
1. Set Up Laravel Project

If you haven't already, create a new Laravel project:

```
composer create-project --prefer-dist laravel/laravel ajax-file-upload
cd ajax-file-upload
```

2. Configure Routes

Edit your `routes/web.php` file:

```
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\FileUploadController;

Route::get('/', function () {
    return view('upload');
});

Route::post('/upload', [FileUploadController::class, 'upload'])->
    name('file.upload');
```

3. Create Controller

Run the command to create a controller:

```
php artisan make:controller FileUploadController
```

Edit `app/Http/Controllers/FileUploadController.php`:

```
namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Validator;

class FileUploadController extends Controller
{
    public function upload(Request $request)
    {
        $validator = Validator::make($request->all(), [
            'file' => 'required|mimes:jpg,jpeg,png,pdf|max:2048'
        ]);
    }
}
```

```

        if ($validator->fails()) {
            return response()->json(['error' => $validator->errors()->first()],
400);
        }

        if ($request->hasFile('file')) {
            $file = $request->file('file');
            $fileName = time().'_'.$file->getClientOriginalName();
            $file->move(public_path('uploads'), $fileName);

            return response()->json(['success' => 'File uploaded successfully!',
'file' => $fileName]);
        }

        return response()->json(['error' => 'File upload failed!'], 500);
    }
}

```

4. Create Blade View

Create a new view file: `resources/views/upload.blade.php`

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>AJAX File Upload</title>
    <meta name="csrf-token" content="{ { csrf_token() } }">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
    <h2>Upload File using AJAX in Laravel</h2>

    <form id="fileUploadForm">
        <input type="file" id="file" name="file">
        <button type="submit">Upload</button>
    </form>

    <div id="message"></div>

    <script>
        $(document).ready(function () {
            $("#fileUploadForm").on("submit", function (e) {
                e.preventDefault();

                let formData = new FormData();
                formData.append("file", $("#file")[0].files[0]);

                $.ajax({

```

```
        url: "{{ route('file.upload') }}",
        type: "POST",
        data: formData,
        processData: false,
        contentType: false,
        headers: {
            'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')
        },
        success: function (response) {
            $("#message").html("<p style='color: green;'>" +
response.success + "</p>");
        },
        error: function (xhr) {
            $("#message").html("<p style='color: red;'>" +
xhr.responseJSON.error + "</p>");
        }
    });
});
});
</script>
</body>
</html>
```

5. Run the Laravel Server

Start your Laravel application:

```
php artisan serve
```

Open <http://127.0.0.1:8000/> in your browser.

6. Testing the AJAX File Upload

- Select a file and click "Upload."
- If the file is successfully uploaded, a success message will appear.
- If there's an error (e.g., invalid file type or size), an error message will be displayed.

7. File Storage

- Uploaded files will be saved in the `public/uploads` directory.
- Make sure the `public/uploads` directory exists. If not, create it manually.

Key Features

- ✓ Uses AJAX to upload files without refreshing the page.
- ✓ Validates file type (`jpg`, `jpeg`, `png`, `pdf`) and size (`max: 2MB`).
- ✓ Displays success and error messages dynamically.