

Pawpal

Tim: TG 13.2 PsićiUOphodnji
Ak. god. 2025./2026.

SADRŽAJ

- Projektni zadatak
- Zaduženja
- Postojeća slična rješenja
- Zahtjevi i UML dijagram obrazaca uporabe
- Arhitektura sustava
- Ispitivanje
- Zaključak

PROJEKTNI ZADATAK

- Sve više vlasnika pasa suočava se s nedostatkom vremena zbog ubrzanog tempa života i poslovnih obaveza
- Riješenje koje olakšava pronalaženje pouzdanih i provjerenih šetača pasa
- Web platforma koja vlasnicima omogućava da jednostavno pronađu šetače u svojoj blizini i rezerviraju termin
- Aplikacija će služiti kao posrednik između vlasnika pasa i šetača, osiguravajući sigurno, jednostavno i transparentno korištenje
- Aplikacija će olakšati organizaciju svakodnevnih obaveza vlasnika, povećavati sigurnost i povjerenje u šetače te poticati razvoj zajednice

ZADUŽENJA

◦ Ema Zidar – voditeljica tima, Frontend + Dizajn

- koordinacija cijelog tima
- planiranje i raspodjela zadataka
- frontend razvoj i dizajn korisničkog sučelja
- pregled i odobravanje pull zahtjeva
- održavanje vizualnog identiteta projekta
- praćenje napretka tima i rokova

◦ Jakov Ramljak – Fronted + Dizajn

- razvoj frontend funkcionalnosti
- implementacija dizajna i UX elemenata
- suradnja s Emom na vizualnom identitetu
- sudjelovanje u code review procesima
- testiranje korisničkog sučelja

◦ Nina Zamberlin – Baza podataka + Dizajn

- projektiranje baze podataka
- implementacija i održavanje schema i tabela
- suradnja na dizajnu aplikacije
- dokumentacija strukture baze
- optimizacija upita i performansi baze

◦ Lana Šapić – Baza podataka + Backend

- razvoj backend funkcionalnosti
- integracija baze podataka s aplikacijom
- pisanje API endpointa
- testiranje backend logike
- dokumentiranje backend procesa

- Luka Zadro – Backend
 - razvoj backend modula i servisa
 - implementacija poslovne logike
 - suradnja s Lanom i Leonardom
 - pisanje testova i debugging
 - održavanje stabilnosti koda
- Leonard Zadro – Backend + Autorizacija
 - implementacija backend modula
 - upravljanje autorizacijom i autentikacijom
 - integracija Oauth 2.0 sustava
 - sigurnosne provjere i kontrola pristupa
 - suradnja s timom na backend arhitekturi
- Viktor Wilder – Frontend
 - razvoj frontend funkcionalnosti
 - implementacija dizajnerskih smjernica
 - sudjelovanje u code review procesu
 - testiranje i optimizacija korisničkog sučelja
 - suradnja s Emom i Jakovom

POSTOJEĆA SLIČNA RJEŠENJA



SLIČNOSTI

- Usluga šetnje pasa
- Sustav rezervacija, ocjena, recenzija i sigurno plaćanje

RAZLIKE

- Rover također nudi i usluge čuvanja te njege pasa

SLIČNOSTI

- Usluga šetnje pasa
- Omogućuje pronalaženje šetača

RAZLIKE

- Primarno tržište je SAD

FUNKCIONALNI ZAHTJEVI

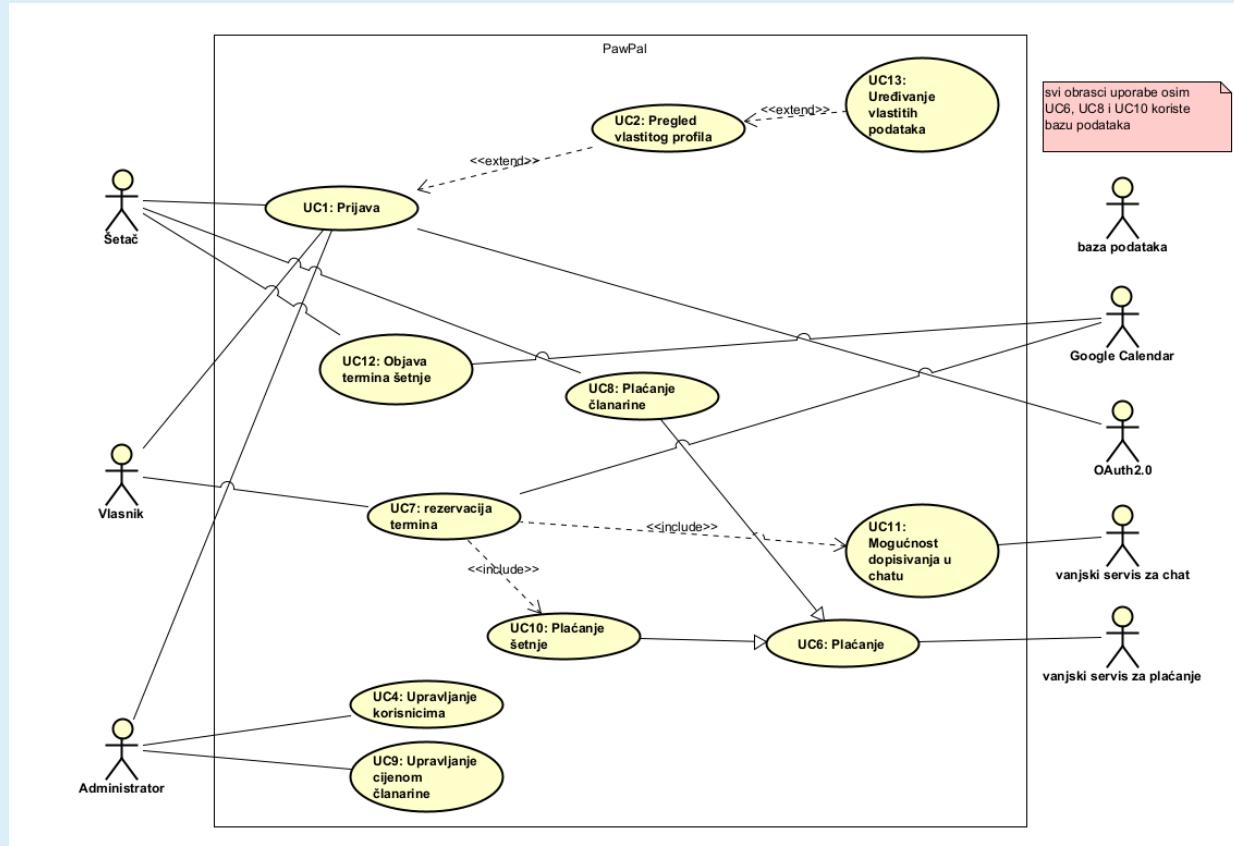
- **25** funkcionalnih zahtjeva
- Broj funkcionalnih zahtjeva po prioritetu
 - **Visoki:** 6
 - **Srednji:** 14
 - **Nizak:** 5

| ID | Naziv zahtjeva | Prioritet | Izvor | Opis / Kriteriji prihvacanja |
|-------|----------------------------|-----------|-----------------|---|
| F-000 | Registracija korisnika | Visok | zahtjev dionika | Korisnik se može registrirati unosom osobnih podataka (ime, prezime, broj mobitela, lozinka) ili putem vanjskog autentifikacijskog servisa. |
| F-001 | Prijava korisnika | Visok | zahtjev dionika | Korisnik se prijavljuje pomocu registriranih vjerodajnica. Sustav provjerava autenticnost i usmjerava korisnika na pocetnu stranicu prema njegovoj ulozi. |
| F-002 | Odabir uloge korisnika | Visok | zahtjev dionika | Tijekom prve registracije korisnik odabire ulogu: šetac ili vlasnik psa. Ova postavka određuje dostupne funkcionalnosti u aplikaciji. |
| F-010 | Plaćanje clanarine (šetac) | Visok | zahtjev dionika | Šetaci placaju clanarinu putem vanjskog servisa (PayPal, kartica). Sustav pohranjuje status clanarine i datum isteka. |
| F-012 | Plaćanje šetnje | Visok | zahtjev dionika | Vlasnik psa placa šetnju gotovinom, karticom ili putem PayPal-a. Sustav bilježi status uplate. |
| F-013 | Rezervacija termina šetnje | Visok | zahtjev dionika | Vlasnik odabire termin u kalendaru šetaca, unosi specifikacije šetnje (datum, vrijeme, trajanje, tip, adresa, napomene) i potvrđuje rezervaciju. |

NEFUNKCIONALNI ZAHTJEVI

| ID zahtjeva | Opis | Prioritet |
|-------------|---|-----------|
| NF-2.1 | Korisnicko sučelje treba biti intuitivno i jednostavno za korištenje za sve tipove korisnika. | Visok |
| NF-3.1 | Sustav treba biti oblikovan tako da omogućuje jednostavno održavanje. | Visok |
| NF-3.2 | Sustav treba imati preciznu dokumentaciju. | Visok |
| NF-3.3 | Sustav treba biti opisan putem dokumenta oblikovanja /SRS/. | Visok |
| NF-4.1 | Aplikacija treba osigurati sigurnost privatnih podataka korisnika. | Visok |
| NF-4.2 | Aplikacija treba biti skalabilna i podržavati proširenje za nove lokacije, korisnicke grupe i objave. | Visok |

UML DIJAGRAM OBRAZACA UPORABE

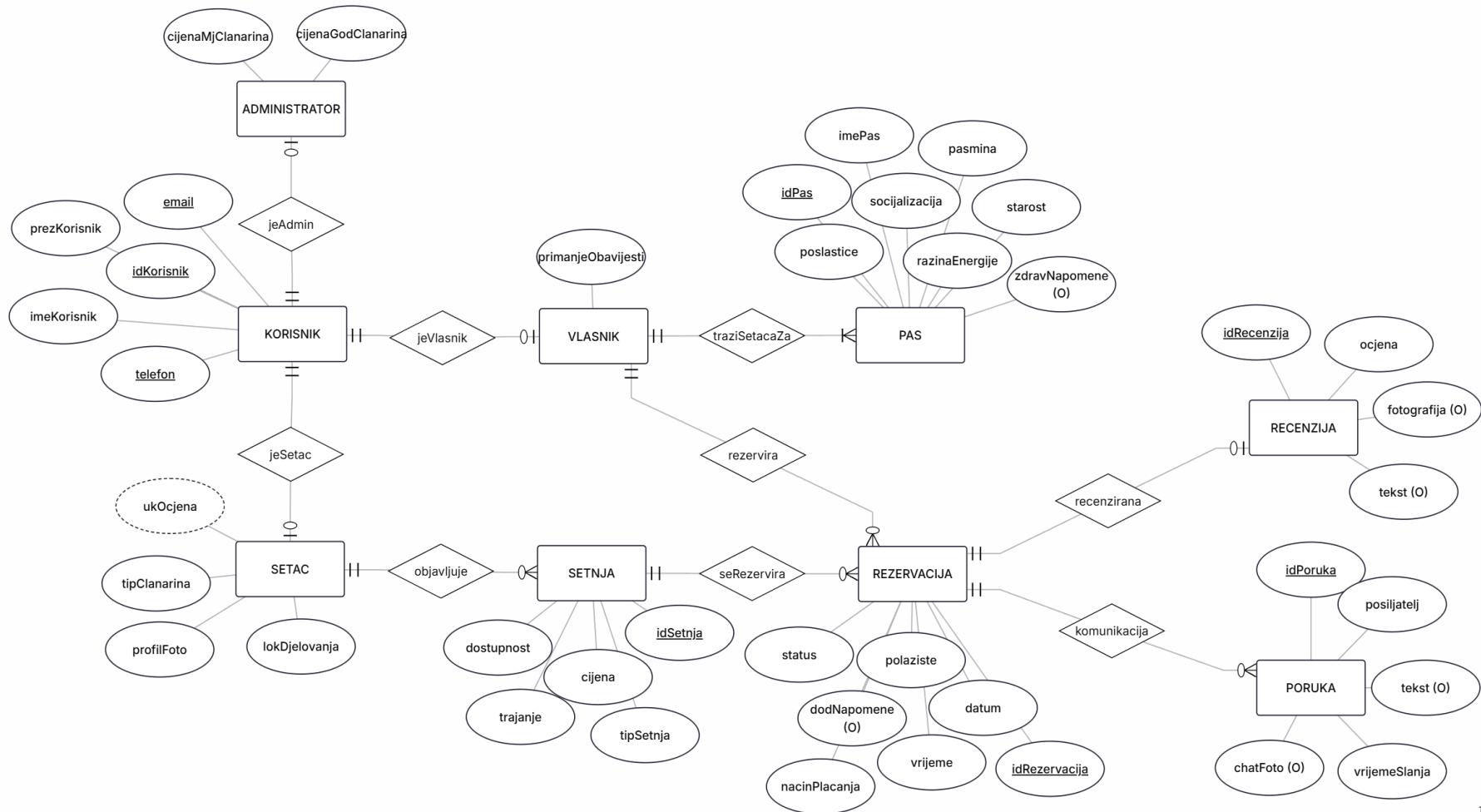


ARHITEKTURA SUSTAVA

- **MVC (Model–View–Controller)** arhitektura
- **Node.js (Express)** za backend i **Reactu** za frontend
- Arhitektura slijedi **klijent–poslužitelj** model
- **REST API** za komunikaciju
- Controller sloj:
 - prima i obrađuje HTTP zahtjeve od frontenda
- Model sloj:
 - definira entitete i upravlja podacima u bazi
- View sloj:
 - Omogućava dinamičko i responzivno korisničko sučelje
- Service sloj:
 - Implementira poslovnu logiku i upravlja podacima

BAZA PODATAKA I POHRANA

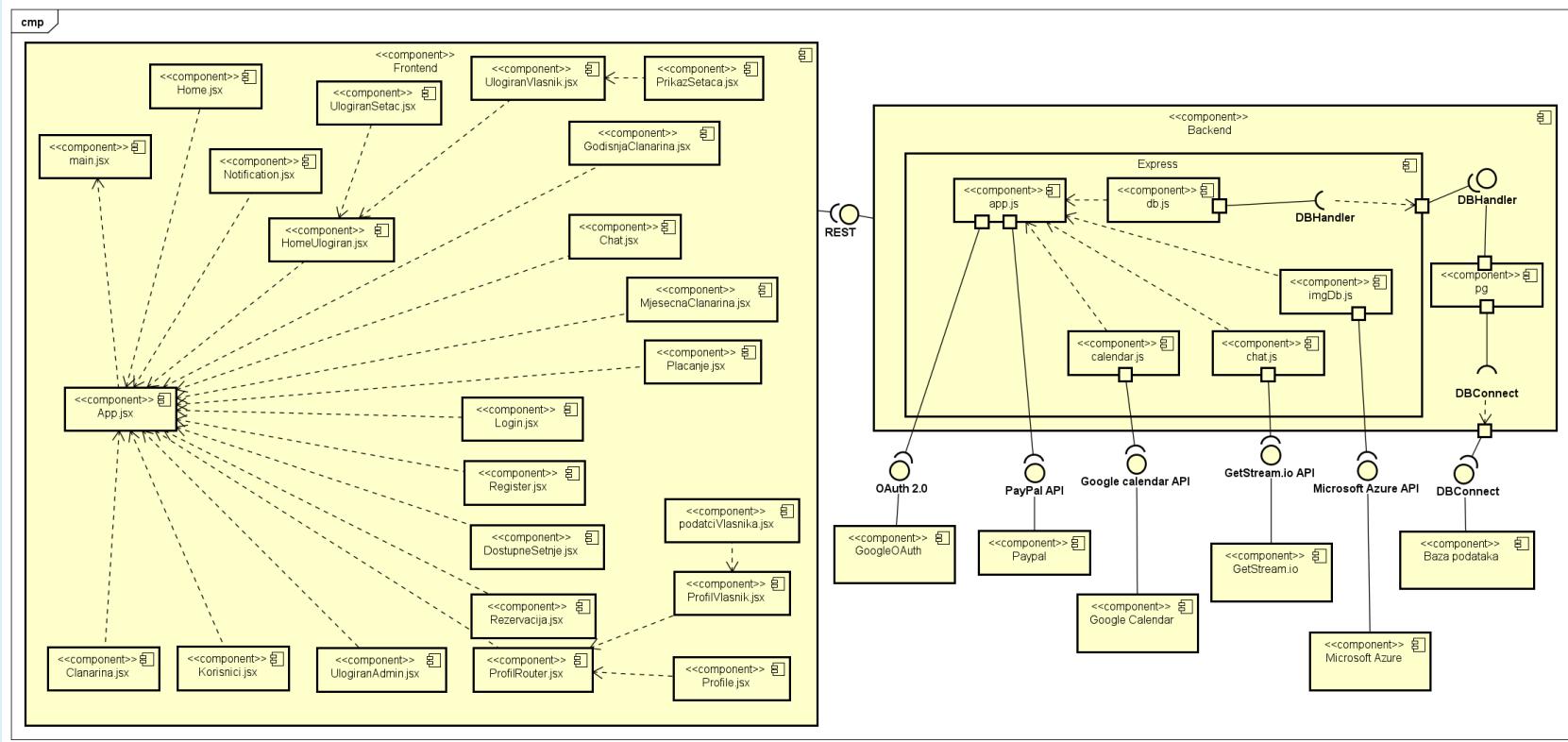
- **Baza podataka**
 - Koristi se **PostgreSQL** kao relacijska baza podataka
 - Node.js backend komunicira s bazom pomoću **node-postgres**
 - Node package **pg** omogućuje jednostavno upravljanje relacijama i transakcijama
- **Baza podataka za slike**
 - Koristi se **Microsoft Azure Blob Storage** za pohranu slika
 - U bazi podataka se pohranjuju samo reference (URL-ovi) na slike



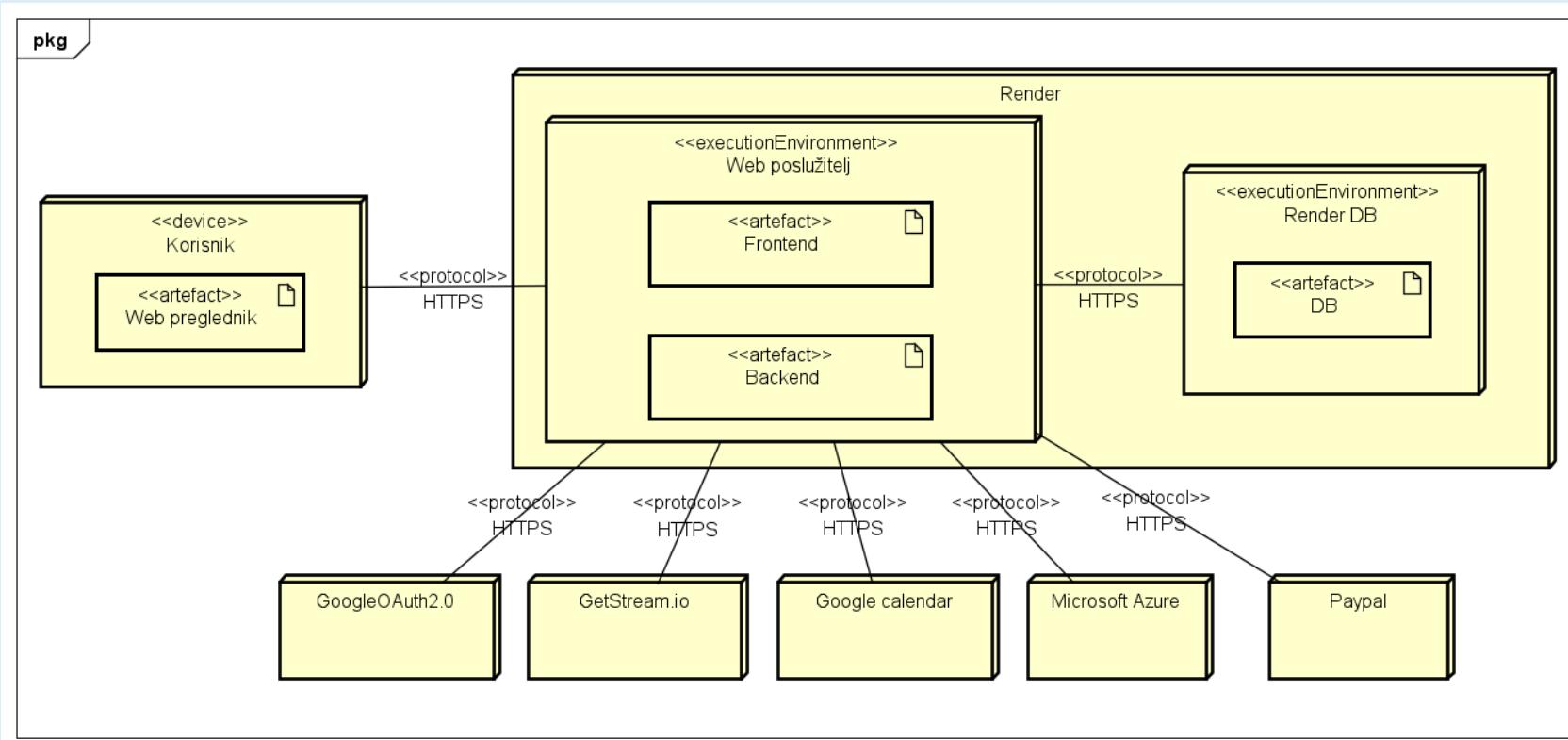
GLOBALNI UPRAVLJAČKI TOK

- **Korisnik** putem React sučelja šalje zahtjev backendu
- **Backend (Controller)** prima zahtjev, proslijeđuje ga **Service** sloju
- **Service sloj** obrađuje zahtjev i prema potrebi komunicira s **Model** slojem
- **Model sloj** dohvaća ili pohranjuje podatke u **PostgreSQL** bazu
- **Backend vraća odgovor** u JSON formatu, koji React frontend prikazuje korisniku

DIJAGRAM KOMPONENTATA



DIJAGRAM RAZMJEŠTAJA



ISPITIVANJE

- **Fokus testiranja:**
 - Testiranje osnovnih funkcionalnosti sustava kao što su registracija, prijava,
- **Opseg testiranja:**
 - Obuhvaćene su sve ključne komponente sustava (frontend, backend i integracija), uz primjenu automatiziranih i ručnih testova te provjeru rubnih i nepredviđenih slučajeva.

KORIŠTENE TEHNOLOGIJE I ALATI

| Redni broj | Ime tehnologije/ alata | Verzija | Kategorija |
|------------|------------------------|---------|------------------------|
| 01 | JavaScript | ES2023 | programske jezik |
| 02 | Node.js | 22.10.5 | radni okvir |
| 03 | Express.js | 4.18.2 | radni okvir |
| 04 | React | 18 | radni okvir |
| 05 | HTML5 | - | frontend markup |
| 06 | CSS3 | - | stiliziranje |
| 07 | PostgreSQL | 16 | baza podataka |
| 08 | Git | 2.43 | razvojni alat |
| 09 | Visual Studio Code | 1.96.3 | razvojni alat |
| 10 | Render | - | alat za razmještavanje |

Nemaš vremena prošetati svog psa?

Poveži se s provjerениm šetačima u svom susjedstvu — brzo, sigurno i jednostavno.

Prijava

Tri koraka do sretnog psa



1. Registriraj se

Registriraj se kao vlasnik ili šetač



2. Odaberij šetnju

Odaberij šetnju koja ti odgovara



3. Plati i uživaj

Plati i uživaj u slobodnom vremenu

ZAKLJUČAK

- Projekt je uspješno realiziran unatoč vremenskim ograničenjima
- Najveći izazov bila je organizacija vremena
- **Tehnički najzahtjevniji dijelovi:**
 - Google OAuth 2.0 autentifikacija
 - Deployment aplikacije na Render
 - Implementacija chat funkcionalnosti
- **Kroz izradu projekta smo stekli:**
 - tehničkih znanja (React, Node.js, PostgreSQL)
 - vještina timskog rada i rješavanja problema