

Univerzitet u Novom Sadu   
Tehnički fakultet „Mihajlo Pupin“   
Zrenjanin

SEMINARSKI RAD

Predmet: Internet alati i servisi

**Evidencija zakazanih termina u mehaničarskoj radionici**

Predmetni nastavnik: Autor rada:

prof.dr Dragana Glušac Stefan Zagorac SI 8/16

Predmetni asistent:

Prof.dr Marko Blažić

Zrenjanin 2022.

**Sadržaj:**

[**Predmetni cilj** 3](#_Toc122586907)

[**Uvod** 4](#_Toc122586908)

[**1.** **Korišćene tehnologije** 5](#_Toc122586909)

[1.1. Microsoft VS Code 5](#_Toc122586910)

[**2.** **Baze podataka** 6](#_Toc122586911)

[**3.** **Jezičke tehnologije** 7](#_Toc122586912)

[3.1. PHP 7](#_Toc122586913)

[3.2. Laravel 7](#_Toc122586914)

[**4.** **Izrada aplikacije** 8](#_Toc122586915)

[4.1. Kreiranje Layout-a 10](#_Toc122586916)

[4.2. Kreiranje kovera 13](#_Toc122586917)

[4.3. Kreiranje logike za CRUD sistem 15](#_Toc122586918)

[4.4. UI Login i stranice za ragistraciju 23](#_Toc122586919)

[4.5. UI stranica za omogućavanje CRUD sistema korisnicima 28](#_Toc122586920)

[4.6. Kreiranje ruta 41](#_Toc122586921)

[**5.** **Zaključak** 43](#_Toc122586922)

[**6.** **Literatura** 44](#_Toc122586923)

[**7.** **Izvori preuzetih ilustracija** 45](#_Toc122586924)

## **Predmetni cilj**

Predmetni cilj predmeta Mobilne tehnologije i programiranje je upoznavanje sa vrstama Internet alata i servisa i njihovom upotrebom.

Predavanja se izvode verbalno - tekstualnim i ilustrativno-demonstrativnim metodama u frontalnom obliku rada gde se  prezentuju osnovne informacije i uputstva (usmenim predavanjima, pomoću video bima).

Vežbe se izvode  laboratorijsko eksperimentalnim metodama gde se sprovodi praktična aktivnost studenata. Oblik rada na vežbama je individualnom rad studenata i rad u paru za izradu projekata za seminarski rad.

## **Uvod**

Internet servisi se zasnivaju na arhitekturi klijenta i servera, server je kompjuter koji je stalno spojen na internet te pruža neku mrežnu uslugu odnosno servis, dok je klijent korisnički kompjuter,odnosno program koji koristi neku od usluga (Web klijent).

Prema današnjem stepenu razvoja Interneta moguće je razlučiti tri najvažnija generička i više izvedenih opštih servisa.

Hronološkim redom njihovog nastajanja, najvažniji generički standardni internetski servisi su:

1. ***Daljinski prenos datoteka***;
2. ***Elektronska pošta***;
3. ***Svetska mreža***.

Postoji i podela na pet opštih internet servisa i to:

1. **Bazični servisi**:
2. **Javni informacioni servisi**:
3. **Diskusioni servisi**:
4. **Konferencijski servisi**:
5. **Servisi za pretraživanje interneta**

## **Korišćene tehnologije**

U ovom seminarskom radu korišćene su tehnologije za razvoj web aplikacija i to su: Microsoft VS Code, CMD, PowerShell, XAMPP.

### 1.1. Microsoft VS Code

Microsoft VS Code je editor izvornog koda koji se može koristiti za razne programske jezike, kao što su Java, Javaskript, Gou, Node.js i C++.

Umesto sistema projekata, VS Code omogućava korisniku da otvori jedan ili više direktorijuma, što zatim može biti sačuvano kao radni prostor za ponovnu upotrebu. Ovo mu omogućava da radi kao jezičko-agnostički uređivač koda za bilo koji jezik. Podržava brojne programske jezike i skup karakteristika koje se razlikuju od jezika do jezika. Neželjene datoteke i direktorijumi se mogu isključiti iz stabla projekta preko podešavanja. Mnoge karakteristike ovog uređivača nisu izložene u menijima ili u korisničkom interfejsu, već im se može pristupiti preko komandne palete.

## **Baze podataka**

Za ovaj seminarski rad korišćena je lokalna MySQL baza podataka i XAMPP kontrolna tabla koja omogućava korišćenje lokalnog servera.

Lokalni server engl. localhost u najjednostavnijoj definiciji je web server koji se nalazi na lokalnom sistemu kao što je vaš računar, što znači da nije dostupan na internetu.

Temeljno testiranje u cilju uklanjanja grešaka mora da se obavi pre nego što programer objavi dodatak ili temu za preuzimanje. Lokalni server daje programeru mogućnost da testira svoj proizvod u kontrolisanom okruženju bez plaćanja hosting kompanije. pošto se datoteke čuvaju lokalno na računaru, pregledač ima sve što je potrebno za prikaz web stranice koja je vidljiva samo programeru. Lokalni server je sigurniji i pruža idealno okruženje za testiranje, daleko od znatiželjnih očiju ili hakera. Nije vam potreban internet da biste mogli da koristite svoj lokalni server bilo gde, u bilo koje vreme, samo vam treba računar.



*Slika 1. MySQL logo.*

## **Jezičke tehnologije**

Za izadu ovog seminarskog rada korišćeni su programski jezik PHP i framework Laravel, HTML, CSS(Tailwind).

### PHP

PHP (PHP: Hypertext Preprocessor) je skriptni jezik koji se izvršava na strani servera i omogućava dinamičko kreiranje HTML sadžaja uz povezivanje sa bazama podataka (u skladu sa specifičnim zahtevima klijenta), kao i obavljanje drugih pozadinskih zadataka na serveru.

Ovaj jezik kreirao je danski programer Rasmus Lerdorf (1994. godine), kao jednostavan sistem za kreiranje dinamičkih sadržaja za manje web prezentacije (da ne kažemo, "za lične potrebe" - čemu je odgovaro i ondašnji pun naziv jezika: Personal Home Page), ali se tokom vremena jezik razvio i postao vrlo moćan i popularan.

Što se tiče popularnosti, smatra se da gotovo dve trećine sajtova na svetu koristi PHP, a što se tiče mogućnosti, jasno je da jezik koji je namenjen kreiranju "ličnih web stranica" ne bismo preporučivali čitaocima i da je PHP dugi niz godina unazad tehnologija preko koje se mogu kreirati krajnje ozbiljni sajtovi.

### Laravel

Laravel je jedan od najpopularnijih PHP framework-a, koji se koristi širom sveta za pravljenje web aplikacija, kako za male, tako i za velike projekte. Laravel je izbor programera zbog svojih performansi, karakteristika i skalabilnosti. takođe prati MVC (Model View Controller) strukturu koja olakšava učenje i brzo kreiranje web aplikacija. Olakšava razvoj web aplikacija pružajući ugrađene funkcije kao što su autentifikacija, pošta, routing, sesije itd.

Framework laravel je jednostavan za prilagođavanje, tako da možete kreirati sopstvenu strukturu projekta koja zadovoljava potrebe vaše web aplikacije. Pomoću Laravelove elegantne sintakse možete pisati kod koji je sam po sebi razumljiv i izražajan. Kreirao ga je Tailor Otvwell još 2011. godine, koji se od tada razvijao i unapređivao i postao kompatibilan sa najnovijim web tehnologijama.

## **Izrada aplikacije**

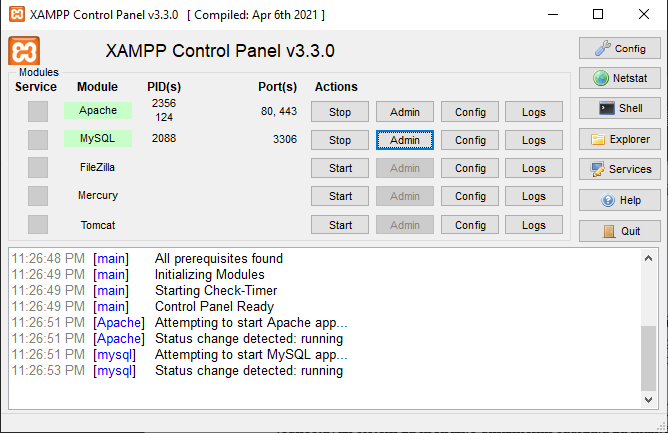
Kod izrade web aplikacije najpre treba kreirati laravel projekat koristeći CMD i composser. Prvenstveno u CMD treba selektovati odredjeni folder (direktorijum) na računaru, pa nakon toga uneti komandu za kreranje projekta.

Komanda za kreiranje laravel projekta:

***composer create-project laravel/laravel evidencija.***

“evidencija” predstavlja naziv projekta.

Nakon kreiranja projetka potrebno je pokrenuti XAMPP kontrolnu tablu i pokrenuti dva servera: Apache i MySQL.



*Slika 1. Pokretanje servera u XAMPP kontrolnoj tabli.*

Zatim klikom na dugme “Admin” otvaramo ”phpMyAdmin” u kome je potrebno kreirati bazu podataka koja će služiti da se podaci skladište, brišu i ažuriraju. Klikom na action text pod naslovom “New” koji se nalazi pri vrhu levog dela stranice otvara se kartica u kojoj unosimo naziv baze pokadataka i u kojoj je mogući podesiti određena podešavanja baze. Naziv je proizvoljan, a sve opcije treba da ostanu podrazumevane (default).

Nakon toga vršimo promene u .env fajlu koji služi za povezivanje sa bazom podataka

DB\_DATABASE=laravel9projekat

DB\_USERNAME=administrator

DB\_PASSWORD=password

*Listing 1. Izvršene promene nad .env fajlom*

Kada je sve podešeno u MS VS Code-u otvaramo folder u kome je kreiran laravel projekat i u terminalu unosimo komandu:

***php artisan serve***

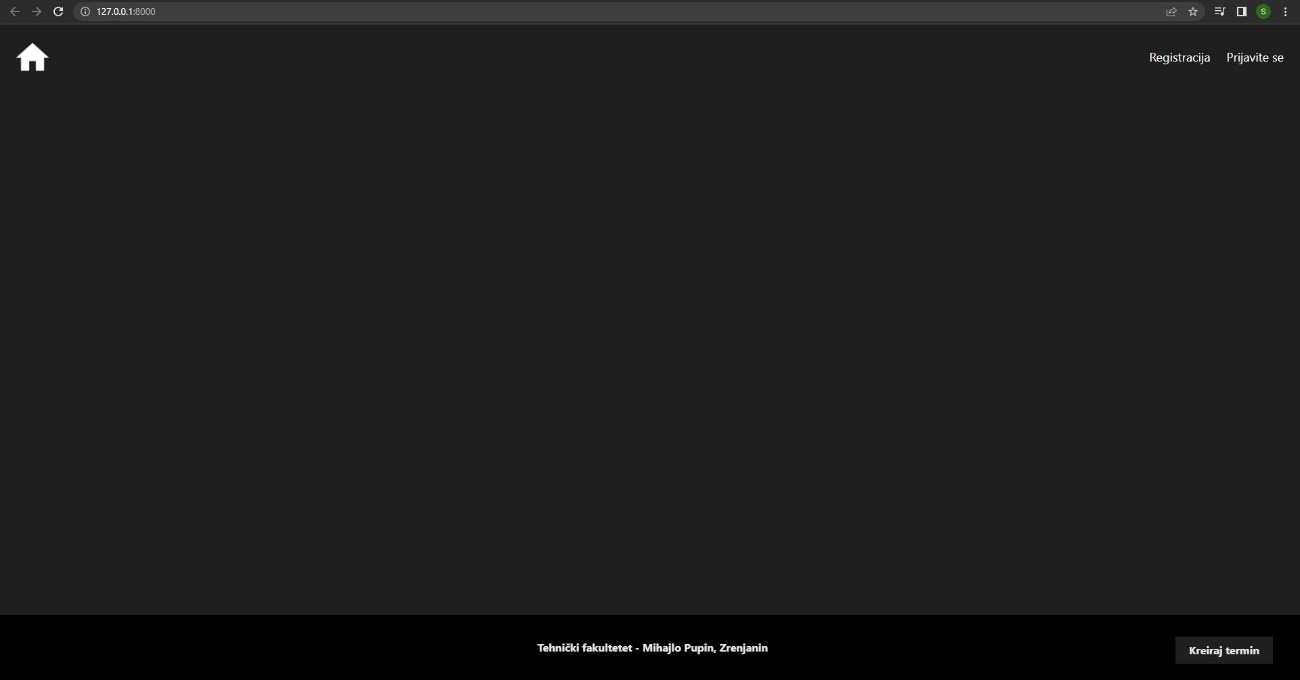
To je radi pokretanja servera na kome se prikazuje pokrenut projekat, i nakon toga još jedan terminal koji će služiti za sve ostale komande.

### Kreiranje Layout-a

Kada je novi laravel projekat kreiran, kreirana je kranje jednostavna web aplikacija koja se sastoji iz jedne stranice sa linkovima koji vode do drugih stranica vezanih za laravel. Ona je potpuno bespotrebna za ovaj seminarski rad pa je iz tog razloga potrebno obrisati sav kod koji se nalazi unutar fajla “welcome.blade.php” i preimenovati ga u “layout.blade.php”.

Ovde se kreirarju fiksni vizuelni elementi aplikacije(navigacija, footer, home button, itd..) i podašavanja za njih (layout) tj. kalup web aplikacije koji prati cela aplikacija tj. ostale stranice u kojima se kasnije koristi.

Nakon toga dobijamo ovakav izgled početne stranice:



*Slika 3. Izgled layout-a*

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8" />

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

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

        <link rel="icon" href="images/favicon.ico" />

        <link

            rel="stylesheet"

            href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css"

            integrity="sha512-KfkfwYDsLkIlwQp6LFnl8zNdLGxu9YAA1QvwINks4PhcElQSvqcyVLLD9aMhXd13uQjoXtEKNosOWaZqXgel0g=="

            crossorigin="anonymous"

            referrerpolicy="no-referrer"

        />

        <script src="https://cdn.tailwindcss.com"></script>

        <script>

            tailwind.config = {

                theme: {

                    extend: {

                        colors: {

                            laravel: "#ef3b2d",

                            darkgrey: "#202020",

                            lightgrey: "#808080",

                        },

                    },

                },

            };

        </script>

        <title>Evidencija</title>

    </head>

    <body class="mb-48 bg-darkgrey">

        <nav class="flex justify-between items-center mb-4">

            <a href="/"

                ><img class="w-24" src="{{asset('images/logo.png')}}" alt="" class="logo"

            /></a>

            <ul class="flex space-x-6 mr-6 text-lg">

                @auth

                    <span class="font-bold uppercase text-white">

                        Dobrodošli {{auth()->user()->name}}

                    </span>

                    <li>

                        <form class="inline" method="POST" action="/logout">

                        @csrf

                        <button type="submit" class="text-white hover:text-laravel">

                            Odjavite se

                        </button>

                    </form>

                    </li>

                @else

                <li>

                    <a href="/register" class="hover:text-laravel text-white"

                        >Registracija</a>

                </li>

                <li>

                    <a href="/login" class="hover:text-laravel text-white"

                        >Prijavite se</a>

                </li>

                @endauth

            </ul>

        </nav>

        <main>

    @yield('content')

</main>

<footer

class="fixed bottom-0 left-0 w-full flex items-center justify-start font-bold bg-black text-white h-24 mt-24 opacity-90 md:justify-center"

>

<p class="ml-2">Tehnički fakultetet - Mihajlo Pupin, Zrenjanin</p>

<a

    href="/listings/create"

    class="absolute top-1/3 right-10 bg-darkgrey text-white py-2 px-5"

    >Kreiraj termin</a

>

</footer>

</body>

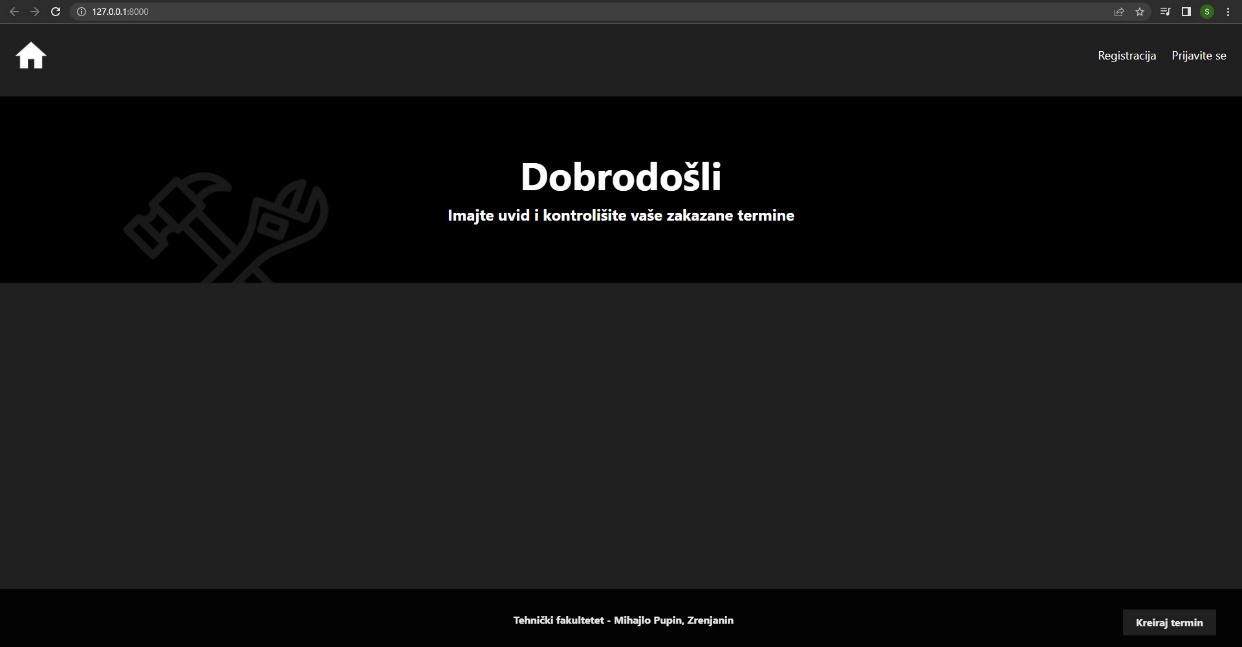
</html>

*Listing 2. Listing koda “layout.blade.php”*

### Kreiranje kovera

Najpre se kreira nov folder pod nazivom “partials“ i u tom folderu stranica pod nazivom “\_cover.blade.php“. ovoj stranci nije potreno dodati elemene layout-a jer služi za kreiranje bannera duž cele stranice. Sastoji se iz sekcije u kojoj su smeštena dva div elementa, jedan sa elementom slike, a u drugom se nalaze heading1 i paragraf elementi

Nakon kreiranja ovog elementa aplikacije dobijamo sledeći UI:



*Slika 4. UI početne stranice.*

Listing koda fajla: “\_cover.blade.php”:

<section

class="relative h-72 bg-black flex flex-col justify-center align-center text-center space-y-4 mb-4"

>

<div

    class="absolute top-0 left-0 w-full h-full opacity-10 bg-no-repeat"

    style="background-image: url('images/laravel-logo.png')"

></div>

<div class="z-10">

    @auth

    <h1 class="text-6xl font-bold text-white">

        Zakazani termini:

    </h1>

    @else

    <h1 class="text-6xl font-bold text-white">

        Dobrodošli

    </h1>

    <p class="text-2xl text-white font-bold my-4">

        Imajte uvid i kontrolišite vaše zakazane termine

    </p>

    @endauth

</div>

</section>

*Listing 3. Listing fajla “\_cover.blade.php”.*

### Kreiranje logike za CRUD sistem

#### **Pravljenje migracije za rad sa bazom**

Da bi aplikacija bila funkcionalna tj. imala backend najpre je potrebno definisati kontroler u drugom pokrenutom terminalu slećem komandom:

***php atrisan make:controller create listings table***

potrebno je obratiti pažnju na zadati naziv kontrolera koji mora biti u množini!

Nakon toga putanjom:

***database > migrations >*** ***2022\_12\_19\_185303\_create\_listings\_table.php***

dolazimo do kreiranog kontrolera koji ima predefinisane dve metode: “up” i “down”, potrebno je promeniti metodu up tako što treba dodati tipove podataka iz tabele sa kojima će da komunicira.

Listing koda migracije:

 <?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('listings', function (Blueprint $table) {

            $table->id();

            $table->string('title');

            $table->string('make');

            $table->string('model');

            $table->string('date');

            $table->longText('note');

            $table->string('price');

            $table->timestamps();

        });

    }

    /\*\*

     \* Reverse the migrations.

     \*

     \* @return void

     \*/

    public function down()

    {

        Schema::dropIfExists('listings');

    }

};

*Listing 4. Listing migracije “2022\_12\_19\_185303\_create\_listings\_table.php”*

#### **Kreiranje modela**

Nakon kreirane migracije potebno je kreirati model takođe koristeći terminal u MS VS Code-u, i to komandom:

***php atrisan make:model Listing***

“Listing” predstavlja naziv modela.

Kada je model kreiran predefinisana je klasa koja ima isti naziv kao i Model i koja proširuje “Model.php” fajl u kome se nalaze razne metode koje nam je laravel omogućio.

Listing modela:

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

class Listing extends Model

{

    use HasFactory;

    protected $fillable = ['title', 'make', 'model', 'date', 'price', 'note'];

}

*Listing 5. listing modela “Lisitng.php”.*

#### **Kreiranje kontrolera**

Kada su migracija i model kreirani potrebno je kreirati kontroler koji će služiti za konkretnu manipulaciju sa podacima iz baze podataka. Radi zahteva aplikacije koja je kao takva osmišljena da mora imati registraciju korisnika da bi imala smisla potrebno je kreirati dva kontrolra, jedan zadužen za rad sa zakazanim teminima, a drugi za rad sa korisnicima aplikacije. Model i migaricju za rad sa korisnicima nije bilo potrebe prethodno kreirati jer su predefinisani.

Kao i migracija i model najbolji način za kreiranje je u terminalu MS VS Code-a gde se pokreće komanda:

***php artisan make:controller*** ***ListingController***

pa zatim:

***php artisan make:controller UserController***

“ListingController” i “UserController” predstavljaju nazive kontrolera.

Kada se kreira kontroler automatski se kreira klasa pod istim nazivom kao i kontroler, isti princip kao i kod kreiranja modela sa tim što se kod kontrolera proširuje fajl “Controller.php” prethodno definisan u laravelu.

Listing kontrolera za termine:

<?php

namespace App\Http\Controllers;

use App\Models\Listing;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Redis;

class ListingController extends Controller

{

    public function index(){

        return view('listings.index', [

            'listings' => Listing::all()

        ]);

    }

    public function create(){

        return view('listings.create');

    }

    public function show(Listing $listing){

        return view('listings.show', [

            'listing' => $listing

        ]);

    }

    public function store(Request $request){

        $formFields = $request->validate([

            'title' => 'required',

            'make' => 'required',

            'model' => 'required',

            'date' => 'required',

            'price' => 'required',

            'note' => 'required'

        ]);

        Listing::create($formFields);

        return redirect('/');

    }

    public function edit(Listing $listing){

        return view('listings.edit', ['listing' =>$listing]);

    }

    public function update(Request $request, Listing $listing){

        $formFields = $request->validate([

            'title' => 'required',

            'make' => 'required',

            'model' => 'required',

            'date' => 'required',

            'price' => 'required',

            'note' => 'required'

        ]);

        $listing->update($formFields);

        return redirect('/');

    }

    public function destroy(Listing $listing) {

        $listing->delete();

        return redirect('/');

    }

}

*Listing 6. Listing koda “ListingController.php”*

Listing kontrolera za korisnike:

<?php

namespace App\Http\Controllers;

use App\Models\User;

use Illuminate\Http\Request;

use Illuminate\Validation\Rule;

class UserController extends Controller

{

    public function create(){

        return view('users.register');

    }

    public function store(Request $request){

        $formFields = $request->validate([

            'name' => ['required', 'min:3'],

            'email' => ['required', 'email', Rule::unique('users', 'email')],

            'password' => 'required|confirmed|min:6'

        ]);

        $formFields['password'] = bcrypt($formFields['password']);

        $user = User::create($formFields);

        auth()->login($user);

        return redirect('/');

    }

    public function logout(Request $request){

        auth()->logout();

        $request->session()->invalidate();

        $request->session()->regenerateToken();

        return redirect('/');

    }

    public function login(){

        return view('users.login');

    }

    public function authenticate(Request $request){

        $formFields = $request->validate([

            'email' => ['required', 'email'],

            'password' => 'required'

        ]);

        if(auth()->attempt($formFields)){

            $request->session()->regenerate();

            return redirect('/');

        }

    }

}

*Listing 7. Listing koda “UserController.php”*

### UI Login i stranice za ragistraciju

Najpre se kreira novi folder pod nazivom “auth”.

#### **Login stranica**

U novokreiranom folderu kreiramo novi fajl tj. stranicu “login.blade.php”.

Koristi elemente iz layout-a, i u njoj se kreira forma za unos podataka korisnika tj e-mail i lozinka korinsika.

Listing login stranice:

@extends(‘layout’)

@section(‘content’)

<div class=”mx-4”>

    <div

        class=”bg-gray-50 border border-gray-200 p-10 rounded max-w-lg mx-auto mt-24”

    >

        <header class=”text-center”>

            <h2 class=”text-2xl font-bold uppercase mb-1”>

                Prijavite se

            </h2>

        </header>

        <form method=”POST” action=”/users/authenticate”>

            @csrf

            <div class=”mb-6”>

                <label for=”email” class=”inline-block text-lg mb-2”

                    >E-mail</label

                >

                <input

                    type=”email”

                    class=”border border-gray-200 rounded p-2 w-full”

                    name=”email”

                    placeholder=”Unesite validan e-mail”

                />

            </div>

            <div class=”mb-6”>

                <label

                    for=”password”

                    class=”inline-block text-lg mb-2”

                >

                    Lozinka

                </label>

                <input

                    type=”password”

                    class=”border border-gray-200 rounded p-2 w-full”

                    name=”password”

                />

            </div>

            <div class=”mb-6”>

                <button

                    type=”submit”

                    class=”bg-laravel text-white rounded py-2 px-4 hover:bg-black”

                >

                    Prijavite se

                </button>

            </div>

            <div class=”mt-8”>

                <p>

                    Niste registrovani?

                    <a href=”/register” class=”text-laravel”

                        >Registrujte se</a

                    >

                </p>

            </div>

        </form>

    </div>

</div>

@endsection

*Listing 8. listing koda “login.blade.php“*

#### **Stranica za registraciju**

Iz razloga što i ova stranica kao i login stranica služi za autentifikaciju korisnika, u folderu “auth” kreiramo fajl pod nazivom “register.blade.php”.

Jako slična stranica kao i stranica za login, takođe korsti elemente layout-a, sa tim što ima 4 polja za unos i treba joj povezati druge rute koje će kasnije biti definisane.

Listing stranice za registraciju:

@extends(‘layout’)

@section(‘content’)

<div class=”mx-4”>

    <div

        class=”bg-gray-50 border border-gray-200 p-10 rounded max-w-lg mx-auto mt-24”

    >

        <header class=”text-center”>

            <h2 class=”text-2xl font-bold uppercase mb-1”>

                Registrujte se

            </h2>

        </header>

        <form method=”POST” action=”/users”>

            @csrf

            <div class=”mb-6”>

                <label for=”name” class=”inline-block text-lg mb-2”>

                    Ime

                </label>

                <input

                    type=”text”

                    class=”border border-gray-200 rounded p-2 w-full”

                    name=”name”

                />

            </div>

            <div class=”mb-6”>

                <label for=”email” class=”inline-block text-lg mb-2”

                    >E-mail</label

                >

                <input

                    type=”email”

                    class=”border border-gray-200 rounded p-2 w-full”

                    name=”email”

                    placeholder=”Unesite validan e-mail”

                />

            </div>

            <div class=”mb-6”>

                <label

                    for=”password”

                    class=”inline-block text-lg mb-2”

                >

                    Lozinka

                </label>

                <input

                    type=”password”

                    class=”border border-gray-200 rounded p-2 w-full”

                    name=”password”

                />

            </div>

            <div class=”mb-6”>

                <label

                    for=”password\_confirmation”

                    class=”inline-block text-lg mb-2”

                >

                    Potvrdite lozinku

                </label>

                <input

                    type=”password”

                    class=”border border-gray-200 rounded p-2 w-full”

                    name=”password\_confirmation”

                />

            </div>

            <div class=”mb-6”>

                <button

                    type=”submit”

                    class=”bg-laravel text-white rounded py-2 px-4 hover:bg-black”

                >

                    Registrujte se

                </button>

            </div>

            <div class=”mt-8”>

                <p>

                    Već ste registrovani?

                    <a href=”/login” class=”text-laravel”

                        >Ulogujte se</a

                    >

                </p>

            </div>

        </form>

    </div>

</div>

@endsection

*Listing 9. listing koda “register.blade.php”*

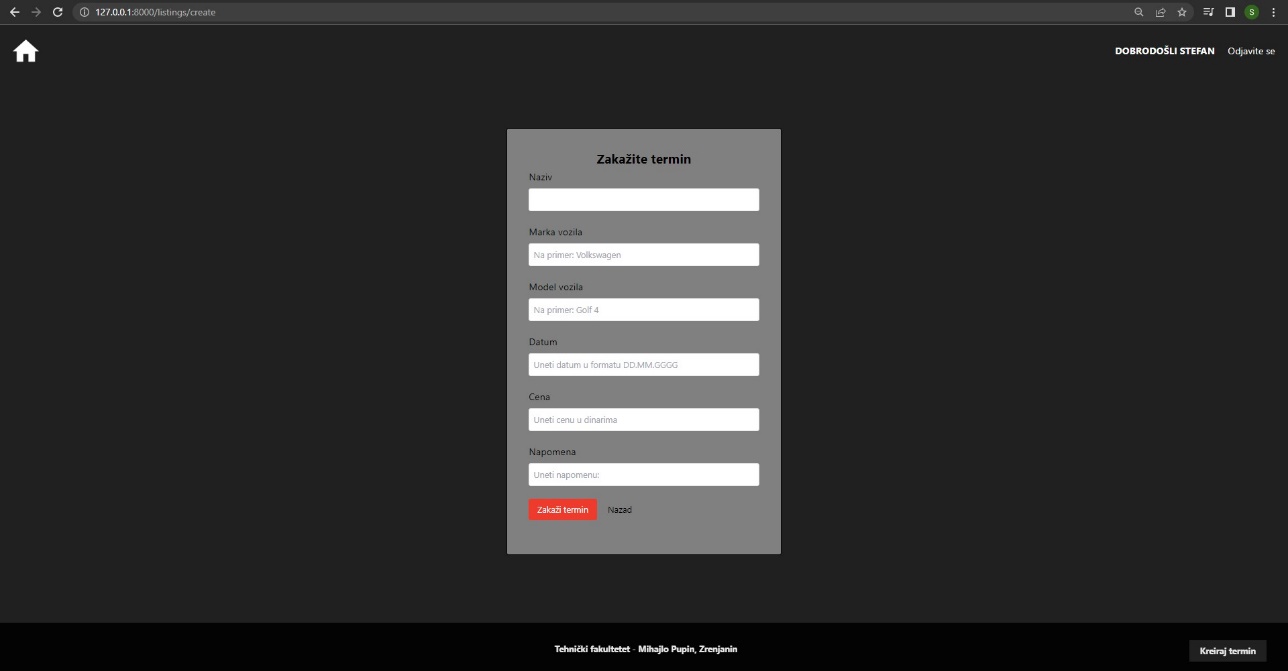
### UI stranica za omogućavanje CRUD sistema korisnicima

Najpre kreiramo folder pod nazivom “listings”.

#### **4.5.1 Stranica za unos termina**

Unutar foldera “listings” kreiramo fajl pod nazivom “create.blade.php”

Sličnog tipa kao i kao stranice za login i registraciju ovaj fajl mora sadržati formu za unos podataka o terminu.



*Slika 5. UI stranice za dodavanje termina.*

Listing stranice za dodavanje termina:

@extends('layout')

@section('content')

<div class="bg-lightgrey border border-black p-10 rounded max-w-lg mx-auto mt-24">

                    <header class="text-center">

                        <h2 class="text-2xl font-bold mb-1">

                            Zakažite termin

                        </h2>

                    </header>

                    <form method="POST" action="/listings">

                        @csrf

                        <div class="mb-6">

                            <label

                                for="title"

                                class="inline-block text-lg mb-2"

                                >Naziv</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="title"

                            />

                        </div>

                        <div class="mb-6">

                            <label for="make" class="inline-block text-lg mb-2"

                                >Marka vozila</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="make"

                                placeholder="Na primer: Volkswagen"

                            />

                        </div>

                        <div class="mb-6">

                            <label

                                for="model"

                                class="inline-block text-lg mb-2"

                                >Model vozila</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="model"

                                placeholder="Na primer: Golf 4"

                            />

                        </div>

                        <div class="mb-6">

                            <label for="date" class="inline-block text-lg mb-2"

                                >Datum</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="date"

                                placeholder="Uneti datum u formatu DD.MM.GGGG"

                            />

                        </div>

                        <div class="mb-6">

                            <label for="price" class="inline-block text-lg mb-2"

                                >Cena</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="price"

                                placeholder="Uneti cenu u dinarima"

                            />

                        </div>

                        <div class="mb-6">

                            <label

                                for="note"

                                class="inline-block text-lg mb-2"

                            >

                                Napomena

                            </label>

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="note"

                                placeholder="Uneti napomenu:"

                            >

                        </div>

                        <div class="mb-6">

                            <button

                                class="bg-laravel text-white rounded py-2 px-4 hover:bg-black"

                                onclick="store()"

                            >

                                Zakaži termin

                            </button>

                            <a href="/" class="text-black ml-4"> Nazad </a>

                        </div>

                    </form>

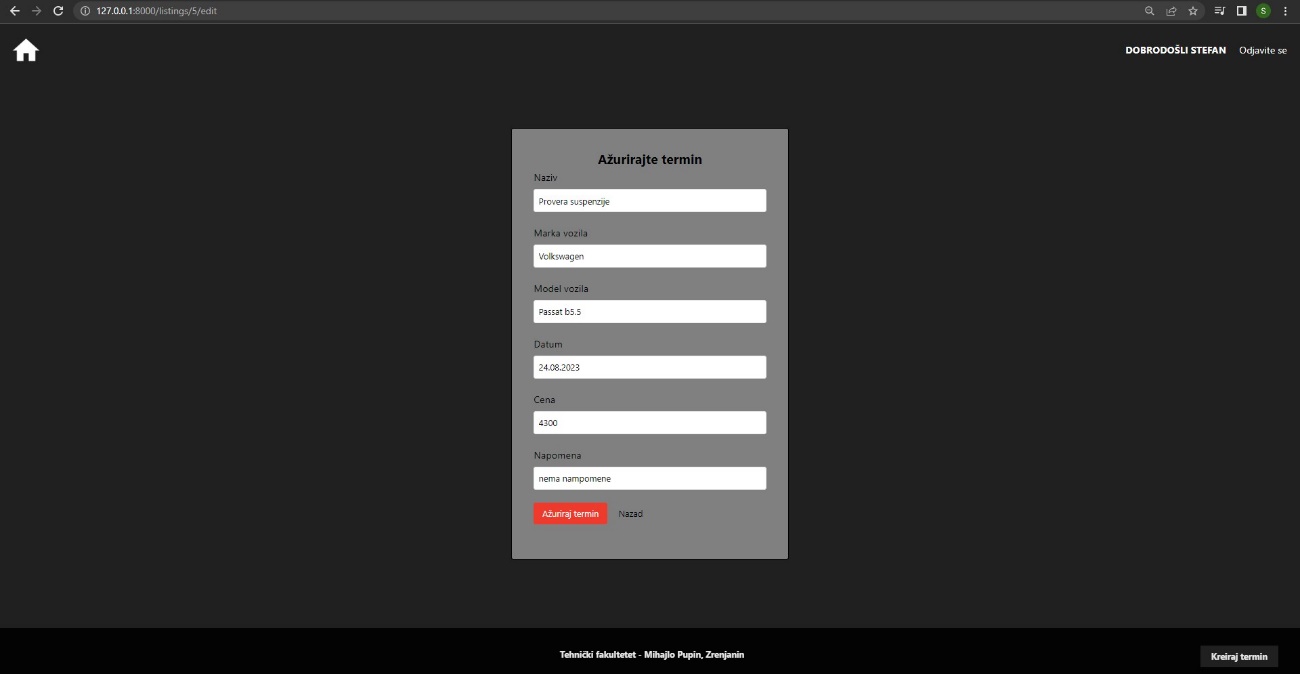
                </div>

                @endsection

*Listing 10. listing “create.blade.php”*

#### **4.5.2 Stranica za ažuriranje termina**

Ova stranica je gotovo identična kao stranica za kreiranje termina sa tim sto su input elementi prethodno podešeni na prethodnu vrednost.



*Slika 5. UI stranice za ažiriranje termina.*

Listing koda za stranicu za ažuriranje:

@extends('layout')

@section('content')

<div class="bg-lightgrey border border-black p-10 rounded max-w-lg mx-auto mt-24">

                    <header class="text-center">

                        <h2 class="text-2xl font-bold mb-1">

                            Ažurirajte termin

                        </h2>

                    </header>

                    <form method="POST" action="/listings/{{$listing->id}}">

                        @csrf

                        @method('PUT')

                        <div class="mb-6">

                            <label

                                for="title"

                                class="inline-block text-lg mb-2"

                                >Naziv</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="title"

                                value="{{$listing->title}}"

                            />

                        </div>

                        <div class="mb-6">

                            <label for="make" class="inline-block text-lg mb-2"

                                >Marka vozila</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="make"

                                placeholder="Na primer: Volkswagen"

                                value="{{$listing->make}}"

                            />

                        </div>

                        <div class="mb-6">

                            <label

                                for="model"

                                class="inline-block text-lg mb-2"

                                >Model vozila</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="model"

                                placeholder="Na primer: Golf 4"

                                value="{{$listing->model}}"

                            />

                        </div>

                        <div class="mb-6">

                            <label for="date" class="inline-block text-lg mb-2"

                                >Datum</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="date"

                          placeholder="Uneti datum u formatu DD.MM.GGGG"

                                value="{{$listing->date}}"

                            />

                        </div>

                        <div class="mb-6">

                            <label for="price" class="inline-block text-lg mb-2"

                                >Cena</label

                            >

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="price"

                                placeholder="Uneti cenu u dinarima"

                                value="{{$listing->price}}"

                            />

                        </div>

                        <div class="mb-6">

                            <label

                                for="note"

                                class="inline-block text-lg mb-2"

                            >

                                Napomena

                            </label>

                            <input

                                type="text"

                                class="border border-gray-200 rounded p-2 w-full"

                                name="note"

                                placeholder="Include tasks, requirements, salary, etc"

                                value="{{$listing->note}}"

                            >

                        </div>

                        <div class="mb-6">

                            <button

                                class="bg-laravel text-white rounded py-2 px-4 hover:bg-black"

                                onclick="store()"

                            >

                                Ažuriraj termin

                            </button>

                            <a href="/" class="text-black ml-4"> Nazad </a>

                        </div>

                    </form>

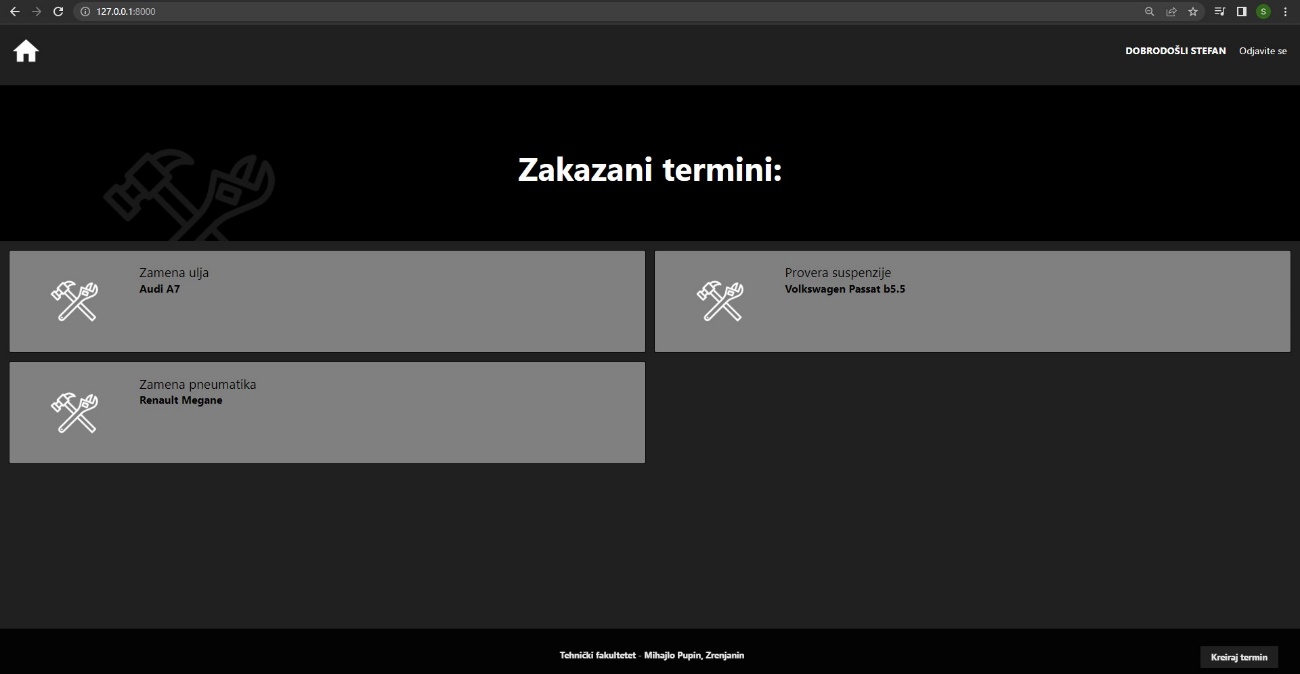
                </div>

                @endsection

*Listing 11. listing koda stranice za ažuriranje*

#### **4.5.3 Stranica za prikazivanje termina na početnoj stranici**

Zapravo ovaj fajl sadži forpetlju preko koje prolazi kroz svaki zakani termin i prikazuje ga na početnu stranicu, tj. samo naziv i model automobila iz termina, ali samo ako je korisnik ulogovan.



*Slika 7. UI početne stranice nakon unetih termina i nakon potvrdne autentifikacije*

Listing koda za prikaz zakazanih termina na početnu stranicu:

@extends('layout')

@section('content')

@include('partials.\_cover')

<div class="lg:grid lg:grid-cols-2 gap-4 space-y-4 md:space-y-0 mx-4">

@unless (count($listings) == 0)

@auth

@foreach($listings as $listing)

<div class="bg-lightgrey border border-black rounded p-6">

    <div class="flex">

        <img

            class="hidden w-48 mr-6 md:block"

            src="{{asset('images/laravel-logo.png')}}"

            alt=""

        />

        <div>

            <h3 class="text-2xl">

                <a href="/listings/{{$listing->id}}">{{$listing->title}}</a>

            </h3>

            <div class="text-xl font-bold mb-4">{{$listing->make}} {{$listing->model}}</div>

        </div>

    </div>

</div>

@endforeach

@endauth

@else

<p>Not Found</p>

@endunless

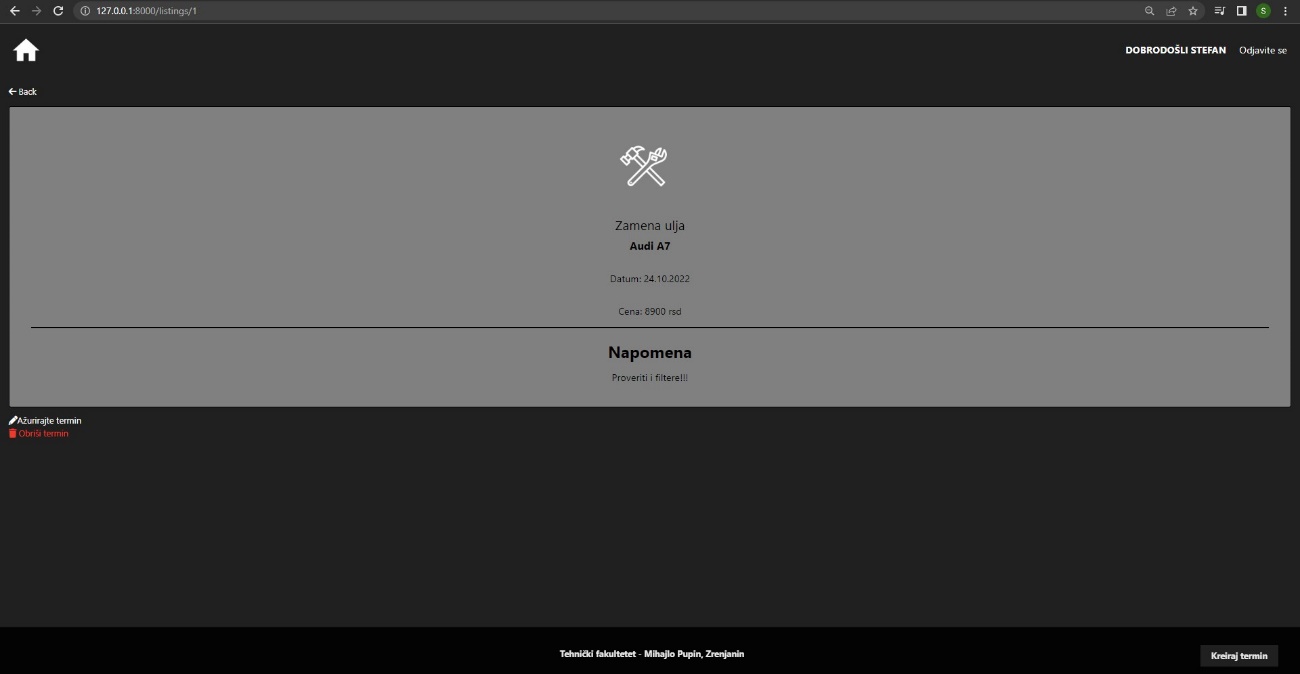
</div>

@endsection

*Listing 12. listing koda “index.blade.php”*

#### **4.5.3 Stranica za prikazivanje termina u zasebnoj stranici**

Ova stranica služi za pun prikaz termina tj. svih njegovih podataka odredjenog termina koji se odabere sa početne stranice i uz to sadrži dva dugmeta od kojih jedno vodi na stranicu za ažuriranje tog termina, dok drugo služi za brisanje tog termina.

****

*Slika 8. UI stranice za potpun prikaz određenog termina.*

Listing koda za stranicu za potpuno prikazivanje određenog termina:

@extends('layout')

@section('content')

<a href="/" class="inline-block text-white ml-4 mb-4"

                ><i class="fa-solid fa-arrow-left"></i> Back

            </a>

            <div class="mx-4">

                <div class="bg-lightgrey border border-black p-10 rounded">

                    <div

                        class="flex flex-col items-center justify-center text-center"

                    >

                        <img

                            class="w-48 mr-6 mb-6"

                            src="{{asset('images/laravel-logo.png')}}"

                            alt=""

                        />

                        <h3 class="text-2xl mb-2">{{$listing->title}}</h3>

                        <div class="text-xl font-bold mb-4">{{$listing->make}} {{$listing->model}}</div>

                        <div class="text-lg my-4">

                            Datum: {{$listing->date}}

                        </div>

                        <div class="text-lg my-4">

                            Cena: {{$listing->price}} rsd

                        </div>

                        <div class="border border-black w-full mb-6"></div>

                        <div>

                            <h3 class="text-3xl font-bold mb-4">

                                Napomena

                            </h3>

                            <div class="text-lg space-y-6">

                                <p>

                                    {{$listing->note}}

                                </p>

                            </div>

                        </div>

                    </div>

                </div>

            </div>

@auth

<div class="text-white mt-3 ml-4">

    <a href="/listings/{{$listing->id}}/edit">

    <i class="fa-solid fa-pencil"></i>Ažurirajte termin</a>

<form method="POST" action="/listings/{{$listing->id}}">

    @csrf

    @method('DELETE')

    <button onclick="destroy()" class="text-laravel"><i class="fa-solid fa-trash mr-1"></i>Obriši termin</button>

</form>

</div>

@endauth

@endsection

*Listing 13. listing koda fajla “show.blade.php”*

### Kreiranje ruta

Naša aplikacija sada ima sve potrebne elemente da bi bila funkcionalna ostalo je još samo konfigurisanje potrebnih ruta sa stranica koje sadrže dugmad ili druge elemente koji pozivaju prethodno definisane funkcije iz kontrolera.

Pristupamo fajlu “web.php” koji se nalazi u “routes” folderu, u njemu je potrebno dodati rute koje odgovaraju logici naše aplikacije.

Listing ruta aplikacije:

<?php

use App\Models\Listing;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\UserController;

use App\Http\Controllers\ListingController;

/\*

|--------------------------------------------------------------------------

| 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('/', [ListingController::class, 'index']);

Route::get('/listings/create', [ListingController::class, 'create'])->middleware('auth');

Route::post('/listings', [ListingController::class, 'store'])->middleware('auth');

Route::get('/listings/{listing}/edit', [ListingController::class, 'edit'])->middleware('auth');

Route::put('/listings/{listing}', [ListingController::class, 'update'])->middleware('auth');

Route::delete('/listings/{listing}', [ListingController::class, 'destroy'])->middleware('auth');

Route::get('/listings/{listing}', [ListingController::class, 'show']);

Route::get('/register', [UserController::class, 'create'])->middleware('guest');

Route::post('/users', [UserController::class, 'store']);

Route::post('/logout', [UserController::class, 'logout'])->middleware('auth');

Route::get('/login', [UserController::class, 'login'])->name('login')->middleware('guest');

Route::post('/users/authenticate', [UserController::class, 'authenticate']);

*Listing 14. listing fajla “web.php”*

## **Zaključak**

Internet alati i servisi nude pregršt korisnih opcija i mogućnosti koje se mogu koristiti svakodnevno u privatne i biznis shvrhe, primena ovih tehnologija je enormna u odnosu na svaku ostalu, gotovo 80% stanovništva ra;unar na kome može za tren oka da prisupi i koristi aplikaciju razvijenu ovim tehnologijama. Konkretno PHP/laravel tehnologija sa MySQL je olakšala programerima još više i oslobodila ih neophodnih stvari za funkcionisanje aplikacija.

Cilj ovog seminarskog rada je bio da se obuhvate osnove za kreiranje dobre web aplikacije koja može imati široku primenu.

Dovoljno je samo biti upoznat sa ovakvom vrstom tehnologije i imati dobru ideju da bi se postigle velike stvari.

## **Literatura**

Literatura:

1. Robin Nikson - Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites, 2012

2. Met Štaufer - Laravel: Up & Running: A Framework for Building Modern PHP Apps, 2020

3. Rasmus Lerdorf - Programming PHP, 2008

Korišćeni web sajtovi:

1. <http://www.tfzr.uns.ac.rs/Predmet/internet-alati-i-servisi>

2. <https://www.php.net/docs.php>

3. <https://laravel.com/docs/9.x/>

4. <https://code.visualstudio.com/docs>

## **Izvori preuzetih ilustracija**

|  |  |
| --- | --- |
| Slika 1 | https://download.logo.wine/logo/MySQL/MySQL-Logo.wine.png |