Storing the form data

Now, we need to write the `store` function to save the data in the database.

```

1 // app/Http/Controllers/FileController.php
2 <?php
3
4 namespace App\Http\Controllers;
5
6 use Illuminate\Http\Request;
7
8 class FileController extends Controller
9 {
10     public function store(Request $request)
11     {
12         $request->validate([
13             'title' => 'required:max:255',
14             'overview' => 'required',
15             'price' => 'required|numeric'
16         ]);
17     }
18 }

```

```

18     auth()->user()->files()->create([
19         'title' => $request->get('title'),
20         'overview' => $request->get('overview'),
21         'price' => $request->get('price')
22     ]);
23
24     return back()->with('message', 'Your file is submitted Successfully');
25 }
26 }

```

We have also included the validation for our form fields.
Also, we need to update our view to see the flash message.

```

1 // File: resources/views/home.blade.php
2
3 @if(session()->get('message'))
4     <div class="alert alert-success">
5         {{ session()->get('message') }}
6     </div>
7 @endif

```

5 Setup File Upload Relationship

Now, we will create another **model** and **migration** for the uploading files.
Type the following command to generate model and migration.

```

1 php artisan make:model Upload -m

```

Write the following schema inside `create_uploads_table.php`.

```

1 // File: database/migrations/YYYY_MM_DD_create_uploads_table.php
2
3 public function up()
4 {
5     Schema::create('uploads', function (Blueprint $table) {
6         $table->increments('id');
7         $table->integer('user_id')->unsigned()->index();
8         $table->string('filename');
9         $table->softDeletes();
10        $table->timestamps();
11
12        $table->foreign('user_id')
13            ->references('id')
14            ->on('users')
15            ->onDelete('cascade');
16    });
17 }

```

Migrate the database using the following command.

```

1 php artisan migrate

```

Set up the relationships between Upload and the other models. Inside `Upload.php` file write the following code.

```

1 // File: app/Upload.php
2 <?php
3
4 namespace App;
5
6 use Illuminate\Database\Eloquent\Model;
7
8 class Upload extends Model
9 {
10
11     protected $fillable = [
12         'filename'
13     ];
14
15     public function user()
16     {
17         return $this->belongsTo(User::class);
18     }
19 }

```

As stated, Upload belongsTo File and User.

So, now write the following relationship methods inside the User.php model.

```
1 // app/User.php
2 public function files()
3 {
4     return $this->hasMany(File::class);
5 }
6
7 public function uploads()
8 {
9     return $this->hasMany(Upload::class);
10 }
```

Finally, write the following relationships inside the File.php model.

```
1 // app/File.php
2
3 public function user()
4 {
5     return $this->belongsTo(User::class);
6 }
```

6 Setup the Dropzone

Install the **Dropzone** library using the following command.

```
1 npm install dropzone --save
```

Now, include this library inside resources/assets/js/bootstrap.js file. We need to add the code inside that file.

```
1 //File: resources/assets/js/bootstrap.js
2 /**
3  * Dropzone
4  */
5
6 window.Dropzone = require('dropzone');
7 Dropzone.autoDiscover = false;
```

We need to import the Dropzone css files, so inside resources/assets/sass/app.scss file, add the following code.

```
1 // resources/assets/sass/app.scss
2 // Fonts
3 @import url('https://fonts.googleapis.com/css?family=Nunito');
4
5 // Variables
6 @import 'variables';
7
8 // Bootstrap
9 @import 'bootstrap/scss/bootstrap';
10
11 // Dropzone
12 @import 'dropzone/dist/min/dropzone.min.css';
13
14 .dropzone {
15     margin-bottom: 20px;
16     min-height: auto;
17 }
```

Compile the file using the following command.

```
1 npm run dev
```

Use the **Dropzone** library inside the home.blade.php file.

```
1 // resources/views/home.blade.php
2 @extends('layouts.app')
3
4 @section('content')
```

```

5 <div class="container">
6   <div class="row justify-content-center">
7     <div class="col-md-8">
8       @if(session()->get('message'))
9         <div class="alert alert-success">
10           {{ session()->get('message') }}
11         </div>
12       @endif
13       <div class="card">
14         <div class="card-header">File Upload</div>
15         <div class="card-body">
16           <form method="POST" action="{ route('file.upload') }}" aria-label="{ __('Upload') }}">
17             {{csrf_field()}}
18             <div class="form-group row">
19               <label for="title" class="col-sm-4 col-form-label text-md-right">{{ __('File Upload') }}</label>
20               <div class="col-md-6">
21                 <div id="file" class="dropzone"></div>
22               </div>
23             </div>
24             <div class="form-group row">
25               <label for="title" class="col-sm-4 col-form-label text-md-right">{{ __('Title') }}</label>
26               <div class="col-md-6">
27                 <input id="title" type="text" class="form-control{{ $errors->has('title') ? ' is-invalid' : '' }}" name="title" value="{{
28                   old('title') }}" required autofocus />
29                 @if ($errors->has('title'))
30                   <span class="invalid-feedback" role="alert">
31                     <strong>{{ $errors->first('title') }}</strong>
32                   </span>
33                 @endif
34               </div>
35             </div>
36             <div class="form-group row">
37               <label for="overview" class="col-sm-4 col-form-label text-md-right">{{ __('Overview') }}</label>
38               <div class="col-md-6">
39                 <textarea id="overview" cols="10" rows="10" class="form-control{{ $errors->has('overview') ? ' is-invalid' : '' }}"
40                   name="overview" value="{{ old('overview') }}" required autofocus></textarea>
41                 @if ($errors->has('overview'))
42                   <span class="invalid-feedback" role="alert">
43                     <strong>{{ $errors->first('overview') }}</strong>
44                   </span>
45                 @endif
46               </div>
47             </div>
48             <div class="form-group row">
49               <label for="price" class="col-md-4 col-form-label text-md-right">{{ __('Price') }}</label>
50               <div class="col-md-6">
51                 <input id="price" type="text" class="form-control{{ $errors->has('price') ? ' is-invalid' : '' }}" name="price" required>
52                 @if ($errors->has('price'))
53                   <span class="invalid-feedback" role="alert">
54                     <strong>{{ $errors->first('price') }}</strong>
55                   </span>
56                 @endif
57               </div>
58             </div>
59             <div class="form-group row mb-0">
60               <div class="col-md-8 offset-md-4">
61                 <button type="submit" class="btn btn-primary">
62                   {{ __('Upload') }}
63                 </button>
64               </div>
65             </div>
66           </form>
67         </div>
68       </div>
69     </div>
70   </div>
71 @endsection
72
73 @section('scripts')
74   <script>
75     var drop = new Dropzone('#file', {
76       url: "{ route('upload') }"
77     });
78   </script>
79 @endsection

```

Define the **POST** route to upload a file.

```

1 // routes/web.php
2
3 Route::post('upload', 'FileController@upload')->name('upload');

```

Create an upload function inside `FileController.php` file.

```

1 // app/Http/Controllers/FileController.php
2 public function upload()

```

```

3 {
4     return 'x';
5 }

```

Save the file, run the server and go to the /home route of your app

7 Handling File Upload Process

First thing, we need to do is that add the headers while sending a POST request to the server.

```

1 // resources/views/home.blade.php
2
3 @section('scripts')
4     <script>
5         var drop = new Dropzone('#file', {
6             createImageThumbnails: false,
7             addRemoveLinks: true,
8             url: "{{ route('upload') }}",
9             headers: {
10                 'X-CSRF-TOKEN': document.head.querySelector('meta[name="csrf-token"]').content
11             }
12         });
13     </script>
14 @endsection

```

Additionally, we have to change the layout to include the scripts section. Add @yield('scripts') at the end of the file.

```

1 // File: resources/views/layouts/app.blade.php
2 <!DOCTYPE html>
3 <html lang="{{ app()->getLocale() }}">
4 <head>
5     <meta charset="utf-8">
6     <meta http-equiv="X-UA-Compatible" content="IE=edge">
7     <meta name="viewport" content="width=device-width, initial-scale=1">
8
9     <!-- CSRF Token -->
10    <meta name="csrf-token" content="{{ csrf_token() }}">
11
12    <title>{{ config('app.name', 'Laravel') }}</title>
13
14    <!-- Styles -->
15    <link href="{{ asset('css/app.css') }}" rel="stylesheet">
16 </head>
17 <body>
18     <div id="app">
19         <nav class="navbar navbar-default navbar-static-top">
20             <div class="container">
21                 <div class="navbar-header">
22
23                     <!-- Collapsed Hamburger -->
24                     <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#app-navbar-collapse"
25                         ↳ aria-expanded="false">
26                         <span class="sr-only">Toggle Navigation</span>
27                         <span class="icon-bar"></span>
28                         <span class="icon-bar"></span>
29                         <span class="icon-bar"></span>
30                     </button>
31
32                     <!-- Branding Image -->
33                     <a class="navbar-brand" href="{{ url('/') }}">
34                         {{ config('app.name', 'Laravel') }}
35                     </a>
36                 </div>
37
38                 <div class="collapse navbar-collapse" id="app-navbar-collapse">
39                     <!-- Left Side Of Navbar -->
40                     <ul class="nav navbar-nav">
41                         &nbsp;
42                     </ul>
43
44                     <!-- Right Side Of Navbar -->
45                     <ul class="nav navbar-nav navbar-right">
46                         <!-- Authentication Links -->
47                         @guest
48                             <li><a href="{{ route('login') }}">Login</a></li>
49                             <li><a href="{{ route('register') }}">Register</a></li>
50                         @else
51                             <li class="dropdown">
52                                 <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-expanded="false" aria-haspopup="true"
53                                     ↳ v-pre>
54                                     {{ Auth::user()->name }} <span class="caret"></span>

```



```

53         </a>
54
55         <ul class="dropdown-menu">
56             <li>
57                 <a href="{{ route('logout') }}"
58                     onclick="event.preventDefault();
59                         document.getElementById('logout-form').submit();">
60                     Logout
61                 </a>
62
63                 <form id="logout-form" action="{{ route('logout') }}" method="POST" style="display: none;"
64                     {{ csrf_field() }}
65                 </form>
66             </li>
67         </ul>
68     </li>
69 @endguest
70 </ul>
71 </div>
72 </div>
73 </nav>
74
75 @yield('content')
76 </div>
77
78 <!-- Scripts -->
79 <script src="{{ asset('js/app.js') }}"></script>
80
81 @yield('scripts')
82 </body>
83 </html>

```

Update the code for the FileController.php to the following:

```

1  // app/Http/Controllers/FileController.php
2  <?php
3
4  namespace App\Http\Controllers;
5
6  use Illuminate\Http\Request;
7  use Illuminate\Support\Facades\Storage;
8  use App\Upload;
9
10 class FileController extends Controller
11 {
12     public function store(Request $request)
13     {
14         $request->validate([
15             'title' => 'required:max:255',
16             'overview' => 'required',
17             'price' => 'required|numeric'
18         ]);
19
20         auth()->user()->files()->create([
21             'title' => $request->get('title'),
22             'overview' => $request->get('overview'),
23             'price' => $request->get('price')
24         ]);
25
26         return back()->with('message', 'Your file is submitted Successfully');
27     }
28
29     public function upload(Request $request)
30     {
31         $uploadedFile = $request->file('file');
32         $filename = time().$uploadedFile->getClientOriginalName();
33
34         Storage::disk('local')->putFileAs(
35             'files/'.$filename,
36             $uploadedFile,
37             $filename
38         );
39
40         $upload = new Upload;
41         $upload->filename = $filename;
42
43         $upload->user()->associate(auth()->user());
44
45         $upload->save();
46
47         return response()->json([
48             'id' => $upload->id
49         ]);
50     }
51 }

```

Now our simple upload works. The database entries can be seen in the tables uploads and files. The uploaded files are saved to the /Storage/app/files/ folder

(MySQL 5.6.38) laravelads/fileupload/uploads

Select Database: fileupload

Structure Content Relations Triggers Table Info Query

Table History Users Console

Search: id = Q Search Filter

TABLES

- files
- migrations
- password_resets
- uploads**
- users

TABLE INFORMATION

- created: 16/08/18
- engine: InnoDB
- rows: 2
- size: 16.0 KiB
- encoding: utf8mb4
- auto_increment: 3

id	user_id	filename	created_at	updated_at
1	1	Laravel Dropzone File Upload Example.png	2018-08-15 20:13:32	2018-08-15 20:13:32
2	1	1534364556Laravel 5.6 File Upload Example Tutorial.png	2018-08-15 20:22:36	2018-08-15 20:22:36

2 rows in table