

El sistema de autenticación con el que contaba Laravel 5 cambio un poco en la versión 5.1, anteriormente tanto las vistas como las rutas y controladores estaban integrados y listos para usar, desde ahora esto requiere un poco de trabajo manual, pero no hay por que preocuparse, en realidad es un proceso muy sencillo.

Hoy veremos como poner en marcha el sistema de autenticación en tan solo unos pocos minutos, vamos a usar boostrap, para ello, recomiendo seguir este tutorial sobre como <u>Integrar Bootstrap 3 en Laravel</u> 5.1.

## Crear formularios de login y registro

Una vez habiendo integrado bootstrap a nuestro proyecto procedemos a modificar un poco la plantilla base (en mi caso app.blade.php) para mostrar los enlaces en la barra de navegación y agregar el elemento que mostrará los errores de validación de cada uno de los formularios.

```
<!DOCTYPE html>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
   <title>Laravel</title>
    {!! Html::style('assets/css/bootstrap.css') !!}
    <!-- Fonts -->
    <link href='//fonts.googleapis.com/css?family=Roboto:400,300'</pre>
rel='stylesheet' type='text/css'>
    <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
        <script
        <script
    <![endif]-->
```

```
<div class="container-fluid">
      <div class="navbar-header">
          <span class="icon-bar"></span>
      </button>
          <a href="/">Home</a>
                 <a href="{{route('auth/login')}}">Login</a>
                 <a href="{{route('auth/register')}}">Register</a>
                    <a href="#">{{ Auth::user()->name }}</a>
                 <a href="{{route('auth/logout')}}">Logout</a>
             @endif
      </div>
   </div>
</nav>
```

También debemos crear las vistas para login y registro en la aplicación

#### login.blade.php

```
</div>
                                 <label>Password</label>
                                 {!! Form::password('password', ['class'=> 'form-
                             </div>
                             <div class="checkbox">
                                 <label><input name="remember" type="checkbox">
Remember me</label>
                             </div>
                             <div>
                             </div>
                         {!! Form::close() !!}
                     </div>
                </div>
            </div>
        </div>
    </div>
@endsection
```

#### register.blade.php

```
{!! Form::open(['route' => 'auth/register', 'class' =>
                               <label>name</label>
                               {!! Form::input('text', 'name', '', ['class'=> 'form-
                           </div>
                               <label>Email</label>
                           </div>
                               <label>Password</label>
                               {!! Form::password('password', ['class'=> 'form-
27 control']) !!}
                               <label>Password confirmation</label>
                                {!! Form::password('password confirmation',
                           </div>
                           <div>
                                {!! Form::submit('send',['class' => 'btn btn-
38 primary']) !!}
                           </div>
                       {!! Form::close() !!}
                   </div>
               </div>
           </div>
       </div>
   </div>
```

### Rutas de acceso

Agregamos las rutas que apunten a los métodos encargados de manejar cada una de las peticiones.

# Base de datos y migración

Recuerda que el lugar para configurar las credenciales de acceso a tu base de datos es el archivo .env de tu proyecto.

```
DB_HOST=localhost
DB_DATABASE=
DB_USERNAME=
DB_PASSWORD=
```

La migración para la tabla users ya viene definida por defecto por lo cual solo debes ejecutar desde la consola

```
1 $ php artisan migrate
```

# Controladores y redireccionamiento

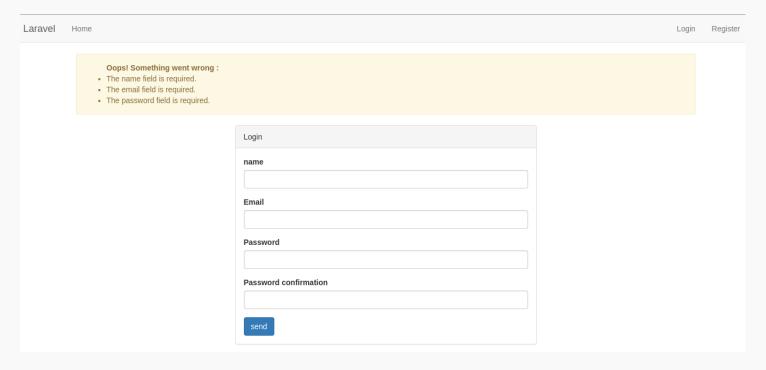
Laravel, si abrimos el archivo app/Controllers/Auth/AuthController.php tan solo encontraremos una función para crear usuarios, para lo demás el controlador hace uso de la clase Illuminate\Foundation\Auth\AuthenticatesAndRegistersUsers.

En este paso no vamos a realizar grandes cambios, tan solo vamos a definir la ruta de redireccionamiento una vez haya ingresado el usuario al sistema, para ello dentro del mismo archivo AuthController.php definimos la siguiente variable.

```
//entre comillas la ruta a la que deseas redireccionar
protected $redirectTo = '/';
```

### Resultado final

Si ingresas a ruta\_de\_tu\_proyecto/public/auth/register vas a ver algo como esto



y de igual forma veremos el formulario de ingreso en ruta\_de\_tu\_proyecto/public/auth/login

Login	
Email	
Password	
□ Remember me	
iogiii	

Con estos sencillos pasos puedes poner a funcionar este modulo en tus aplicaciones de forma muy rápida.

He creado un proyecto en GitHub con un <u>Bolierplate de Laravel 5.1</u> que incluye Bootstrap 3, Laravel debugbar, y Laravel Collective 5.1 , listo para descargar y usar.