Laravel - Forms

Topics : Laravel
Written on December 25, 2023

In Laravel, working with forms is made easy with the help of the Blade templating engine and Laravel Collective's HTML and Form Builders. Here's a guide on working with forms in Laravel:

1. Creating Forms:

Forms can be created using the `form` method provided by Laravel Collective. Laravel Collective provides a set of HTML and Form helper methods that simplify the process of creating and working with HTML forms.

First, you need to install Laravel Collective:

```
composer require laravelcollective/html
```

After installation, you should add the service provider and facade in the `config/app.php` file:

```
'providers' => [
    // ...
    Collective\Html\HtmlServiceProvider::class,
],

'aliases' => [
    // ...
    'Form' => Collective\Html\FormFacade::class,
    'Html' => Collective\Html\HtmlFacade::class,
],
```

2. Creating a Form:

In your Blade view, you can use the `Form` facade to create forms.

```
{!! Form::open(['url' => '/submit', 'method' => 'post']) !!}
    // Form fields and controls go here
{!! Form::close() !!}
```

3. Form Fields:

Laravel Collective provides methods for various form fields:

```
{!! Form::text('name', null, ['class' => 'form-control']) !!}
{!! Form::password('password', ['class' => 'form-control']) !!}
{!! Form::email('email', null, ['class' => 'form-control']) !!}
{!! Form::select('role', ['admin' => 'Admin', 'user' => 'User'], null,
['class' => 'form-control']) !!}
```

4. CSRF Protection:

Laravel automatically adds CSRF protection to your forms. Use the @csrf directive to include the CSRF token.

```
{!! Form::open(['url' => '/submit', 'method' => 'post']) !!}
    @csrf
    // Form fields and controls go here
{!! Form::close() !!}
```

5. Form Validation:

You can validate form data using Laravel's built-in validation features. The validate method can be called in your controller to validate the incoming request.

```
 public function store(Request $request)
{
    $request->validate([
        'name' => 'required|string|max:255',
        'email' => 'required|email|unique:users',
        'password' => 'required|min:8',
    ]);

    // Process the form data
}
```
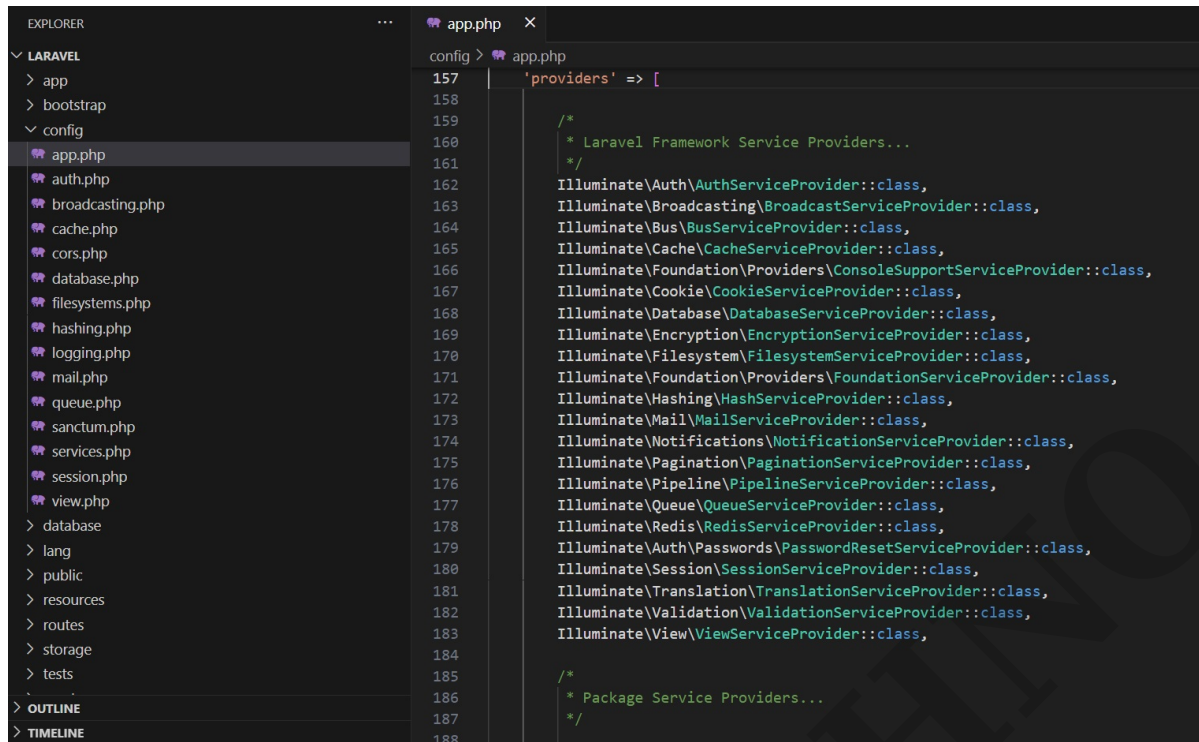
6. Old Input:

After a form submission with validation errors, you can repopulate the form fields with old input using the old method.

```
{!! Form::text('name', old('name'), ['class' => 'form-control']) !!}
```

7. Form Actions:

Set the action attribute of the form to the URL where the form data should be submitted.

```
{!! Form::open(['url' => '/submit', 'method' => 'post']) !!}
    // Form fields and controls go here
    {!! Form::submit('Submit', ['class' => 'btn btn-primary']) !!}
{!! Form::close() !!}
```

```php
      'providers' => [

          /*
          * Laravel Framework Service Providers...
          */
          Illuminate\Auth\AuthServiceProvider::class,
          Illuminate\Broadcasting\BroadcastServiceProvider::class,
          Illuminate\Bus\BusServiceProvider::class,
          Illuminate\Cache\CacheServiceProvider::class,
          Illuminate\Foundation\Providers\ConsoleSupportServiceProvider::class,
          Illuminate\Cookie\CookieServiceProvider::class,
          Illuminate\Database\DatabaseServiceProvider::class,
          Illuminate\Encryption\EncryptionServiceProvider::class,
          Illuminate\Filesystem\FilesystemServiceProvider::class,
          Illuminate\Foundation\Providers\FoundationServiceProvider::class,
          Illuminate\Hashing\HashServiceProvider::class,
          Illuminate\Mail\MailServiceProvider::class,
          Illuminate\Notifications\NotificationServiceProvider::class,
          Illuminate\Pagination\PaginationServiceProvider::class,
          Illuminate\Pipeline\PipelineServiceProvider::class,
          Illuminate\Queue\QueueServiceProvider::class,
          Illuminate\Redis\RedisServiceProvider::class,
          Illuminate\Auth\Passwords\PasswordResetServiceProvider::class,
          Illuminate\Session\SessionServiceProvider::class,
          Illuminate\Translation\TranslationServiceProvider::class,
          Illuminate\Validation\ValidationServiceProvider::class,
          Illuminate\View\ViewServiceProvider::class,

          /*
          * Package Service Providers...
          */
```