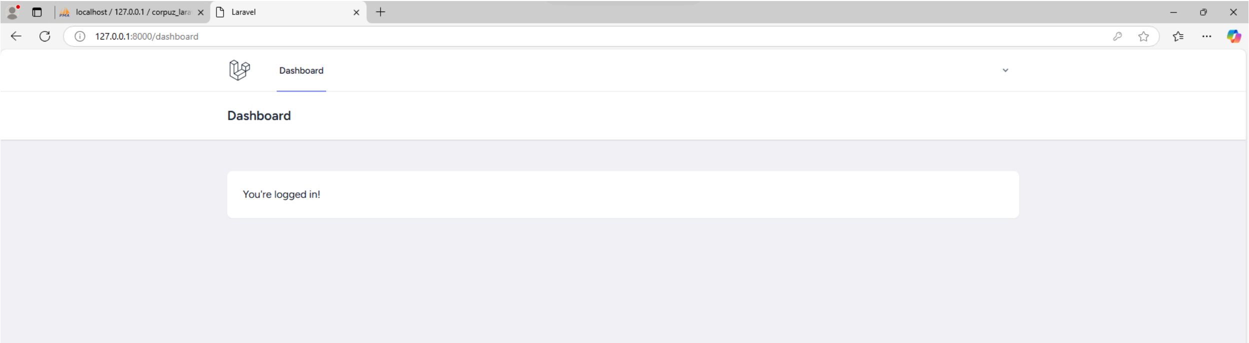
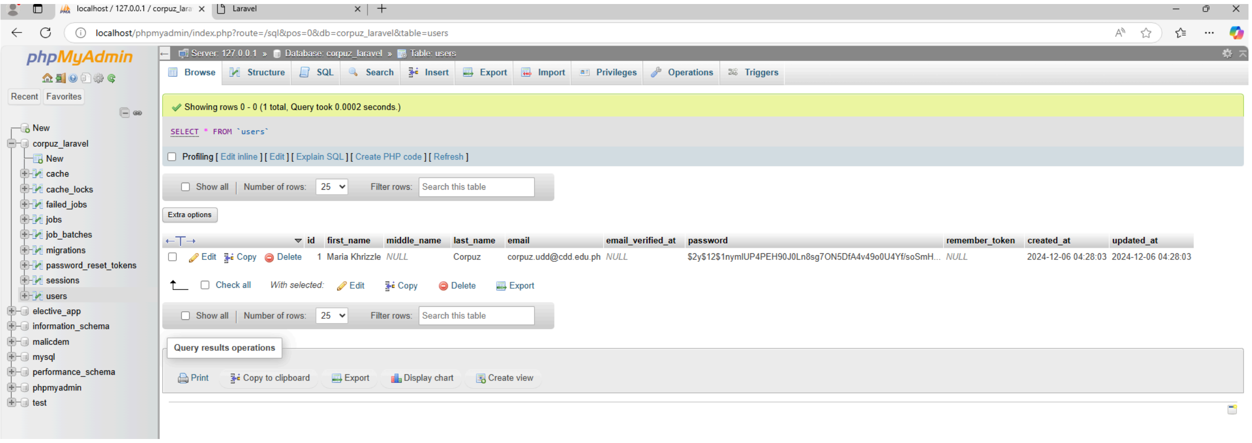
Dashboard



Dashboard

You're logged in!

Database



✓ Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)

```
SELECT * FROM `users`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

←T→	id	first_name	middle_name	last_name	email	email_verified_at	password	remember_token	created_at	updated_at
□ Edit Copy Delete	1	Maria Khrizzle	NULL	Corpuz	corpuz.udd@cdd.edu.ph	NULL	$2y$12$1nymlUP4PEH90J0Ln8sg7ON5DfA4v49o0U4Yf/soSmH...	NULL	2024-12-06 04:28:03	2024-12-06 04:28:03

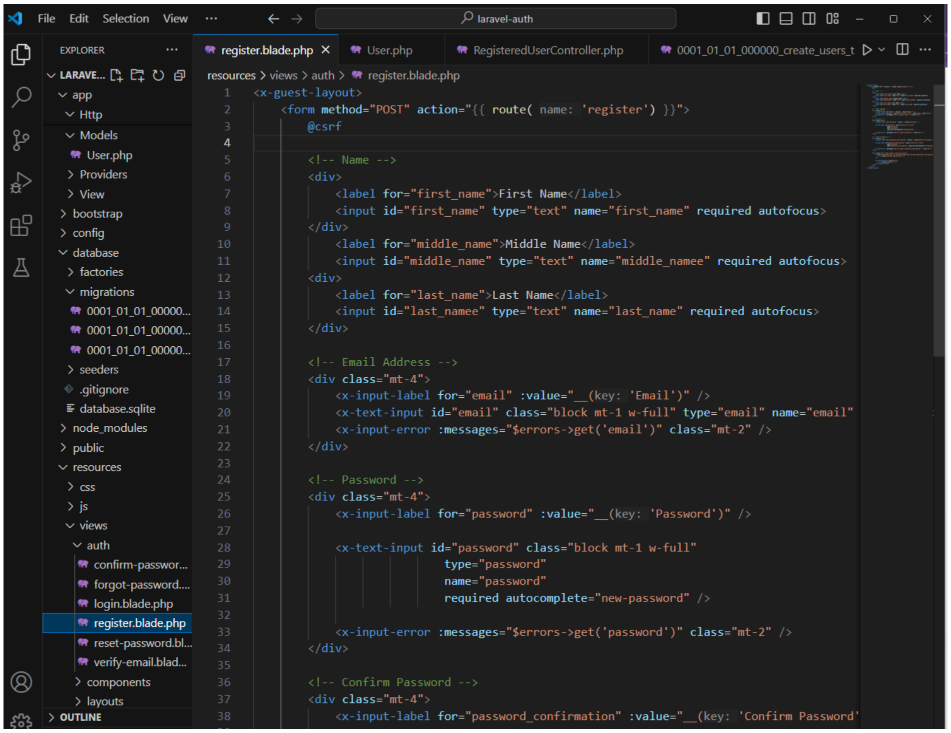
□ Check all *With selected:* Edit Copy Delete Export

□ Show all | Number of rows: 25 | Filter rows: Search this table

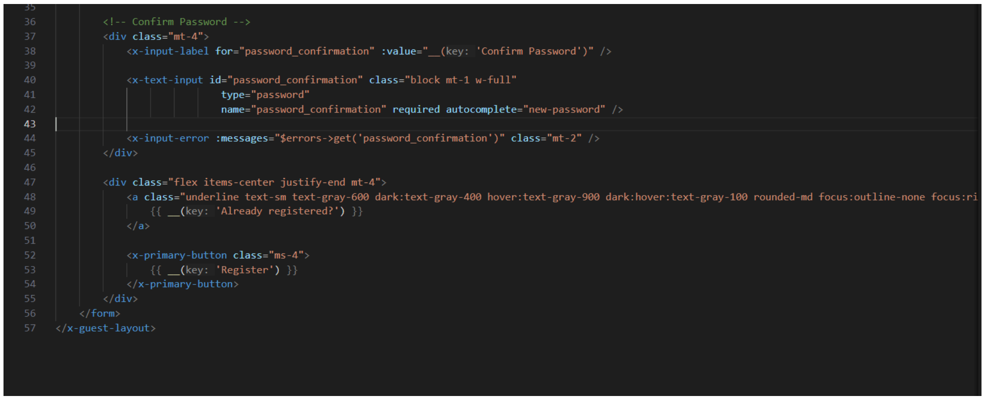
Query results operations

Print Copy to clipboard Export Display chart Create view

Register.blade

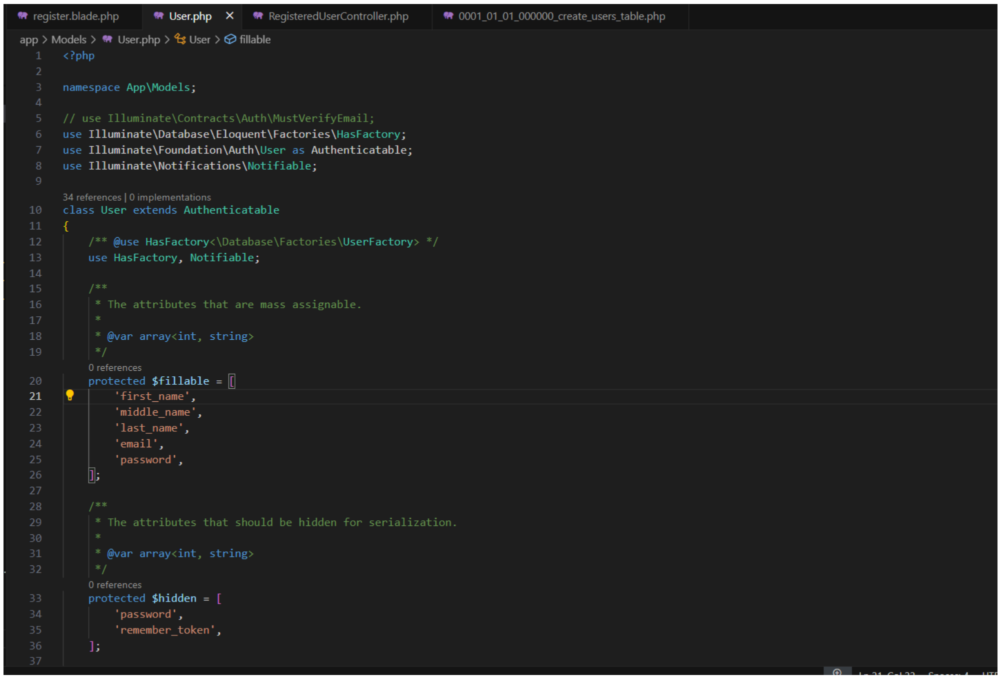


```
<x-guest-layout>
    <form method="POST" action="{{ route( name: 'register') }}">
        @csrf

        <!-- Name -->
        <div>
            <label for="first_name">First Name</label>
            <input id="first_name" type="text" name="first_name" required autofocus>
        </div>
            <label for="middle_name">Middle Name</label>
            <input id="middle_name" type="text" name="middle_namee" required autofocus>
        <div>
            <label for="last_name">Last Name</label>
            <input id="last_namee" type="text" name="last_name" required autofocus>
        </div>

        <!-- Email Address -->
        <div class="mt-4">
            <x-input-label for="email" :value="__(key: 'Email')" />
            <x-text-input id="email" class="block mt-1 w-full" type="email" name="email"
            <x-input-error :messages="$errors->get('email')" class="mt-2" />
        </div>

        <!-- Password -->
        <div class="mt-4">
            <x-input-label for="password" :value="__(key: 'Password')" />

            <x-text-input id="password" class="block mt-1 w-full"
                            type="password"
                            name="password"
                            required autocomplete="new-password" />

            <x-input-error :messages="$errors->get('password')" class="mt-2" />
        </div>

        <!-- Confirm Password -->
        <div class="mt-4">
            <x-input-label for="password_confirmation" :value="__(key: 'Confirm Password'
```

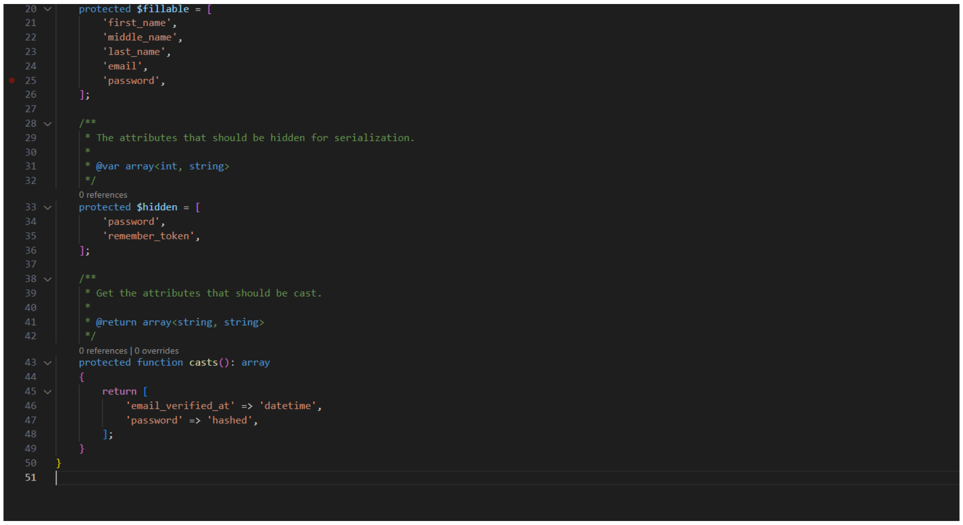


```
        <!-- Confirm Password -->
        <div class="mt-4">
            <x-input-label for="password_confirmation" :value="__(key: 'Confirm Password')" />

            <x-text-input id="password_confirmation" class="block mt-1 w-full"
                            type="password"
                            name="password_confirmation" required autocomplete="new-password" />

            <x-input-error :messages="$errors->get('password_confirmation')" class="mt-2" />
        </div>

        <div class="flex items-center justify-end mt-4">
            <a class="underline text-sm text-gray-600 dark:text-gray-400 hover:text-gray-900 dark:hover:text-gray-100 rounded-md focus:outline-none focus:ri
                {{ __(key: 'Already registered?') }}
            </a>

            <x-primary-button class="ms-4">
                {{ __(key: 'Register') }}
            </x-primary-button>
        </div>
    </form>
</x-guest-layout>
```

User.php

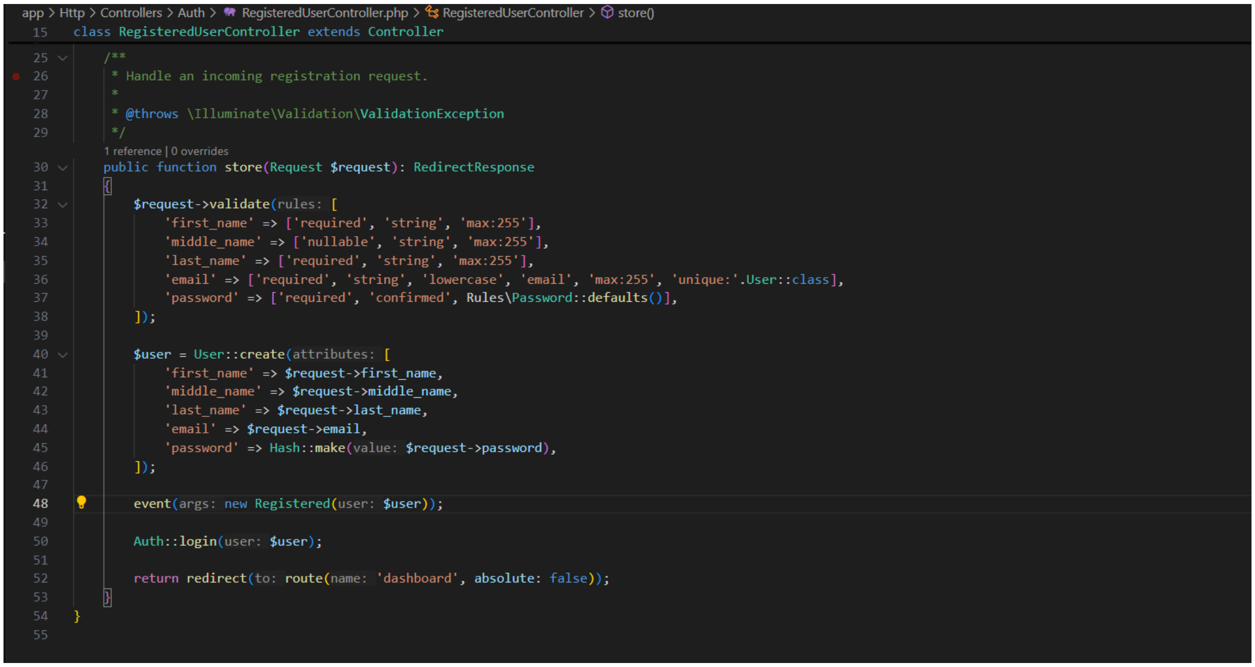

register.blade.php
User.php
RegisteredUserController.php
0001_01_01_000000_create_users_table.php
app > Models > User.php > User > fillable

```
<?php

namespace App\Models;

// use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;

34 references | 0 implementations
class User extends Authenticatable
{
    /** @use HasFactory<\Database\Factories\UserFactory> */
    use HasFactory, Notifiable;

    /**
     * The attributes that are mass assignable.
     *
     * @var array<int, string>
     */
    0 references
    protected $fillable = [
        'first_name',
        'middle_name',
        'last_name',
        'email',
        'password',
    ];

    /**
     * The attributes that should be hidden for serialization.
     *
     * @var array<int, string>
     */
    0 references
    protected $hidden = [
        'password',
        'remember_token',
    ];

```



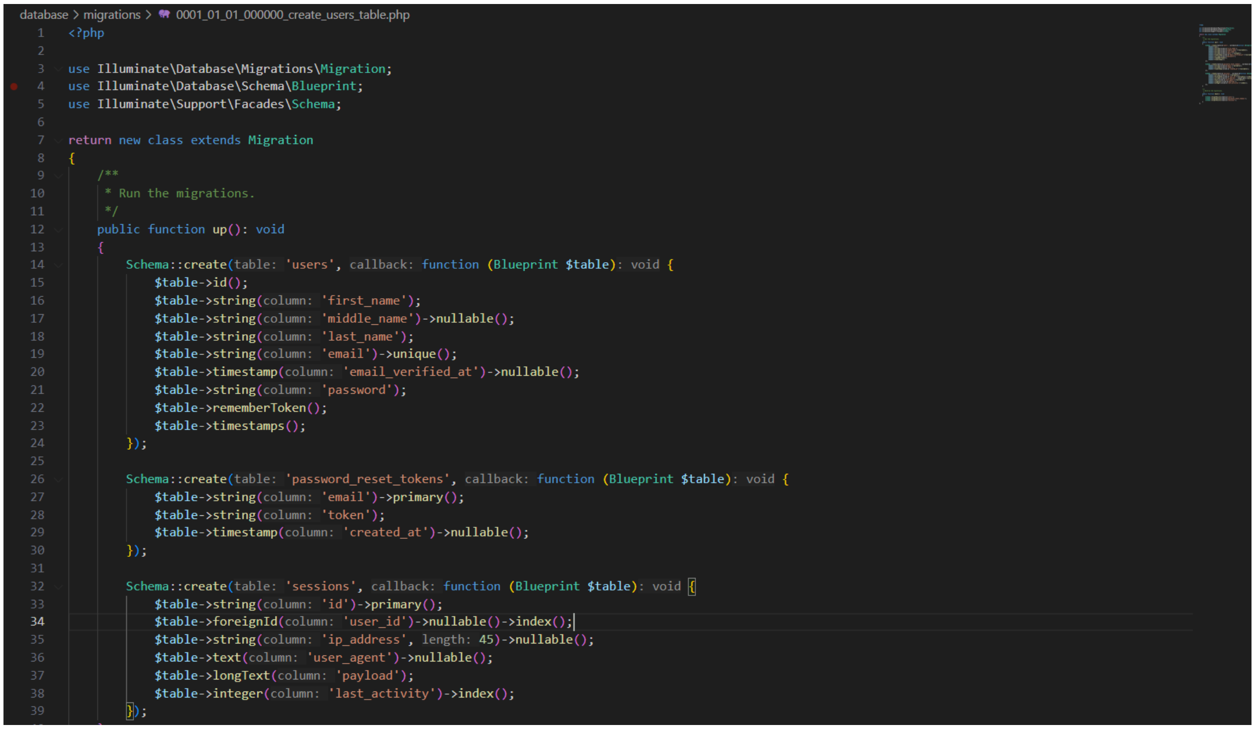
```
    protected $fillable = [
        'first_name',
        'middle_name',
        'last_name',
        'email',
        'password',
    ];

    /**
     * The attributes that should be hidden for serialization.
     *
     * @var array<int, string>
     */
    0 references
    protected $hidden = [
        'password',
        'remember_token',
    ];

    /**
     * Get the attributes that should be cast.
     *
     * @return array<string, string>
     */
    0 references | 0 overrides
    protected function casts(): array
    {
        return [
            'email_verified_at' => 'datetime',
            'password' => 'hashed',
        ];
    }
}

```

RegisteredUserController


app > Http > Controllers > Auth > RegisteredUserController.php > RegisteredUserController > store()

```php
class RegisteredUserController extends Controller

    /**
     * Handle an incoming registration request.
     *
     * @throws \Illuminate\Validation\ValidationException
     */
    1 reference | 0 overrides
    public function store(Request $request): RedirectResponse
    {
        $request->validate(rules: [
            'first_name' => ['required', 'string', 'max:255'],
            'middle_name' => ['nullable', 'string', 'max:255'],
            'last_name' => ['required', 'string', 'max:255'],
            'email' => ['required', 'string', 'lowercase', 'email', 'max:255', 'unique:'.User::class],
            'password' => ['required', 'confirmed', Rules\Password::defaults()],
        ]);

        $user = User::create(attributes: [
            'first_name' => $request->first_name,
            'middle_name' => $request->middle_name,
            'last_name' => $request->last_name,
            'email' => $request->email,
            'password' => Hash::make(value: $request->password),
        ]);

        event(args: new Registered(user: $user));

        Auth::login(user: $user);

        return redirect(to: route(name: 'dashboard', absolute: false));
    }
}
```

0001_01_01_000000_create_users_table


database > migrations > 0001_01_01_000000_create_users_table.php

```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.
     */
    public function up(): void
    {
        Schema::create(table: 'users', callback: function (Blueprint $table): void {
            $table->id();
            $table->string(column: 'first_name');
            $table->string(column: 'middle_name')->nullable();
            $table->string(column: 'last_name');
            $table->string(column: 'email')->unique();
            $table->timestamp(column: 'email_verified_at')->nullable();
            $table->string(column: 'password');
            $table->rememberToken();
            $table->timestamps();
        });

        Schema::create(table: 'password_reset_tokens', callback: function (Blueprint $table): void {
            $table->string(column: 'email')->primary();
            $table->string(column: 'token');
            $table->timestamp(column: 'created_at')->nullable();
        });

        Schema::create(table: 'sessions', callback: function (Blueprint $table): void {
            $table->string(column: 'id')->primary();
            $table->foreignId(column: 'user_id')->nullable()->index();
            $table->string(column: 'ip_address', length: 45)->nullable();
            $table->text(column: 'user_agent')->nullable();
            $table->longText(column: 'payload');
            $table->integer(column: 'last_activity')->index();
        });
```