

**CENTRO DE ESTUDIOS TECNOLÓGICO
INDUSTRIAL Y DE SERVICIOS NO. 115**

Proyecto 3 CRUD en Laravel



MODULO IV APLICACIONES WEB

**Ing. Verónica Robles
Robles**

01/12/2021



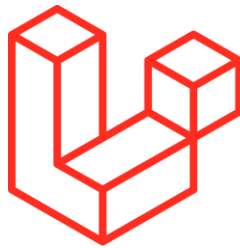
- *Arteaga Hernandez Yoselin*
- *Gutierrez Guerrero Christian Adrián*
- *Macías Rodriguez Yessica Dallan*
- *Martínez Rivera Luis Fernando*



Antes de comenzar...

El proyecto completo con su documentación y libre para descargar se encuentra en el siguiente repositorio:

<https://github.com/TheVampi/ProyectoFinal3Laravel.git>



Laravel

Creación del CRUD desde cero

Antes de comenzar a este punto debe ser mas que obvio que necesitamos tener instalado Laragon, PHP, Laravel y Composer.

Ahora procedemos a crear nuestro proyecto desde cero con Laravel, desde la terminal con el siguiente comando:

```
MAIN - EMPTY
PS C:\laragon\www>
```

Primero nos situamos en la carpeta raíz de Laragon

```
MAIN - EMPTY EMPTY
PS C:\laragon\www> laravel new proyecto_zoologico
```

Y creamos un nuevo proyecto llamado proyecto_zoologico

```
Application key set successfully.

Application ready! Build something amazing.
PS C:\laragon\www>
```

Una vez creado el proyecto empezamos a explorar y modificar el archivo .env desde visual studio code.

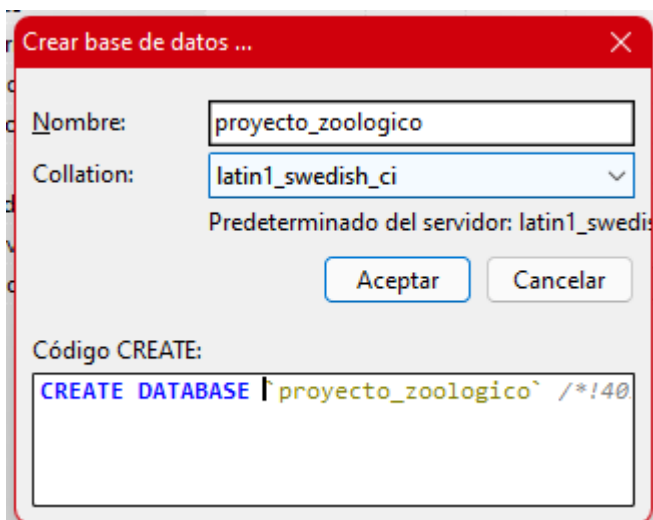
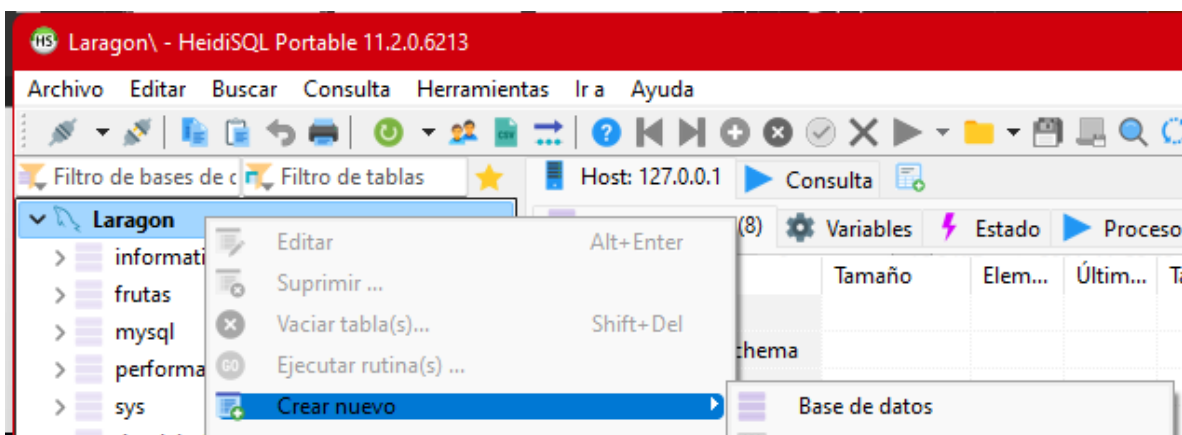
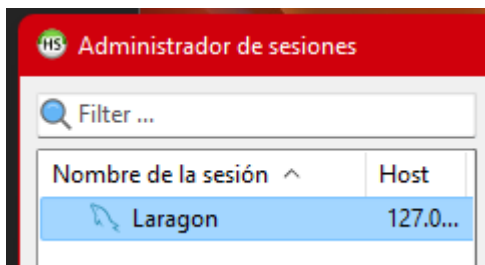
```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=proyecto_zoologico
DB_USERNAME=root
DB_PASSWORD=
```

La función de modificar este código es establecer los parámetros para que el proyecto de Laravel se pueda conectar a la base de datos.

Específicamente en DB_DATABASE se especifica

el nombre de la base de datos a usar en MySQL.

Ahora procedemos a crear una base de datos llamada de igual manera en el archivo .env en Laragon

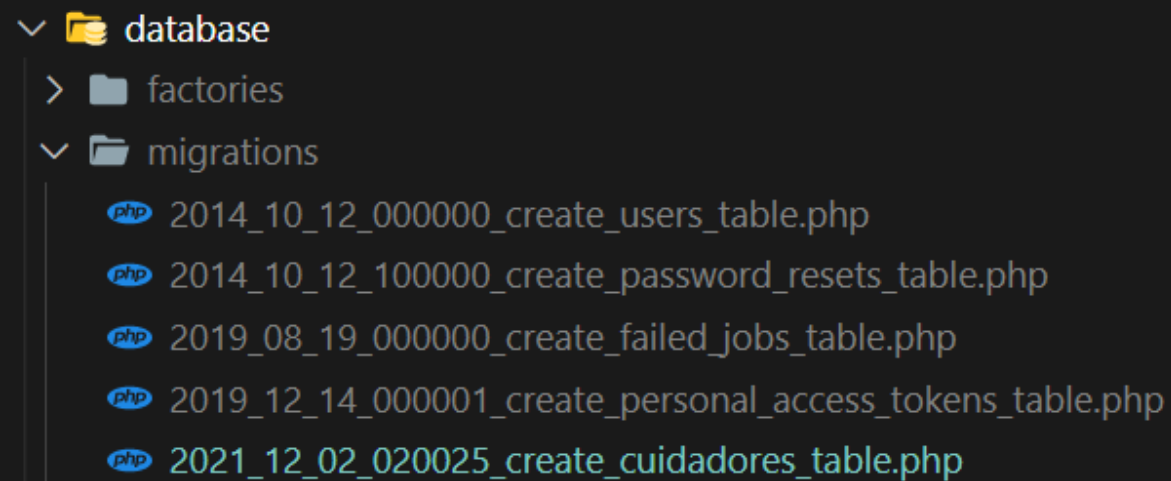


Nos ubicamos en el directorio del proyecto y creamos nuestra migración llamada cuidadores

```
PS C:\laragon\www> cd proyecto_zoologico
PS C:\laragon\www\proyecto_zoologico>
```

```
PS C:\laragon\www\proyecto_zoologico> php artisan make:migration create_cuidadores_table
Created Migration: 2021_12_02_020025_create_cuidadores_table
```

Comprobamos que se haya creado en el siguiente directorio:



```

database
├── factories
└── migrations
    ├── 2014_10_12_000000_create_users_table.php
    ├── 2014_10_12_100000_create_password_resets_table.php
    ├── 2019_08_19_000000_create_failed_jobs_table.php
    ├── 2019_12_14_000001_create_personal_access_tokens_table.php
    └── 2021_12_02_020025_create_cuidadores_table.php
```

Basándonos en la siguiente tabla vamos a hacer la migración:

cuidador		
	Nombre del campo	Tipo de datos
	clave_cuidador	Número
	nombre	Texto corto
	direccion	Texto corto
	telefono	Número
	fecha_ingreso	Fecha/Hora

Modificamos el código de la migración para hacer las columnas correspondientes junto con sus tipos de datos, en este caso se van a utilizar cadenas de texto, enteros grandes y fechas, se pone el nombre de las columnas

```
14     public function up()
15     {
16         Schema::create('cuidadores', function (Blueprint $table)
17         {
18             $table->integer('id');
19             $table->string('nombre', 50);
20             $table->string('direccion', 100);
21             $table->bigInteger('telefono');
22             $table->datetime('fecha_ingreso', $precision = 0);
23             $table->timestamps();
24         });
25     }
```

y en el caso de los strings se pone el numero de caracteres específicos.

Después de guardar el código de la migración procedemos a ejecutar todas las migraciones utilizando los comandos de consola:

Y comprobamos en nuestra base de datos que los campos se crearon correctamente.

Ahora vamos a crear el modelo para el proyecto con la regla de nombrarlo con Mayuscula.

The screenshot shows a terminal window at the top with the command `php artisan migrate` and the output `Migration table created successfully.` Below the terminal is a table structure view for the `proyecto_zoologico` database. The table `cuidadores` has the following columns:

#	Nombre	Tipo de datos	Longitud/Co...	Sin signo	Permitir...	Relle...	Predeterminado	Comen
1	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sin valor predeter...	
2	nombre	VARCHAR	50	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sin valor predeter...	
3	direccion	VARCHAR	100	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sin valor predeter...	
4	telefono	BIGINT	20	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sin valor predeter...	
5	fecha_ingreso	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sin valor predeter...	
6	created_at	TIMESTAMP		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	
7	updated_at	TIMESTAMP		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	

Below the table structure view, the terminal shows the migration process:

```
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (36.33ms)
Migrating: 2021_12_02_020025_create_cuidadores_table
Migrated: 2021_12_02_020025_create_cuidadores_table (12.80ms)
```

```
PS C:\laragon\www\proyecto_zoologico> php artisan make:model Cuidador
Model created successfully.
```

Dentro del modelo ponemos una línea de código que nos va a servir para que los campos especificados sean de manera obligatoria ingresarlos.

```
8 class Cuidador extends Model
9 {
10     use HasFactory;
11
12     //Necesario para la operacion de creacion de registro puesto que indica que campos son requeridos
13     protected $fillable = ['nombre', 'direccion', 'telefono', 'fecha_ingreso'];
14 }
15 |
```

Ahora creamos el controlador con ayuda del siguiente comando:

```
PS C:\laragon\www\proyecto_zoologico> php artisan make:controller CuidadorController --resource
Controller created successfully.
```

Dentro del controlador agregamos líneas de código para hacer funcional el index, el create y el show.

```
public function index()
{
    //
    $cuidadores = Cuidador::all() -> sortBy('nombre');
    return view('cuidador.index', compact('cuidadores'));
}
```

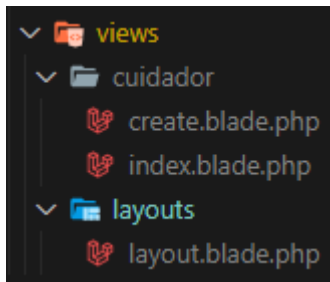
```
public function create()
{
    //
    return view('cuidador.create');
}
```

```
public function store(Request $request)
{
    //
    $this->validate($request, [ 'nombre'=>'required', 'direccion'=>'required', 'telefono'=>'required', 'fecha_ingreso'=>'required']);
    Cuidador::create($request->all());
    return redirect()->route('cuidador.index')->with('success','Registro creado satisfactoriamente');
}
```

Ahora pasamos al; archivo de routes para modificar las rutas a las que vamos acceder en la página.

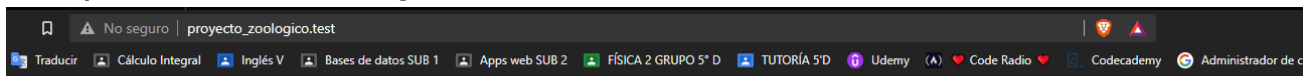
```
15 Route::get('/', 'App\Http\Controllers\CuidadorController@index') -> name('cuidador.index');
16 Route::get('/cuidador/create', 'App\Http\Controllers\CuidadorController@create') -> name('cuidador.create');
17 Route::post('/cuidador/', 'App\Http\Controllers\CuidadorController@store') -> name('cuidador.store');
18
```

Después creamos estos archivos para las views:



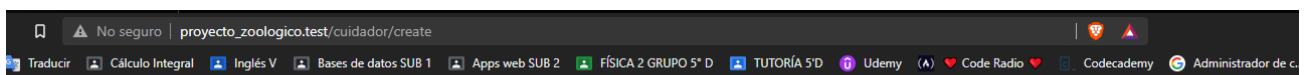
En create.blade.php y en el archivo index.blade.php se hicieron multiples cambios para que fuera funcional, tanto como código, variables y elementos. Todo diseñado para hacer funcional y adaptarlo a la tabla.

Capturas de ejecución:



Mis cuidadores uwu [Agregar cuidador](#)

Nombre	Direccion	Telefono	Fecha de Ingreso	Editar	Eliminar
Capitan Price	Downtown Verdansk	4611234567	2021-12-01 00:00:00		
Daxter	Las Arboledas	4618451682	2021-08-25 00:00:00		
El Progrepapu	Blimasakti Strip	4612602067	2021-12-13 00:00:00		



Nuevo cuidador

[Guardar](#)

[Atrás](#)

Vamos a hacer una prueba para comprobar que funciona la inserción de datos.

Nombre del cuidador: Chika Fujiwara

Dirección: Celayansk

Número de contacto: 4612602067

Fecha de ingreso: 02/12/2021

Nuevo cuidador

Chika Fujiwara

Celayansk

4612602067

02/12/2021

Guardar

Atrás

Damos click en guardar y comprobamos que se agregó exitosamente a nuestra base de datos:

Mis cuidadores uwu [Agregar cuidador](#)

Nombre	Dirección	Teléfono	Fecha de ingreso	Editar	Eliminar
Capitan Price	Downtown Verdansk	4611234567	2021-12-01 00:00:00		
Chika Fujiwara	Celayansk	4612602067	2021-12-02 00:00:00		
Daxter	Las Arboledas	4618451682	2021-08-25 00:00:00		
El Progrepapu	Bimasakti Strip	4612602067	2021-12-13 00:00:00		

Vista en la Base de datos:

Laragon\proyecto_zoologico\cuidadores - HeidiSQL Portable 11.2.0.6213

Host: 127.0.0.1 Base de datos: proyecto_zoologico Tabla: cuidadores Datos Consulta

proyecto_zoologico.cuidadores: 4 filas en total (aproximadamente)

id	nombre	direccion	telefono	fecha_ingreso	created_at	updated_at
1	Capitan Price	Downtown Verdansk	4,611,234,567	2021-12-01 00:00:00	2021-12-02 04:26:22	2021-12-02 04:26:22
2	El Progrepapu	Bimasakti Strip	4,612,602,067	2021-12-13 00:00:00	2021-12-02 04:43:36	2021-12-02 04:43:36
3	Daxter	Las Arboledas	4,618,451,682	2021-08-25 00:00:00	2021-12-02 04:47:21	2021-12-02 04:47:21
4	Chika Fujiwara	Celayansk	4,612,602,067	2021-12-02 00:00:00	2021-12-02 06:28:02	2021-12-02 06:28:02

Host: 127.0.0.1 Base de datos: proyecto_zoologico Tabla: cuidadores Datos Consulta

proyecto_zoologico.cuidadores: 4 filas en total (aproximadamente)

id	nombre	direccion	telefono	fecha_ingreso	created_at	updated_at
1	Capitan Price	Downtown Verdansk	4,611,234,567	2021-12-01 00:00:00	2021-12-02 04:26:22	2021-12-02 04:26:22
2	El Progrepapu	Bimasakti Strip	4,612,602,067	2021-12-13 00:00:00	2021-12-02 04:43:36	2021-12-02 04:43:36
3	Daxter	Las Arboledas	4,618,451,682	2021-08-25 00:00:00	2021-12-02 04:47:21	2021-12-02 04:47:21
4	Chika Fujiwara	Celayansk	4,612,602,067	2021-12-02 00:00:00	2021-12-02 06:28:02	2021-12-02 06:28:02

Código completo de CuidadorController.php

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Cuidador;

class CuidadorController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        //
        $cuidadores = Cuidador::all() -> sortBy('nombre');
        return view('cuidador.index', compact('cuidadores'));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        //
        return view('cuidador.create');
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param \Illuminate\Http\Request $request
     * @return \Illuminate\Http\Response
     */
    public function store(Request $request)
```

```

{
    //
    $this->validate($request, [ 'nombre'=>'required',
'direccion'=>'required', 'telefono'=>'required',
'fecha_ingreso'=>'required']);
    Cuidador::create($request->all());
    return redirect()->route('cuidador.index')->with('success','Registro
creado satisfactoriamente');
}

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function show($id)
{
    //
}

/**
 * Show the form for editing the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function edit($id)
{
    //
}

/**
 * Update the specified resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function update(Request $request, $id)
{
    //
}

/**

```

```

        * Remove the specified resource from storage.
        *
        * @param int $id
        * @return \Illuminate\Http\Response
        */
    public function destroy($id)
    {
        //
    }
}

```

Código completo del modelo Cuidador.php

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Cuidador extends Model
{
    use HasFactory;

    //Necesario para la operacion de creacion de registro puesto que indica
    //que campos son requeridos
    protected $fillable = ['nombre', 'direccion', 'telefono',
    'fecha_ingreso'];
}

```

Código completo de la migración

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateCuidadoresTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('cuidadores', function (Blueprint $table) {
            $table->increments('id');
            $table->string('nombre', 50);
            $table->string('direccion', 100);
            $table->bigInteger('telefono');
            $table->datetime('fecha_ingreso', $precision = 0);
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('cuidadores');
    }
}
```

Código completo de las rutas

```
<?php

use Illuminate\Support\Facades\Route;

/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Route::get('/', 'App\Http\Controllers\CuidadorController@index') ->
name('cuidador.index');
Route::get('/cuidador/create',
'App\Http\Controllers\CuidadorController@create') ->
name('cuidador.create');
Route::post('/cuidador/', 'App\Http\Controllers\CuidadorController@store') -
> name('cuidador.store');
```

Código completo create.blade.php

```
@extends('layouts.layout')
@section('content')
<div class="row">
    <section class="content">
        <div class="col-md-8 col-md-offset-2">
            @if (count($errors) > 0)
                <div class="alert alert-danger">
                    <strong>Error!</strong> Revise los campos
obligatorios.<br><br>
                    <ul>
                        @foreach ($errors->all() as $error)
                            <li>{{ $error }}</li>
                        @endforeach
                    </ul>
                </div>
            @endif
            @if(Session::has('success'))
                <div class="alert alert-info">
                    {{Session::get('success')}}
                </div>
            @endif

            <div class="panel panel-default">
                <div class="panel-heading">
                    <h3 class="panel-title">Nuevo cuidador</h3>
                </div>
                <div class="panel-body">
                    <div class="table-container">
                        <form method="POST" action="{{
route('cuidador.store') }}" role="form">
                            {{ csrf_field() }}
                            <div class="row">
                                <div class="col-xs-6 col-sm-6 col-md-6">
                                    <div class="form-group">
                                        <input type="text" name="nombre"
id="nombre" class="form-control input-sm" placeholder="Nombre del cuidador">
                                    </div>
                                </div>
                                <div class="col-xs-6 col-sm-6 col-md-6">
                                    <div class="form-group">
                                        <input type="text" name="direccion"
id="direccion" class="form-control input-sm" placeholder="Direccion">
                                    </div>
                                </div>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </section>
</div>
</div>
</div>
</div>
```

```

        </div>
    </div>

    <div class="row">

        <div class="col-xs-6 col-sm-6 col-md-6">
            <div class="form-group">
                <input type="text" name="telefono"
id="telefono" class="form-control input-sm" placeholder="Telefono de
contacto">

            </div>
        </div>
        <div class="col-xs-6 col-sm-6 col-md-6">
            <div class="form-group">
                <input type="date"
name="fecha_ingreso" id="fecha_ingreso" class="form-control input-sm"
placeholder="Fecha de Ingreso">

            </div>
        </div>
    </div>
    <div class="row">

        <div class="col-xs-12 col-sm-12 col-md-12">
            <input type="submit" value="Guardar"
class="btn btn-success btn-block">
            <a href="{{ route('cuidador.index') }}"
class="btn btn-info btn-block" >Atrás</a>
        </div>

    </div>
</form>
</div>
</div>

</div>
</div>
</section>
@endsection

```


Código completo index.blade.php

```
@extends('layouts.layout')
@section('content')
<div class="row">
    <section class="content">
        <div class="col-md-8 col-md-offset-2">
            <div class="panel panel-default">
                <div class="panel-body">
                    <div class="pull-left">
                        <h3>Mis cuidadores uwu</h3>
                    </div>
                    <div class="pull-right">
                        <div class="btn-group">
                            <a href="{ route('cuidador.create') }}"
class="btn btn-info">Agregar cuidador</a>
                        </div>
                    </div>
                    <div class="table-container">
                        <table id="mytable" class="table table-bordered
table-striped">

                            <thead>
                                <th>Nombre</th>
                                <th>Direccion</th>
                                <th>Telefono</th>
                                <th>Fecha de Ingreso</th>
                                <th>Editar</th>
                                <th>Eliminar</th>
                            </thead>
                            <tbody>
                                @if($cuidadores->count())
                                @foreach($cuidadores as $cuidador)
                                    <tr>
                                        <td>{{ $cuidador->nombre }}</td>
                                        <td>{{ $cuidador->direccion }}</td>
                                        <td>{{ $cuidador->telefono }}</td>
                                        <td>{{ $cuidador->fecha_ingreso }}</td>

                                        </tr>
                                @endforeach
                                @else
                                    <tr>
                                        <td colspan="8">No hay registro !!</td>
                                    </tr>
                                @endif
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
    </section>
</div>
</div>
</div>
```

```
        </tbody>
      </table>
    </div>
  </div>
</div>
</section>

@endsection
```

Código completo layout.blade.php

```
<!DOCTYPE html>
<html lang="es">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0" ,
user-scalable="yes">
    <link
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css"
rel="stylesheet">
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></
script>
</head>

<body>
    <div class="container-fluid" style="margin-top: 100px">

        @yield('content')
    </div>
    <style type="text/css">
        .table {
            border-top: 2px solid #ccc;
        }
    </style>
</body>

</html>
```