**Laravel Database Configuration**

**Step-01:**

Install Laravel using command:

composer create-project laravel/laravel notificationDatabase

**Step-02:**

setup the .env file  
 DB\_DATABASE=notificationDatabase

**Step-03:**

php artisan notifications:table  
**Step-04:**

php artisan migrate

**step-05:**

composer require Laravel/breeze –dev

php artisan breeze:install

**Step-06:**

npm install

npm run dev

**Step-07:**

npm run watch

In the **package.json** file, the code below needs to be updated to execute this command.

    "scripts": {

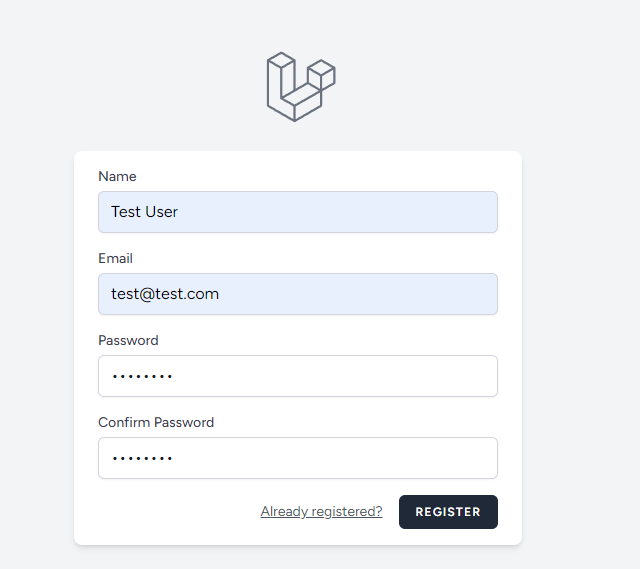
        "dev": "vite",

        "build": "vite build",

        "watch": "vite build --watch"

    },

**Step-08:**

****

**Step-09:**

php artisan make:notification UserFollowNotification

**Step-10:**

php artisan make:controller HomeController

**HomeController.php**<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Notifications\UserFollowNotification;

use App\Models\User;

class HomeController extends Controller

{

    public function index()

    {

        if(auth()->user()){

        $user = User::first();

        auth()->user()->notify(new UserFollowNotification($user));

        }

        dd('done');

    }

    public function markasread($id)

    {

        if($id){

            auth()->user()->notifications()->find($id)->markAsRead();

        }

        return redirect()->back();

    }

}

**web.php**

<?php

use App\Http\Controllers\ProfileController;

use Illuminate\Support\Facades\Route;

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

    return view('welcome');

});

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

    return view('dashboard');

})->middleware(['auth', 'verified'])->name('dashboard');

Route::middleware('auth')->group(function () {

    Route::get('/profile', [ProfileController::class, 'edit'])->name('profile.edit');

    Route::patch('/profile', [ProfileController::class, 'update'])->name('profile.update');

    Route::delete('/profile', [ProfileController::class, 'destroy'])->name('profile.destroy');

});

require \_\_DIR\_\_.'/auth.php';

Route::get('/notify', [\App\Http\Controllers\HomeController::class, 'index']);

Route::get('/markasread/{id}', [\App\Http\Controllers\HomeController::class, 'markasread'])->name('markasread');

**Dashboard.blade.php**

<x-app-layout>

    <x-slot name="header">

        <h2 class="font-semibold text-xl text-gray-800 leading-tight">

            {{ \_\_('Dashboard') }}

        </h2>

    </x-slot>

    <div class="py-12">

        <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">

            <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">

                <div class="p-6 bg-white border-b border-gray-200">

                    @foreach( auth()->user()->notifications as $notification)

                    <div class="bg-blue-300 p-3 m-2">

                       <b> {{ $notification->data['name'] }} started following you !!!</b>

                       <a href="{{ route('markasread', $notification->id) }}" class="p-2 bg-red-400 text-white rounded-lg">Mark as read</a>

                    </div>

                    @endforeach

                </div>

            </div>

        </div>

    </div>

</x-app-layout>

auth()->user()->notifications/unreadnotification/readnotification

**UserFollowNotification.php**

<?php

namespace App\Notifications;

use Illuminate\Bus\Queueable;

use Illuminate\Contracts\Queue\ShouldQueue;

use Illuminate\Notifications\Messages\MailMessage;

use Illuminate\Notifications\Notification;

class UserFollowNotification extends Notification

{

    use Queueable;

    public $user;

    /\*\*

     \* Create a new notification instance.

     \*/

    public function \_\_construct($user)

    {

        $this->user = $user;

    }

    public function via(object $notifiable): array

    {

        return ['database'];

    }

    /\*\*

     \* Get the array representation of the notification.

     \*

     \* @return array<string, mixed>

     \*/

    public function toArray(object $notifiable): array

    {

        return [

            'user\_id' => $this->user['id'],

            'name' => $this->user['name'],

            'email' => $this->user['email']

        ];

    }

}

**Final Output:**

