Authentification Simple LARAVEL

composer create-project laravel/laravel auth

cd auth

php artisan serve

.env =>db=authracine

supprimer

auth\app\Models\User.php

\database\migrations\2014\_10\_12\_000000\_create\_users\_table.php

//Nb:ici 'login' et 'resgistration' sont des fonction

------------------------------------------------------------------------------

php artisan make:controller CustomAuthController

app>http>controllers>CustomAuthController.php

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\User; //333333333333333333333333

use Hash; //4444444444444

use Session;

class CustomAuthController extends Controller

{

public function login(){ //11111111111111111111111111111

return view("auth.login");

}

public function registration(){ //111111111111111111111111111111

return view("auth.registration");

}

public function registerUser(Request $request){ //222222222222222222222222222222

$request ->validate([ //echo 'value posted';

'name'=>'required',

'email'=>'required|email|unique:users', //tb

'password'=>'required|min:5|max:12'

]);

$user = new User(); //333333333333333333

$user ->name = $request->name;

$user ->email =$request->email;

$user ->password = Hash::make($request ->password) ; //444444444444444

$res =$user ->save();

if($res){

return back()->with('success', 'you have registered successfuly');

}

else{

return back()->with('fail', 'you have not registered ');

}

}

public function loginUser(Request $request){ // 555555555555555555555555

$request ->validate([

'email'=>'required|email',

'password'=>'required|min:5|max:12'

]);

$user = User::where('email','=',$request->email)->first();

if ($user){

if(Hash::check($request->password, $user->password)){

$request->session()->put('loginId',$user->id);

return redirect('dashboard');

}else{

return back()->with('fail', 'passwor not matches ');

}

}else {

return back()->with('fail', 'this email is not registered ');

}

}

public function dashboard(){ //66666666666666666666666666666666666666

$data =array();

if (Session::has('loginId')){

$data = User::where('id','=',Session::get('loginId'))->first();

}

return view('dashboard',compact('data'));

}

public function logout(){ //77777777777777777777777777

if (Session::has('loginId')){

Session::pull('loginId');

return redirect('login');

}

}

}

------------------------------------------------------------------------------

routes>web.php

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\CustomAuthController; //11111111111111111111111111111111111111

Route::get('/', function () {

return view('welcome');

});

Route::get('/login',[CustomAuthController::class,'login'])->middleware('alreadyLoggedIn'); //Route::get('/login',[CustomAuthController::class,'login']); // 111111111111111111

Route::get('/registration',[CustomAuthController::class,'registration'])->middleware('alreadyLoggedIn'); //Route::get('/registration',[CustomAuthController::class,'registration']); // 11111111111

Route::post('/register-user',[CustomAuthController::class,'registerUser'])->name('register-user'); //222222222222222222222222222222

Route::post('/login-user',[CustomAuthController::class,'loginUser'])->name('login-user'); //4444444444444444444444

Route::get('/dashboard',[CustomAuthController::class,'dashboard'])->middleware('isLoggedIn'); //Route::get('/dashboard',[CustomAuthController::class,'dashboard']);

Route::get('/logout',[CustomAuthController::class,'logout']);

------------------------------------------------------------------------------

resources>views>auth (nv)+>login.blade.php (nv)

registration.blade.php (nv)

------------------------------------------------------------------------------

login.blade.php

<center><H1>Login...</H1></center>

<CENTER>

<form action="{{route('login-user')}}" method="post"> //44444444444444

@if (Session::has('success'))

{{Session::get('success')}}

@endif

@if (Session::has('fail'))

{{Session::get('fail')}}

@endif

<br>

@csrf

email <input type="text" name="email" value=""><span>@error ('email') {{$message}} @enderror</span> <br>

Password <input type="password" name="password" value=""><span>@error ('password') {{$message}} @enderror</span> <br>

<button> Login</button><br>

<a href="registration">new user</a>

</form>

</CENTER>

------------------------------------------------------------------------------

registration.blade.php

<center><H1>Registration...</H1></center>

<CENTER>

<form action="{{route('register-user')}}" method="post"> //222222222222222222222

@if (Session::has('success')) //3333333333333

{{Session::get('success')}}

@endif

@if (Session::has('fail'))

{{Session::get('fail')}}

@endif

<br>

@csrf

full name <input type="text" name="name" value="{{old('name')}}"> <span>@error ('name') {{$message}} @enderror</span> <br>

email <input type="text" name="email" value="{{old('email')}}"> <span>@error ('email') {{$message}} @enderror</span> <br>

Password <input type="password" name="password" value=""><span>@error ('password') {{$message}} @enderror</span> <br>

<button> ENREGISTRER</button><br>

<a href="login">login here</a>

</form>

</CENTER>

------------------------------------------------------------------------------

php artisan make:model User -m

------------------------------------------------------------------------------

database>migration>2021\_...create\_users\_table.php

public function up(): void

{

Schema::create('users', function (Blueprint $table) {

$table->id();

$table->string('name');

$table->string('email')->unique();

$table->string('password');

$table->timestamps();

});

}

------------------------------------------------------------------------------

php artisan migrate

------------------------------------------------------------------------------

resources>views>dashboard.blade.php

<center>

<h1>welcome to dashboard</h1>

<hr>

<h2>name : <p> {{$data->name}}</p></h2>

<h2>email : <p> {{$data->email}}</p></h2>

<br>

<a href="logout">logout</a>

</center>

------------------------------------------------------------------------------protection //50:00

php artisan make:middleware AuthCheck

------------------------------------------------------------------------------

app\http\kernel.php

protected $middlewareAliases = [

'isLoggedIn' => \App\Http\Middleware\AuthCheck::class,

'auth' => \App\Http\Middleware\Authenticate::class,

------------------------------------------------------------------------------

app\http\Middleware\AuthCheck.php

public function handle(Request $request, Closure $next): Response

{

if(!Session()->has('loginId')){

return redirect('login')->with('fail','you have to login first');

}

return $next($request);

}

------------------------------------------------------------------------------

routes>web.php //permet de prevenir l'utilisateur qu'il faut faire dabord login

Route::get('/dashboard',[CustomAuthController::class,'dashboard'])->middleware('isLoggedIn');

------------------------------------------------------------------------------

php artisan make:middleware AlreadyLoggedIn

app\http\kernel.php

protected $middlewareAliases = [

'alreadyLoggedIn' => \App\Http\Middleware\AlreadyLoggedIn::class,

'isLoggedIn' => \App\Http\Middleware\AuthCheck::class,

------------------------------------------------------------------------------

routes>web.php //permet de prevenir l'utilisateur qu'il est deja fait login

Route::get('/login',[CustomAuthController::class,'login'])->middleware('alreadyLoggedIn');

------------------------------------------------------------------------------

app>http>middleware>AlreadyLoggedIn.php

public function handle(Request $request, Closure $next): Response

{

if(Session()->has('loginId') && (url('login')==$request->url() || url('registration')==$request->url() )){

return back();

}

return $next($request);

}

------------------

------------------

resources>views>dashboard1.blade.php

<center>

hello {{$data->name}}

<hr>

<br><center><h1>

to day is the day ..

</h1></center>

<a href="dashboard">retour</a>

<a href="logout">logout</a>

</center>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

routes>web.php

Route::get('/dashboard1',[CustomAuthController::class,'dashboard1'])->middleware('isLoggedIn');

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

app>http>controllers>customAuthController.php

public function dashboard1(){

$data =array();

if (Session::has('loginId')){

$data = User::where('id','=',Session::get('loginId'))->first();

}

return view('dashboard1',compact('data'));

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

------------------------------------------------------image

php artisan make:controller ImageController

app>http>controllers>imagecontroller.php

use Illuminate\View\View;

public function index():View

{

return view('image');

}

-------

routes>web.php

use App\Http\Controllers\ImageController;

Route::controller(ImageController::class)->group(function(){

Route::get('image-upload','index');

});

---------------

resources>views>image.blade.php

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

LARAVEL CRUD

composer create-project laravel/laravel crud

cd crud

php artisan serve => Laravel v10.37.3 (PHP v8.1.25)

.env

DB\_DATABASE=bd14

php artisan migrate

php artisan make:migration create\_prodcts\_tables

2023\_12\_21\_114602\_create\_prodcts\_tables.php

Schema::create('products', function (Blueprint $table) {

$table->id();

$table->strling('name');

$table->integer('qty');

$table->decimal('price');

$table->text('description');

$table->timestamps();

php artisan migrate

php artisan make:model Product

app>Models>Product.php

use HasFactory;

protected $fillable = [

'name',

'qty',

'price',

'description'

];

php artisan make:controller ProductController

app>Http>Controllers>ProductController.php

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class ProductController extends Controller

{

public function index(){

return view('products.index');

}

}

creation d'un dossier et un fichier

resources>views>products>index.blade.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

</head>

<body>

<h1>Product</h1>

hello tina lalaina

</body>

</html>

-----------------------------

routes>web.php

use App\Http\Controllers\ProductController;

Route::get('/product', [ProductController::class, 'index'])->name('product.index');

http://localhost:8000/product

routes>web.php

Route::get('/product/create', [ProductController::class, 'create'])->name('product.create');

app>Http>Controllers>ProductController.php

public function create(){

return view('products.create');

}

creation d'un dossier et un fichier

resources>views>products>create.blade.php

<h1>Create a Product</h1>

<form method="post" action="">

<div>

<label>Name</label>

<input type="text" name="name" placeholder="Name" />

</div>

<div>

<label>Qty</label>

<input type="text" name="qty" placeholder="Qty" />

</div>

<div>

<label>Price</label>

<input type="text" name="price" placeholder="Price" />

</div>

<div>

<label>Description</label>

<input type="text" name="description" placeholder="Description" />

</div>

<div>

<input type="submit" value="Save a New Product" />

</div>

</form>

http://localhost:8000/product/create

routes>web.php

Route::post('/product', [ProductController::class, 'store'])->name('product.store');

app>Http>Controllers>ProductController.php

use App\Models\Product;

public function store(Request $request){

$data = $request->validate([

'name' => 'required',

'qty' => 'required|numeric',

'price' => 'required|numeric', ici ca doit etre decimal fait une recherche

'description' => 'nullable'

]);

$newProduct = Product::create($data);

return redirect(route('product.index'));

}

modification sur

resources>views>products>create.blade.php

<form method="post" action="{{route('product.store')}}">

@csrf

@method('post')

app>Http>Controllers>ProductController.php modifier

public function index(){

$products = Product::all();

return view('products.index', ['products' => $products]);

}

modification sur

index.blade.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

</head>

<body>

<h1>Product</h1>

<div>

<table border="1">

<tr>

<th>ID</th>

<th>Name</th>

<th>Qty</th>

<th>Price</th>

<th>Description</th>

<th>Edit</th>

<th>Delete</th>

</tr>

@foreach($products as $product )

<tr>

<td>{{$product->id}}</td>

<td>{{$product->name}}</td>

<td>{{$product->qty}}</td>

<td>{{$product->price}}</td>

<td>{{$product->description}}</td>

<td>

<a href="">Edit</a>

</td>

<td>

<input type="submit" value="Delete" />

</td>

</tr>

@endforeach

</table>

</div>

</body>

</html>

web.php

Route::get('/product/{product}/edit', [ProductController::class, 'edit'])->name('product.edit');

ProductController.php

public function edit(Product $product){

return view('products.edit', ['product' => $product]);

}

creation d'un nouveau fichier dans

Product>edit.blade.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

</head>

<body>

<h1>Edit a Product</h1>

<form method="post" action="">

@csrf

@method('put')

<div>

<label>Name</label>

<input type="text" name="name" placeholder="Name" value="{{$product->name}}" />

</div>

<div>

<label>Qty</label>

<input type="text" name="qty" placeholder="Qty" value="{{$product->qty}}" />

</div>

<div>

<label>Price</label>

<input type="text" name="price" placeholder="Price" value="{{$product->price}}" />

</div>

<div>

<label>Description</label>

<input type="text" name="description" placeholder="Description" value="{{$product->description}}" />

</div>

<div>

<input type="submit" value="Update" />

</div>

</form>

</body>

</html>

Route::put('/product/{product}/update', [ProductController::class, 'update'])->name('product.update');

ProductController.php

public function update(Product $product, Request $request){

$data = $request->validate([

'name' => 'required',

'qty' => 'required|numeric',

'price' => 'required|numeric',

'description' => 'nullable'

]);

$product->update($data);

return redirect(route('product.index'))->with('success', 'Product Updated Succesffully');

}

edit.blade.php modifier

<form method="post" action="{{route('product.update', ['product' => $product])}}">

@csrf

@method('put')

index.blade.php modifier

<td>

<form method="post" action="">

@csrf

@method('delete')

<input type="submit" value="Delete" />

</form>

</td>

web.php

Route::delete('/product/{product}/destroy', [ProductController::class, 'destroy'])->name('product.destroy');

ProductController.php

public function destroy(Product $product){

$product->delete();

return redirect(route('product.index'))->with('success', 'Product deleted Succesffully');

}

index.blad.php modifier

<form method="post" action="{{route('product.destroy', ['product' => $product])}}">

index.blade.php

<a href="{{route('product.edit', ['product' => $product])}}">Edit</a>

-----------------------------------------------

ENREGISTREMENT D’IMAGE AVEC LARAVEL

.env db=>DB\_DATABASE=bd16

-----------------------------------------------------

php artisan make:migration create\_employees\_table

database>migrations>create\_employees\_table.php

<?php

use Illuminate\Database\Migrations\Migration;

use Illuminate\Database\Schema\Blueprint;

use Illuminate\Support\Facades\Schema;

return new class extends Migration

{

/\*\*

\* Run the migrations.

\*

\* @return void

\*/

public function up()

{

Schema::create('employees', function (Blueprint $table) {

$table->id();

$table->string('name');

$table->string('address');

$table->string('mobile');

$table->string('photo', 300);

$table->timestamps();

});

}

/\*\*

\* Reverse the migrations.

\*

\* @return void

\*/

public function down()

{

Schema::dropIfExists('employees');

}

};

---------------------------------------------

php artisan migrate

---------------------------------------------

php artisan make:controller EmployeeController --resource

app\Http\Controllers\EmployeeController.php

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Employee;

class EmployeeController extends Controller

{

/\*\*

\* Display a listing of the resource.

\*

\* @return \Illuminate\Http\Response

\*/

public function index()

{

$employees = Employee::all();

return view ('employees.index')->with('employees', $employees);

}

/\*\*

\* Show the form for creating a new resource.

\*

\* @return \Illuminate\Http\Response

\*/

public function create()

{

return view('employees.create');

}

/\*\*

\* Store a newly created resource in storage.

\*

\* @param \Illuminate\Http\Request $request

\* @return \Illuminate\Http\Response

\*/

public function store(Request $request)

{

$requestData = $request->all();

$fileName = time().$request->file('photo')->getClientOriginalName();

$path = $request->file('photo')->storeAs('images', $fileName, 'public');

$requestData["photo"] = '/storage/'.$path;

Employee::create($requestData);

return redirect('employee')->with('flash\_message', 'Employee Addedd!');

}

/\*\*

\* 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)

{

//

}

}

---------------------------------------

php artisan make:model Employee

---------------------------------------

app\Models\Employee.php

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

class Employee extends Model

{

use HasFactory;

protected $table = 'employees';

protected $primaryKey = 'id';

protected $fillable = ['name', 'address', 'mobile','photo'];

}

--------------------------------------

resources\views\employee\layout.blade.php

<head>

<title>Laravel 9 Image Upload and Display in Datatable | File Storage</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

</head>

<body>

<div class="container">

@yield('content')

</div>

</body>

</html>

---------------------------------------

resources\views\employee\create.blade.php

@extends('employees.layout')

@section('content')

<div class="card" style="margin: 20px;">

<div class="card-header">Create New Employee</div>

<div class="card-body">

<form action="{{ url('employee') }}" method="post" enctype="multipart/form-data">

{!! csrf\_field() !!}

<label>Name</label></br>

<input type="text" name="name" id="name" class="form-control"></br>

<label>Address</label></br>

<input type="text" name="address" id="address" class="form-control"></br>

<label>Mobile</label></br>

<input type="text" name="mobile" id="mobile" class="form-control"></br>

<input class="form-control" name="photo" type="file" id="photo">

<input type="submit" value="Save" class="btn btn-success"></br>

</form>

</div>

</div>

@stop

-----------------------------------------------

resources\views\employee\index.blade.php

@extends('employees.layout')

@section('content')

<div class="container">

<div class="row">

<div class="col-12" style="padding:20px;">

<div class="card">

<div class="card-header">Laravel 9 Image Upload and Display in Datatable | File Storage</div>

<div class="card-body">

<a href="{{ url('/employee/create') }}" class="btn btn-success btn-sm" title="Add New Contact">

<i class="fa fa-plus" aria-hidden="true"></i> Add New

</a>

<br/>

<br/>

<div class="table-responsive">

<table class="table">

<thead>

<tr>

<th>#</th>

<th>Name</th>

<th>Address</th>

<th>Telephone</th>

<th>Photo</th>

</thead>

</thead>

<tbody>

@foreach($employees as $item)

<tr>

<td>{{ $loop->iteration }}</td>

<td>{{ $item->name }}</td>

<td>{{ $item->address }}</td>

<td>{{ $item->mobile }}</td>

<td>

<img src="{{ asset($item->photo) }}" width= '50' height='50' class="img img-responsive" />

</td>

</tr>

@endforeach

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

</div>

@endsection

----------------------------------

routes\web.php

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\EmployeeController; //add the ControllerNameSpace

Route::get('/', function () {

return view('welcome');

});

Route::resource("/employee", EmployeeController::class);

---------------------------------------

php artisan storage:link

php artisan serve

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*