

# File Upload Process in Laravel Framework

We can easily integrate file upload process in Laravel as follows:

1. Add following codes to create.blade.php file:

```
1 <div class="box box-primary">
2   <div class="box-body">
3     <div class="row">
4       {!! Form::open(['route' => 'applications.store', 'files'=>true]) !!}
5       <div class="form-group col-sm-6">
6         {!! Form::label('image', 'Image:') !!}
7         @if(isset($application->image))
8           image')}}" width="125" height="125"/>
9           <input type="hidden" name="image-old" value="{{ $application->image }}">
10          @endif
11          <div class="form-control">
12            {!! Form::file('image') !!}
13          </div>
14          {!! Form::close() !!}
15        </div>
16      </div>
17    </div>
18  </div>
```

2. Add following codes to edit.blade.php file:

```
1 <div class="box box-primary">
2   <div class="box-body">
3     <div class="row">
4       {!! Form::model($application,
5         ['route' => ['applications.update', $application->id], 'method' =>
6         'patch', 'files'=>true]) !!}
7
8       <div class="form-group col-sm-6">
9         {!! Form::label('image', 'Image:') !!}
10        @if(isset($application->image))
11          image')}}" width="125" height="125"/>
12          <input type="hidden" name="image-old" value="{{ $application->image }}">
13          @endif
14          <div class="form-control">
15            {!! Form::file('image') !!}
16          </div>
17          {!! Form::close() !!}
18        </div>
19      </div>
20    </div>
21  </div>
```

3. In **Controller** class:

```
1 public function store(CreateApplicationRequest $request)
2 {
3     $fileUtil = new FileUtil();
4     $filePath = $fileUtil->saveSingleFile($request, "image");
5     $input = $request->all();
6     $input["image"] = $filePath;
7     $application = $this->applicationRepository->create($input);
8     alert()->success('Application saved successfully.');
```

```
9     return redirect(route('applications.index'));
10 }
11
12 public function update($id, UpdateApplicationRequest $request)
13 {
14     $application = $this->applicationRepository->findWithoutFail($id);
15 }
```

```

16     if (empty($application)) {
17         Flash::error('Application not found');
18
19         return redirect(route('applications.index'));
20     }
21     $fileUtil = new FileUtil();
22     $filePath = $fileUtil->saveSingleFile($request, "image");
23
24     if ($filePath == null) $filePath = $request->input("image-old");
25     $input = $request->all();
26     $input["image"] = $filePath;
27
28     $application = $this->applicationRepository->update($input, $id);
29     alert()->success('Application updated successfully.');
```

## FileUtil class:

```

1 use Illuminate\Http\Request;
2
3 class FileUtil
4 {
5     /**
6      * @param Request $request
7      * @param string $inputName
8      * @return string saved file path
9      */
10    public static function saveSingleFile(Request $request, $inputName = "file")
11    {
12        $file = $request->file($inputName);
13        if ($file != null) {
14            $fileName = Storage::disk("uploads")->put("", $file);
15            if ($request->secure()){
16                $filePath = "https://" . $request->getHost() . "/uploads/" . $fileName;
17            }
18            else{
19                $filePath = "http://" . $request->getHost() . "/uploads/" . $fileName;
20            }
21            return $filePath;
22        } else {
23            return null;
24        }
25    }
26 }
27 }
```

## config/filesystem.php file for Storage:

```

1 'disks' => [
2
3     'local' => [
4         'driver' => 'local',
5         'root' => storage_path('app'),
6     ],
7
8     'public' => [
9         'driver' => 'local',
```

```
10         'root' => storage_path('app/public'),
11         'visibility' => 'public',
12     ],
13     'uploads' => [
14         'driver' => 'local',
15         'root' => public_path().'/uploads' ,
16         'visibility' => 'public',
17     ],
18     's3' => [
19         'driver' => 's3',
20         'key' => 'your-key',
21         'secret' => 'your-secret',
22         'region' => 'your-region',
23         'bucket' => 'your-bucket',
24     ],
25 ],
26 ],
27 ],
```