

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

Dominik Černjević

DJ TOUR MANAGER

PROJEKT

Varaždin, 2026.

**SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN**

Dominik Černjević

Matični broj: 00000

Studij: IPI

**DJ TOUR MANAGER
PROJEKT**

Mentor:

Ime Mentora

Varaždin, 15. siječnja 2026.

Sadržaj

1	Opis aplikacijske domene	1
1.1	Uvod i definicija problema	1
1.2	Cilj projekta	1
1.3	Motivacija za odabir tehnologije	1
2	Teorijski uvod	2
2.1	Objektno-relacijske baze s prostornim proširenjem	2
2.2	Prostorno indeksiranje (GIST)	2
2.3	Koncept aktivne baze podataka	2
3	Model baze podataka	3
3.1	Entiteti i relacije	3
3.2	ERA Dijagram	3
4	Implementacija	4
4.1	Implementacija Baze Podataka (SQL)	4
4.2	Implementacija Backenda (Flask)	4
4.3	Implementacija Frontenda (Leaflet)	4
4.4	Sigurnost i Integritet Podataka	5
5	Primjeri korištenja	6
5.1	Planiranje nove turneje (Unos)	6
5.2	Analiza profitabilnosti (Turneja)	6
5.3	Unos nove lokacije	6
6	Zaključak	7
6.1	Osobni osvrt i perspektiva razvoja	7
6.1.1	Integracija u širi ekosustav	7

1. Opis aplikacijske domene

1.1 Uvod i definicija problema

U modernoj glazbenoj industriji, upravljanje turnejama predstavlja kompleksan logistički izazov koji nadilazi puku vremensku organizaciju događaja. Menadžeri turneja i samostalni izvođači, poput DJ-eva, suočavaju se s problemom raspršenih podataka: informacije o lokacijama (klubovima) često su odvojene od financijskih podataka i vremenskih rasporeda.

Klasični alati, poput tabličnih kalkulatora (Excel) ili generičkih kalendara, ne posjeduju prostornu svijest (*eng. spatial awareness*). Oni ne mogu automatski odgovoriti na pitanja poput: "Kolika je ukupna udaljenost između nastupa u lipnju?" ili "Koji se klubovi nalaze u Splitsko-dalmatinskoj županiji?" bez ručne intervencije korisnika.

1.2 Cilj projekta

Cilj ovog projekta je razvoj specijaliziranog Geografskog Informacijskog Sustava (GIS) pod nazivom "DJ Tour Manager". Sustav je dizajniran da omogućiti:

1. Centralizaciju podataka: Povezivanje statičkih podataka o lokacijama (klubovi) s dinamičkim podacima o događajima (gaže).
2. Prostornu analitiku: Vizualizaciju ruta, klasterizaciju lokacija i automatsko određivanje administrativne pripadnosti.
3. Financijsko praćenje: Analizu profitabilnosti (prihodi naspram putnih troškova) u vremenskom i prostornom kontekstu.

1.3 Motivacija za odabir tehnologije

Za implementaciju rješenja odabran je tehnološki stog koji osigurava robusnost, skalabilnost i pridržavanje industrijskih standarda:

Sloj podataka je spoj PostgreSQL s ekstenzijom PostGIS-om. PostgreSQL je napredni objektno-relacijski sustav upravljanja bazom podataka (ORDBMS). Odabran je zbog svoje stabilnosti i proširivosti. Ekstenzija PostGIS transformira PostgreSQL u moćnu prostornu bazu podataka koja podržava OGC (Open Geospatial Consortium) standarde [1]. Korištenje PostGIS-a ključno je jer omogućuje izvođenje složenih geometrijskih operacija (poput presjeka poligona i generiranja linija ruta) na razini baze podataka, što je znatno brže od izvođenja istih operacija u aplikacijskom sloju [2].

Aplikacijski sloj je realiziran s Python-om (Flask). Python je standard za GIS skriptiranje i analizu podataka. Flask je odabran kao mikro-okvir koji omogućuje brz razvoj RESTful API sučelja uz minimalno dodatno opterećenje (overhead), što osigurava brzi odaziv aplikacije [3].

Prezentacijski sloj se temelji na Leaflet.js. Za razliku od rješenja poput Google Maps API, Leaflet je rješenje otvorenog koda koje nudi fleksibilnost, podršku za grupiranje markera (*eng. clustering*) i jednostavnu integraciju s GeoJSON formatom podataka [4].

2. Teorijski uvod

2.1 Objektno-relacijske baze s prostornim proširenjem

Tradicionalne relacijske baze podataka (RDBMS) optimizirane su za alfanumeričke podatke. Međutim, prostorni podaci posjeduju inherentnu složenost odnosno oni su višedimenzionalni. Projekt se temelji na konceptu prostornih baza podataka (eng. Spatial Databases) koje uvode apstrakciju geometrijskih tipova podataka kao što su `POINT`, `LINESTRING` i `POLYGON` [2].

Ključna prednost ovog pristupa u odnosu na spremanje koordinata kao običnih brojeva (`float x`, `float y`) je mogućnost korištenja prostornih indeksa i funkcija.

2.2 Prostorno indeksiranje (GIST)

Standardni B-stablo (eng. B-Tree) indeksi, koji se koriste za sortiranje brojeva i teksta, neučinkoviti su za dvodimenzionalne podatke. U ovom projektu koristi se GIST (Generalized Search Tree) indeks [5]. GIST indeks funkcionira na principu R-stabla, koje grupira objekte u hijerarhiju minimalnih obuhvatnih pravokutnika (MBR).

Prilikom upita "Pronađi u kojoj se županiji nalazi ovaj klub", baza ne mora provjeravati geometriju svake županije detaljno. Umjesto toga, GIST indeks brzo eliminira sve županije čiji MBR ne siječe točku kluba, drastično smanjujući vrijeme izvršavanja upita [2].

2.3 Koncept aktivne baze podataka

Implementirani sustav koristi paradigmu aktivne baze podataka. Umjesto da baza služi samo kao pasivno spremište, ona preuzima dio poslovne logike kroz okidače (eng. Triggers) [5].

U kontekstu ovog projekta, to se manifestira kroz automatsko određivanje stranog ključa `regija_id` na temelju prostornog odnosa (presjeka točke i poligona) u trenutku unosa podataka (`BEFORE INSERT`). Prednost ovog pristupa je očuvanje integriteta podataka i centralizacija logike.

3. Model baze podataka

Baza podataka modelirana je u trećoj normalnoj formi (3NF) kako bi se smanjila redundancija podataka, posebno u pogledu lokacija klubova.

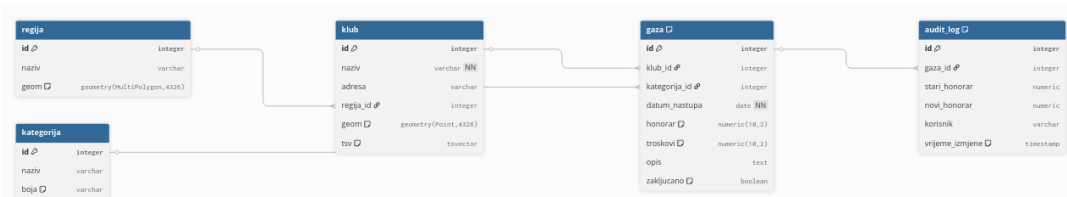
3.1 Entiteti i relacije

Sustav se sastoji od pet glavnih entiteta:

- **REGIJA**: Predstavlja administrativnu jedinicu (županiju). Sadrži primarni ključ i geometriju (**MultiPolygon**). U relaciji je 1:N s entitetom **KLUB**.
- **KLUB**: Predstavlja fizičku lokaciju. Sadrži naziv, adresu i geometriju točke (**Point**). Sadrži vanjski ključ prema **REGIJA**. U relaciji je 1:N s entitetom **GAŽA**.
- **KATEGORIJA**: Klasifikacijski entitet (npr. Festival, Klub). U relaciji je 1:N s entitetom **GAŽA**.
- **GAŽA**: Predstavlja događaj u vremenu. Sadrži datum, financijske atribute (honorar, trošak) i referencu na **KLUB**.
- **AUDIT_LOG**: Bilježi povijest promjena financijskih podataka.

3.2 ERA Dijagram

Na slici 3.1 prikazan je konceptualni model baze podataka.



Slika 3.1: ERA model baze podataka DJ Tour Manager

4. Implementacija

Implementacija je podijeljena na sloj baze podataka (SQL logika), aplikacijski sloj (Backend) i prezentacijski sloj (Frontend).

4.1 Implementacija Baze Podataka (SQL)

Ključni dio implementacije su okidači za automatsko lociranje kluba. Kako bi se izbjegli problemi s tipovima podataka u PL/pgSQL-u, korištena je proceduralna logika iteracije [5]:

```
CREATE OR REPLACE FUNCTION trg_klub_regija() RETURNS TRIGGER AS $$
DECLARE r RECORD;
BEGIN
    -- Iteracija kroz poligone regija
    FOR r IN SELECT id, geom FROM regija LOOP
        -- Prostorna provjera: Siječe li točka kluba poligon regije?
        IF ST_Intersects(r.geom, NEW.geom) THEN
            NEW.regija_id := r.id;
            EXIT;
        END IF;
    END LOOP;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;
```

Za generiranje rute turneje koristi se pohranjena funkcija koja dinamički generira GeoJSON objekt spajanjem točaka kronološkim redom koristeći funkciju `ST_MakeLine` [2].

4.2 Implementacija Backenda (Flask)

Python backend služi kao REST API. Koristi biblioteku SQLAlchemy za ORM (Object-Relational Mapping), ali za složene prostorne upite koristi izravne SQL naredbe radi performansi. Posebnost implementacije je pametni unos: API endpoint `/api/add` provjerava postoji li već klubId. Ako ne postoji, prvo kreira novi zapis u tablici KLUB, a zatim kreira zapis u tablici GAZA.

4.3 Implementacija Frontenda (Leaflet)

Frontend koristi napredne tehnike vizualizacije temeljene na Leaflet biblioteci [3]:

- MarkerCluster: Za rješavanje problema preklapanja točaka u gusto naseljenim područjima (npr. više klubova u centru Zagreba), koristi se klasterizacija koja grupira markere
- Layers Control: Implementirana je logika za toggle (uključivanje/isključivanje) sloja granica županija
- Dva moda rada: Logičko razdvajanje sučelja na "Pregled turneje" (fokus na rute i statistiku) i "Bazu Klubova" (fokus na unos i administraciju)

4.4 Sigurnost i Integritet Podataka

Osim prostornih funkcionalnosti, sustav implementira stroga pravila za očuvanje integriteta financijskih podataka [5]:

- Mehanizam zaključavanja: Procedura `zakljucaj_mjesec` postavlja zastavicu `zakljucano = TRUE` za sve nastupe u odabranom mjesecu. Nad tablicom *gaza* postavljen je `BEFORE UPDATE` okidač koji sprječava izmjene zaključanih zapisa podizanjem iznimke (Exception).
- Revizijski trag: Okidač `trg_audit_honorar` automatski bilježi promjene financijskih atributa u zasebnu tablicu *audit_log*.
- Ograničenja (Constraints): Definirana su `CHECK` ograničenja koja sprječavaju unos negativnih novčanih iznosa.

5. Primjeri korištenja

5.1 Planiranje nove turneje (Unos)

Korisnik želi unijeti nastupe za srpanj 2026. Odabire tab "Unos" i na mapi vidi sive markere svih klubova gdje je ikad nastupao (baza znanja).

1. Korisnik klikne na postojeći marker (npr. "Papaya" na Zrću). Forma se automatski popunjava podacima o klubu.
2. Korisnik samo unosi datum i honorar.
3. Sustav sprema novu gažu vezanu za postojeći klub, čime se izbjegava redundancija.

5.2 Analiza profitabilnosti (Turneja)

Korisnik želi vidjeti financijsko stanje za lipanj. U tabu "Turneja" odabire "Lipanj 2025".

1. Aplikacija automatski iscrtava crnu isprekidanu liniju koja povezuje nastupe redom kako se odvijaju (Ruta).
2. U sidebaru se prikazuje izračun: $Prihod - Troškovi = Profit$. Podaci se vuku iz SQL pogleda `view_statistika_mjesec`.

5.3 Unos nove lokacije

Korisnik nastupa na novoj lokaciji u Istri koja nije u bazi:

1. Klikne na prazno mjesto na mapi u Puli
2. Pojavljuje se marker
3. Korisnik unosi ime "Arena Pula"
4. Nakon spremanja, PostGIS trigger automatski detektira da točka pripada poligonu "Istarska županija"

6. Zaključak

U sklopu ovog projekta razvijen je funkcionalan i robustan GIS sustav "DJ Tour Manager". Implementacija je potvrdila da je PostgreSQL s PostGIS ekstenzijom superiorna platforma za upravljanje prostornim podacima u odnosu na standardne baze podataka.

Ključna postignuća:

- Uspješna implementacija aktivne baze putem okidača, čime se osigurava integritet prostornih relacija
- Normalizacija modela (odvajanje Kluba od Gaže) omogućila je stvaranje "baze znanja" lokacija, što dugoročno štedi vrijeme unosa
- Korištenje prostornih pogleda omogućilo je efikasno generiranje statistike u realnom vremenu

Trenutna implementacija računa zračne udaljenosti (`ST_Length`). Buduća nadogradnja mogla bi uključivati integraciju s pgRouting ekstenzijom za izračun stvarnih cestovnih udaljenosti, što bi dodatno preciziralo izračun putnih troškova.

6.1 Osobni osvrt i perspektiva razvoja

Kao osoba koja se aktivno bavi DJ-ingom, razvoj sustava "DJ Tour Manager" za mene nije bio samo akademski zadatak, već rješavanje stvarnog problema s kojim se susrećem u svakodnevnom radu. Do sada se moja evidencija nastupa svodila na bilješke u mobitelu odnosno tekstualne dokumente. Takav način rada često je rezultirao gubitkom informacija, otežanim praćenjem financija i nemogućnošću sagledavanja šire slike gaža.

Implementacijom ovog sustava, po prvi put imam alat koji mi omogućuje vizualno upravljanje nastupima. Mogućnost da na interaktivnoj mapi vidim logički slijed putovanja, automatski izračunam profitabilnost po regijama i detektiram potencijalne logističke probleme prije nego se dogode, predstavlja ogroman korak naprijed u profesionalizaciji mog poslovanja. Posebno bih istaknuo vrijednost baze znanja klubova. Činjenica da jednom uneseni klub ostaje trajno zapamćen s točnom geolokacijom štedi vrijeme i smanjuje mogućnost greške pri budućim bookingima.

6.1.1 Integracija u širi ekosustav

Gledajući u budućnost, vidim veliki potencijal u integraciji ovog sustava s mobilnom platformom "Spinspot" [6], čiji je razvoj također iniciran u sklopu fakultetskog projekta. Dok je "DJ Tour Manager" primarno alat za internu organizaciju i logistiku (Business to Business), mobilna aplikacija fokusirana je na interakciju s publikom (Business to Consumer).

Ova dva sustava mogla bi se realizirati na sljedeći način:

- Automatizacija Kalendara (F03): Kada DJ unese novi nastup u "DJ Tour Manager", podatak se putem API-ja može automatski sinkronizirati s "Kalendarom događanja" u mobilnoj aplikaciji. Time DJ ne mora dvaput unositi iste podatke, a publika u realnom vremenu dobiva informaciju o novom nastupu u svojoj blizini.

- Geolokacijske Notifikacije (F07): Korištenjem PostGIS logike iz ovog projekta (npr. *ST_DWithin*), mobilna aplikacija "Vaomil jeni DJ nastupaveerasuradi jusu od 20km od vas!".
- Napredna Analitika (F05): Mobilna aplikacija trenutno nudi osnovni export statistike. Integracijom s ovim sustavom, DJ bi mogao dobiti "Heatmap" prikaz svoje popularnosti, odnosno vizualizaciju gdje korisnici najčešće naručuju pjesme ili ostavljaju recenzije, što bi mu pomoglo u planiranju budućih turneja.

Zaključno, ovaj projekt postavlja čvrste temelje (backend infrastrukturu s prostornom inteligencijom) na koje se mogu nadograđivati složeni korisnički servisi, čime se stvara cjelovit ekosustav za modernog glazbenog profesionalca.

Bibliografija

- [1] Open Geospatial Consortium, *OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture*, 2011.
- [2] R. O. Obe and L. S. Hsu, *PostGIS in Action*. Manning Publications, third ed., 2021.
- [3] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*. Sebastopol, CA: O'Reilly Media, 2018.
- [4] Leaflet JS Team, *Leaflet - an open-source JavaScript library for mobile-friendly interactive maps*, 2023. Pristupljeno: 12.01.2026.
- [5] PostgreSQL Global Development Group, *PostgreSQL 16 Documentation: PL/pgSQL - SQL Procedural Language*, 2024. Pristupljeno: 10.01.2026.
- [6] D. Černjević, “Spinspot - mobilna aplikacija za interakciju s publikom,” 2025. GitHub repozitorij.

Popis slika

3.1	ERA model baze podataka DJ Tour Manager	3
-----	---	---