Univerzitet u Beogradu

Fakultet organizacionih nauka

Katedra za elektronsko poslovanje

Laravel

Domaći zadatak 2

|  |  |  |
| --- | --- | --- |
| Ime | Prezime | Broj indeksa |
|  |  |  |
| Link ka Github-u |  | |

Sadržaj

[1 Korisnički zahtev 3](#_Toc154095131)

[2 Implementacija 4](#_Toc154095132)

[2.1 Model 4](#_Toc154095133)

[2.2 Migracije 4](#_Toc154095134)

[2.3 Kontroleri 6](#_Toc154095135)

[2.4 Api rute 7](#_Toc154095136)

[3 REST API 8](#_Toc154095137)

[4 Korisničko uputstvo 17](#_Toc154095138)

# Korisnički zahtev

# Implementacija

## Model

U aplikaciji imamo sledece klase

1. **ZdravstveniKarton**: Predstavlja zdravstveni karton korisnika, sadrži informacije kao što su istorija bolesti, alergije i prethodni pregledi. Povezan je sa korisnikom kroz **korisnik\_id**.

Slika na kojoj se nalazi tekst, snimak ekrana, ekran, softver

Opis je automatski generisan

1. **Anketa**: Opisuje anketu sa atributima poput naslova, opisa, datuma početka i kraja, i statusa. Sadrži veze sa odgovorima ankete i pitanjima ankete.

Slika na kojoj se nalazi tekst, snimak ekrana, Font

Opis je automatski generisan

1. **OdgovorAnkete**: Sadrži informacije o odgovorima na anketu, uključujući korisnika koji odgovara, pitanje ankete i sam odgovor. Povezan je sa anketom, korisnikom i pitanjem ankete.

Slika na kojoj se nalazi tekst, snimak ekrana, softver

Opis je automatski generisan

1. **PitanjeAnkete**: Definiše pitanja unutar ankete.

Slika na kojoj se nalazi tekst, snimak ekrana, ekran, softver

Opis je automatski generisan

1. **User**: Opisuje korisnika, verovatno sa standardnim atributima kao što su ime, email, lozinka itd. Povezan je sa više različitih klasa kao što su ZdravstveniKarton, Zakazivanje, Zahtev, i OdgovorAnkete.

Slika na kojoj se nalazi tekst, snimak ekrana, softver, Multimedijski softver

Opis je automatski generisan

1. **Zahtev**: Predstavlja zahtev korisnika, sa atributima poput tipa zahteva, opisa, statusa, datuma podnošenja, i odgovora na zahtev. Povezan je sa korisnikom.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, softver

Opis je automatski generisan

1. **Zakazivanje**: Opisuje zakazivanje, uključujući datum i vreme, tip pregleda, status i napomenu. Takođe je povezano sa korisnikom.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, ekran

Opis je automatski generisan

## Migracije

U nasoj aplikaciji imamo sledece migracije

1. Migracija koja menja ime kolone 'name' u 'imePrezime' u tabeli 'users'. Ovo omogućava jasniju identifikaciju korisnika kroz njegovo puno ime i prezime.

Slika na kojoj se nalazi tekst, snimak ekrana, Font

Opis je automatski generisan

1. Dodaje dodatna polja u tabelu 'users', uključujući datum rođenja, pol, adresu, grad, državu, telefon, JMBG, broj lične karte, status, ulogu, sliku profila, i vreme poslednjeg prijavljivanja. Ovo proširuje informacije dostupne za svakog korisnika.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, softver

Opis je automatski generisan

1. Kreira tabelu 'ankete' koja sadrži detalje o anketama kao što su naslov, opis, datum početka i kraja, i status. Ova tabela omogućava organizovanje anketa unutar sistema.

Slika na kojoj se nalazi tekst, snimak ekrana, ekran, softver

Opis je automatski generisan

1. Kreira tabelu 'zahtevi' koja uključuje informacije o različitim zahtevima korisnika, kao što su tip zahteva, opis, status, datum podnošenja, i odgovor. Ovo omogućava efikasnije upravljanje korisničkim zahtevima.

Slika na kojoj se nalazi tekst, snimak ekrana, ekran, softver

Opis je automatski generisan

1. Tabela 'zakazivanje' koja omogućava korisnicima da zakažu sastanke ili pregled, sa detaljima kao što su datum i vreme, tip pregleda, status i napomena.

Slika na kojoj se nalazi tekst, snimak ekrana, ekran

Opis je automatski generisan

1. Kreira tabelu 'zdravstveni\_karton' koja sadrži ključne zdravstvene informacije korisnika, uključujući istoriju bolesti, alergije i prethodne preglede.

Slika na kojoj se nalazi tekst, snimak ekrana, Font

Opis je automatski generisan

1. Uklanja kolonu 'doktor\_id' iz tabele 'zakazivanjes', verovatno zbog reorganizacije načina na koji se zakazivanja upravljaju ili povezuju sa doktorima.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, linija

Opis je automatski generisan

1. Kreira tabelu 'pitanje\_anketes' koja sadrži pitanja koja se postavljaju u okviru anketa. Ovo omogućava dinamično kreiranje anketa sa različitim pitanjima.

Slika na kojoj se nalazi tekst, snimak ekrana, Font

Opis je automatski generisan

1. Uvođenje tabele 'odgovori\_ankete' koja beleži odgovore korisnika na pitanja iz anketa, uključujući informacije o anketi, korisniku, pitanju ankete i samom odgovoru.

Slika na kojoj se nalazi tekst, snimak ekrana, Font

Opis je automatski generisan

1. Dodaje strane ključeve u tabelu 'odgovori\_ankete', povezujući je sa tabelama 'anketas', 'users', i 'pitanje\_anketes'. Ovo osigurava integritet podataka i omogućava povezivanje odgovora sa konkretnim anketama, korisnicima i pitanjima. ​

Slika na kojoj se nalazi tekst, snimak ekrana, Font

Opis je automatski generisan

## Kontroleri

Kod ispod predstavlja kontroler nazvan AnketaController, koji upravlja operacijama povezanim sa anketama. U okviru kontrolera, implementirane su metode koje omogućavaju različite operacije sa anketama, kao što su pregled, kreiranje, ažuriranje, brisanje i izvoz podataka ankete.

Prva metoda, index(), dohvata sve ankete iz baze podataka i vraća ih u JSON formatu. Slično tome, show($id) metoda dohvata pojedinačnu anketu na osnovu njenog identifikacionog broja (ID) i takođe vraća podatke u JSON formatu. Ove metode omogućavaju pregled svih anketa, odnosno pojedinačnih anketa.

Metoda store(Request $request) služi za kreiranje nove ankete. Prvo se vrši validacija unetih podataka koristeći Laravelov Validator, gde se proverava da li su sva neophodna polja popunjena i da li su u odgovarajućem formatu. U slučaju da validacija ne prođe, vraća se greška. Ako je validacija uspešna, kreira se nova anketa sa unetim podacima i vraća se u JSON formatu. Metoda update(Request $request, $id) funkcioniše na sličan način, ali ažurira postojeću anketu na osnovu njenog ID-a.

Metoda destroy($id) omogućava brisanje ankete na osnovu njenog ID-a. Nakon što se anketu pronađe i obriše, vraća se odgovor sa statusom 204, što označava uspešno brisanje bez povratnih podataka.

Metoda getAnketaWithPitanjaIOdgovori($id) je posebno zanimljiva jer dohvata detalje o anketi zajedno sa pitanjima i odgovorima vezanim za tu anketu. Koristi Eloquentovu metodu with() za učitavanje povezanih modela, što omogućava dohvatanje složenih podataka u jednom upitu.

Poslednja metoda, exportAnketaToCsv($id), izvozi podatke ankete u CSV formatu. Prvo dohvata anketu sa pitanjima i odgovorima, zatim kreira CSV datoteku sa tim podacima, uključujući zaglavlje sa osnovnim informacijama o anketi i detalje o svakom pitanju i odgovoru. Datoteka se čuva na definisanu lokaciju i na kraju se vraća odgovor sa porukom o uspešnom sačuvanju i putanjom do CSV datoteke.

<?php

namespace App\Http\Controllers;

use App\Models\Anketa;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Validator;

use Illuminate\Support\Facades\Response;

class AnketaController extends Controller

{

    public function index()

    {

        $ankete = Anketa::all();

        return response()->json($ankete);

    }

    public function show($id)

    {

        $anketa = Anketa::findOrFail($id);

        return response()->json($anketa);

    }

    public function store(Request $request)

    {

        $validator = Validator::make($request->all(), [

            'naslov' => 'required|string|max:255',

            'opis' => 'required|string',

            'datum\_pocetka' => 'required|date',

            'datum\_kraja' => 'required|date',

            'status' => 'required|string|max:255',

        ]);

        if ($validator->fails()) {

            return response()->json($validator->errors(), 400);

        }

        $anketa = Anketa::create($validator->validated());

        return response()->json($anketa, 201);

    }

    public function update(Request $request, $id)

    {

        $anketa = Anketa::findOrFail($id);

        $validator = Validator::make($request->all(), [

            'naslov' => 'required|string|max:255',

            'opis' => 'required|string',

            'datum\_pocetka' => 'required|date',

            'datum\_kraja' => 'required|date',

            'status' => 'required|string|max:255',

        ]);

        if ($validator->fails()) {

            return response()->json($validator->errors(), 400);

        }

        $anketa->update($validator->validated());

        return response()->json($anketa);

    }

    public function destroy($id)

    {

        $anketa = Anketa::findOrFail($id);

        $anketa->delete();

        return response()->json(null, 204);

    }

    public function getAnketaWithPitanjaIOdgovori($id)

    {

        // Pronađi anketu sa datim ID-om i učitaj pitanja i odgovore

        $anketa = Anketa::with(['pitanja', 'pitanja.odgovori'])

            ->findOrFail($id);

        return response()->json($anketa);

    }

    public function exportAnketaToCsv($id)

    {

        $anketa = Anketa::with(['pitanja', 'pitanja.odgovori'])->findOrFail($id);

        $csvData = [];

        // Dodavanje podataka ankete u zaglavlje

        $csvData[] = [

            'ID Ankete', $anketa->id,

            'Naslov', $anketa->naslov,

            'Opis', $anketa->opis,

            'Datum Početka', $anketa->datum\_pocetka,

            'Datum Kraja', $anketa->datum\_kraja,

            'Status', $anketa->status

        ];

        // Dodavanje prazne linije između zaglavlja i pitanja

        $csvData[] = [];

        // Dodavanje naslova za pitanja i odgovore

        $csvData[] = ['ID Pitanja', 'Tekst Pitanja', 'ID Odgovora', 'Odgovor'];

        foreach ($anketa->pitanja as $pitanje) {

            if (count($pitanje->odgovori) > 0) {

                foreach ($pitanje->odgovori as $odgovor) {

                    $csvData[] = [

                        $pitanje->id,

                        $pitanje->tekst,

                        $odgovor->id,

                        $odgovor->odgovor,

                    ];

                }

            } else {

                $csvData[] = [

                    $pitanje->id,

                    $pitanje->tekst,

                    '', // ID odgovora

                    '', // Odgovor

                ];

            }

        }

        $fileName = 'anketa\_' . $anketa->id . '.csv';

        $filePath = storage\_path('/app/public/ankete\_csv/' . $fileName);

        $directory = dirname($filePath);

        if (!is\_dir($directory)) {

            mkdir($directory, 0777, true);

        }

        $file = fopen($filePath, 'w');

        foreach ($csvData as $row) {

            fputcsv($file, $row);

        }

        fclose($file);

        return response()->json(['message' => 'CSV file saved', 'path' => $filePath]);

    }

}

AuthController upravlja autentifikacijom i registracijom korisnika. Metoda register služi za registraciju novog korisnika, gde se prvo vrši validacija unetih podataka koristeći Laravelov Validator. Ukoliko validacija prođe, kreira se novi korisnik sa šifrovanom lozinkom i, ako je dostavljena, sa slikom profila. Ukoliko validacija ne uspe, vraća se greška. Metoda login omogućava korisniku da se uloguje, verifikujući unete kredencijale (e-mail i lozinku). Ako su kredencijali tačni, korisniku se dodeljuje token za autentifikaciju. U suprotnom, vraća se poruka o neautorizovanom pristupu. Metoda logout omogućava odjavu trenutno autentifikovanog korisnika, brišući njegov trenutni token za pristup. Na kraju, metoda userProfile vraća podatke o profilu trenutno autentifikovanog korisnika. Ovaj kontroler je ključan za upravljanje procesima registracije, prijave, odjave i pristupa profilu korisnika, što je osnovni aspekt bezbednosti i korisničkog iskustva u web aplikacijama.

<?php

namespace App\Http\Controllers;

use App\Models\User;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth;

use Illuminate\Support\Facades\Validator;

class AuthController extends Controller

{

    /\*\*

     \* Register a new user.

     \*

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

     \* @return \Illuminate\Http\JsonResponse

     \*/

    public function register(Request $request)

    {

        $validator = Validator::make($request->all(), [

            'imePrezime' => 'required|string|max:255',

            'email' => 'required|string|email|unique:users,email',

            'password' => 'required|string|min:6',

            'datum\_rodjenja' => 'required|date',

            'pol' => 'required|string|max:255',

            'adresa' => 'required|string|max:255',

            'grad' => 'required|string|max:255',

            'drzava' => 'required|string|max:255',

            'telefon' => 'required|string|max:255',

            'JMBG' => 'required|string|max:255|unique:users,JMBG',

            'broj\_licne\_karte' => 'required|string|max:255|unique:users,broj\_licne\_karte',

            'status' => 'required|string|max:255',

            'uloga' => 'required|string|max:255',

            'slika\_profila' => 'nullable|image|max:2048',

        ]);

        if ($validator->fails()) {

            return response()->json($validator->errors(), 400);

        }

        $pathSlike = null;

        if ($request->hasFile('slika\_profila')) {

            $pathSlike = $request->file('slika\_profila')->store('slike\_profila', 'public');

        }

        $user =  User::create([

            'imePrezime' => $request->input('imePrezime'),

            'email' => $request->input('email'),

            'password' => bcrypt($request->input('password')),

            'datum\_rodjenja' => $request->input('datum\_rodjenja'),

            'pol' => $request->input('pol'),

            'adresa' => $request->input('adresa'),

            'grad' => $request->input('grad'),

            'drzava' => $request->input('drzava'),

            'telefon' => $request->input('telefon'),

            'JMBG' => $request->input('JMBG'),

            'broj\_licne\_karte' => $request->input('broj\_licne\_karte'),

            'status' => $request->input('status'),

            'uloga' => $request->input('uloga'),

            'slika\_profila' => $pathSlike,

        ]);

        return response()->json(['message' => 'Registration successful'], 201);

    }

    /\*\*

     \* Log in a user and issue a token.

     \*

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

     \* @return \Illuminate\Http\JsonResponse

     \*/

    public function login(Request $request)

    {

        $credentials = $request->only('email', 'password');

        if (Auth::attempt($credentials)) {

            $token = Auth::user()->createToken('auth-token')->plainTextToken;

            return response()->json(['token' => $token]);

        }

        return response()->json(['message' => 'Unauthorized'], 401);

    }

    /\*\*

     \* Log out the authenticated user.

     \*

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

     \* @return \Illuminate\Http\JsonResponse

     \*/

    public function logout(Request $request)

    {

        $request->user()->currentAccessToken()->delete();

        return response()->json(['message' => 'Logged out']);

    }

    public function userProfile(Request $request)

        {

            $user = Auth::user(); // Dobijamo trenutno ulogovanog korisnika

            return response()->json($user);

        }

}

## Api rute

Prvo su definisane rute za AnketaController, gde ruta /ankete/export/{id} omogućava izvoz podataka ankete u CSV format, dok ruta /ankete/{id}/pitanjaodgovori omogućava dohvatanje anketa sa pripadajućim pitanjima i odgovorima. Osim toga, postoji i ruta za pretragu zakazivanja /zakazivanja/search, koja omogućava filtriranje zakazivanja na osnovu određenih kriterijuma, kao što su identifikacioni broj korisnika i status zakazivanja.

Za autentifikaciju korisnika, postoje rute /register i /login, koje koristi AuthController. Ruta /register se koristi za registraciju novih korisnika, dok se ruta /login koristi za prijavljivanje korisnika u sistem. Nakon uspešne autentifikacije, korisnik dobija token koji se koristi za pristup zaštićenim rutama.

Zaštićene rute su grupisane unutar middleware koji zahteva autentifikaciju. U ovom delu, ruta user-profile omogućava korisniku da pristupi svojim profilnim informacijama. Dodatno, postoje posebne rute za administratore i obične korisnike. Administratori imaju pristup ruti /admin, koja omogućava upravljanje anketama, pitanjima i odgovorima. Za obične korisnike, ruta /user omogućava pristup samo osnovnim funkcijama, poput pregleda anketa i zakazivanja.

Na kraju, definisana je ruta za odjavu /logout, koja omogućava korisnicima da se odjave iz sistema i time invalidiraju svoj trenutni token.

# REST API

U ovom poglavlju ce biti prikazani detalji o api rutama za kontrolere.

|  |  |
| --- | --- |
| Opis funkcije | Prikaz svih anketa sa svim pitanjima I odgovorima |
| HTTP metoda | GET |
| URL | /api/ankete/1/pitanjaodgovori |
| URL parametri | nema |
| HTTP body parametri | nema |
| Format HTTP body parametara | JSON |
| Izlazni parametri | {      "id": 1,      "naslov": "Anketa o javnom mnenju",      "opis": "Ova anketa ima za cilj ispitivanje javnog mnenja o aktuelnim pitanjima.",      "datum\_pocetka": "2024-01-29 23:46:06",      "datum\_kraja": "2024-02-05 23:46:06",      "status": "otvoreno",      "created\_at": "2024-01-29T23:46:06.000000Z",      "updated\_at": "2024-01-29T23:46:06.000000Z",      "pitanja": [          {              "id": 1,              "anketa\_id": 1,              "tekst": "Kako ocenjujete rad lokalne samouprave?",              "created\_at": "2024-01-29T23:46:06.000000Z",              "updated\_at": "2024-01-29T23:46:06.000000Z",              "odgovori": [                  {                      "id": 2,                      "anketa\_id": 1,                      "korisnik\_id": 5,                      "pitanje\_ankete\_id": 1,                      "odgovor": "da",                      "created\_at": **null**,                      "updated\_at": **null**                  },                  {                      "id": 3,                      "anketa\_id": 1,                      "korisnik\_id": 2,                      "pitanje\_ankete\_id": 1,                      "odgovor": "da",                      "created\_at": **null**,                      "updated\_at": **null**                  }              ]          },          {              "id": 2,              "anketa\_id": 1,              "tekst": "Da li podržavate mere za zaštitu životne sredine?",              "created\_at": "2024-01-29T23:46:07.000000Z",              "updated\_at": "2024-01-29T23:46:07.000000Z",              "odgovori": [                  {                      "id": 1,                      "anketa\_id": 1,                      "korisnik\_id": 5,                      "pitanje\_ankete\_id": 2,                      "odgovor": "da",                      "created\_at": **null**,                      "updated\_at": **null**                  }              ]          },          {              "id": 3,              "anketa\_id": 1,              "tekst": "Koje su vaše glavne brige u vezi sa infrastrukturom?",              "created\_at": "2024-01-29T23:46:07.000000Z",              "updated\_at": "2024-01-29T23:46:07.000000Z",              "odgovori": []          }      ]  } |
| Format izlaznih parametara | Application/json |
|  |  |
| Opis funkcije | Kreiranje ankete |
| HTTP metoda | POST |
| URL | /api/ankete |
| URL parametri | naslov=anketa1&opis=opis&datum\_pocetka=2024-01-01&datum\_kraja=2024-06-01&status=otvoreno |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "naslov": "anketa1",      "opis": "opis",      "datum\_pocetka": "2024-01-01",      "datum\_kraja": "2024-06-01",      "status": "otvoreno",      "updated\_at": "2024-01-30T17:37:10.000000Z",      "created\_at": "2024-01-30T17:37:10.000000Z",      "id": 3  } |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Ažuriranje ankete |
| HTTP metoda | PUT |
| URL | /api/ankete |
| URL parametri | 3?naslov=anketa1&opis=opis&datum\_pocetka=2024-01-01&datum\_kraja=2024-06-01&status=otvoreno |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "id": 3,      "naslov": "anketa1",      "opis": "opis",      "datum\_pocetka": "2024-01-01",      "datum\_kraja": "2024-06-01",      "status": "otvoreno",      "created\_at": "2024-01-30T17:37:10.000000Z",      "updated\_at": "2024-01-30T17:37:47.000000Z"  } |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Brisanje ankete |
| HTTP metoda | DELETE |
| URL | /api/ankete |
| URL parametri | 3 |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | -No Content- |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Prikaz svih pitanja |
| HTTP metoda | GET |
| URL | /api/pitanja |
| URL parametri | nema |
| HTTP body parametri | nema |
| Format HTTP body parametara | JSON |
| Izlazni parametri | [      {          "id": 1,          "anketa\_id": 1,          "tekst": "Kako ocenjujete rad lokalne samouprave?",          "created\_at": "2024-01-29T23:46:06.000000Z",          "updated\_at": "2024-01-29T23:46:06.000000Z"      },      {          "id": 2,          "anketa\_id": 1,          "tekst": "Da li podržavate mere za zaštitu životne sredine?",          "created\_at": "2024-01-29T23:46:07.000000Z",          "updated\_at": "2024-01-29T23:46:07.000000Z"      },      {          "id": 3,          "anketa\_id": 1,          "tekst": "Koje su vaše glavne brige u vezi sa infrastrukturom?",          "created\_at": "2024-01-29T23:46:07.000000Z",          "updated\_at": "2024-01-29T23:46:07.000000Z"      },      {          "id": 4,          "anketa\_id": 2,          "tekst": "Koja politička stranka ima vašu podršku?",          "created\_at": "2024-01-29T23:46:07.000000Z",          "updated\_at": "2024-01-29T23:46:07.000000Z"      },      {          "id": 5,          "anketa\_id": 2,          "tekst": "Da li verujete u ishod predstojećih izbora?",          "created\_at": "2024-01-29T23:46:07.000000Z",          "updated\_at": "2024-01-29T23:46:07.000000Z"      },      {          "id": 6,          "anketa\_id": 2,          "tekst": "Kako ocenjujete trenutno političko stanje u zemlji?",          "created\_at": "2024-01-29T23:46:07.000000Z",          "updated\_at": "2024-01-29T23:46:07.000000Z"      }  ] |
| Format izlaznih parametara | Application/json |
|  |  |
| Opis funkcije | Kreiranje pitanja |
| HTTP metoda | POST |
| URL | /api/pitanja |
| URL parametri | ?anketa\_id=1&tekst=pitanje za anketu? |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "anketa\_id": "1",      "tekst": "pitanje za anketu?",      "updated\_at": "2024-01-30T17:40:07.000000Z",      "created\_at": "2024-01-30T17:40:07.000000Z",      "id": 7  } |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Ažuriranje pitanja |
| HTTP metoda | PUT |
| URL | /api/ pitanja |
| URL parametri | 7?anketa\_id=1&tekst=pitanje za anketu? |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "id": 7,      "anketa\_id": "1",      "tekst": "pitanje za anketu?",      "created\_at": "2024-01-30T17:40:07.000000Z",      "updated\_at": "2024-01-30T17:40:07.000000Z"  } |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Brisanje pitanja |
| HTTP metoda | DELETE |
| URL | /api/ pitanja |
| URL parametri | 7 |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | --No Content-- |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Registracija korisnika |
| HTTP metoda | POST |
| URL | /api/register |
| URL parametri | imePrezime=korisnik1&email=korisnik11@gmail.com&password=korisnik1&datum\_rodjenja=2000-01-01&pol=Muški&adresa=adresa&drzava=Srbija&telefon=065421241&JMBG=1254789632514&broj\_licne\_karte=14785236&status=Aktivan&uloga=korisnik&grad=beograd |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "message": "Registration successful"  } |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Prijava korisnika |
| HTTP metoda | POST |
| URL | /api/login |
| URL parametri | ?email=korisnik11@gmail.com&password=korisnik1 |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "token": "2|nkLidx2hcaNPUGOt6D370eZS6ZvvY79UbfBghd2w2e4b4428"  } |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Odjava korisnika |
| HTTP metoda | POST |
| URL | /api/logout |
| URL parametri | nema |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | {      "message": "Logged out"  } |
| Format izlaznih parametara | application/json |

# Korisničko uputstvo

Registracija I prijava korisnika

Slika na kojoj se nalazi tekst, snimak ekrana, broj, softver

Opis je automatski generisan

Slika na kojoj se nalazi tekst, snimak ekrana, broj, softver

Opis je automatski generisan

Pregled svih anketa

Slika na kojoj se nalazi tekst, snimak ekrana, broj, softver

Opis je automatski generisan

Kreiranje ankete

Slika na kojoj se nalazi tekst, snimak ekrana, broj, Font

Opis je automatski generisan

Azuriranje ankete

Slika na kojoj se nalazi tekst, snimak ekrana, softver, broj

Opis je automatski generisan

**Brisanje ankete**

Slika na kojoj se nalazi tekst, snimak ekrana, softver, broj

Opis je automatski generisan

Prikaz svih pitanja

Slika na kojoj se nalazi tekst, snimak ekrana, broj, softver

Opis je automatski generisan

Kreiranje pitanja

Slika na kojoj se nalazi tekst, linija, Font, broj

Opis je automatski generisan

Azuriranje pitanja

Slika na kojoj se nalazi tekst, snimak ekrana, softver, Font

Opis je automatski generisan

Brisanje pitanja

Slika na kojoj se nalazi tekst, snimak ekrana, broj, Font

Opis je automatski generisan

Logout

Slika na kojoj se nalazi tekst, snimak ekrana, broj, softver

Opis je automatski generisan