



# 04 UPORABNA INFORMATIKA

2023 < ŠTEVILKA 4 < LETNIK XXXI < ISSN 1318-1882

# Vabilo na 31. konferenco Dnevi slovenske informatike

**13. – 15. 5. 2024, Hotel Bernardin, Portorož**

V Slovenskem društvu Informatika nas že več kot 30 let povezuje strast do informatike. Vabimo vas, da si v svojem koledarju rezervirate in se nam pridružite na 31. konferenci Dnevi slovenske informatike, ki se bo osredotočila na **ključne teme** informatike in digitalizacije v Sloveniji in širše. Pridružite se nam, da predstavite svoje uspešne projekte, da skupaj razpravljamo o novih tehnologijah, da spoznavamo najnovejše trende ter da se povezujete z vodilnimi strokovnjaki s področja informatike zasebnega in javnega sektorja, doma in v tujini.

Priložnost sodelovati imate vsi, ki vas zanima **odmev na vaše rešitve, izkušnje, pristope, strategije, pridobljene pri zasnovi, razvoju in uvajanju na informacijskih tehnologijah temelječih storitev in sistemov.**

Konferenca DSI 2024 bo potekala v luči rdeče niti »**Z umetno inteligenco gradimo svet prihodnosti**«. V središču konference bo tako ideja uporabe in vpliva umetne inteligence in z njo povezanimi temami, ki prispevajo k napredku Slovenije in postaja ključno orodje za izboljšanje učinkovitosti v različnih sektorjih, bodisi v zasebnem ali javnem okolju.

Povzetke predstavitev vaših rezultatov in dela pričakujemo **do vključno 17. marca 2023** v elektronski obliki – vsi podatki glede oddaje so na voljo na <https://dsi2024.dsi-konferenca.si/stran/za-avtorje>. Avtorji bodo imeli kasneje tudi možnost za objavo v reviji Uporabna informatika.

Pri odločitvi so vam lahko v pomoč spodnje tematike. Seveda pa so dobrodošle tudi druge, za katere ocenjujete, da tematsko sodijo v vsebino:

Podatkovna družba Podatkovni prostori Digitalna in zelena odpornost Digitalne javne storitve Napredne digitalne tehnologije Inovacije v IKT Podatkovna jezera Inteligentne aplikacije	Podpora pri odločanju Digitalizacija izobraževanj in usposabljanj Izobraževanje in usposabljanje za digitalno družbo Odprti in dostopni podatki Tehnologije upravljanja znanja IT v managementu Kibernetska varnost
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Pomembni datumi:

**17. marec 2024:** Oddaja povzetka

**31. marec 2024:** Povratna informacija avtorjem

**5. maj 2024:** Oddaja predstavitev

Več o prispevkih, datumih, priporočilih in ostalih napotkih o sodelovanju s prispevkom, si lahko preberete na **strani konference** (<https://dsi2023.dsi-konferenca.si/>).

Že sedaj pa v svojem koledarju rezervirajte termin konference, ki bo potekala med **13. in 15. majem 2024 v Hotelu Bernardin v Portorožu.**

# U P O R A B N A I N F O R M A T I K A

2023 ŠTEVILKA 4 OKT/NOV/DEC LETNIK XXXI ISSN 1318-1882

## Znanstveni prispevki

- Miha Frangež, Matevž Pesek  
**Načrtovalnik poti za javni potniški promet in storitve mikromobilnosti** 171
- Živa Hudobivnik, Maja Savinek, Marina Trkman  
**Uporaba pogovornega robota za podporo intranetu: primer Elektro Ljubljana** 179

## Kratki znanstveni prispevki

- Lidija Stanovnik, Miha Mraz  
**Naslavljanje DNA-pomnilnikov** 192

## Strokovni prispevki

- Timotej Knez  
**NORA: Napredno orodje za poučevanje relacijske algebre** 196

## Informacije

- Iz Islovarja** 202
- Vabilo na konferenco DSI**

#### Ustanovitelj in izdajatelj

Slovensko društvo INFORMATIKA  
Litostrojska cesta 54, 1000 Ljubljana

#### Predstavniki

Niko Schlamberger

#### Odgovorni urednik

Mirjana Kljajić Borštnar

#### Uredniški odbor

Andrej Kovačič, Anton Manfreda, Evelin Krnac, Jan Mendling, Jan von Knop, John Taylor, Lili Nemeč Zlatolas, Marko Hölbl, Miodrag Popović, Mirjana Kljajić Borštnar, Mirko Vintar, Pedro Simões Coelho, Saša Divjak, Sjaak Brinkkemper, Tatjana Welzer Družovec, Vesna Bosilj-Vukšič, Vida Groznik, Vida Groznik, Vladislav Rajkovič

#### Recenzentski odbor

Aleksander Sadikov, Alenka Baggia, Alenka Brezavšček, Aljaž Košmerlj, Andrej Brodnik, Andrej Kovačič, Andreja Pucihar, Anton Manfreda, Benjamin Urh, Blaž Rodič, Borut Batagelj, Borut Werber, Boštjan Šumak, Božidar Potočnik, Branko Kavšek, Branko Šter, Ciril Bohak, Damjan Fujs, Damjan Strnad, David Jelenc, Dejan Lavbič, Denis Trček, Domen Mongus, Drago Bokal, Eva Jereb, Evelin Krnac, Inna Novalija, Irena Nančovska Šerbec, Ivan Gerlič, Jernej Vičič, Jure Žabkar, Jurij Mihelič, Lovro Šubelj, Luka Pavlič, Luka Tomat, Maja Pušnik, Marina Trkman, Marjeta Marolt, Marko Bajec, Marko Hölbl, Marko Robnik Šikonja, Martin Šavc, Matej Klemen, Matjaž Divjak, Mirjana Kljajić Borštnar, Mladen Borovič, Muhamed Turkanovič, Niko Schlamberger, Nikola Ljubešič, Patricio Bulić, Polona Rus, Robert Leskover, Samed Bajrić, Sandi Gec, Saša Divjak, Slavko Žitnik, Tatjana Welzer Družovec, Tomaž Hovelja, Uroš Rajkovič, Vida Groznik, Vladislav Rajkovič, Živa Rant

#### Tehnični urednik

Timotej Knez

#### Lektoriranje angleških izvlečkov

Marvelingua (angl.)

#### Oblikovanje

KOFEIN DIZAJN, d. o. o.

#### Prelom in tisk

Boex DTP, d. o. o., Ljubljana

#### Naklada

110 izvodov

#### Naslov uredništva

Slovensko društvo INFORMATIKA  
Uredništvo revije Uporabna informatika  
Litostrojska cesta 54, 1000 Ljubljana  
www.uporabna-informatika.si

Revija izhaja četrtletno. Cena posamezne številke je 20,00 EUR. Letna naročnina za podjetja 85,00 EUR, za vsak nadaljnji izvod 60,00 EUR, za posameznike 35,00 EUR, za študente in seniorje 15,00 EUR. V ceno je vključen DDV.

Revija Uporabna informatika je od številke 4/VII vključena v mednarodno bazo INSPEC.

Revija Uporabna informatika je pod zaporedno številko 666 vpisana v razvid medijev, ki ga vodi Ministrstvo za kulturo RS.

Revija Uporabna informatika je vključena v Digitalno knjižnico Slovenije (dLib.si).

Izid publikacije je finančno podprla Javna agencija za znanstvenoraziskovalno in inovacijsko dejavnost Republike Slovenije.

© Slovensko društvo INFORMATIKA

## Vabilo avtorjem

V reviji Uporabna informatika objavljamo kakovostne izvirne prispevke domačih in tujih avtorjev z najširšega področja informatike, ki se nanašajo tako na poslovanju podjetij, javno upravo, družbo in posameznika. Prispevki so lahko znanstvene, strokovne ali informativne narave, še posebno spodbujamo objavo interdisciplinarnih prispevkov. Zato vabimo avtorje, da prispevke, ki ustrezajo omenjenim usmeritvam, pošljejo uredništvu revije po elektronski pošti na naslov [ui@društvo-informatika.si](mailto:ui@društvo-informatika.si).

Avtorje prosimo, da pri pripravi prispevka upoštevajo navodila, ki so objavljena na naslovu <http://www.uporabna-informatika.si>.

Za kakovost prispevkov skrbi mednarodni uredniški odbor. Prispevki so anonimno recenzirani, o objavi pa na podlagi recenzij samostojno odloča uredniški odbor. Recenzenti lahko zahtevajo, da avtorji besedilo spremenijo v skladu s priporočili in da popravljeni prispevek ponovno prejmejo v pregled. Sprejeti prispevki so pred izidom revije objavljeni na spletni strani revije (predobjava), še prej pa končno verzijo prispevka avtorji dobijo v pregled in potrditev. Uredništvo lahko še pred recenzijo zavrne objavo prispevka, če njegova vsebina ne ustreza vsebinski usmeritvi revije ali če prispevek ne ustreza kriterijem za objavo v reviji.

Pred objavo prispevka mora avtor podpisati izjavo o avtorstvu, s katero potrjuje originalnost prispevka in dovoljuje prenos materialnih avtorskih pravic. Avtorji prejmejo enoletno naročnino na revijo Uporabna informatika, ki vključuje avtorski izvod revije in še nadaljnje tri zaporedne številke. S svojim prispevkom v reviji Uporabna informatika boste pomagali k širjenju znanja na področju informatike. Želimo si čim več prispevkov z raznoliko in zanimivo tematiko in se jih že vnaprej veselimo

Uredništvo revije

## Navodila avtorjem člankov

Članke objavljamo praviloma v slovenščini, članke tujih avtorjev pa v angleščini. Besedilo naj bo jezikovno skrbno pripravljeno. Priporočamo zmernost pri uporabi tujk in, kjer je mogoče, njihovo zamenjavo s slovenskimi izrazi. V pomoč pri iskanju slovenskih ustreznih priporočamo uporabo spletnega terminološkega slovarja Slovenskega društva Informatika, Islovar ([www.islovar.org](http://www.islovar.org)).

Znanstveni prispevek naj obsega največ 40.000 znakov, kratki znanstveni prispevek do 10.000 znakov, strokovni članki do 30.000 znakov, obvestila in poročila pa do 8.000 znakov.

Prispevek naj bo predložen v urejevalniku besedil Word (\*.doc ali \*.docx) v enojnem razmaku, brez posebnih znakov ali poudarjenih črk. Za ločilom na koncu stavka napravite samo en presledek, pri odstavkih ne uporabljajte zamika.

Naslovu prispevka naj sledi polno ime vsakega avtorja, ustanova, v kateri je zaposlen, naslov in elektronski naslov. Sledi naj povzetek v slovenščini v obsegu 8 do 10 vrstic in seznam od 5 do 8 ključnih besed, ki najbolje opredeljujejo vsebinski okvir prispevka. Sledi naj prevod naslova povzetka in ključnih besed v angleškem jeziku. V primeru, da oddajate prispevek v angleškem jeziku, velja obratno. Razdelki naj bodo naslovljeni in oštevilčeni z arabskimi številkami.

Slike in tabele vključite v besedilo. Opremite jih z naslovom in oštevilčite z arabskimi številkami. Na vsako sliko in tabelo se morate v besedilu prispevka sklicevati in jo pojasniti. Če v prispevku uporabljate slike ali tabele drugih avtorjev, navedite vir pod sliko oz. tabelo. Revijo tiskamo v črno-beli tehniki, zato barvne slike ali fotografije kot original niso primerne. Slikam zaslonov se v prispevku izogibajte, razen če so nujno potrebne za razumevanje besedila. Slike, grafikoni, organizacijske sheme ipd. naj imajo belo podlago. Enačbe oštevilčite v oklepajih desno od enačbe.

V besedilu se sklicujte na navedeno literaturo skladno s pravili sistema IEEE navajanja bibliografskih referenc, v besedilu to pomeni zaporedna številka navajenega vira v oglatem oklepaju (npr. [1]). Na koncu prispevka navedite samo v prispevku uporabljeno literaturo in vire v enotnem seznamu, urejeno po zaporedni številki vira, prav tako v skladu s pravili IEEE. Več o sistemu IEEE, katerega uporabo omogoča tudi urejevalnik besedil Word 2007, najdete na strani [https://owl.purdue.edu/owl/research\\_and\\_citation/ieee\\_style/ieee\\_general\\_format.html](https://owl.purdue.edu/owl/research_and_citation/ieee_style/ieee_general_format.html).

Prispevku dodajte kratek življenjepis vsakega avtorja v obsegu do 8 vrstic, v katerem poudarite predvsem strokovne dosežke.



# Načrtovalnik poti za javni potniški promet in storitve mikromobilnosti

Miha Frangež, Matevž Pesek

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Večna pot 113, 1000 Ljubljana  
mf7094@student.uni-lj.si, matevz.pesek@fri.uni-lj.si

## Izvleček

V luči trenutne okoljske in energetske krize je uporaba javnega in nizkoogljičnega prevoza ključnega pomena za doseganje zadanih državnih in evropskih ciljev do konca tekočega desetletja. Za razliko od osebne avtomobila pa potovanje z drugimi načini prevoza zahteva vsaj nekaj načrtovanja, ki se lahko izkaže za preveliko prilagoditev za marsikaterega potencialnega uporabnika. Razlogi za nizko uporabo javnega prevoza in deljenih prevoznih sredstev ležijo v veliki meri v razdrobljenosti dostopa do informacij, potrebnih za njihovo učinkovito rabo. Cilj članka je načrtovanje poti olajšati z uvedbo inteligentnega načrtovalnika poti, ki na enem mestu omogoča samodejno iskanje možnosti prevoza in načrtovanje poti z več prevoznimi sredstvi z javno dostopnimi oblikami mobilnosti, kot so avtobusi, vlaki, souporabo avtomobilov in mestnimi kolesi. Problem rešujemo v treh korakih: (1) razvijemo enotni podatkovni model za podatke javnega prevoza in storitve mobilnosti v Sloveniji ter za vse ponudnike implementiramo adapterje iz posameznih zasebnih formatov v skupno strukturo, (2) postavimo odprtokodni multimodalni načrtovalnik poti *OpenTripPlanner* in (3) razvijemo hibridno spletno in mobilno aplikacijo, ki omogoča pregled nad vsemi podatki ter interakcijo z načrtovalnikom.

**Ključne besede:** javni potniški promet, načrtovanje poti, odprti podatki, trajnostna mobilnost

## Route Planner for Public Transport and Micromobility

### Abstract

In light of the current environmental and energy crisis, the use of public transport and other low-carbon means of transportation is crucial for achieving the set national and European goals by the end of the current decade. Unlike traveling by car, however, traveling by other means requires at least some planning, which may prove to be too much of an adjustment for many potential users. The reasons for the low level of use of public transport and shared mobility solutions lie largely in the fragmentation of access to information about the possibilities of travel. The aim of the article is to make route planning easier by introducing an intelligent route planner that enables automatic search for transport options and multi-modal route planning with publicly available forms of mobility such as buses, trains, car sharing and city bikes in one place. There are three steps to solving the problem: (1) development of a unified data model for data on public transport and mobility services in Slovenia and implementation of adapters from their internal formats into a common format for all providers, (2) setting up of the open source multimodal route planner *OpenTripPlanner*, and (3) the development of a hybrid web and mobile application that presents the aggregated information in one place.

**Keywords:** public transport, route planning, open data, sustainable mobility

### 1 UVOD

V zadnjih letih je Slovenija doživela razrast novih storitev mikromobilnosti. Mednje sodijo naslednje vrste souporabe prevoznih sredstev: kratkotrajna izposoja avtomobilov, sistemi mestnih koles in skirojev. Prav tako smo priča večjemu napredku na

področju javnega potniškega prometa z dodanimi mestnimi prevozniki v manjših mestih (npr. Celje), uvedbo kartomatov kot alternativne točke za nakup vozovnic in uvedbo enotne »karte Slovenija«. Okoljske prednosti pogostejše hoje in kolesarjenja ter uporabe javnega potniškega prometa so splošno znane

že desetletja, v zadnjih letih pa so pozitivni vpliv na okolje demonstrirali tudi sistemi souporabe avtomobilov (Nijland & van Meerkerk, 2017). V Sloveniji so se slednji pojavili komaj leta 2016 z uvedbo sistema Avant2Go (Avant2Go, 2023), v zadnjih letih pa sta se mu na trgu pridružila še dva ponudnika: podjetji Share'N'go in GreenGo. Pozitiven vpliv na okolje so študije odkrile tudi pri sistemih souporabe koles (Zhang & Mi, 2018), ki pri nas delujejo že dlje. Prvi v Sloveniji je bil sistem BicikeLJ, v Ljubljani postavljen leta 2011 (Stregar, 2019), danes pa podobne sisteme skupno petih ponudnikov najdemo po skoraj vseh večjih mestih (Društvo za elektronsko in računalniško pismenost, 2023).

Čeprav mnoge študije kažejo, da je uporaba tako javnega potniškega prometa, kot tudi storitev souporabe avtomobilov in koles zelo koristna pri zmanjšanju izpustov toplogrednih plinov in onesnaževanja nasploh, pa pogosto pridejo do enakega zaključka – delež uporabe teh sistemov je v primerjavi z osebni avtomobili tako majhen, da je njihov realni vpliv precej manjši, kot bi lahko bil (Peng in sod., 2021). Razlogov, zakaj se ljudje v večini primerov raje odločijo za vožnjo z osebnim avtomobilom, je mnogo. Na področju javnega prevoza se med njimi najpogosteje omenja časovno konkurenčnost (Ogrin & Dovečar, 2014), ki je sestavljena iz časa prevoza in časa čakanja. Izboljšava prvega dela seveda zahteva velike investicije s strani operaterjev, študije v ZDA pa so pokazale, da uporaba mobilne aplikacije s podatki v realnem času skrajša tako občuten, kot tudi dejanski čas čakanja (Brakewood in sod., 2014; Watkins in sod., 2011).

Razlogi za manjšo uporabo javnega prevoza se izkazujejo tudi v sami kompleksnosti procesa pregleda možnih načinov uporabe in uporabe večih tipov prevoza v okviru ene poti. V zadnjem času je razrast novih ponudnikov souporabe in javnih prevozov doprinesel k še večji fragmentaciji trga mobilnosti z vidika individualnih procesov nakupa vozovnic in planiranja poti. Uporabnik v Ljubljani ima na primer na voljo devet načinov mobilnosti, vsaka z lastno mobilno aplikacijo. Uporabnik mora med aplikacijami preklapljati in ročno iskati optimalno pot. Tudi storitve Google, ki jih v večjih državah uporabniki smatrajo kot skupno alternativo vsaj na nivoju planiranja, v nekateri primerih pa tudi nakupa vozovnic, v Sloveniji ne povzemajo podatkov o vseh razpoložljivih prevoznikih. Zaradi dinamične narave mikromobilnosti se prav tako ni mogoče zanašati na ustaljene

vzorke, kot je to mogoče pri javnem potniškem prometu, saj obstoj postaj še ne pomeni, da je pot med njima mogoča. To je posebej očitno v kontekstu souporabe sredstev, ki jih je potrebno prevzeti in pustiti na za to določenih postajah. V teh primerih je mogoče stanje, ko začetna postaja nima razpoložljivih vozil, ali pa končna postaja nima prostih parkirnih mest.

Iz navedenih razlogov se med uporabniki teh storitev pojavlja vedno večja potreba po integraciji med sistemi. Integracija sistemov mobilnosti je možna na več nivojih:

1. agregacija podatkov v obliki prikaza podatkov vseh ponudnikov v skupnem uporabniškem vmesniku;
2. multimodalni načrtovalnik, ki omogoča samodejno iskanje poti z uporabo storitev več ponudnikov;
3. integracija plačil omogoča plačilo vseh storitev z enim plačilnim sredstvom oz. plačilo celotne poti z enim računom.

Cilj pričujočega članka je pregled trenutnega stanja integracije v Sloveniji, pregled možnosti izboljšav na tem področju in implementacija prvih dveh stopenj integracije – agregatorja podatkov in načrtovalnika poti.

## 2 SORODNA DELA

Tako v Sloveniji kot po svetu se je delo integracije sistemov mobilnosti pričelo skoraj sočasno z njihovim prihodom v mesta. Integracijo med ponudniki najpogosteje izvajajo mesta oz. občine, nacionalni organi in podjetja, ki sama niso ponudniki storitev mobilnosti.

### 2.1 Enotne mestne kartice in aplikacije

Začetki digitalne integracije mobilnosti segajo v zgodnja devetdeseta leta, s prvimi primeri »pametnih mestnih kartic« na Finskem in v Argentini (Oorni in sod., 2001), kjer so mesta uvedla pametne kartice kot enotno plačilno sredstvo, ki so ga sprejemale vsi ponudniki javnega potniškega prometa v pokritih mestih (Oulu na Finskem in Cordoba v Argentini). Tovrstne kartične sisteme so v naslednjih dveh desetletjih uvedla mnoga mesta po svetu, vanje pa integrirala še več mestnih storitev (na primer plačevanje parkirnine, storitve mikromobilnosti in komunalne storitve).

V Sloveniji je prvo mestno kartico uvedla Mestna občina Ljubljana pod imenom Urbana. Ta je v uporabi še danes in imetnikom omogoča vožnjo z mestnimi avtobusi, plačilo parkirnine, članstvo v mestni knjižnici, izposajo koles in vožnjo z žičnico na mestni grad. Na kartico je možno nalaganje dobroimetja v

obliki gotovine ali z bančno kartico na enem izmed prodajnih mest, med katerimi je skoraj 60 avtomatov.

Na nivoju države je bil uveden Integriran javni potniški promet (IJPP) kot projekt Ministrstva za infrastrukturo, ki omogoča uporabo pametnih kartic za vožnjo z javnim potniškim prometom v Sloveniji. Del sistema predstavlja tudi podatkovna baza vozniških redov, ki vsebuje vse medkrajevne linije, ki operirajo v sklopu državne koncesije. Ko je bil sistem razvit leta 2015, je bila potnikom na voljo spletna stran, na kateri so si lahko vse vozne rede v sistemu ogledali, danes pa več ni na voljo. Podatki, ki jih sistem vsebuje, so na voljo na Nacionalni točki dostopa<sup>(1)</sup> v formatu *GTFS*, prav tako pa je na voljo spletna storitev *SOAP*, ki omogoča dostop do podatkov v prvotni obliki (pri preslikavi v *GTFS* se namreč nekateri podatki izgubijo).

Od leta 2020 naprej so na voljo tudi osnovni podatki v realnem času, dostopni po standardnem protokolu *SIRI* (podprte so le surove lokacije vozil – profil *SIRI-VM*). Prav tako pa v času pisanja pričujočega članka poteka javni razpis za spletno storitev, ki bo podatke izvažala v formatu *NeTEx* (Network Timetable Exchange), saj so trenutni *NeTEx* podatki le pretvorjeni iz *GTFS* in vsebujejo mnogo napak.

## 2.2 Obstoječe rešitve v Evropi in Sloveniji

Pravo multimodalno načrtovanje (tako, ki pokriva vse načine prevoza in vse ponudnike storitev mobilnosti) je že na voljo marsikje po svetu, tako v neodvisnih aplikacijah tehnoloških velikanov, kot tudi s strani samih ponudnikov storitev in nacionalnih agencij. Največja komercialna ponudnika aplikacij, ki omogočajo multimodalno načrtovanje poti, sta podjetja Google in Citymapper. Oba v mnogih svetovnih mestih omogočata navigacijo z uporabo javnega prevoza, soupornosti avtomobilov, mestnimi kolesi in skiroji, v nekaterih mestih pa imata integracijo tudi s taksi storitvami. Svoje storitve oba financirata s prodajo dostopa do programskih vmesnikov za načrtovanje poti, kar pa ju omejuje le na večja mesta, kjer je povpraševanje po takšnih storitvah dovolj veliko. Aplikacija Google Maps tako v Sloveniji podpira le dva mestna prometa in medkrajevne IJPP linije (s pogostimi napakami), Citymapper pa pri nas sploh ni na voljo zaradi zaradi majhnosti največjih slovenskih mest.

Google Maps je brezplačna mobilna in spletna aplikacija podjetja Google, ki omogoča multimodalno navigacijo po celem svetu. V najbolj podprtih mestih aplikacija omogoča navigacijo s hojo, kolesom, avtomobilom, javnim potniškim prometom, storitvami mikromobilnosti (izposoja koles in skirojev) ter storitvami za prevoz na klic. Slovenija je v aplikaciji med slabše podprtimi državami, saj so pri nas na voljo le navigacija s hojo, avtomobilom in javnim prevozom, pri čemer so podatki o javnem prevozu na voljo le za mestna prometa v Mariboru in Ljubljani ter linije v sistemu IJPP.

Aplikacija Citymapper je ena izmed najbolj naprednih aplikacij za javni prevoz in storitve mikromobilnosti trenutno na tržišču. Boljšo pokritost od konkurence dosega z obširno uporabo podatkov, ki jih vnesejo uporabniki sami. Za razliko od aplikacije Google Maps, Moovit od uporabnikov ne sprejema le prijavi in obvestil v realnem času (na primer prijavi obvoza, zamude ali zasedenosti vozila), ampak omogoča uporabnikom tudi vnos vozniških redov za regije, kjer uradnih podatkov ni na voljo. Zraven javnega prevoza (avtobus, vlak, tramvaj, metro) podpira tudi navigacijo s taksi službami, vključno s platformami kot sta Uber in Lyft, ter z lastnimi ali izposojenimi kolesi, avtomobili in skiroji.

Digitransit je odprtokoden projekt javne agencije za promet finske regije Helsinki, sestavljen iz spletnega vmesnika za načrtovanje poti in pregledovanje zemljevida ter več zalednih storitev. Za načrtovanje poti uporablja odprtokoden strežniški program *OpenTripPlanner* z nekaj spremembami, prav tako pa se nanj zanaša za hrambo in dostop do podatkov (nima lastne podatkovne baze).

Transitland je odprtokoden sistem za agregacijo podatkov o javnem potniškem prometnem. Sprejema podatke v formatu *GTFS*, navadno v obliki povezave do najnovejše različice, ki jo periodično prenese in podatke posodobi v podatkovni bazi. Viri podatkov so definirani v opisnih datotekah *JSON*, hranjenih na platformi *GitHub*, kjer jih lahko uporabniki dodajajo in spreminjajo. Najnovejša različica podpira tudi sisteme mikromobilnosti v formatu *GBFS*.

Leta 2020 je bila izdana aplikacija *Travana*, namenjena Ljubljanskemu potniškemu prometom. Ključna funkcionalnost, ki je aplikacijo razločevala od ostalih (vključno z uradno aplikacijo *Urbana*), je bil prikaz

<sup>1</sup> www.nap.si

lokacij avtobusov v realnem času. Aplikacija je po uporabniških ocenah hitro prehitela uradno aplikacijo (povprečno 4,6 zvezdic v Trgovini Play, v primerjavi z 2,0 za aplikacijo Urbana), med pozitivnimi komentarji pa je pogosto omenjen ravno dostop do lokacij. Leta 2023 je bila aplikacija objavljena pod odprtokodno licenco, kmalu za tem pa je nastala tudi neuradna različica aplikacija za Maribor.

Aplikacija *Točen.si*, razvita v Laboratoriju za grafiko in multimedije na Fakulteti za računalništvo in informatiko ima najširšo pokritost Slovenije izmed aplikacij, ki so trenutno na voljo. Vsebuje statične in realnočasovne informativne podatke iz avtobusnega omrežja LPP, car-sharing ponudnika *Avant2Go* ter sistemov mestnih koles *BicikeLJ*, *MBajk*, *Nomago Bikes* in *Mobiln.si*. Podatki so prikazani na zemljevidu, po katerem aplikacija omogoča tudi iskanje, samodejno načrtovanje poti pa ni na voljo. Aplikacija izstopa tudi po dejstvu, da je na voljo na vseh mobilnih in namiznih operacijskih sistemih, saj je zgrajena v ogrodju *Flutter*. Iz istega razloga pa na starejših ali počasnejših mobilnih napravah v spletnih vmesnikih ne deluje optimalno.

### 3 IMPLEMENTACIJA AGREGATORJA PODATKOV

Načrtovanje agregatorja podatkov smo začeli z definicijo skupnega modela podatkov, ki bo lahko zajel vse podatke vseh ponudnikov, ki jih nameravamo pokrivati. Ker med seboj nimata dosti skupnega, bomo ločeno obravnavali sisteme souporabe (*car-, bike-, scooter-sharing*) ter sisteme linijskega javnega prevoza.

Zaradi enostavnejšega uvoza in izvoza podatkov ter integracije z zunanjimi orodji smo posebno težo dali obstoječim odprtim standardom na teh področjih. Najpogosteje uporabljen standard za izmenjavo podatkov o linijskem potniškem prometu je *General Transit Feed Specification (GTFS)*, ki ga je leta 2006 razvil Google za uporabo v aplikaciji Google Maps, trenutno pa njegov nadaljni razvoj koordinira ne-profitna organizacija *MobilityData*. Zraven tega pa se v Sloveniji v medkrajevnem prevozu uporablja tudi podatkovni model sistema integriranega javnega potniškega prometa (IJPP). Sistem IJPP sicer ima funkcijo za izvoz podatkov po standardu GTFS, a ti podatki ne vsebujejo vseh potrebnih informacij.

Na podlagi standardov GTFS in IJPP smo definirali podatkovni model za linijski javni prevoz (Tabela 1).

Tabela1: Preslikovalna tabela entitet za linijski javni prevoz

entiteta	GTFS	IJPP
Stop Location	stop	Postajna Tocka
Stop Group	stop	Postajalisce
Operator	agency	Prevoznik
Route Route Group	route	Vozni Red
Trip	route (parent) trip	Voznja
StopTime	stop_time	VoznjaOpis
Schedule	calendar	Rezim
Timetable Period	calendar (date)	Rezim Interval
Timetable Exception	calendar (exceptions)	Rezim Termin
Vehicle	Vehicle Update	
Vehicle Location		

V svetu souporabe vozil obstaja le en odprt standard – General Bikeshare Feed Specification (GBFS). Razvila ga je ameriška skupina ponudnikov sistemov deljenja koles North American Bikeshare Association (NABSA), kasneje pa prav tako predala organizaciji *MobilityData*. Na podlagi tega standarda smo definirali podatkovni model za souporabo vozil (Tabela 2).

Tabela2: Preslikovalna tabela entitet za souporabo vozil

entiteta	GTFS	IJPP
StopLocation	stop	PostajnaTocka
StopGroup	stop	Postajalisce
Operator	agency	Prevoznik
Route	route	VozniRed
RouteGroup	route (parent)	
Trip	trip	Voznja
StopTime	stop_time	VoznjaOpis
Schedule	calendar	Rezim
TimetablePeriod	calendar (date)	RezimInterval
TimetableException	calendar (exceptions)	RezimTermin
Vehicle	VehicleUpdate	
VehicleLocation		

#### 3.1 Moduli za pretvorbo podatkov

Za prenos podatkov v sistem smo razvili množico modulov za pretvorbo podatkov, ki skrbijo za uvoz podatkov iz strežnikov ponudnikov. Vsak modul je implementiran kot ločena aplikacija v ogrodju Django, poznamo pa dve vrsti: uvozne in izvozne module. Za vsakega izmed ponudnikov podatkov smo implementirali en uvozni modul, ki praviloma definira eno ali več periodičnih opravil, ki se samodejno



izvajajo na strežniku. Za hrambo dodatnih podatkov lahko uvozni moduli definirajo tudi lastne entitete, prav tako pa je možna uporaba n:1 povezav s primarnimi entitetami (na primer eno postajališče se lahko nahaja v bazi kot Marprom\_StopLocation in IJPP\_StopLocation, oba objekta pa, po združevanju, s tujim ključem kažeta na le en objekt Vehicle).

Posodobitvene intervale smo določili na podlagi narave podatkov in zmogljivosti vira, od koder jih prenašamo. Podatki o voznih redih se ne spreminjajo pogosto, zato se uvoz podatkov o linijah iz IJPP in LPP strežnikov izvede le enkrat tedensko (Marprom podatke pošilja ročno, zato posodobitvenega intervala ni). Za prenos podatkov v realnem času smo se odločili za polovico intervala, s katerim se podatki posodablja na strani ponudnika (na primer za LPP je to 5 sekund).

Podatki v realnem času so po prenosu potrebovali še dodatno obdelavo. Zraven GPS koordinat se hranijo še čas prejema pozicije, ID trenutne vožnje in smer vožnje vozila. Za vire, ki ne vsebujejo podat-

ka o smeri vožnje, se ta izračuna na podlagi prejšnje prejete pozicije. Ker čas prejema pozicije ni na voljo, se kot čas upošteva čas prevzema s strežnika ponudnika, ker pa bi to pomenilo, da bi se isti podatek v bazo shranil večkrat z različnimi časi, uvozni modul preskoči lokacije, ki se v koordinatah popolnoma ujemajo z zadnjo prejeto lokacijo.

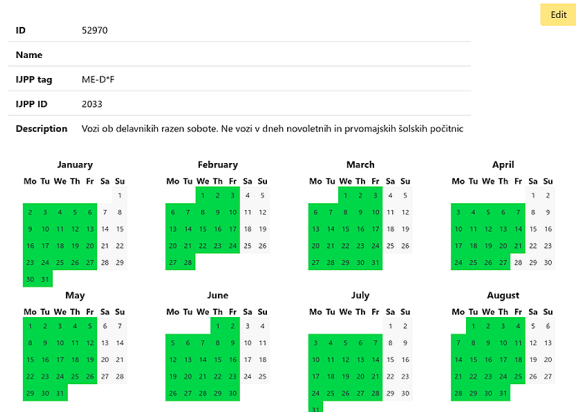
Za preverjanje točnosti podatkov smo implementirali nekaj spletnih prikazov podatkov (Slika 1) in podatke naključno primerjali z uradnimi viri in opaževanjem na terenu.

### 3.2 Združevanje postaj

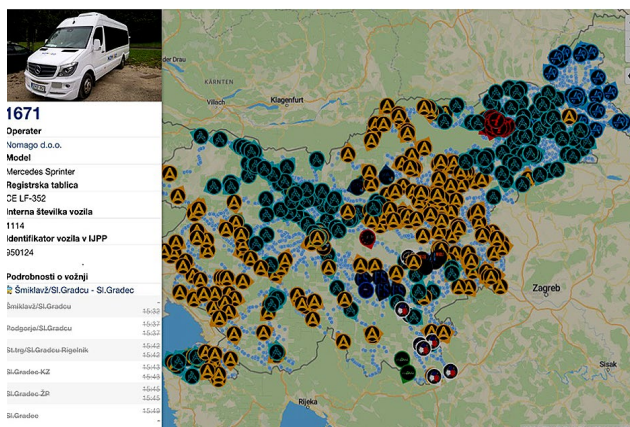
Ker agregator podatkov sprejema podatke iz mnogih različnih virov, se pogosto pojavljajo podvojeni podatki. Najpogostejši primer so avtobusne postaje, ki se uporabljajo tako v enem izmed mestnih avtobusnih omrežij, kot tudi za medkrajevne linije v sistemu IJPP. Ker različni ponudniki lokacijske podatke o postajah vnašajo na različne načine in z različno na-

Operator	Name	IJPP Sifra	Active	Vehicle count	Route count
Nomago d.o.o.	Nomago d.o.o.	A10	646	2045	
Arriva d.o.o.	Arriva d.o.o.	A22	527	877	
Javno podjetje ljubljanski potniški promet d.o.o.	Javno podjetje ljubljanski potniški promet d.o.o.	A15	290	125	
Javno podjetje za mestni potniški promet Marprom, d.o.o.	Javno podjetje za mestni potniški promet Marprom, d.o.o.	A98	95	0	
Avtobusni promet Murska Sobota d.d.	Avtobusni promet Murska Sobota d.d.	A20	73	183	
MPOV d.o.o. Vinica	MPOV d.o.o. Vinica	A34	23	29	
FS prevozi, storitve in trgovina d.o.o.	FS prevozi, storitve in trgovina d.o.o.	A57	10	20	
Prevozi Prijatelj d.o.o.	Prevozi Prijatelj d.o.o.	A23	6	7	
AP Novak d.o.o.	AP Novak d.o.o.	A97	5	4	
Avtobusni prevozi Novak bus d.o.o.	Avtobusni prevozi Novak bus d.o.o.	A30	5	5	
Prevozi Rok Jaklič s.p.	Prevozi Rok Jaklič s.p.	A42	5	24	

(a) Prikaz seznama prevoznikov



(b) Prikaz urnika vožnje na koledarju



(c) Zemljevid lokacij z odprtim oknom za vožnjo v teku



(d) Prikaz podatkov o vozilu in zgodovine lokacij

Slika 1: Zaslonske slike spletnega vmesnika agregatorja podatkov

tančnostjo se za združevanje dvojnikov ne moremo zanašati izključno na razdaljo med dvema postajama. Prav tako ni mogoča prepoznava po imenih, saj tudi ta niso enaka med ponudniki (razlike se pojavljajo predvsem pri okrajšavah, na primer »AP Ljubljana« in »Ljubljana – Glavna avtobusna postaja«).

Ker samodejno združevanje ni zanesljivo, ročno pa pri taki količini podatkov ne pride v poštev, smo se odločili za hibridni pristop. Razvili smo interaktivno aplikacijo v okolju *Jupyter Notebook*, ki za vsak par točk v podatkovni bazi izračuna razdaljo med njima ter vsak par, pri katerem je razdalja pod določeno mejo, posamezno prikaže uporabniku na zemljevidu in ga vpraša za odločitev. Če uporabnik odgovori, da sta postaji enaki, njuna identifikatorja vpiše v datoteko in prikaže naslednji par postaj. Hkrati aplikacija shranjuje seznam že pregledanih postaj, da jih izloči s seznama, ko uporabnik naslednjič požene program.

S pomočjo programa smo v približno eni uri ustvarili seznam za združitev več kot 500 postaj, ki smo ga uvozili v podatkovno bazo.

## 4 NAČRTOVALNIK POTI

Samodejno načrtovanje poti je še posebej uporabno za potnike, ki sistema javnega prevoza ne poznajo dobro, na primer turiste in potnike, ki potujejo izven domačega kraja. Omogoča enostavno izbiro točke

začetka in konca, uporabniku pa vrne vse možnosti za pot med točkama. Ker želimo skozi načrtovalnik omogočiti ne le uporabo vseh oblik mobilnosti v Sloveniji, ampak tudi njihovo poljubno kombinacijo, potrebujemo posebno vrsto načrtovalnika, t. i. multimodalni načrtovalnik. Čeprav obstaja več kvalitetnih odprtokodnih načrtovalnikov poti (*Open Source Routing Machine, Pyroute, OpenRouteService, GraphHopper* idr.), ima le eden izmed njih podporo za pravo multimodalno načrtovanje: *OpenTripPlanner*.

Projekt *OpenTripPlanner* je odprtokodna platforma za načrtovanje potovanj, ki omogoča uporabnikom, da načrtujejo poti z uporabo javnega prevoza in drugih možnosti prevoza. Projekt je bil začel leta 2009 v Portlandu (Oregon, ZDA) s podporo tamkajšnjega prevoznega podjetja *TriMet* (*Open Trip Planner project, 2023*). Že od začetka se razvija pod odprtokodno licenco *GNU LGPL*, leta 2013 je bil projekt sprejet v organizacijo *Software Freedom Conservancy*, ki skrbi za njegovo pravno in finančno upravljanje.

Preizkus delovanja načrtovalnika je možen z uporabo vgrajenega spletnega vmesnika (Slika 2), ki je dostopen na korenem spletnem naslovu strežnika OTP. Do načrtovalnika dostopa preko spletnega programskega vmesnika (API), ki je namenjen integraciji z zunanji aplikacijami.

The screenshot displays the 'My OTP Instance' web interface. On the left, three itineraries are listed, with the third one selected. The selected itinerary details are as follows:

- Start:** 2:04pm, 08/28/2023
- WALK:** 1.08 miles, 22 mins to Kranj Orehek nas.
- BUS:** Arriva d.o.o., PR A22 4201 13 (S2104)
- 2:27pm Board at Kranj Orehek nas.** (Stop #1:1121203)
- Time in transit:** 39 mins
- Route ID:** 1:32104
- Trip ID:** 1:312374
- Service Date:** 2023-08-28
- 20 Intermediate Stops**
- 3:06pm Alight at Ljubljana AP.** (Stop #1:112954)
- WALK:** 0.24 miles, 5 mins to Ljubljana
- TRAIN:** SŽ - Potniški promet, d.o.o., SŽ Z01 1490420 O (S0157)
- WALK:** 0.31 miles, 7 mins to Maribor AP
- BUS:** Arriva d.o.o., PR A22 1017 19 (S1557)
- WALK:** 0.42 miles, 9 mins to Destination
- End:** 5:53pm, 08/28/2023

The **Trip Summary** at the bottom shows:

- Travel Time:** 3 hrs, 48 mins
- Weight / Cost:** 18010
- Total Walk:** 2.05 miles
- Elevation:** 0 feet
- Gain:** 2 feet
- Loss:** 0 feet
- Transfers:** 2
- Fare:** N/A

Slika 2: Spletni vmesnik OTP, ki prikazuje pot med Kranjem in Mariborom



## 5 RAZVOJ HIBRIDNE SPLETNE APLIKACIJE

Spletno aplikacijo smo zasnovali na modularen način, ki omogoča enostavno dodajanje in spreminjanje funkcij, pri čemer obstoj in delovanje ene funkcije čim manj vpliva na obstoječe funkcionalnosti. Ker je uporabniški vmesnik, z izjemo zemljevida, zelo enostaven, smo se odločili proti uporabi JavaScript in CSS knjižnic ter vso funkcionalnost in izgled vmesnika implementirali sami (Slika 3).

Da bi bila aplikacija lažje dostopna uporabnikom, smo se odločili za implementacijo modela t.i. »hibridne spletne aplikacije«, ki jo je možno uporabljati tako v brskalniku, kot tudi namestiti kot samostojno aplikacijo. Da bi uporabnikom prepustili čim več možnosti, smo to izvedli na dva načina hkrati. Najprej smo po principu t. i. progresivnih spletnih aplikacij (*Progressive Web Apps – PWA*) napisali opisno datoteko *Web Application Manifest* ter implementirali minimalno *Service Worker* skripto, kar v podprtih brskalnikih uporabniku omogoči »namestitve« aplikacije. Za platforme, kjer progresivne spletne aplikacije niso na voljo (npr. iOS) ter za objavo v trgovine aplikacij pa smo aplikacijo zapakirali tudi z orodjem *Capacitor*.

Ker aplikacija ne potrebuje lastnega zalednega sistema, smo se odločili za popolnoma statično gostovanje datotek aplikacije. Z uporabo platforme za gostovanje izvorne kode *GitLab* in njenih funkcij *GitLab CI/CD* in *GitLab Pages* smo omogočili samodejno izgradnjo aplikacije iz izvorne kode in njeno objavo na spletu. Tako z gostovanjem spletne aplikacije nimamo dodatnih stroškov, omogoča pa nam tudi enostavno vzporedno objavo in preizkus razvojnih različic, kar bo v primeru, da projekt v prihodnosti

pridobi pozornost drugih razvijalcev, skupinski razvoj precej olajšalo.

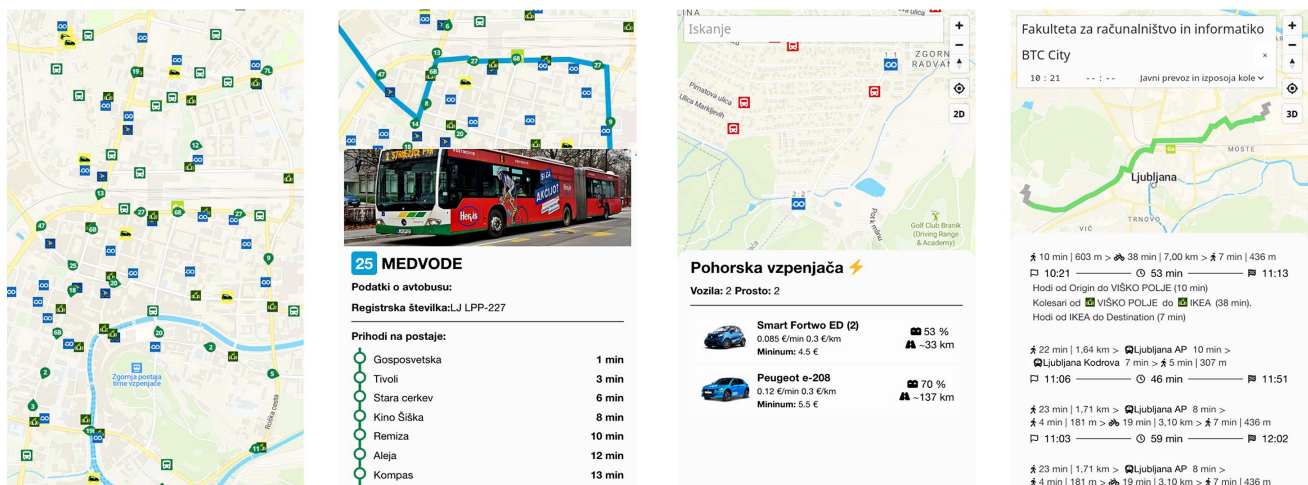
## 6 ZAKLJUČEK

V tem članku smo predstavili trenutno stanje podatkov za multimodalno planiranje v Sloveniji in implementacijo agregatorja ter mobilne aplikacije za multimodalno planiranje poti. Agregator podatkov vsebuje podatke skoraj vseh ponudnikov storitev mikromobilnosti v Sloveniji, pri čemer je edina izjema sistem souporabe skirojev Bolt. Med storitvami javnega potniškega prometa je pokritost medkrajevnega prometa zahvaljujoč sistemu IJPP popolna, med mestnimi avtobusnimi omrežji pa trenutno pokriva le Maribor in Ljubljano. Preostanek mestnih avtobusov operirajo le trije ponudniki: AP Murska Soba (podatki niso v celoti digitalizirani), Nomago in Arriva. Kljub nizki pokritosti mestnih omrežij je ta še vedno primerljiva s konkurenco (Google Maps prav tako vsebuje podatke le za Maribor in Ljubljano), na področju mikromobilnosti pa je naša pokritost nepriemerljivo višja od ostalih rešitev na trgu.

### 6.1 Nadaljnji razvoj

Med razvojem smo dobili več idej za nadaljnji razvoj agregatorja podatkov in aplikacije, ki niso bili med načrtovanimi cilji, prav tako pa smo od uporabnikov testne različice aplikacije prejeli nekaj predlogov, ki bi povečali uporabnost aplikacije.

Že v prvih tednih testiranja aplikacije smo naleteli na situacijo, ko je zaradi del na cesti avtobus vozil po spremenjeni poti in na določeni postaji sploh ni ustavljal, načrtovalnik prihodov pa jo je kljub temu



Slika 3: Zaslonske slike mobilne različice aplikacije

predlagal kot vstopno postajo. Načrtovalnik OTP omogoča uvoz podatkov o zamudah in začasno izključenih postajah v formatu GTFS-RT, ki pa ga noben izmed slovenskih avtobusnih prevoznikov še ne objavlja. Prav tako tudi na uradnih programskih vmesnikih, ki smo jih prejeli od prevoznikov, podatkov o obvozih ni, saj jih objavljajo le v obliki novic na spletni strani in jih nekateri sploh ne vnašajo v svoje sisteme za vodenje prometa.

Aplikacija je v trenutni obliki najbolj prilagojena iskanju informacij pred začetkom poti, ko je uporabnik enkrat na avtobusu ali vlaku pa je dosti težje dobiti informacije o prihodnjih postajah, možnostih prestopa in morebitnih zamudah. Aplikaciji želimo v prihodnosti dodati funkcijo »sledenja« vozilu, ki bi uporabniku omogočila hitri dostop do informacij o poti v poteku ter ga samodejno obveščala o spremembah. Eden izmed uporabnikov je predlagal tudi možnost nastavitve alarma, ki bi uporabnika obvestil nekaj minut pred prihodom na ciljno postajo.

Uporaba teh funkcij bi bila še lažja, če bi aplikacija lahko samodejno zaznala, da je uporabnik na določenem avtobusu ali vlaku. Najlažja implementacija bi bila z uporabo GPS lokacije telefona, kar pa ne bi delovalo v mestnem prometu, kjer se različni avtobusi nahajajo precej blizu med seboj.

Najbolj zanesljivo zaznavanje bi bilo možno z uporabo brezžičnih oddajnikov, ki bi po protokolu Bluetooth Low Energy na vozilih oddajali identifikator vozila, kar pa bi zahtevalo precejšnjo investicijo in sodelovanje prevoznikov. Cenejša možnost bi bila uporaba že obstoječih brezžičnih naprav na vozilih, od katerih lahko aplikacija zahteva nek enolični identifikator. Na avtobusih, ki uporabljajo najnovejšo generacijo validatorjev podjetja Margento, je možna identifikacija vozila po BLE MAC naslovu validatorja, na vlakih in avtobusih z Wi-Fi omrežji za potnike pa je identifikacija možna po MAC naslovu dostopne točke. Zaradi umetnih omejitev upora-

ba MAC naslovov ni mogoča na napravah podjetja Apple, vzdrževanje tabele MAC naslovov vozil pa bi zahtevalo precej terenskega dela, zato se za ta pristop nismo odločili.

## LITERATURA

- [1] Avant2Go. (2023). *Ljubljana / Avant2Go*. <https://avant2go.si/car-sharing/cities/ljubljana>
- [2] Brakewood, C., Barbeau, S., & Watkins, K. (2014). An experiment evaluating the impacts of real-time transit information on bus riders in Tampa, Florida. *Transportation Research Part A: Policy and Practice*, 69, 409–422. <https://doi.org/10.1016/j.tra.2014.09.003>
- [3] Društvo za elektronsko in računalniško pismenost. (2023). *Souporaba koles – Wiki - oJPP Docs*. Pridobljeno 14. avgusta 2023, <https://gitlab.com/derp-si/ojpp-docs/-/wikis/Souporaba-koles>
- [4] Nijland, H., & van Meerkerk, J. (2017). Mobility and environmental impacts of car sharing in the Netherlands [Sustainability Perspectives on the Sharing Economy]. *Environmental Innovation and Societal Transitions*, 23, 84–91. <https://doi.org/10.1016/j.eist.2017.02.001>
- [5] Ogrin, M., & Dovečar, M. (2014). Vrednotenje sistemov javnega potniškega prometa v izbranih občinah Slovenije. *Dela*, 115–127. <https://doi.org/10.4312/dela.42.115-127>
- [6] Oorni, S., Vehvilainen, J., & Lappalainen, L. D. I. (2001). FINNISH CITY CARD PROGRAMME 1993-2000. <https://api.semanticscholar.org/CorpusID:106481841>
- [7] Open TripPlanner project. (2023). *History – OpenTripPlanner*. Pridobljeno 14. julija 2023, <https://docs.opentripplanner.org/en/dev-2.x/History/>
- [8] Peng, H., Nishiyama, Y., & Sezaki, K. (2021). Estimation of Greenhouse Gas Emission Reduction from Shared Micro-mobility System. *2021 IEEE Green Energy and Smart Systems Conference (IGESSC)*, 1–6. <https://doi.org/10.1109/IGESSC53124.2021.9618701>
- [9] Stregar, A. (2019). Po osmih letih je Bicikelj vedno večji hit. *Delo*. Pridobljeno 14. avgusta 2023, <https://www.delo.si/lokalno/ljubljana-in-okolica/po-osmih-letih-je-bicikelj-vedno-vecji-hit/>
- [10] Watkins, K. E., Ferris, B., Borning, A., Rutherford, G. S., & Layton, D. (2011). Where Is My Bus? Impact of mobile real-time information on the perceived and actual wait time of transit riders. *Transportation Research Part A: Policy and Practice*, 45(8), 839–848. <https://doi.org/10.1016/j.tra.2011.06.010>
- [11] Zhang, Y., & Mi, Z. (2018). Environmental benefits of bikes haring: A big data-based analysis. *Applied Energy*, 220, 296–301. <https://doi.org/10.1016/j.apenergy.2018.03.101>

**Miha Frangež** je magistrski študent na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Je vodja iniciative *Odprt javni potniški promet*, ki se zavzema za večjo odprtost in dostopnost podatkov ponudnikov storitev mobilnosti.

**Matevž Pesek** je docent in raziskovalec na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, kjer je diplomiral (2012) in doktoriral (2018). Od leta 2009 je član Laboratorija za računalniško grafiko in multimedije. Njegovi raziskovalni interesi so iskanje glasbenih informacij, vključno z glasbenim e-učenjem, biološko navdihnjeni modeli in globoke arhitekture. Raziskuje tudi večmodalno zaznavanje glasbe, nove vmesnike in interakcijo ter vizualizacijo podatkov v obliki mobilnih, spletnih in VR aplikacij.

# Uporaba pogovornega robota za podporo intranetu: primer Elektro Ljubljana

Živa Hudobivnik<sup>1</sup>, Maja Savinek<sup>1</sup>, Marina Trkman<sup>2</sup>

<sup>1</sup>Elektro Ljubljana, d. d., Slovenska cesta 56, 1000 Ljubljana, <sup>2</sup>Univerza v Ljubljani, Fakulteta za upravo, Gosarjeva 5, 1000 Ljubljana  
ziva.hudobivnik@elektro-ljubljana.si, maja.savinek@elektro-ljubljana.si, marina.trkman@fu.uni-lj.si

## Izvleček

Ena izmed pomembnih tehnologij, ki so ključne za današnjo popularnost pogovornih robotov, so veliki jezikovni modeli. Ti pogovornim robotom omogočajo, da lahko s pomočjo kontekstualnega spomina nudijo podporo v dialogu, ki je na las podoben tistemu med ljudmi. Pogovorni robot prinaša koristne strankam na področju zabave, industrije in izobraževanja. Le malo člankov obravnava uporabo pogovornega robota kot intraneta podjetja. Namen članka je predstaviti potek načrtovanja pogovornega robota v podjetju Elektro Ljubljana. Prototip Belka smo ustvarili na modelu GPT-3.5-turbo. S pomočjo fokusnih skupin smo pripravili njeno primarno bazo znanja. Skupina zaposlenih je Belko testirala in ugotavljala, ali lahko nudi oporo intranetu podjetja. Na podlagi ugotovitev, smo pripravili seznam uporabniških zahtev za profesionalen razvoj pogovornega robota Belke. Študija primera je lahko v pomoč podjetjem, ki se odločajo za uporabo pogovornega robota kot intraneta podjetja, saj nudi unikaten vpogled v njegovo implementacijo.

**Ključne besede:** pogovorni robot, ChatGPT, intranet, fokusne skupine

## The use of conversational robots to support intranet: a case study in Elektro Ljubljana

### Abstract

One of the important technologies that are key to the popularity of conversational robots today are the large language models. These allow conversational robots to use contextual memory to support dialogue in a way that is similar to that between humans. Conversational robots benefit customers in entertainment, industry and education. Few papers have addressed the use of a conversational robot in a corporate setting as support to intranet. The aim of this paper is to present the process of designing a conversational robot in Elektro Ljubljana. The Belko prototype was created based on the GPT-3.5-turbo model. We used focus groups to prepare its primary knowledge base. A group of employees tested Belko to see if it could support the company's intranet. Based on the findings, a list of user requirements for the professional development of the Belka conversational robot was drawn up. The case study can be useful to companies deciding to use a conversational robot as a company intranet, as it provides a unique insight into its creation.

**Keywords:** chatbot, ChatGPT, intranet, focus groups

## 1 UVOD

V zadnjem času pogovorni roboti in njihove napredne zmožnosti pritegujejo veliko pozornosti. Pogovorni robot je pogovorni agent, ki omogoča dostop do storitev in informacij s pomočjo interakcij z uporabnikom v naravnem jeziku ter tako nudi pomoč pri opravljanju vsakodnevnih nalog [1]. V prihodnosti

bodo pogovorni roboti prisotni na različnih področjih [2]. Gartner [3] napoveduje, da bodo pogovorni roboti postali primarni kanal za podporo strankam za približno četrtino vseh organizacij.

Podjetja pri uvajanju pogovornih robotov stremijo k povečanju produktivnosti, saj pogovorni roboti pomagajo pri pridobivanju pravočasne in učinko-

vite pomoči [4]. Zato je smiselno razmisliti o uporabi pogovornih robotov za pomoč zaposlenim pri vsakdanjem delu v podjetjih. Kot opisujejo Casillo in drugi [5], je uporaba pogovornega robota kot način za izobraževanje novih zaposlenih učinkovita in pri naša številne prednosti, kot je poenostavitev učenja zaposlenih.

**Namen** članka je pridobiti razumevanje o enostavnosti privzemanja znanega orodja GPT-3.5-turbo (na katerem temelji ChatGPT) kot pogovornega robota, katerega namen je podpreti informacijske potrebe zaposlenih znotraj slovenskega podjetja. Odgovorili bomo na sledeča **raziskovalna vprašanja**:

- 1) Katere vsebine in znanja bi lahko imel pogovorni robot v svoji bazi znanja, da bi pomagal zaposlenim pri izvajanju vsakodnevnih nalog?
- 2) Ali delovanje znanega orodja pogovornega robota GPT-3.5-turbo ustreza potrebam intraneta slovenskega podjetja?

Ker so vsebine in znanja pogovornega robota tesno povezani s kontekstom uporabe, smo se povezali s podjetjem Elektro Ljubljana. **Cilji** raziskave so sledeči: 1) z izvedbo fokusnih skupin med zaposlenimi pridobiti razumevanje, katere informacijske potrebe bi lahko podprl pogovorni robot, 2) pripraviti prototip pogovornega robota na znanem orodju GPT-3.5-turbo za podporo interne komunikacije; 3) pridobiti povratno informacijo uporabnikov.

Struktura članka je sledeča. Članek v drugem poglavju predstavi področje pogovornih robotov in v tretjem pristop k njihovem načrtovanju. V četrtem poglavju predstavimo konkretno problemsko domeno podjetja Elektro Ljubljana, izvedbo fokusnih skupin ter pripravo prototipa pogovornega robota in rezultate njegovega testiranja. V zadnjem poglavju podamo ključne ugotovitve, ki bodo v pomoč tudi drugim podjetjem, ki se bodo odločila za uporabo pogovornega robota za potrebe intraneta.

## 2 POGOVORNI ROBOTI

Pogovorni robot (angl. chatbot) je računalniški program, ki prek glasovnih ukazov ali besedilnih sporočil simulira človeški pogovor. Obstaja več različnih izrazov, s katerim poimenujemo pogovorne robote. To so na primer pogovorni bot (angl. talkbot), bot, bot s takojšnjim sporočanjem (angl. Instant messaging bot), interaktivni agent (angl. interactive agent) ali umetna entiteta za pogovor (angl. artificial conversa-

tion entity) [6]. Pomembna tehnologija, ki povečuje popularnost pogovornih robotov, so veliki jezikovni modeli (angl. large language models – LLM). LLM pogovornim robotom omogoča, da lahko s pomočjo kontekstualnega spomina nudijo podporo v pogovoru, podobno kot ljudje [7].

Prvi znani pogovorni robot Eliza je bil razvit leta 1966 [8]. Gre za program, ki omogoča pogovor v naravnem jeziku z računalnikom. Analizira vhodne stavke s pomočjo pravil razstavljanja na podlagi ključnih besed, ki se pojavljajo v besedilu. Odgovore vrača s pomočjo pravil ponovnega sestavljanja [9]. ELIZA je uporabljala preprosto ujemanje vzorcev in odgovore na podlagi predloge. Njena pogovorna zmožnost sicer ni bila visoka, vendar je v času, ko ljudje še niso bili vajeni računalnikov, lahko zavedla uporabnike. Skozi leta so se razvijale vse bolj zmogljive različice pogovornih robotov. V letu 1972 je nastala izboljšana različica pogovornega robota ELIZA po imenu PARRY. V letu 1995 je bila razvita ALICE. Kasneje so se razvili pogovorni roboti, ki so bili dostopni prek dopisovalnih aplikacij, na primer pogovorni robot SmarterChild v letu 2001. Za tem pa je sledil razvoj virtualnih osebnih asistentov, kot denimo Apple Siri, Google Assistant in Amazon Alexa [8].

Pogovorne robote se uporablja za različne namene. Brandtzaeg in Følstad [4] ugotavljata, da v večji meri ljudje uporabljajo pogovorne robote za povečanje produktivnosti posameznika. V manjši meri jih uporabljajo tudi za razvedrilo ter za družabne stike, saj lahko pogovorni robot pomaga pri tem, da se ljudje ne počutijo osamljeni. Ljudi tudi zanima, kakšne zmožnosti in omejitve ima pogovorni robot. Georgescu [10] pravi, da so koristi pogovornih robotov predvsem na področju razvedrila, industrije in izobraževanja. Pogovorni robot lahko na primer priporoča, kako se obleči, in svetuje glede potovanja. V industriji je bistveno povezovanje strank s podjetji, predvsem za storitve za stranke, deljenje vsebine ali partnersko trženje. Omogočajo zmanjšanje stroškov v klicnih centrih, prevzemajo zamudne naloge zaposlenih in omogočajo dostopnost na internetu 24 ur na dan, vse dni v tednu. Na področju izobraževanja so lahko pogovorni agenti koristni, če uporabniku pomagajo pri reševanju težav in doseganju ciljev. Zanimanje za pogovorne robote je začelo naraščati v letu 2016, še bolj pa v letu 2023, ob množični uporabi ChatGPT-ja.



### 3 NAČRTOVANJE POGOVORNIH ROBOTOV

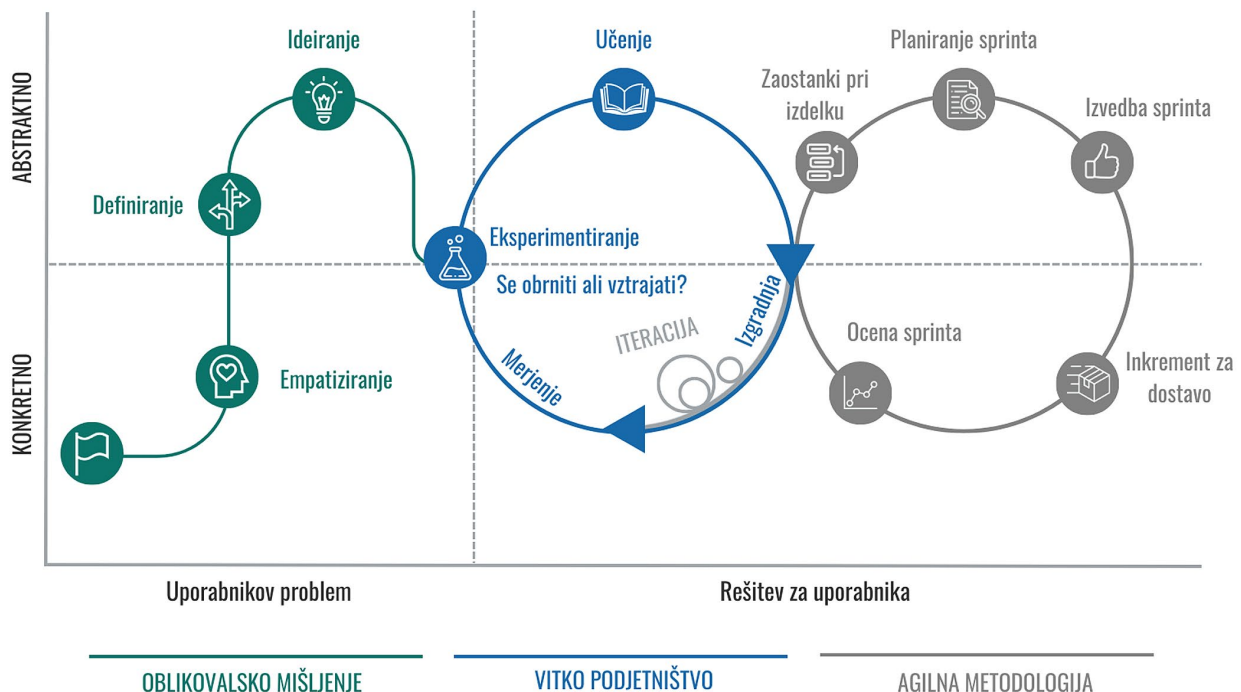
Življenjski cikel razvoja programske opreme (angl. software development life cycle) tipično sestavljajo: planiranje, analiza, načrtovanje in implementacija [11]. Literatura ponuja več različnih modelov pristopov k razvoju programske opreme, kot so: model slapa, V-model, agilni model, inkrementalni model, iterativni model in spiralni model. Gartner [12] predlaga združevanje iterativnih in eksperimentalnih pristopov za načrtovanje inovativnih digitalnih produktov. Agilna metodologija, ki se nanaša na ustvarjanje produktov ob nenehnih spremembah s postopnimi inovacijami, je bila tako izboljšana z uporabo oblikovalskega mišljenja (angl. design thinking), ki je celovit pristop raziskovanja in identificiranja problema uporabnikov ter poudarja verificiranje rešitve s prototipi. Obema pristopoma je dodana še vitka metodologija (angl. lean methodology) oziroma pristop vitkega podjetništva (angl. lean startup), ki izpostavlja stalno pridobivanje povratnih informacij ciljnih uporabnikov. Celosten Gartnerjev pristop (Slika 1) izpostavlja ciklični tok med idejami in izdelkom ter med problemom in rešitvijo [13].

Pri načrtovanju pogovornega robota je pomembno, da sledimo sedmim ključnim korakom [14]:

- Ustvari se osebnost pogovornega robota, ki najbolje predstavlja podjetje in da ga uporabniki doje-

majo kot človeka. Lahko se določi na primer ime pogovornega robota, ali bo vljuden, profesionalen ali duhovit.

- Izbere se pravi ton in jezik, ki lahko pomembno vplivata na uporabniško izkušnjo. Pogovorni robot je lahko večjezičen, lahko je uraden ali pa se pogovarja bolj sproščeno.
- Določi se delovne tokove in scenarije, kar med drugim vključuje oblikovanje odzivov na različne scenarije. Po možnosti se lahko integrira tudi baza znanja, iz katere se bo učil.
- Izbere se orodja in elemente za izboljšanje uporabniške izkušnje in avtomatizacijo rutinskih procesov, to so na primer integrirani koledarji za rezervacije storitev in možnost hitrega odgovaranja uporabnikov z enim klikom.
- Vključi se možnosti posredovanja človeškemu agentu. Alternativna možnost je preusmeritev do obrazca, kjer uporabnik lahko podrobno pojasni svoje vprašanje in mu kasneje pomaga oseba, odgovorna za podporo.
- Vključi se metrike za merjenje uspešnosti. To je pomembno za spremljanje, ali pogovorni robot dosega svoj namen in kako ga dojemajo uporabniki.
- Glede na uspešnost se izvaja nadaljnje spremembe, da je pogovorni robot bolj uspešen.



Slika 1: Pristop k načrtovanju programske opreme [12].

Spletna stran ChatBot [15], ki ponuja platformo za izdelavo pogovornih robotov, opisuje najboljše prakse in korake za izdelavo pogovornega robota:

1. faza: načrtovanje. V fazi načrtovanja opredelimo težave, ki jih želimo rešiti. To so lahko težave s časom, proračunom, ekipo ali pa z uporabniki. Na podlagi teh se odločimo, ali bo naš pogovorni robot služil podpori, prodaji ali marketingu. Nato izberemo komunikacijske kanale, na katerih bo dostopen naš pogovorni robot, in izberemo pravo rešitev za pogovornega robota. To se najlažje ugotovi s testiranjem različnih produktov oziroma platform za izdelavo pogovornih robotov. Nato glede na težave, ki smo jih identificirali, poiščemo funkcije, ki bi nam jih pomagale rešiti, in izberemo ustrezno platformo. Določimo tudi kazalnike uspešnosti za spremljanje učinkovitosti, izberemo zgodbo pogovornega robota in oblikujemo scenarij.
2. faza: izdelava. V fazi izdelave izberemo pozdrave pogovornega robota. Ustvarimo tudi osebnost pogovornega robota, kjer je pomembno, da pogovorni robot lahko služi tudi kot maskota podjetja. To storimo zato, da ne bodo uporabniki mislili, da govorijo s človekom, in kasneje ugotovili, da to ne drži. Določi se tudi ton pogovornega robota, glede na to, za kakšno skupino ljudi je namenjen. Pokažemo tudi njegovo vrednost, in sicer s tem, da pogovorni robot že na začetku vedno obvesti uporabnike, kako jim lahko pomaga. Pogovorni robot izjemno hitro pridobi informacije, vendar je dobro, da uporabimo nekaj zamika med sporočili, da celotna izkušnja pogovora izpade bolj naravno. Treba je sestaviti tudi pravo nadomestno sporočilo, če pogovorni robot ne ve odgovora, pri čemer je pomembno, da se izognemo splošnim odgovorom, kot je na primer »ne razumem«. Naradimo več različnih vrst nadomestnih sporočil in preusmerimo uporabnikovo pozornost k drugim možnim rešitvam, kot je povabilo uporabniku, da preoblikuje vprašanje, ali pa preusmeritev k drugi vrsti pomoči. Pogovornega robota testiramo, najbolje s testnimi uporabniki.
3. faza: nadzor. V tej fazi spremljamo opredeljene kazalnike uspešnosti. Opazujemo, katere interakcije so najbolj popularne, po čem uporabniki sprašujejo, tako jih tudi spoznamo, in po potrebi spremenimo scenarij pogovornega robota [15].

## 4 PODJETJE ELEKTRO LJUBLJANA

Elektro Ljubljana, d. d., se ukvarja z distribucijo električne energije in zagotavlja preskrbo z električno energijo za več kot tretjino Slovenije. Podjetje upravlja največje distribucijsko omrežje v Sloveniji, opravlja omrežninsko dejavnost in tržne storitve, povezane z elektroenergetsko infrastrukturo. Podjetje deluje na prehodu v nizkoogljično družbo z željo po zagotovitvi trajnostne in zelene prihodnosti. Pri tem ustvarja nova partnerstva z vsemi, ki so pripravljene sodelovati in slediti enakim ciljem. Aktivne uporabnike spodbuja tako, da jim omogoča prilagajanje porabe in uporabo lastnih virov energije. Podjetje vlaga v razvoj zaposlenih, da ti lahko razvijajo svoje talente in uporabljajo svoje znanje ter tako prispevajo k uspešnosti družbe. Slogan Elektra Ljubljana je »Mrežimo svetlo prihodnost« [16]. Podjetje veliko deluje na področju digitalizacije, saj trenutno izvaja projekt Digit Bela, ki ga sofinancira Evropska unija iz naslova Sklada za okrevanje in odpornost – Next-GenerationEU, s katerim bo digitalno preoblikovalo ekosistem Elektra Ljubljana, ki bo izboljšal uporabniško izkušnjo in podpiral energetske prehode [17].

Glede na vizijo, poslanstvo in cilje podjetja se veliko razmišlja o novih digitalnih rešitvah, ki bodo lahko izboljšale in tudi avtomatizirale različne poslovne procese v podjetju. Služba za napredno analitiko (SNA) se med drugim ukvarja z masovnimi obdelavami podatkov, pripravo podatkovnih setov in razvojem naprednih analitičnih rešitev za podporo delovnim procesom v podjetju. V SNA smo prišli do ideje, da bi zaposlenim v podjetju olajšali iskanje pomoči pri njihovem delu z razvojem pogovornega robota. S preizkušanjem in kreiranjem novih inovativnih in naprednih pristopov pomembno pripomore k razvijanju področja elektroenergetike.

Slika 2 prikazuje pretekli proces iskanja informacij za opravljanje delovnih nalog. Ko zaposleni nečesa ne ve, običajno kontaktira nekoga za pomoč, na primer sodelavca. Lahko traja precej časa, preden pridobi ustrezne informacije ali odgovor osebe, ki bi mu lahko pomagala pri njegovem vprašanju.

V podjetju želijo pomagat zaposlenim pri učinkovitem iskanju informacij potrebnih za delo. Nov pogovorni robot v podjetju bo izboljšal poslovni proces iskanja informacij za opravljanje delovnih nalog. Kot pri vsaki informatizaciji poslovnih procesov, so tudi tu predvidene spremembe v poslovanju oziroma,



bolj konkretno, pri iskanju informacij za vsakdanje delo. Slika 3 prikazuje bodoči proces, ki se ga želi doseči z uvedbo pogovornega robota. Ko zaposleni nečesa ne ve, vpraša pogovornega robota, ta pa mu takoj vrne odgovor in poda informacije. Če so iskane informacije ustrezne, je zaposleni našel pravo rešitev, če ne, pa se obrne na podporo SNA.

## 5 RAZISKOVALNA METODOLOGIJA

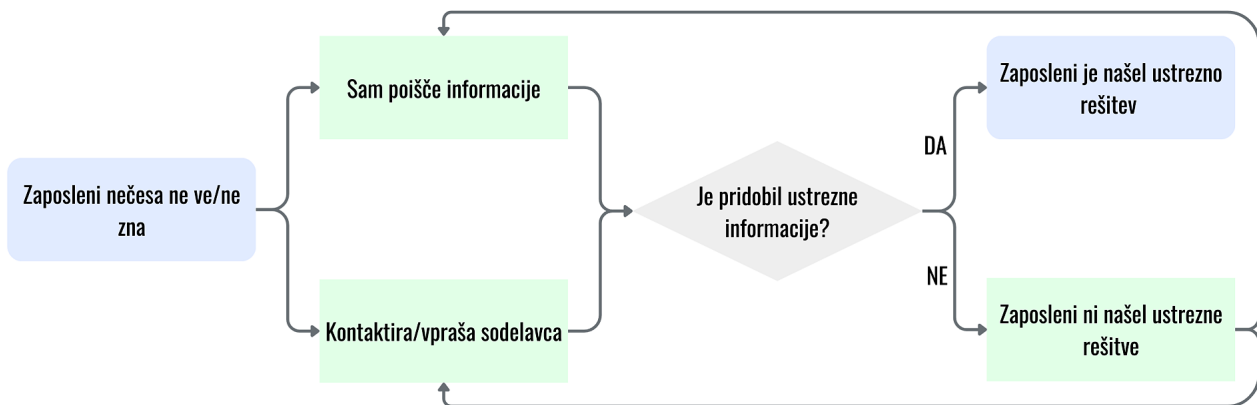
### 5.1 Raziskovalni pristop

Uporabljen raziskovalni pristop je opisan na Sliki 4. Začne se z izvedbo fokusnih skupin za prepoznavo informacijskih potreb različnih služb v podjetju. Fokusna skupina je skupina ljudi, ki so neformalno intervjuvani in sodelujejo pri kvalitativni raziskovalni tehniki [18]. Gre za družbeno-skupinsko kvalitativno metodo raziskovanja. Z njo spoznamo razmišljanja posameznika in njegova mnenja o različnih področjih iz njegovega življenja. To metodo lahko opišemo tudi kot obliko skupinskega intervjuja, kjer gre za pridobivanje podatkov predvsem na podlagi interakcije med udeleženci. S fokusnimi skupinami ne pridobimo samo podatkov o tem, kaj udeleženci mislijo o neki določeni zadevi, ampak tudi, kako in zakaj tako razmišljajo [19]. Prav zato je to ena izmed najbolj primernih metod za ugotavljanje potreb zaposlenih in smo jo izbrali za izvedbo razi-

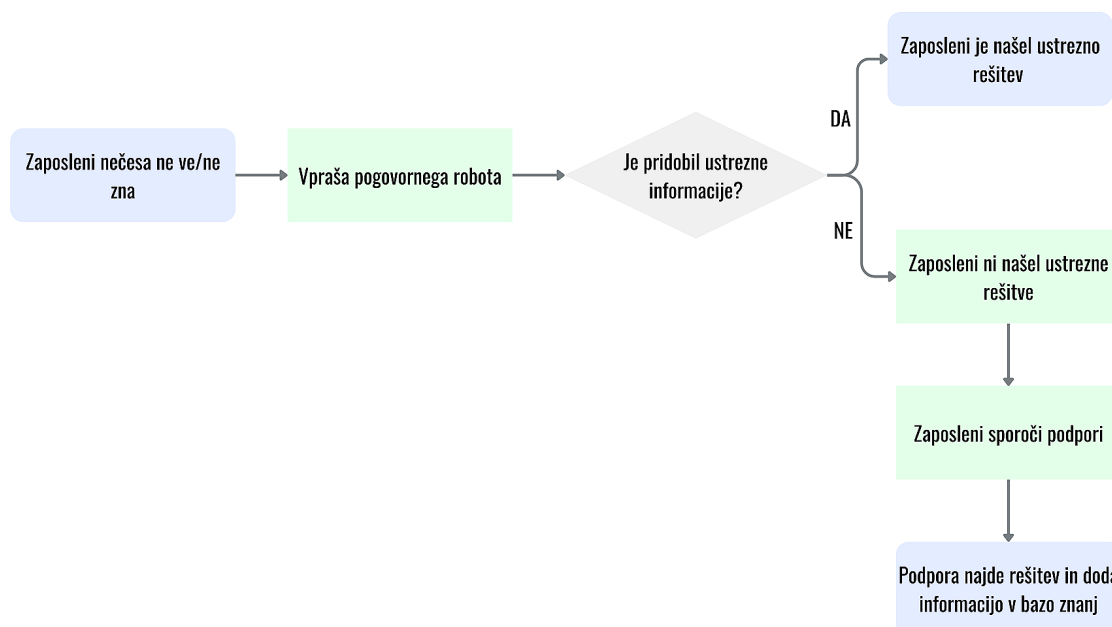
skave med zaposlenimi za načrtovanje pogovornega robota. Fokusne skupine so tako omogočile pripravo primarne baze znanja za novega pogovornega robota.

Nato je sledil korak prototipiziranja. Uporaba prototipov je v okviru oblikovalskega mišljenja prepoznana kot najučinkovitejši način za reševanje problematik, kot so interne težave pri načrtovanju, sprememba zahtev s strani stranke, težava menjave zaposlenih, kršitve specifikacij programske opreme in nizka produktivnost razvojne ekipe. Prototipi pomagajo pri nedvoumni interpretaciji uporabniških zahtev in hitrem pridobivanju povratnih informacij o produktu [20]. Lauff in drugi [21] definirajo prototip kot digitalno ali fizično utelešenje kritičnih elementov zasnove, ki je bila načrtovana in kot iterativno okolje za omogočanje učenja, sprejemanje odločitev na katerikoli točki v procesu oblikovanja in za izboljšanje komunikacije. Z oblikovanjem prototipa se izpopolni koncept izdelka, preden je izdelan, s tem pa se med drugim prihrani čas in denar. S tem se pomaga validirati zgodnje koncepte, olajša komunikacijo med osebami, ki sodelujejo pri projektu in se izboljša funkcije ter tokove [22]. Tako smo pripravili prototip na osnovi orodja GPT-3.5-turbo in mu dali v prejšnjem koraku pripravljeno primarno bazo znanja.

Medtem, ko je bil prototip pogovornega robota v uporabi med ciljnim uporabniki, so lahko ti podali



Slika 2: Pretekli procesa iskanja informacij za opravljanje delovnih nalog v podjetju Elektro Ljubljana, d. d.



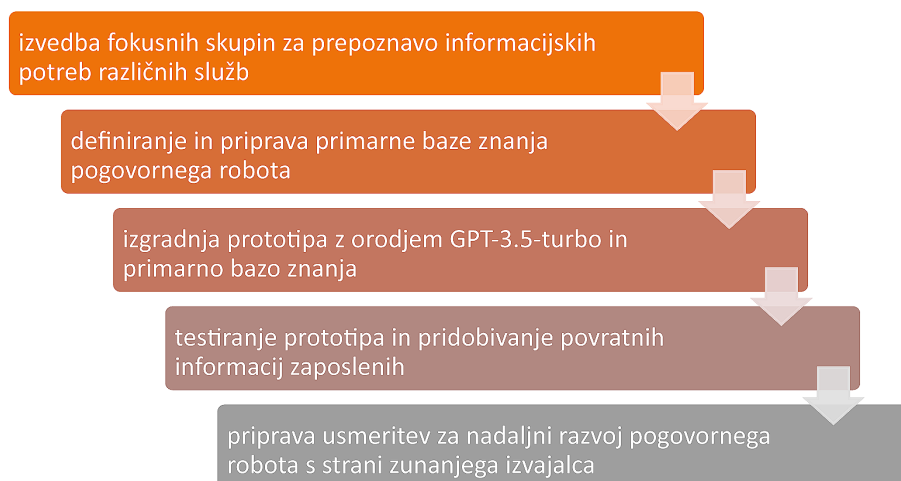
Slika 3: Bodoči procesa iskanja informacij za opravljanje delovnih nalog v podjetju Elektro Ljubljana, d. d., ki temelji na uporabi pogovornega robota

tehnične in vsebinske predloge za izboljšave. Člani tima za načrtovanje pogovornega robota so v nekaj tednih uporabe zbirali povratne informacije uporabnikov in pripravili usmeritve za nadaljni razvoj pogovornega robota s strani zunanjega ponudnika.

## 5.2 Sodelujoči v fokusnih skupinah

Pri definiranju vloge pogovornega robota v podjetju je pomembno razumeti: kaj zaposleni delajo, kaj uporabljajo pri svojem delu, kako si pri delu pomagajo, kakšno pomoč bi pri delu še potrebovali in kako bi jim lahko pri tem pomagal pogovorni robot. Vpraša-

nja za fokusne skupine in intervju smo načrtovali v treh sklopih: spoznavanje dela na oddelkih, priprava informacijskih potreb, ki bi jih lahko podprl nov pogovorni robot, in priprava seznama uporabniških zahtev pogovornega robota kot intraneta v podjetju. S pomočjo teh treh sklopov smo oblikovali scenarij fokusne skupine in intervjuja (Priloga A). Skupaj z vodjo projekta načrtovanja pogovornega robota smo izbrali tri različne oddelke v podjetju Elektro Ljubljana, d. d. in sicer: Oddelek za obračun, Služba za merjenje električne energije in Klicni center. Nato smo po e-pošti kontaktirali vodje oddelkov in jih skupaj z za-



Slika 4: Koraki raziskovalnega pristopa

poslenimi povabili k sodelovanju v raziskavi. Vsi so sprejeli povabilo za sodelovanje. Nato smo se z njimi dogovorili za izvedbo raziskave. Oblikovali smo pet fokusnih skupin (Tabela 1).

Fokusne skupine so potekale v živo, na delovnih mestih zaposlenih, torej v njihovih pisarnah. V **prvi**, **drugi** in **tretji** fokusni skupini so sodelovali zaposleni iz Službe za obračun. V **prvi** fokusni skupini so opisali, da njihovo delo v večji meri zajema kontrolo popisov števecv za gospodinjstvi odjem in poslovni odjem električne energije. Urejajo menjave števecv, spremembe plačnika, lastnika in pogodbe v povezavi s tem. Člani **druge** fokusne skupine obravnavajo odjemalce velikih organizacij s priključno močjo nad 43 kW in odjemalce električne energije, ki spadajo v kategorijo med gospodinjstvom in veleodjemom električne energije. V **tretji** fokusni skupini so opisali, da delajo evidenco uporabnikov. Občasno se ukvarjajo tudi z reklamacijami, sicer pa stik s stranko opravlja dobavitelj. V službi za obračun imajo torej razporejeno delo obračuna podatkov in vsaka skupina ljudi se ukvarja z določenim področjem obračuna. Celotna služba za obračun pri delu uporablja namenske programe, razvite za obračun električne energije, programe s podatki o uporabnikih in merilnih mestih, Microsoftova orodja in aplikacije Službe za napredno analitiko.

V **četrti** fokusni skupini so sodelovali iz Službe za merjenje električne energije. Služba je sestavljena iz več oddelkov. En oddelek se ukvarja z zbiranjem podatkov daljinskih števecv. Obvladujejo podatke 15-minutnih meritev električne energije in drugih parametrov ter iščejo anomalije. Drugi oddelek skrbi za obvladovanje merilnih mest ter prevzemanje in izdajanje navodil vsem monterjem na terenu. Tretji oddelek je Oddelek za kontrolo, ki se ukvarja se z razvojnimi projekti in skrbijo, da se dobi čim več podatkov z merilnih naprav ter z daljinskim nadzorom nad števci. Celotna Služba za merjenje električne

energije pri svojem delu uporablja sisteme za zajem merilnih podatkov in programe za njihovo obdelavo, za pripravo obračunskih podatkov, za parametriranje podatkov in za nadzor delovnih nalogov ter aplikacije Službe za napredno analitiko.

V **peti** fokusni skupini je sodeloval klicni center. So prva linija za uporabnike. Rešujejo težave uporabnikov po telefonu, nekateri pa odgovarja na e-pošto. Med drugim rešujejo raznorazna vprašanja, zahteve, soglasja, pogodbe in naročilnice. Ob javljanju uporabnikov o napakah in nedelovanju elektrike pošiljajo tudi monterje na teren. Vodja službe skrbi tudi za določene informacije uporabnikom na spletni strani podjetja. Pri svojem delu uporabljajo sistem klicnega centra, dokumentni sistem, aplikacije s podatki o merilnih mestih in uporabnikih ter aplikacije Službe za napredno analitiko.

### 5.3 Analiza fokusnih skupin

Na začetku so udeleženci diskutirali uporabo informacijskega sistema podjetja. Udeleženec izpostavi pomemben vidik **iskanja informacij** in sicer karakter zaposlenega: »Kako bo nekdo iskal informacije je odvisno od njega samega. Nekdo je malo bolj proaktiven, bo šel brskat, bo prišel do cilja, drug bo pa ali niti ne bo iskal ali bo pa mislil, da tega ni na voljo.« Podobno izpostavi udeleženka iz tretje fokusne skupine: »Odvisno od osebe, eni to vse sami naštudirajo pa pač vidijo, kako pa kaj, eni pač rabijo še kakšno informacijo. Eni malo več, eni malo manj.« Strinjali so se, da ne potrebujejo veliko pomoči pri rabi ustaljenih aplikacij pri delu. Radi si med seboj pomagajo, imajo pa tudi napisane dokumente, kjer so popisane specifikke dela in protokoli.

V fokusnih skupinah so zaznali potencialno korist pogovornega robota **pri uvajanju novo zaposlenih**. Udeleženec to opiše z besedami: »Ko se [nekdo na novo] zaposli bi bila takšna zadeva enkratna!« V tretji fokusni skupini so udeleženci izpostavili po-

Tabela1: Pregled fokusnih skupin

Številka fokusne skupine	Služba, ki je sodelovala v fokusni skupini	Število sodelujočih
1	Služba za obračun	3
2	Služba za obračun	2
3	Služba za obračun	3
4	Služba za merjenje električne energije	4
5	Klicni center	2

membnost medsebojnega sodelovanja. Udeleženec je rekel: »najboljša navodila so tista, ki jih skupaj pri-delamo.« Pomoč pri uporabi licenčnih aplikacij nudijo ponudniki sami in razumeti je, da so z njihovo podporo zadovoljni. Udeleženka iz druge skupine to opiše z besedami: »Te programe v bistvu dosti obvladamo, tako da ne rabimo nobene pomoči. Zdaj, ko pa se karkoli na novo posodablja ali pa ko se na novo ureja, pa ja, takrat so navodila zelo pomembna.« Podjetje pa ima tudi lastniške aplikacije. Običajno zaposleni sodelujejo že pri razvoju aplikacije in se učijo s testiranjem. Podjetje organizira izobraževanja za končne uporabnike aplikacij. Udeleženec iz četrte fokusne skupine pravi: »V bistvu so to bolj prezentacije; predstavitev s strani proizvajalca in potem tiste najnujnejše stvari pač v sodelovanju hitro osvojiš. Če pa rabiš kaj novega, je pa najlažje poklicat ponudnika pa se z njim pogovoriti kako pa kaj [...].«

Udeleženci menijo, da bi jim prav prišla **pomoč pri interpretaciji (novih) zakonov in aktov**, denimo pri poznavanju in razumevanju novega Akta o metodologiji za obračunavanje omrežnine za elektrooperaterje. Udeleženka druge fokusne skupine sta poudarila, da v zakonih in aktih ponavadi iščejo, kaj opredeljuje določen člen. Prav bi jim prišla tudi pomoč pri razumevanju kratic, ki se uporabljajo v elektroenergetiki in v podjetju. Udeleženka iz prve fokusne skupine pravi: »Včasih ti res ni jasno, kaj so tiste kratice.« Ob vprašanju, kaj bi vprašali pogovornega robota, so se udeleženci v večji meri navezovali na zakonodajo in akte ali pa na znanja, ki niso specifična za njihovo službo. Kot opiše udeleženka iz prve fokusne skupine, je to iskanje tovrstnih informacij precej zamudno: »Jaz včasih vse sorte preiskujem, ko kaj iščem. Pa traja preden kaj najdem, sploh pa v katerem (členu/aktu) je informacija.« Drugi se strinjajo. Zakonodajo veliko uporabljajo na dnevni ravni.

Udeleženci izpostavljajo potencialno koristnost pogovornega robota za **deljenje znanja med posameznimi oddelki**. Eden to opiše z besedami: »Sploh za tisti del, ki ga mi [na našem oddelku] ne poznamo toliko.« Drugi reče: »[...] da ne bom tam enega na drugem oddelku spraševal nekaj glede soglasij! To, da bi mi robot ponudil odgovor, pa bom rekel, okej, ja, sej ne rabim ga sploh klicat, pa ga obremenjavat, ker ima mogoče že preveč dela.« S pogovornim robotom bi se lahko delilo tudi medsebojna znanja na različnih oddelkih. Tako bi se znanje preneslo tudi

na druge zaposlene, in ko bi ga potrebovali, jim ne bi bilo treba iskati osebe, ki bi jim pri tem pomagala.

Udeleženci iz klicnega centra in tisti, kjer imajo opravka z odjemalci električne energije, so predlagali razvoj **pogovornega robota tudi za stranke**, predvsem za odgovore na osnovna vprašanja, na primer: »Kaj je jalova energija?« Enako ugotavljajo v tretji fokusni skupini, kjer kot primer vprašanja uporabnikov navajajo: »Kaj potrebujem za prvi priklop elektrike?« Sicer omenijo, da so pogosta vprašanja na voljo na spletni strani podjetja, vendar bi bila tudi pomoč v obliki pogovornega robota že lahko koristna.

Nekateri zaposleni v fokusnih skupinah so imeli **željo preizkusiti njim namenjenega pogovornega robota**. Menijo, da bodo potem imeli več idej, kaj vse bi lahko še vsebovala baza znanja. Udeleženci v fokusnih skupinah so oblikovali nekaj vprašanj, ki bi jih že v tistem trenutku lahko postavili pogovornemu robotu: »Kaj je dogovorjena moč?«; »V katerem členu je kaj?«; »V katerem aktu (je kaj)?«; »Kako se naredi menjava števca?«; »Sprememba priključne moči.«; »Izklop od omrežja.« Udeleženka iz prve fokusne je tako rekla: »Ko bi ga enkrat imel in uporabljal, bi potem lažje podal mnenje, kaj nam prav pride, kaj nam manjka.« Udeleženec iz tretje fokusne skupine je povedal: »Z uporabo bi mogoče dobil še kakšno idejo.« Udeleženka ga dopolni: »Zdaj si mogoče vsak drugače robota predstavlja in to, kaj vse zmore.« Je pa udeleženka iz prve fokusne skupine izpostavila: »Potrebno se bo naučit, kako ga spraševati!« Pomembno je, da ko bo pogovorni robot razvit, da se bo zaposlenim pojasnilo, kako se ga uporablja, na kakšen način se ga sprašuje ter kaj vse obsega.

V nadaljevanju je pogovor tekkel o pogovornih robotih na splošno. Nekateri udeleženci so že preizkusili pogovornega robota, drugi vedo, da obstaja, pa ga, če se pojavi na spletni strani, ugasnejo. Večina je preizkusila pogovornega robota za podporo strankam, kjer so morali izbirati možnosti. Nekateri so bili z njim zadovoljni, drugi nekoliko manj. Udeleženka iz druge fokusne skupine opiše svojo pozitivno izkušnjo: »Jaz sem imela reklamacijo na spletni nakup. Dokler sem imela splošna vprašanja, je pogovorni robot odgovarjal, če je bilo pa kaj bolj zapletenega, te je itak povezal z dejansko osebo.« Nekateri so izpostavili dodatne omejitve pogovornih robotov. Udeleženec iz pete fokusne skupine, ki jih je nekaj preizkusil, pravi: »Ko uporabnik vidi, da v doglednem času ne pride do odgovora oziroma po dolgem času dobi na-

pako (angl. error), potem napišite mail. Vendar, uporabnik bi v bistvu lahko že v začetku email napisal.« Pri tovrstni slabi izkušnji uporabnik naslednjič takoj napiše email. Tudi udeleženec iz četrte fokusne skupine opiše svojo izkušnjo: »V bistvu te samo usmerja na njihove spletne strani [...], to je premal.« Drugi udeleženec iz četrte fokusne skupine je preizkusil ChatGPT. Izkušnja z njim je bila pozitivna.

## 5.4 Izdelava prototipa

S pomočjo fokusnih skupin smo identificirali potencialne koristi pogovornega robota: 1) kot podporo pri deljenju znanj med zaposlenimi iz različnih oddelkov znotraj podjetja, 2) kot podporo novo zaposlenemu pri spoznavanju delovnega okolja, 3) kot podporo pri informiranju strank. Podjetje se je odločilo za testiranje uporabe pogovornega robota za potrebe interne komunikacije. Primarna baza znanja prototipa pogovornega robota za podporo intranetu podjetja je zajemala izbrane teme:

- dokumente z zakonodajo in akti, ki se uporabljajo:
  - Zakon o oskrbi z električno energijo (ZOEE)
  - Energetski zakon (EZ)
  - Zakon o spodbujanju rabe obnovljivih virov energije (ZSROVE)
  - Sistemska obratovalna navodila za distribucijski sistem električne energije (SONDSEE)
  - Zakon o učinkoviti rabi energije (ZURE)
  - Akt o metodologiji za določitev regulativnega okvira za elektrooperaterje
  - Akt o metodologiji za obračunavanje omrežnine za elektrooperaterje
- enostavne razlage zakonov, pogosta vprašanja in odgovori glede zakonov in aktov
- pogosta vprašanja uporabnikov
- razlage kratic v elektroenergetiki in kratic, ki se uporabljajo v podjetju

Za izdelavo pogovornega robota je bil izbran pogovorni robot, ki temelji na procesiranju naravnega jezika. ChatGPT ni bil primeren za ta namen, saj ne more odgovoriti na vprašanja, ki se nanašajo na zasebne podatke, saj njegova baza zajema podatke, ki so v večji meri dostopni na spletu [23]. Ne more pojasniti rabe specifičnih kratic, izrazov, internih dokumentov ipd. Tako smo uporabili model GPT-3.5-turbo, na katerem je bil ChatGPT narejen, in ga prilagodili glede na našo zasebno bazo znanja ter podkrepili dialog z uporabo učenja s človeškimi povra-

tnimi informacijami. Omenjen model smo prilagodili v orodju za testiranje različnih AI modelov OpenAI Playground. Tako smo preizkusili, kako se obnaša in koliko se ga da spreminjati. Za izdelavo prototipa pogovornega robota, katerega baza znanja temelji na podlagi dokumentov, smo uporabili način, ki ga Sha [24] opisuje po korakih:

1. korak: Na računalnik naložimo Python
2. korak: Posodobimo upravitelja paketov Pip in preverimo, ali je nameščen pravilno.
3. korak: Naložimo knjižnice: OpenAI, GPT Index, PyPDF in Gradio.
4. korak: Naredimo mapo z dokumenti za treniranje pogovornega robota, ki jo napolnimo z izbranimi dokumenti PDF.
5. korak: Ustvarimo brezplačen račun na platformi OpenAI, pridobimo ključ OpenAI API in ga kopiramo.
6. korak: Naložimo in odpremo urejevalnik kode. V našem primeru smo uporabili Visual Studio Code. Ustvarimo datoteko z imenom app.py in dodamo kodo, ki jo uporabi Sha [24].
7. korak: Z ukazom se pomaknemo na lokacijo, kjer se nahajata mapa »docs« in dokument »app.py«. Nato z ukazom poženemo datoteko »app.py«. Pridobimo lokalni URL, ki ga kopiramo v brskalnik.
8. korak: Preizkusimo pogovornega robota.

Pripravili smo vizualno predstavitev Belke (Slika 5). Belka je prijazna pogovorna robotka in lepo pozdravi uporabnike ter jim ponudi pomoč. Njeni odgovori so profesionalni, saj temeljijo iz baze znanja danih dokumentov. Po pozdravu uporabnik vnese svoje vprašanje. Če Belka pozna odgovor nanj, odgovori, če ne, pa napiše, da odgovora ne pozna, saj vsebine ni v njeni bazi znanja, ter naj se uporabnik obrne na podporo SNA prek portala za zahteve. Belka je tako prisotna na vseh straneh izbranih aplikacij (Slika 6). Poleg tega je Belka na voljo tudi na ločeni nadzorni plošči, kjer so opisane njene sposobnosti. Na Sliki 7 je primer vprašanja, ki ga postavi uporabnik, in odgovora pogovornega robota, ki jih je pogovorni robot generiral s pomočjo besedila iz Energetskega zakona.

## 5.5 Analiza prototipa

V naslednjih tednih so zaposleni pri testiranju prototipa spoznali, kako zelo je pomembno, da so do-



kumenti v bazi znanja dobro strukturirani. Ugotovili so, da bo potrebno preoblikovati strukturo internih dokumentov, da bo Belka lahko hitro našla potrebno vsebino in oblikovala pravi odgovor. Izkušnje so pokazale, da Belka najbolje odgovarja, če so v bazi znanja dokumenti napisani v obliki vprašanj in odgovorov. Prav tako je potrebno dodatno predelati kontekst dokumentov. Določeni pojmi se lahko nahajajo v več različnih dokumentih in če se kontekst dokumentov ne specificira, lahko pogovorni robot poda napačen odgovor. Zato je pomembno, da se bo uporabnike izobrazilo, da bodo pri svojih vprašanjih specificirali kontekst.

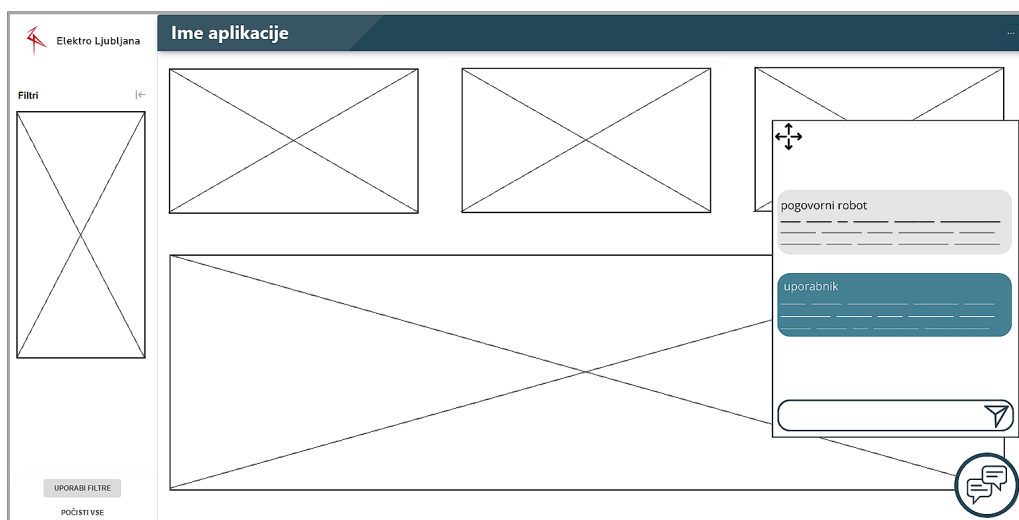


Slika 5: Slika pogovorne robotke Belke – pridobljena na iStock [25].

Podjetje je ugotovilo, da so potrebne večje spremembe osnovnega orodja, ki podpira Belko, da bi lahko podala točne in zanesljive odgovore. Organiziralo se je več sestankov z zunanjim izvajalcem. Izbrane uporabniške zahteve za pogovornega robota, ki podpira intranet podjetja, so bile:

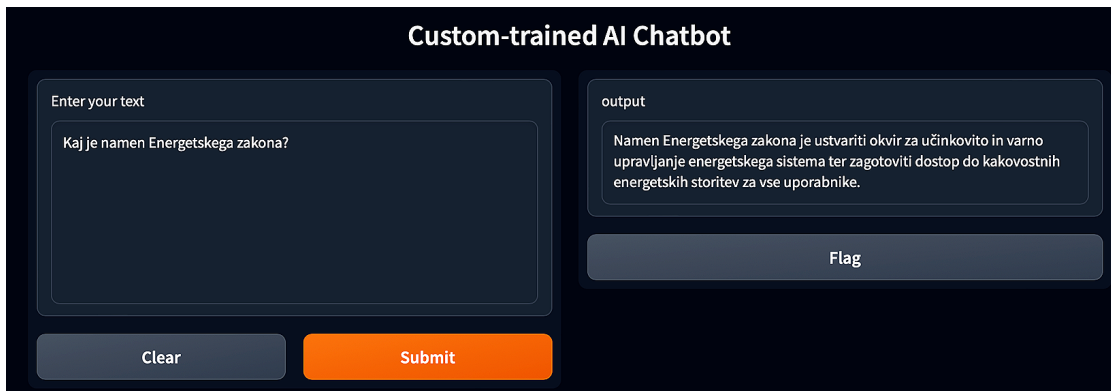
Nov pogovorni robot naj bo podoben ChatGPT-ju in bo sposoben v slovenščini odgovarjati na vprašanja uporabnikov.

- Pogovorni robot bo poznal odgovore na vprašanja o vsebini, ki se bo nahajala v bazi znanja.
- Bazo znanja predstavlja mapa z dokumenti.
- Podjetju se omogoči, da zaposleni dodajajo in brišejo dokumente v bazi znanja ter spreminjajo osnovne parametre modela.
- Pogovorni robot se integrira v platformo Apache Superset, kjer bo na voljo v vseh internih aplikacijah ter na eni ločeni nadzorni plošči, namenjeni samo pogovornemu robotu.
- Uporabi se izbrano ime Belka.
- Belka se mora vesti kot pogovorni asistent, znati pozdravljati in odgovoriti na osnovna vprašanja, kot so: kdo si, kako si, se posloviti itd., poleg tega da bo znal odgovarjati na vprašanja uporabnikov.
- Belka ima barvno shemo pogovornega okna, ki je ujemajoča z barvno shemo nadzornih plošč v SNA.
- Uporabnikom se omogoči možnost razširitve pogovornega okna, da si ga uporabniki lahko po želji povečajo ali pomanjšajo.



Slika 6: Prototip pogovornega robota v aplikaciji





Slika 6: Primer vprašanja in odgovora

- Omogoči se neprestano testiranje novih različic Belke.

## 6 DISKUSIJA IN ZAKLJUČEK

Namen članka je bil pridobiti razumevanje o enostavnosti in uspešnosti privzemanje orodja GPT-3.5-turbo kot osnovo za delovanje pogovornega robota, katerega namen je služiti interni komunikaciji zaposlenih v slovenskem podjetju. Udeleženci fokusnih skupin so sami prepoznali različne vloge, ki jih lahko pogovorni robot zasede v podjetju. Podjetje ga lahko uporabi za informiranje tako strank kot zaposlenih. Slovensko podjetje Elektro Ljubljana se je odločilo, da podpre delo zaposlenih s pogovornim robotom.

V okviru načrtovanja pogovornega robota smo dosegli vse 3 cilje. Z izvedbo fokusnih skupin smo pridobili razumevanje zaposlenih o zmožnostih pogovornega robota in tudi konkretne ideje za kaj vse bi lahko uporabljali pogovorni robot v dotičnem podjetju. Na osnovi orodja GPT-3.5-turbo smo pripravili Belko, ki je lahko uporabnikom ponujala odgovore na omejen nabor tem oziroma dokumentov. Ciljni uporabniki so jo testirali in od njih smo pridobili dragocene informacije za nadaljni razvoj Belke.

Odgovor na prvo raziskovalno vprašanje smo ugotovili, da si zaposleni v dotičnem podjetju želijo pogovornega robota, ki jim bo pomagal pri iskanju in razumevanju zakonov, aktov in kratic ter pri učenju novosti. Prav tako bi se radi s pomočjo pogovornega robota informirali o vsebinah drugih služb in oddelkov v podjetju. Pogovorni robot bo lahko pomagal novo zaposlenim pri uvajanju na novo delovno mesto. Relevantne vsebine so torej zakoni, akti, razlage, pogosta vprašanja in pomembni interni dokumenti različnih služb in oddelkov.

Odgovor na drugo raziskovalno vprašanje smo pridobili s testiranjem prototipa Belke. Ugotovili smo, da uporaba GPT-3.5-turbo tehnično pomanjkljivo zadovoljuje potrebe slovenskega podjetja pri podpori intranetu. Skladno s tem se je podjetje odločilo, da nadaljni razvoj Belke prepusti zunanjim izvajalcem. Ključne uporabniške zahteve so: 1) da bo Belka sposobna dobro odgovarjati v slovenščini, 2) da bodo zaposleni lahko dodajali dokumente v bazo znanja in 3) da bo ustrezno integrirana v obstajajoče aplikacije podjetja. Ključno pri oblikovanju uporabniških zahtev pogovornega robota kot intraneta pa je bilo sodelovanje zaposlenih v podjetju.

V bodoče se bo zaposlene večkrat prosilo za povratne informacije o uporabi izboljšanih različic Belke. Preverjalo se bo: zadovoljstvo z uporabo pogovornega robota ugotavljalo, kdaj in kje se pojavljajo napake, ter katere dokumente bi bilo še smiselno dodati v bazo znanja. Tako se bo pogovornega robota stalno prilagajalo novim zahtevam uporabnikov in redno posodabljal bazo znanja.

## 7 ZAHVALA

Raziskovalno delo soavtorice Marine Trkman je financirala Agencija za raziskovalno dejavnost Republike Slovenije (ARIS) v okviru raziskovalnega projekta J7-50185. Omenjena soavtorica deluje v okviru programske skupine P5-0399.

Živa Hudobivnik se za možnost izvedbe raziskovalnega projekta v okviru magistrske naloge in študentskega dela iskreno zahvaljuje podjetju Elektro Ljubljana, d. d., še posebej Maji Savinek in Tadeju Šinkovcu.

## LITERATURA

- [1] A. Følstad *idr.*, Future directions for chatbot research: an interdisciplinary research agenda, *Computing*, let. 103, št. 12, str. 2915–2942, 2021, doi: 10.1007/s00607-021-01016-7
- [2] M. Dahiya, A tool of conversation: Chatbot, *International Journal of Computer Sciences and Engineering*, let. 5, št. 5, str. 158–161, 2017.
- [3] Gartner predicts Chatbots will become a primary customer service channel within five years, Gartner, <https://www.gartner.com/en/newsroom/press-releases/2022-07-27-gartner-predicts-chatbots-will-become-a-primary-customer-service-channel-within-five-years>
- [4] P. B. Brandtzaeg in A. Følstad, Why People Use Chatbots, v *Internet Science*, I. Kompatsiaris, J. Cave, A. Satsiou, G. Carle, A. Passani, E. Kontopoulos, S. Diplaris, in D. McMillan, ur., Cham: Springer International Publishing, str. 377–392, 2017. [https://doi.org/10.1007/978-3-319-70284-1\\_30](https://doi.org/10.1007/978-3-319-70284-1_30)
- [5] M. Casillo, F. Colace, L. Fabbri, M. Lombardi, A. Romano, in D. Santaniello, Chatbot in Industry 4.0: An Approach for Training New Employees, v *2020 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, str. 371–376, 2020. doi: 10.1109/TALE48869.2020.9368339
- [6] J. Frankenfield, Chatbot definition, types, Pros & Cons, examples. Pridobljeno 16. januarja 2024, <https://www.investopedia.com/terms/c/chatbot.asp>
- [7] J. Chua, How will large language models (llms) change chatbots?, Simplr, <https://www.simplr.ai/blog/how-will-large-language-models-llms-change-chatbots>
- [8] E. Adamopoulou in L. Moussiades, An Overview of Chatbot Technology, v *Artificial Intelligence Applications and Innovations*, I. Maglogiannis, L. Iliadis, in E. Pimenidis, ur., Cham: Springer International Publishing, str. 373–383, 2020. [https://doi.org/10.1007/978-3-030-49186-4\\_31](https://doi.org/10.1007/978-3-030-49186-4_31)
- [9] J. Weizenbaum, ELIZA—a computer program for the study of natural language communication between man and machine, *Commun. ACM*, let. 9, št. 1, str. 36–45, jan. 1966, doi: 10.1145/365153.365168
- [10] A. A. Georgescu, Chatbots for Education - trends, benefits and challenges, *14th International Conference eLearning and Software for Education*, apr. 2018. doi:10.12753/2066-026x-18-097
- [11] S. Balaji in M. Sundararajan, Waterfall vs v-model vs agile: A comparative study on SDLC, *International Journal of Information Technology and Business Management*, let. 2., št. 1 str. 26–30, 2012.
- [12] Enterprise architects combine design thinking, Lean Startup and agile to drive Digital Innovation. Pridobljeno 16. januarja 2024, <https://www.gartner.com/en/documents/3941917>
- [13] N. Patel, Understanding how design thinking, Lean and agile work together. Pridobljeno 16. januarja 2024, <https://www.digi-corp.com/blog/understanding-how-design-thinking-lean-and-agile-work-together/>
- [14] A. Suresh, The A to Z of chatbot design: How to plan your chatbot. Pridobljeno 16. januarja 2024, <https://www.freshworks.com/live-chat-software/chatbots/chatbot-design-blog/>
- [15] Chatbot best practices. Pridobljeno 16. januarja 2024, <https://www.chatbot.com/chatbot-best-practices/>
- [16] Dovolite, da vas omrežimo. Pridobljeno 16. januarja 2024, <https://www.elektro-ljubljana.si/elektro-ljubljana>
- [17] Elektro Ljubljana s projektom Digit Bela do digitalne prenove. Pridobljeno 16. januarja 2024, <https://www.elektro-ljubljana.si/medijske-objave/ArtMID/921/ArticleID/2063/Elektro-Ljubljana-s-projektom-Digit-Bela-do-digitalne-prenove>
- [18] W. L. Neuman, *Social research methods: Qualitative and quantitative approaches*. Pearson Education Limited, 2014
- [19] Ž. Lep, Fokusne skupine, v *Kako spodbujati zaposlene: psihološki pristopi od A do Ž*, E. Boštjančič in A. Petrovčič, ur., Znanstvena založba Filozofske fakultete Univerze v Ljubljani, str. 33–49, 2020.
- [20] M. N. Orobey, A. Alchinov in I. M. Daudov, Theoretical aspects of prototyping, *Journal of Physics: Conference Series*, let. 1582, št. 1, jul. 2020. doi:10.1088/1742-6596/1582/1/012068
- [21] C. A. Lauff, D. Kotys-Schwartz in M. E. Rentschler, What is a prototype? What are the roles of prototypes in companies?, *Journal of Mechanical Design*, let. 140, št. 6, mar. 2018. doi:10.1115/1.4039340
- [22] What is prototyping?. Pridobljeno 16. januarja 2024, <https://www.figma.com/resource-library/what-is-prototyping/>
- [23] K. Muehmel, What is a large language model, the Tech behind chatgpt?. Pridobljeno 16. januarja 2024, <https://blog.dataiku.com/large-language-model-chatgpt>
- [24] A. Sha, How to train an AI chatbot with custom knowledge base using CHATGPT API. Pridobljeno 16. januarja 2024, <https://beebom.com/how-train-ai-chatbot-custom-knowledge-base-chatgpt-api/>
- [25] Cute artificial intelligence robot with notebook stock photo. Pridobljeno 16. januarja 2024, <https://www.istockphoto.com/photo/cute-artificial-in-telligence-robot-with-notebook-gm1279564226-378132643>

■

**Živa Hudobivnik** je magistrica družboslovne informatike. Magistrirala je z delom Načrtovanje pogovornega robota za pomoč zaposlenim v podjetju Elektro Ljubljana, d. d. Zaposlena je v podjetju Elektro Ljubljana, d. d., v Službi za napredno analitiko, kjer dela na področju podpore uporabnikom, analitike in uporabe umetne inteligence.

■

**Maja Savinek** je doktorska študentka na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Zaposlena je v podjetju Elektro Ljubljana, d. d., kjer se ukvarja z razvojem napredne analitike in algoritmov na področju elektroenergetike, s poudarkom na tehnologijah velepodatkov. Njeni strokovni prispevki so s področja umetne inteligence, podatkovnega rudarjenja in analize podatkov v elektroenergetiki.



**Marina Trkman** je docentka na Univerzi v Ljubljani; zaposlena na Fakulteti za upravo in Fakulteti za družbene vede. Doktorat je opravila na Fakulteti za računalništvo. Svoj podoktorski projekt je vodila na Institutu Jožefa Štefana. Raziskovalno se ukvarja: 1) s preiskovanjem dejavnikov, ki vplivajo na privzemanje informacijsko komunikacijskih tehnologij in 2) z analiziranjem poslovnih procesov za optimizacijo poslovanja. Kot prvi avtor je objavila članke v prestižnih revijah kot sta *International journal of information management in pa Government information quarterly*.

## **PRILOGA A: SCENARIJ ZA INTERVJU IN FOKUSNE SKUPINE**

### **Uvod**

*[Se predstavim in povem, da izvajam raziskavo za namen magistrskega dela z naslovom Načrtovanje pogovornega robota za pomoč zaposlenim v Elektro Ljubljana, d. d. Udeležencem razdelim obrazce z obveščnim soglasjem za sodelovanje v raziskavi. Ko si ga preberejo in podpišejo, začnem z izvedbo raziskave.]*

### **Vprašanja in podvprašanja**

#### **1. SKLOP: Delo na oddelku in raba aplikacij pri delovnih procesih**

1. Za začetek bi vas vprašala, kaj delate na tem oddelku?
2. Kaj vse pa uporabljate pri vašem delu (na primer programe, aplikacije itd.)? Opišite mi svojo izkušnjo z njimi.

#### **2. SKLOP: Pomoč pri delu**

3. Ali bi potrebovali kakšno pomoč pri rabi aplikacij in programov?
  - 3.1. Podvprašanja za pomoč: Ne znate uporabljati, ne veste, kje kaj najti, ne razumete izrazov?
4. Kaj oziroma katere informacije bi še potrebovali pri svojem delu, da bi vam bilo lažje?
  - 4.1. Podvprašanje za pomoč: Morda razlage kratic, izrazov, navodila, videi itd.?

#### **3. SKLOP: Pogovorni robot**

5. Ali veste, kaj je pogovorni robot? Kako si ga predstavljate?
  - 5.1 Za pomoč, če ne vedo, razlaga s praktičnimi primeri:  
*Ste kdaj nakupovali v kakšni spletni trgovini, na primer Big Bang, Merkur, ki ima desno spodaj klepetalnik? Ali pa na primer na Viberju ali Facebook Messengerju? Na Facebook Messengerju je na primer Občina Ljubljana imela svojega pogovornega robota.*
6. Če si predstavljate pogovornega robota v podjetju za zaposlene, kaj bi ga vprašali oziroma kaj bi želeli izvedeti?
  - 6.1. Podvprašanje za pomoč: Morda navodila, tehnična pomoč, odgovori na vprašanja itd.?
7. Se vam zdi, da bi bil pogovorni robot z bazo znanja uporaben pri vašem delu?

Na koncu vsak udeleženec prejme simbolično nagrado.

# ■ Naslavljanje DNA-pomnilnikov

Lidija Stanovnik, Miha Mraz

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Večna pot 113, 1000 Ljubljana

lidija.stanovnik@fri.uni-lj.si, miha.mraz@fri.uni-lj.si

## Izveček

Potrebe po cenovno učinkovitih pomnilnih medijih za dolgotrajno arhiviranje podatkov so povzročile razvoj nekonvencionalnih platform, med katere sodi tudi trajno pomnjenje podatkov v DNA. Z vsakim novim medijem pa se pojavijo tudi novi problemi. Zaradi drugačnih lastnosti bioloških sistemov se v mediju DNA pojavljajo drugačni tipi napak, kot smo jih vajeni iz digitalnih sistemov, prav tako se razlikuje tudi njihova porazdelitev in pogostost. Da bi težave ustrezno naslovili, je potrebno razviti ustrezne (nove) kode za odkrivanje in odpravljanje napak. V članku spoznamo kode brez prekritij, ki se uporabljajo za dostop do podatkov v DNA-pomnilnikih, in opišemo pristop, s katerim lahko konstruiramo velik naslovni prostor biološkega pomnilnika.

**Ključne besede:** DNA-pomnilniki, celoštevilaska optimizacija, kodi brez prekritij

## Addressing of DNA memory

### Abstract

Due to the increasing demand for price efficient long-term storage media, over the last decade, unconventional platforms such as the DNA memory have been developed. Every novel medium, however, brings up new problems. Biological systems have specific characteristics that lead to errors in the DNA medium that differ from the ones that occur in the digital systems. Their frequency and distribution also vary. To address these problems accordingly, new error detecting and correcting codes should be developed. Herein, we present non-overlapping codes that are used to access the data stored in DNA memory, and describe a method to construct a large address space of a biological storage device.

**Keywords:** DNA memory, integer optimization, non-overlapping codes

## 1 UVOD

Zaradi vse večjih potreb po trajnem shranjevanju podatkov se proizvajalci konvencionalnih pomnilnih platform spopadajo s težavo zagotavljanja zadostnih pomnilnih kapacitet. Eno od možnih rešitev predstavlja razvoj novih pomnilnih platform, med katere sodi tudi trajno pomnjenje podatkov v DNA. Ker je DNA robusten medij in je posledično obstoj podatkov na njem dolgotrajen ter omogoča veliko gostoto zapisa, v teoriji do 455 EB/g [5], je še posebej primeren za pomnjenje arhivskih podatkov [7].

Biološki pomnilni sistemi se po načinu dostopa do podatkov razlikujejo od klasičnih elektronskih sistemov. Do podatkov namreč ne moremo dostopati neposredno preko vodil, ki bi povezovala pošiljatelja in prejemnika, saj so podatki shranjeni v obliki mole-

kul DNA, ki znotraj biološkega pomnilnika niso prostorsko urejene [3]. Dostop do podatkov je omogočen z uporabo tehnik, ki iz nabora vseh molekul DNA, ki so shranjene v pomnilniku, izlušči tiste, v katerih se nahaja nek določen vnaprej izbran podniz. Slednjemu rečemo naslov. Velikost pomnilnika je torej omejena s številom različnih naslovov.

Za kreiranje množice naslovov znotraj posameznega pomnilnika uporabimo kode za odkrivanje in odpravljanje napak [1, 2, 6, 11, 17]. Ena izmed družin kodov, ki se uporabljajo v ta namen, so kodi brez prekritij in njihova posplošitev v šibko nekorelirane kode z dodatnimi zahtevami, da je v naslovu največ pet zaporednih enakih simbolov ter da je v njem delež simbolov iz nabora  $\{G, C\}$  blizu 50 % [9, 15, 18, 10]. Slednji zahtevi izvirata iz lastnosti tehnologije, ki

se uporablja za dostop do podatkov, saj je pogostost napačnega dostopa pri zaporedjih, ki zahtevama ne zadoščajo, bistveno večja [12, 13].

Znano je, da velikost največjega koda brez prekritij narašča eksponentno z dolžino kodne besede [8], vendar natančna formula za njen izračun v splošnem ne obstaja. Še manj je znanega o kodih brez prekritij, ki izpolnjujejo dodatne kriterije za uporabo v DNA-pomnilnikih. Zanima nas, ali lahko vseeno (učinkovito) generiramo kode z velikim številom različnih besed.

## 2 DOLOČANJE NASLOVNEGA PROSTORA POMNILNIKA

Kod brez prekritij je množica besed, za katero velja, da se nobena netrivialna predpona kodne besede ne sme pojaviti kot netrivialna pripona iste ali druge kodne besede [8]. Chee in sodelavci [4] so določili velikosti največjih dvojiških kodov brez prekritij za manjše dolžine s pomočjo iskanja največje klike v grafu. Sestavili so graf, katerega vozlišča so dvojiške besede, za katere velja, da se znotraj iste besede nobena predpona ne pojavi kot pripona. Dve vozlišči so povezali, kadar predpone ene besede niso nastopale kot pripone druge besede in obratno. Čeprav se je v dvojiškem primeru pristop obnesel in bi se ga dalo razširiti tako, da vozlišča sestavljajo le kodne besede, ki imajo kvečjemu pet zaporednih enakih simbolov ter približno polovico simbolov iz nekega nabora, je v splošnem gradnja takega grafa prostorsko in časovno zahtevna. Zaradi oblike grafa lahko tudi pričakujemo, da bodo algoritmi za iskanje največje klike v grafu v praksi počasni [16]. V grafu so številne simetrije, ki bi jih bilo potrebno upoštevati, da bi bil računski problem lažje obvladljiv.

Sami smo ubrali nekoliko drugačen pristop. Problem smo prevedli na celoštevilski optimizacijski problem s polinomsko kriterijsko funkcijo in polinomskimi omejitvami (1). Vsako njegovo optimalno rešitev  $(x^*, y^*)$  preslikamo v pripadajočo množico največjih kodov brez prekritij po naslednjem postopku. Množico  $\{C, G, T, A\}$  razbijemo na poljubna dva dela tako, da ima del  $L_1 x_1$  elementov, del  $R_1$  pa  $y_1$  elementov. Naj  $(L_j R_{j-1})$  označuje množico vseh besed, ki jih dobimo, če staknemo besedo iz množice  $L_j$  z besedo iz množice  $R_{j-1}$ . Za  $i \in \{2, \dots, n-1\}$  razbijemo množico  $\cup_{j=1}^{i-1} (L_j R_{j-1})$  na poljubna dva dela tako, da ima del  $L x$  elementov, del  $R_i$  pa  $y_i$  elementov. Kodne besede ustrezajo množici  $\cup_{i=1}^{n-1} (L_i R_{n-i})$ .

Spremenljivka  $n$  torej označuje dolžino kodnih besed. Če se omejimo na kode, ki se uporabljajo v DNA-pomnilnikih, so smiselne vrednosti za  $n$  manjše od 30. Vedoželjen bralec si lahko podrobno izpeljave in reševanja optimizacijskega problema prebere v [14].

$$\max \sum_{i=1}^{n-1} x_i y_{n-i}$$

pri pogojih

$$\begin{aligned} x_1 + y_1 &= 4 \\ x_1 + y_1 &= \sum_{j=1}^{i-1} x_j y_{i-j} \quad \forall i > 1 \\ x_i, y_i &> 0 \\ x_i, y_i &\geq 0 \quad \forall i > \\ x_i, y_i &\in \mathbb{Z} \quad \forall i \geq 1 \end{aligned}$$

Če bi želeli v optimizacijski problem vključiti tudi obe dodatni zahtevi za DNA-pomnilnike, ki smo ju opisali v uvodu, bi morali množice  $L_i, R_i$  razbiti na podmnožice tako, da posamezna podmnožica vsebuje zgolj besede z enakim številom pojavitev črk iz množice  $\{G, C\}$ , ki se začnejo z isto črko in enakim številom ponovitev le-te, ter se končajo z isto črko in enakim številom ponovitev le-te. Vsaki podmnožici bi morali pripisati novo spremenljivko ter ustrezno nadomestiti kriterijsko funkcijo in omejitve. Tudi v tem primeru bi lahko nekatere neoptimalne rešitve odstranili iz preiskovalnega prostora s podobnimi matematičnimi izpeljavami, kot so predstavljene v [14] za kode brez prekritij brez dodatnih omejitev.

## 3 REZULTATI

Eksaktno reševanje opisanega optimizacijskega problema je navkljub zmanjšanju preiskovalnega prostora časovno zelo zahtevno [14]. Dodatne zahteve množico rešitev še dodatno eksponentno povečajo in s tem otežijo reševanje optimizacijskega problema. Če je delež besed, ki kršijo posamezno zahtevo, majhen, je zato bolj smiselno vzeti dobro rešitev osnovnega optimizacijskega problema in iz dobljenega koda brez prekritij izločiti kodne besede, ki ji ne zadoščajo.

Vzeli smo vse optimalne rešitve optimizacijskega problema za štiriške kode dolžine od 3 do 15 simbolov povzete po viru [14]. Za vsak kod smo izračunali delež kodnih besed, ki imajo največ pet zaporednih enakih simbolov, ter delež kodnih besed, ki imajo med  $0,4n$  in  $0,6n$  simbolov iz množice  $\{C, G\}$ .



Najmanjši in največji dobljeni deleži so prikazani v tabeli 1. Vidimo, da skoraj vse kodne besede zadoščajo kriteriju o omejenem številu zaporednih enakih simbolov, medtem ko le približno polovica kodnih

besed zadošča kriteriju o omejenem deležu simbolov iz nabora {C,G}. Slednjega je torej smiselno vključiti v optimizacijski problem kljub povečanju časovne zahtevnosti reševanja problema.

Tabela1: Najmanjši in največji deleži kodnih besed, ki izpolnjujejo kriterije. Oznaka  $n$  označuje dolžino kodnih besed, oznaka  $N(n)$  število različnih največjih štiriških kodov brez prekritij dolžine  $n$  in  $S(n)$  število kodnih besed v največjem štiriškem kodu brez prekritij dolžine  $n$ .

$n$	$N(n)$	$S(n)$	delež kodnih besed z največ 5 enakimi zaporednimi simboli		delež kodnih besed z GC-deležem med 40 in 60 %	
			najmanjši	največji	najmanjši	največji
3	8	9	100,000 %	100,000 %	44,444 %	44,444 %
4	8	27	100,000 %	100,000 %	69,136 %	69,136 %
5	8	81	100,000 %	100,000 %	32,922 %	32,922 %
6	24	251	99,759 %	99,759 %	54,041 %	63,691 %
7	24	829	99,598 %	99,598 %	26,297 %	31,519 %
8	24	2753	99,643 %	99,643 %	51,086 %	64,620 %
9	24	9805	99,533 %	99,533 %	67,384 %	80,364 %
10	24	34921	99,425 %	99,425 %	46,158 %	57,067 %
11	24	124373	99,317 %	99,317 %	62,152 %	73,050 %
12	120	446496	99,283 %	99,305 %	41,527 %	52,160 %
13	120	1619604	99,187 %	99,201 %	56,957 %	68,453 %
14	240	5941181	99,170 %	99,189 %	69,606 %	81,392 %
15	240	21917583	99,083 %	99,096 %	54,170 %	65,376 %

## 4 ZAKLJUČEK

Ugotovili smo, da lahko problem največjega naslovnega prostora DNA pomnilnika, ki uporablja kode brez prekritij, rešujemo eksaktno s pomočjo celoštevilske optimizacije. Ker nam pristop vrne vse največje kode, lahko izmed njih izberemo takega, ki ima najbolj ugodne biokemijske lastnosti. Ugotovili smo, da lahko pogoj o omejenem številu enakih zaporednih simbolov preverjamo sprotno, medtem ko bi bilo kriterij o omejenem GC-deležu smiselno vključiti v matematični model in prilagoditi optimizacijski problem.

Reševanje celoštevilskih optimizacijskih problemov s polinomsko kriterijsko funkcijo in polinomskimi omejitvami omogočajo tudi nekatera splošno namenska orodja za reševanje optimizacijskih problemov, ki jih nameravamo preizkusiti v nadaljevanju raziskovalnega dela. Omenjeni pristop ima sicer že znane pomankljivosti. Prva je, da številna orodja ne podpirajo nekonveksnih omejitev, ki so se pojavile v naši formulaciji. Pogosto je tudi taka orodja moč uporabiti, če zna uporabnik kreirati zaporedje relaksacij svojega optimizacijskega problema, ki bo konvergiralo k pra-

vilni rešitvi. Lahko se zgodi, da rešitev konvergira tako počasi, da je pristop neuporaben, ali da se velikost relaksacij poveča čez mejo, ki jo orodja še podpirajo.

## LITERATURA

- [1] Daniella Bar-Lev, Itai Orr, Omer Sabary, Tuvi Etzion, and Eitan Yaakobi. Deep DNA storage: Scalable and robust DNA storage via coding theory and deep learning. *arXiv preprint arXiv:2109.00031*, 2021.
- [2] Meinolf Blawat, Klaus Gaedke, Ingo Huetter, Xiao-Ming Chen, Brian Turczyk, Samuel Inverso, Benjamin W Pruitt, and George M Church. Forward error correction for DNA data storage. *Procedia Computer Science*, 80:1011–1022, 2016.
- [3] Avital Boruchovsky, Daniella Bar-Lev, and Eitan Yaakobi. DNA-Correcting Codes: End-to-end Correction in DNA Storage Systems. *arXiv preprint arXiv:2304.10391*, 2023.
- [4] Yeow Meng Chee, Han Mao Kiah, Punarbasu Purkayastha, and Chengmin Wang. Cross-bifix-free codes within a constant factor of optimality. *IEEE Transactions on Information Theory*, 59(7):4668–4674, 2013.
- [5] George M Church, Yuan Gao, and Sriram Kosuri. Next-generation digital information storage in DNA. *Science*, 337(6102):1628–1628, 2012.
- [6] Robert N Grass, Reinhard Heckel, Michela Puddu, Daniela Paunescu, and Wendelin J Stark. Robust chemical preservation of digital information on DNA in silica with error-correcting codes. *Angewandte Chemie International Edition*, 54(8):2552–2555, 2015.



- [7] Reinhard Heckel, Ilan Shomorony, Kannan Ramchandran, and NC David. Fundamental limits of DNA storage systems. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 3130–3134. IEEE, 2017.
- [8] V.I. Levenshtein. Maximum number of words in codes without overlaps. *Problemy Peredachi Informatsii*, 6(4):88–90, 1970.
- [9] Maya Levy and Eitan Yaakobi. Mutually uncorrelated codes for DNA storage. *IEEE Transactions on Information Theory*, 65(6):3671–3691, 2018.
- [10] Xiaozhou Lu and Sunghwan Kim. Weakly mutually uncorrelated codes with maximum run length constraint for dna storage. *Computers in Biology and Medicine*, 165:107439, 2023.
- [11] Lee Organick, Siena Dumas Ang, Yuan-Jyue Chen, Randolph Lopez, Sergey Yekhanin, Konstantin Makarychev, Miklos Z Racz, Govinda Kamath, Parikshit Gopalan, Bichlien Nguyen, et al. Scaling up DNA data storage and random access retrieval. *BioRxiv*, page 114553, 2017.
- [12] Michael G Ross, Carsten Russ, Maura Costello, Andrew Hollinger, Niall J Lennon, Ryan Hegarty, Chad Nusbaum, and David B Jaffe. Characterizing and measuring bias in sequence data. *Genome biology*, 14:1–20, 2013.
- [13] Jerrod J Schwartz, Choli Lee, and Jay Shendure. Accurate gene synthesis with tag-directed retrieval of sequence-verified dna molecules. *Nature methods*, 9(9):913–915, 2012.
- [14] Lidija Stanovnik, Miha Moškon, and Miha Mraz. In search of maximum non-overlapping codes. *Designs, Codes and Cryptography*, pages 1–28, 2024.
- [15] SM Tabatabaei Yazdi, Yongbo Yuan, Jian Ma, Huimin Zhao, and Olgica Milenkovic. A rewritable, random-access DNA-based storage system. *Scientific reports*, 5(1):1–10, 2015.
- [16] Jose L Walteros and Austin Buchanan. Why is maximum clique often easy in practice? *Operations Research*, 68(6):1866–1895, 2020.
- [17] SM Yazdi, Ryan Gabrys, and Olgica Milenkovic. Portable and error-free DNA-based data storage. *Scientific reports*, 7(1):1–6, 2017.
- [18] SMH Tabatabaei Yazdi, Han Mao Kiah, Ryan Gabrys, and Olgica Milenkovic. Mutually uncorrelated primers for DNA-based data storage. *IEEE Transactions on Information Theory*, 64(9):6283–6296, 2018.

■

**Lidija Stanovnik** je diplomirala in magistrirala na Fakulteti za računalništvo in informatiko ter Fakulteti za matematiko in fiziko Univerze v Ljubljani. Zaposlena je kot mlada raziskovalka na Fakulteti za računalništvo in informatiko Univerze v Ljubljani.

■

**Miha Mraz** je diplomiral, magistriral in doktoriral na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Na isti ustanovi je zaposlen kot redni profesor in poučuje predmete Zanesljivost in zmogljivost računalniških sistemov, Modeliranje računalniških omrežij in Nekonvencionalne platforme in metode procesiranja. Je (so)avtor več kot 200 raziskovalnih del.

# ❏ NORA: napredno orodje za poučevanje relacijske algebre

Timotej Knez

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Večna pot 113, 1000 Ljubljana  
timotej.knez@fri.uni-lj.si

## Izvleček

V članku predstavljamo orodje NORA za avtomatsko delo z relacijsko algebro, ki smo ga razvili in pilotno uporabili na univerzitetnem predmetu „Osnove podatkovnih baz“. Sodelovalo je približno 170 študentov, ki so orodje uspešno uporabljali pri vajah, reševanju domačih nalog ter na preverjanju znanja znotraj. Orodje omogoča pisanje, izvajanje in avtomatsko ocenjevanje poizvedb na pripravljenih podatkovnih bazah. Omogoča tudi uporabo znotraj spletne učilnice Moodle. Pilotna izvedba je pokazala, da je orodje učinkovito, prihrani čas pri ocenjevanju in omogoča prilagodljivo ocenjevanje študentskih rešitev. S tem verjamemo, da orodje predstavlja pomemben prispevek k izboljšanju učnega procesa relacijske algebre v okviru študijskih programov računalništva in informatike.

**Ključne besede:** Avtomatsko preverjanje poizvedb, E-učenje, Ocene na daljavo, Relacijska algebra

## NORA: Advanced Tool for Relational Algebra Teaching

### Abstract

The article introduces an innovative tool for the automatic validation of relational algebra queries piloted in the university course, the "Introduction to Databases." Approximately 170 students actively utilized the tool during practical exercises, homework assignments, and a midterm quiz. The tool enables query writing, execution, and automatic assessment on prepared databases, with a notable emphasis on its Moodle integration. The pilot implementation demonstrated the tool's effectiveness, time-saving benefits in evaluation, and adaptability in grading student solutions. This suggests that the tool represents a significant contribution to enhancing the learning process of relational algebra within computer science and informatics study programmes.

**Keywords:** Automatic Query Verification, E-Learning, Remote Grading, Relational Algebra

### 1 UVOD

V sodobnem izobraževalnem okolju, kjer se računalništvo in informatika neprestano razvijata, postaja ključno zagotoviti študentom učinkovite in inovativne načine učenja temeljnih konceptov. Eden izmed takšnih konceptov, ki ga obravnavamo pri predmetu osnove podatkovnih baz je relacijska algebra. Eden izmed ključnih izzivov pri poučevanju relacijske algebre je prikaz delovanja posameznih operacij ter izvajanja izrazov. Za razliko od ostalih poizvedovalnih orodij, ki jih predstavljamo pri predmetu, relacijska algebra ni neposredno uporabljena v nobeni podatkovni bazi, zaradi česar je njeno uporabo težko demonstrirati na praktičnem primeru.

Za reševanje te težave smo razvili orodje, ki študentom omogoča pisanje in preverjanje poizvedb relacijske algebre na enostaven in dostopen način. To inovativno orodje ne zahteva namestitve ali vzdrževanja podatkovnih baz ali strežnikov, saj v celoti deluje v brskalniku. Orodje omogoča, da študent med sestavljanjem izraza relacijske algebre na vsakem koraku preveri, kako trenutna poizvedba deluje ter ali je prišlo do kakšne napake. Orodje študentu pomaga tudi pri reševanju morebitnih sintaktičnih napak, kar omogoča, da se študent osredotoči na razumevanje konceptov, namesto da bi se ukvarjal z načinom zapisa izrazov.

Poleg tega orodje ponuja izvajalcem predmeta možnost avtomatskega preverjanja študentskih po-

izvedb, kar omogoča bolj učinkovito ocenjevanje in zagotavljanje povratnih informacij. Orodje smo že vključili v izvedbo vaj pri predmetu osnove podatkovnih baz na Fakulteti za računalništvo in informatiko Univerze v Ljubljani.

V nadaljevanju članka bomo podrobno predstavili funkcionalnosti orodja, opisali uporabo na univerzi ter ponudili vpogled v tehnične podrobnosti. Celoten projekt je prosto dostopen ter objavljena na platformi GitHub, s čimer želimo spodbuditi sodelovanje in prispevke širše skupnosti. Orodje je dostopno na povezavi <https://github.com/TimotejK/OPB-RA>

## 2 RELACIJSKA ALGEBRA

Relacijska algebra je matematična struktura, ki se uporablja za opisovanje in manipulacijo podatkov v relacijskih bazah podatkov. Prvotno jo je opisal Codd leta 1970 [1]. Razvila se je kot teoretično orodje za podporo relacijskim bazam podatkov, ki so osnova za večino sodobnih sistemov za upravljanje s podatkovnimi bazami.

Glavni cilj relacijske algebre je omogočiti enostavno in formalno opisovanje poizvedb, filtriranja, združevanja in drugih operacij nad relacijskimi podatki. V osnovi vključuje pet osnovnih operacij:

**Selekcija ( $\sigma$ ):** Izbor vrstic, ki izpolnjujejo določen pogoj.

**Projekcija ( $\pi$ ):** Izbor določenih atributov iz relacije, običajno za zmanjšanje kompleksnosti podatkov.

**Unija ( $\cup$ ):** Povezovanje dveh relacij, kjer rezultat vsebuje vse vrstice, ki se pojavijo v vsaj eni izmed začetnih relacij.

**Razlika ( $-$ ):** Odstranjevanje vrstic iz ene relacije, ki se pojavljajo tudi v drugi.

**Produkt ( $\times$ ):** Povezovanje dveh relacij, tako da se vsaka vrstica ene relacije združi z vsako vrstico druge relacije.

Poleg teh osnovnih operacij obstajajo tudi druge sestavljene operacije, ki omogočajo bolj kompleksne poizvedbe in so sestavljene iz osnovnih operacij. Relacijska algebra zagotavlja formalno orodje za preučevanje in analizo poizvedb nad relacijskimi podatki, s čimer se doseže boljše razumevanje njihove strukture in omogoča optimizacijo izvajanja poizvedb.

V praksi se relacijska algebra uporablja v jezikih za poizvedovanje podatkov, kot je SQL (Structured Query Language), ki omogoča enostavno branje in spreminjanje podatkov v relacijskih bazah.

Poleg osnovnih operacij naše orodje podpira tudi ostale operacije:

**Presek ( $\cap$ ):** Povezovanje dveh relacij, kjer rezultat vsebuje vrstice, ki se pojavijo v obeh začetnih relacijah.

**Naravni in pogojni stik ( $\bowtie$ ):** Združi vrstice iz dveh relacij na podlagi pogoja oziroma ujemanja istoimenskih stolpcev.

**Odpri naravni in pogojni stik ( $\ltimes$ ,  $\ltimes$ ,  $\ltimes$ ):** Deluje podobno kot naravni in pogojni stik, s to razliko, da doda tudi vrstice iz odprte strani, ki se ne ujemajo z nobeno vrstico z druge strani.

**Pol-stik ( $\triangleright$ ):** Združi vrstice med dvema relacijama na podlagi pogoja in vrne samo attribute iz leve relacije.

**Deljenje ( $/$ ):** Izpiše skupine podatkov iz prve relacije, ki se ujemajo z vsemi podatki iz druge relacije.

**Agregacija ( $\tau$ ):** Grupira vrstice iz relacije ter ovrednoti agregacijske funkcije nad podatki iz relacije.

**Preimenovanje ( $\rho$ ):** Preimenuje relacijo ter njene attribute.

**Prerejanje ( $\leftarrow$ ):** Relaciji priredi vrednost izraza.

Orodje NORA tako podpira vse operacije relacijske algebre, ki jih uporabljamo za poizvedovanje po relacijskih podatkovnih bazah.

## 3 SORODNA ORODJA

Pri razvoju našega orodja smo upoštevali tudi druge projekte, ki se ukvarjajo z avtomatskim preverjanjem poizvedb in olajševanjem učenja relacijske algebre. Eno izmed takšnih orodij je orodje Relax, ki se je uveljavilo v podobnem okolju.

Orodje Relax je namenjeno učenju in preverjanju poizvedb relacijske algebre ter je bilo razvito na univerzi v Innsbrucku. Cilj orodja je poenostavljanje učenja konceptov relacijske algebre ter SQL jezikov.

Orodje Relax je v mnogo pogledih podobno našemu orodju. Obe orodji si prizadevata zagotoviti izboljšano učno izkušnjo študentom, omogočata pisanje poizvedb ter njihovo preverjanje na pripravljenih podatkovnih bazah. Prav tako sta bili obe orodji razviti za uporabo na fakulteti in sta prosto dostopni. Obe orodji sta napisani v jeziku JavaScript, vendar se tehnične podrobnosti, kot so arhitektura brez strežnikov, lahko razlikujejo.

Medtem ko se obe orodji trudita doseči podobne cilje, naše orodje prinaša nekatere dodatne funkcionalnosti. Naše orodje ima prilagojeno različico, ki omogoča integracijo v Moodle spletno učilnico. To

omogoča, da lahko študenti znotraj kvizov vpisujejo izraze relacijske algebre ter jih preizkušajo.

Obstaja tudi več drugih orodij, ki rešujejo sorodne probleme. Soler in ostali [3] so razvili orodje, ki omogoča avtomatsko pripravo nalog iz relacijske algebre za podporo poučevanja na področju podatkovnih baz.

Podobno orodje je tudi Relational Algebra at your fingertips (ALF) [4]. Orodje omogoča izvedbo izrazov relacijske algebre v terminalu. V primerjavi z našim orodjem, alf ne ponuja grafičnega uporabniškega vmesnika ter integracije z Moodle spletno učilnico. ALF uporablja izraze zapisane v prilagojeni obliki, ki je sestavljena kot (operacija izraz1, izraz2). Sintaksa, ki smo jo uporabili v našem orodju omogoča bolj naraven zapis izrazov, ki je bližji zapisu, ki se običajno uporablja v literaturi.

## 4 OPIS ORODJA

Orodje NORA omogoča bolj prilagodljivo in dostopno poučevanje relacijske algebre. NORA omogoča avtomatsko izvajanje poizvedb, kar uporabniku omogoča, da na vsakem koraku med grajenjem izraza preveri, kako izraz deluje ter ali je kje prišlo do napake. Orodje smo razvili s ciljem, da bi študentom omogočili čimbolj preprosto uporabo. Prav tako smo želeli omogočiti integracijo orodja v zunanje sisteme, kot je sistem Moodle in s tem podpreti uporabo v celotnem učnem procesu.

## 4.1 Funkcionalnosti orodja

Orodje NORA študentom omogoča pisanje poizvedb relacijske algebre preko uporabniku prijaznega vmesnika. Uporabnik lahko izbira med vsemi operacijami relacijske algebre.

Orodje trenutno ponuja dva uporabniška vmesnika. Prvi je postavljen kot samostojna spletna stran na portalu Github Pages in omogoča, da uporabniki kar se da enostavno vpisujejo ter preverjajo svoje poizvedbe. Drugi ponuja dodajanje funkcionalnosti orodja NORA v vprašanje na Moodle spletni učilnici. Ta omogoča uporabo orodja znotraj kvizov in preverjanj znanj v učnem procesu.

### 4.1.1 Uporaba na spletni strani

NORA orodje je na voljo na Github pages strani na naslovu <https://timotejk.github.io/OPB-RA/>. Stran omogoča izbiro ene izmed štirih različnih domen, ki vsebujejo različne podatke. Po izbiri domene, se na strani samodejno izpišejo vse relacije ter njihovi atributi. Uporabnik lahko nato v vnosno polje zapiše izraz ter ga evalvira. Rezultat izraza se izpiše v obliki tabele na spletni strani.

Spletna stran je zasnovana tako, da jo je mogoče uporabiti za demonstracijo izrazov relacijske algebre. Domene, ki so na voljo smo prilagodili tako, da ustrezajo domenam, ki se pojavljajo v nalogah ki jih uporabljamo na vajah pri predmetu Osnove podatkovnih baz. Tako lahko študenti poizvedbe, ki jih pišemo na vajah neposredno preizkušajo v aplikaciji NORA. Uporabniški vmesnik je prikazan na Sliki 1.

## Relacijska algebra

Podatkovna domena  
Facebook & Twitter

oseba

- ID number
- Ime string
- Rojen date
- SID number

facebook

- OID number
- PID number

twitter

- OID number
- SID number

stan

- SID number
- Stan string

π σ ρ τ × ⋈ ⋉ ⋊ ⋋ ⋌ ⋍ ⋎ ⋏ ⋐ ⋑ ⋒ ⋓ ⋔ ⋕ ⋖ ⋗ ⋘ ⋙ ⋚ ⋛ ⋜ ⋝ ⋞ ⋟ ⋠ ⋡ ⋢ ⋣ ⋤ ⋥ ⋦ ⋧ ⋨ ⋩ ⋪ ⋫ ⋬ ⋭ ⋮ ⋯ ⋰ ⋱ ⋲ ⋳ ⋴ ⋵ ⋶ ⋷ ⋸ ⋹ ⋺ ⋻ ⋼ ⋽ ⋾ ⋿ ⋰ ⋱ ⋲ ⋳ ⋴ ⋵ ⋶ ⋷ ⋸ ⋹ ⋺ ⋻ ⋼ ⋽ ⋾ ⋿

oseba ⋈ facebook

Evalviraj

(oseba) ⋈ (facebook)

ID	Ime	Rojen	SID	OID	PID
1	Jill	1990-03-09	1	1	2
1	Jill	1990-03-09	1	2	1

Slika 1: Primer uporabniškega vmesnika na spletni strani.



```

<script type="text/javascript" src="https://opb-tools.data-lab.si/client-lib/jquery-1.11.1.min.js"></script>
<script type="text/javascript" src="https://timotejk.github.io/OPB-RA/operacije.js"></script>
<script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/underscore.js/1.9.1/underscore-min.js">
</script>
<script type="text/javascript" src="https://timotejk.github.io/OPB-RA/knjiznica.js"></script>

```

Slika 2: Primer značk za dodajanje knjižnice.

Orodje omogoča uporabo znotraj Moodle spletne učilnice. Ta način uporabe deluje tako, da v kviz dodamo esejsko vprašanje, pri katerem uporabnik v odgovor vpisuje izraz relacijske algebre. K vprašanju dodamo html značko za dodajanje javascript kode in uvozimo orodje. Potrebno je dodati datoteki knjižnica.js, operacije.js ter splošnji knjižnici jquery in underscore. Primer potrebnih značk je prikazan na Sliki 2.

Ko se vprašanje prikaže študentu, orodje nad besedilno polje doda gumbe za vstavljanje posebnih znakov, pod besedilno polje pa doda gumb za poganjanje poizvedbe. Ko uporabnik požene poizvedbo, se izraz iz besedilnega polja ovrednoti, njegov rezultat pa se izpiše v obliki tabele pod besedilno polje. Na ta način lahko študenti sproti pregledujejo kaj njihovi izrazi naredijo in ali so sintaktično pravilni.

V našem primeru smo po zaključku kviza iz učilnice avtomatsko naložili vse odgovore študentov ter izraze pognali lokalno na prilagojenih zbirkah relacij. Rezultat vsake izmed poizvedb smo nato avtomatsko primerjali s pričakovanim rezultatom in tako točkovali študentske odgovore. Na ta način orodje omogoča tudi avtomatsko popravljanje kvizov, kar lahko izvajalcem predmeta prihrani veliko časa.

## 4.2 Tehnične podrobnosti

Orodje za preverjanje poizvedb relacijske algebre je v celoti napisano v programskem jeziku JavaScript. Izvajanje poizvedb poteka lokalno v brskalniku uporabnika.

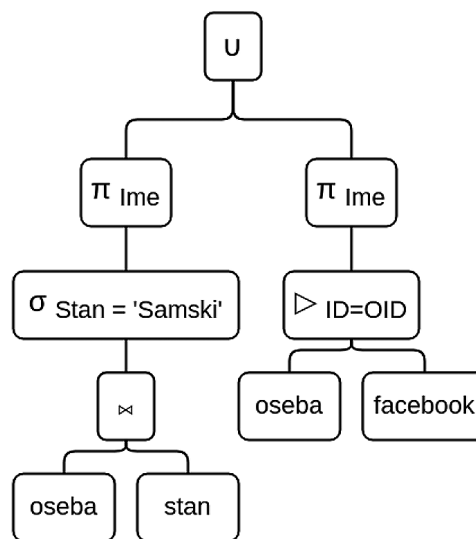
### 4.2.1 Implementacija

Preverjanje poizvedb poteka v več korakih. V prvem koraku se uporabnikov vnos razdeli na posamezne poizvedbe, saj orodje omogoča izvajanje več zaporednih izrazov, kjer je rezultate izraza mogoče uporabiti znotraj vseh naslednjih izrazov.

V drugem koraku vsak izraz razčlenimo na operacijo ter izraza pred in po operaciji. V primeru da gre za unarno operacijo, bo izraz pred operacijo prazen. Nato rekurzivno ovrednotimo vse izraze nad katerimi se operacija izvaja. Primer rekurzivnega izvajanja

je prikazan v Sliki 3. V sliki je prikazan izračun vrednosti izraza .

V primeru, da trenutni izraz vsebuje ime relacije ali spremenljivke, ga lahko ovrednotimo tako, da uporabimo vrednost relacije oziroma spremenljivke. Ta pogoj služi kot ustavitveni pogoj rekurzije.

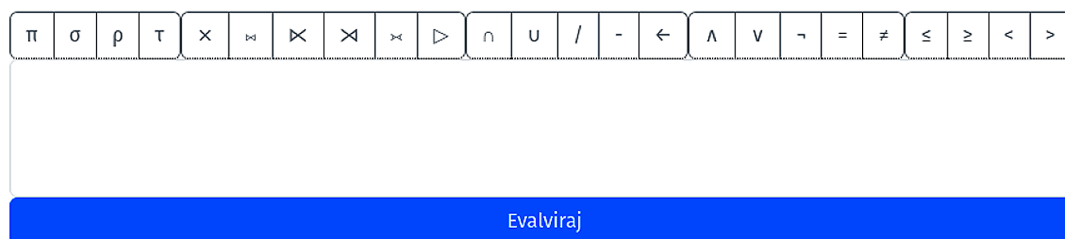


Slika 3: Primer rekurzivnega izračuna vrednosti izraza.

### 4.2.2 Zapis izrazov

Ena izmed težav, ki jih je bilo pri zasnovi orodja potrebno rešiti, je kako zapisati izraze na način, ki omogoča nedvoumen zapis v besedilnih poljih. Prva težava na katero smo naleteli je, da izrazi relacijske algebre vsebujejo matematične simbole, ki jih ni enostavno zapisati s tipkovnico. To težavo smo rešili z dodatkom gumbov nad vnosnim poljem, ki omogočajo vpis vseh matematičnih simbolov, ki se uporabljajo znotraj relacijske algebre. Za lažje vpisovanje izrazov smo omogočili tudi zapis nekaterih simbolov z znaki, ki so na voljo na tipkovnici. Na primer namesto znaka  $\neq$  lahko uporabnik zapiše  $\neq$ . Gumbi za vnos posebnih znakov so prikazani na Sliki 4.

Druga težava na katero smo naleteli je zapisovanje argumentov operacij. Navadno se pri zapisu izrazov argumente operacije zapisuje podpisano. S tem vemo, da gre za argumente operacij in ne na-



Slika 4: Vnosno polje za zapis izrazov v predstavljenem orodju. Nad vnosnim poljem imamo gumbje za zapis operacij relacijske algebre.

slednji izraz. Ker vnosna polja v brskalniku ne podpirajo vnosa podpisanega besedila, smo se odločili, da bomo argumente operacij označevali z uporabo oglatih oklepajev.

Tako izraz zapišemo kot  $(\pi[Ime] \sigma[Stan='Samski'] (oseba \bowtie stan)) \cup (\pi[Ime] (oseba \triangleright [ID = OID] facebook))$ .

#### 4.2.3 Preverjanje sintaktičnih napak

Ena izmed ključnih funkcionalnosti orodja je tudi dobro poročanje o sintaktičnih napakah v izrazih. Z dobrim poročanjem napak lahko študentom pomagamo popraviti napačno zapisane izraze ter s tem poenostavimo uporabo orodja. Pri zasnovi orodja smo si prizadevali da bi za vsako izjemo, do katere bi lahko prišlo, ustvarili sporočilo o napaki, poleg sporočila pa smo dodali tudi začetek ter konec dela poizvedbe, ki je napako povzročil. S tem lahko uporabniku prikažemo sporočilo o napaki ter označimo, del poizvedbe ki je problematičen. Primer takšnega sporočila je prikazan na Sliki 5.

## 5 UPORABA ORODJA V PRAKSI

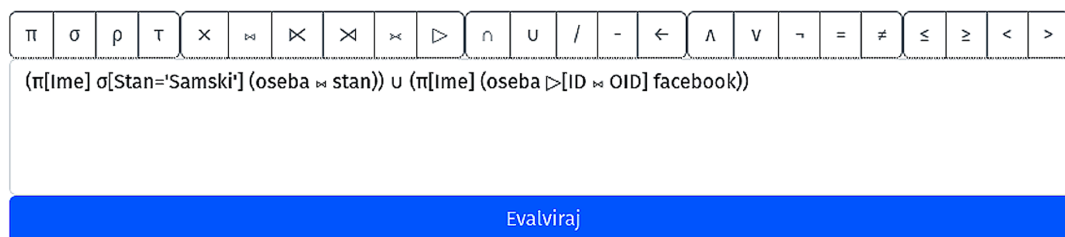
V letu 2023 smo uspešno pilotno uporabili naše orodje pri predmetu Osnove podatkovnih baz na univerzitetnem programu računalništva in informatike. Pri predmetu je sodelovalo približno 170 študentov, ki

so imeli možnost aktivne uporabe orodja tako na vajah kot pri reševanju domačih nalog.

Študentom smo omogočili, da so si med reševanjem nalog na vajah pomagali z orodjem. Ta pristop je zagotavljal takojšnjo povratno informacijo o pravilnosti njihovih poizvedb, kar je izboljšalo učni proces. Z orodjem so si lahko pomagali tudi pri reševanju domačih nalog, kjer so študenti samostojno preverjali svoje rešitve in odpravljali morebitne napake.

Po uspešni uporabi na vajah smo se odločili uporabiti orodje tudi na kolokviju, ki smo ga izvedli v obliki spletnega kviza. Kviz je zajemal tri naloge iz relacijske algebre, ki so bile ocenjene s pomočjo našega orodja. Uporaba orodja se je izkazala kot zelo koristno, saj je omogočilo avtomatsko ocenjevanje nalog. Pri avtomatskem ocenjevanju smo pregledovali pravilnosti rezultatov, ki jih poizvedba vrne, namesto ocenjevanja same poizvedbe. S tem pristopom smo študentom omogočili, da rešujejo en problem na več različnih načinov, brez da bi to vplivalo na število doseženih točk.

Na podlagi rezultatov pilotne izvedbe smo ugotovili, da so izračunane ocene ustrezne in odražajo pravilno razumevanje relacijske algebre s strani študentov. Orodje je uspešno integrirano v izobraževalni proces, študentje pa so ga sprejeli kot koristno orodje pri svojem učenju.



Znotraj logičnih izrazov ni mogoča uporaba relacijskih operacij

$(\pi[Ime] \sigma[Stan='Samski'] (oseba \bowtie stan)) \cup (\pi[Ime] (oseba \triangleright [ID \bowtie OID] facebook))$

Slika 5: Primer izpisanega sporočila o sintaktični napaki v izrazu. Pod gumbom "Evalviraj" je zapisano do kakšne napake je prišlo, spodaj pa je v izrazu z rdečo barvo pobarvan element, ki je napako povzročil.

## 6 ZAKLJUČEK

Razvoj in implementacija orodja za avtomatsko preverjanje poizvedb relacijske algebre sta odprla nova vrata za učinkovitejše in prilagodljivejše učenje v akademskem okolju. Naša pilotna izvedba na predmetu »Osnove podatkovnih baz« je pokazala, da lahko orodje prinese številne prednosti tako študentom kot izvajalcem predmeta.

Z uporabo orodja smo študentom omogočili aktivno preizkušanje svojega znanja v realnem času med vajami, kar je povečalo razumevanje relacijske algebre in olajšalo reševanje domačih nalog. Orodje je postalo nepogrešljivo tudi pri kolokviju v obliki spletnega kviza, kjer smo z avtomatskim ocenjevanjem dosegli visoko raven doslednosti in prihranili dragocen čas pri ročnem pregledu.

Pomembna dodana vrednost orodja je njegova integracija v Moodle spletno učilnico, ki študentom omogoča še boljšo izkušnjo in hkrati poenostavlja delo izvajalcev predmeta. Prilagodljivost pri ocenjevanju rezultatov poizvedb je študentom omogočila

večjo kreativnost pri reševanju problemov, ne da bi vplivala na končni rezultat.

Zavedamo se, da so pred nami izzivi in nadaljnje izboljšave, vendar smo s to izkušnjo trdno prepričani, da orodje za avtomatsko preverjanje poizvedb relacijske algebre prinaša vrednost v izobraževalnem procesu. Z nadaljnjim razvojem, odprtim sodelovanjem in prilagajanjem potrebam skupnosti verjamejo, da lahko obogati učno izkušnjo na področju podatkovnih baz in računalništva na splošno.

## 7 LITERATURA

- [1] Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6), 377-387.
- [2] Rampl M in drugi. Relax. dostopno na: <https://github.com/dbis-uibk/relax>, [pregledano 18. 12. 2023].
- [3] Soler, J., Boada, I., Prados, F., Poch, J., & Fabregat, R. (2007, August). An automatic correction tool for relational Algebra Queries. In *International Conference on Computational Science and Its Applications* (pp. 861-872). Berlin, Heidelberg: Springer Berlin Heidelberg.
- [4] Daloz B. Alf. dostopno na: <https://github.com/eregon/alf>, [pregledano 18. 12. 2023]

■

**Timotej Knez** je leta 2021 pridobil magisterij iz računalništva in informatike. Trenutno zaseda položaj mladega raziskovalca na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Ukvarja se z raziskovanjem uporabe znanstvenih baz v obdelavi naravnega jezika. Dela tudi kot asistent pri predmetu osnove podatkovnih baz na fakulteti.

# Iz Islovarja

Islovar je spletni terminološki slovar informatike, ki ga že več kot 20 let ureja jezikovna sekcija Slovenskega društva INFORMATIKA. Slovar je javno dostopen za vpoglede in vnašanje novih izrazov na naslovu <http://www.islovar.org>

**inženiring uporabnosti -a -- m** (*angl. usability engineering*)

večdisciplinarno področje, ki se ukvarja z interakcijo človek-računalnik in posebej z oblikovanjem zelo uporabnih in uporabniku prijaznih računalniških vmesnikov

**lógica vrata -ih vrát ž** (*angl. logic gate*)

vezje, ki predstavlja osnovno logično funkcijo (NOT, AND, OR, NAND, NOR)

**méšanje podatkov -a -- s** (*angl. data blending*)

postopek, pri katerem se velepodatki iz več virov združujejo v eno podatkovno skladišče; sin. zlivanje podatkov

**napád níčelnega dné -a -- -- m** (*angl. Zero-day attack*)

kibernetski napad, kjer napadalec izkorišča varnostno ranljivost, preden jo lahko razvijalci odpravijo

**negíbljiva elektrónika -e -e ž** (*angl. solid-state electronics, semiconductor electronics*)

elektronika, sestavljena iz polprevodniških elementov, npr. negíbljivi disk, polprevodniški pomnilnik; sin. polprevodniška elektronika

**nèvidni splét -ega -a m** (*angl. invisible web, deep web, undernet, hidden web*)

del spleta, ki ga običajni iskalniki ne indeksirajo in ne zaznavajo; sin. globoki splet; prim. temni splet

**obnovljívi pomnílnik -ega - a m** (*angl. regenerative memory, regenerative capacitor memory*)

pomnilnik, ki za shranjevanje podatkov uporablja kondenzator, katerega električni naboj je treba osveževati; sin. regenerativni pomnilnik

**programábilna lógica napráva -e -e -e ž** (*angl. programmable logic device, PLD*)

elektronska komponenta, ki jo lahko uporabnik sam programira; sin. PLN, programirljiva logična naprava

**regeneratívno bránje -ega -a s** (*angl. read and restore cycle, regenerative read*)

branje iz pomnilnika, pri katerem se podatki samodejno zapišejo nazaj na mesta, s katerih so bili prebrani; sin. obnovljivo branje

**tèmni splét -ega -a m** (*angl. dark web, darknet websites*)

del svetovnega spleta, ki temelji na posebnem šifriranem usmerjevalnem protokolu, dostopen le z uporabo namenskih brskalnikov, namenjen pa je nenadzorovanemu povezovanju in pogosto nezakonitemu poslovanju



# STE PRIPRAVLJENI NA KIBERNETSKE GROŽNJE?

V TELEKOMU SLOVENIJE POSKRBITIMO ZA NAJVIŠJO  
STOPNJO VARNOSTI V KIBERNETSKEM PROSTORU



Operativni center  
kibernetske varnosti

Obveščanje o  
kibernetskih grožnjah

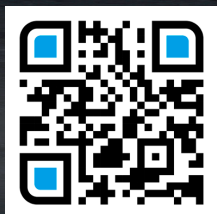
Varen poslovni splet

Testi socialnega  
inženiringa

Varnostni pregledi in  
penetracijski testi

Varna poslovna  
mobilnost

POSLUJTE VARNO. POSLUJTE DIGITALNO.



TelekomSlovenije



# SOPHOS

Cybersecurity delivered.



## Sophos Managed Threat Response

DRUGI SE USTAVIJO SAMO PRI OBVESTILU O GROŽNJI.

**SOPHOS MTR STROKOVNJAKI  
GROŽNJO TUDI ODSTRANIJO - 24/7!**

Distributer: Sophos d.o.o., [www.sophos.si](http://www.sophos.si), [slovenija@sophos.si](mailto:slovenija@sophos.si), T: 07/39 35 600

**AKCIJA**

**ZA BRALCE REVIJE**

**ŽIVLJENJE IN TEHNIKA**

Prejmite sveže novice iz sveta računalništva vsakih 14 dni v vaš poštni nabiralnik!



Letna naročnina na revijo Računalniške novice (21 števil) skupaj z darilom za samo

**33,50 €**

IZBIRATE LAHKO MED:

Ključavnica Masterlock 8391  
(90 cm)



USB ključ Apacer 3.1  
128 GB



Kalkulator Casio  
DJ-120D Plus



**T: 01 620 88 00**

**M: narocnine@stromboli.si**





Fakulteta za  
informatijske študije  
Faculty of information studies

# ŠTUDIJ PRIHODNOSTI

## na Fakulteti za informatijske študije

### MAGISTRSKI ŠTUDIJ

- INFORMATIKA V SODOBNI DRUŽBI - povežite družboslovje z informatiko.
- RAČUNALNIŠTVO IN SPLETNE TEHNOLOGIJE - razvijte napredne programske rešitve.
- PODATKOVNE ZNANOSTI - odkrijte velike skrivnosti v velikih podatkih.
- KIBERNETSKA VARNOST - z najnovejšimi znanji proti kibernetским grožnjam.

### DOKTORSKI ŠTUDIJ

- INFORMACIJSKA DRUŽBA - ustvarite novo znanje s področja informacijske družbe.

Sodelujte na predavanjih  
in vajah tudi VIRTUALNO!



Fakulteta za informatijske študije v Novem mestu (FIŠ)  
Ljubljanska cesta 31a, 8000 Novo mesto, 07 37 37 870, [vpis@fis.unm.si](mailto:vpis@fis.unm.si)



# Izpitni centri ECDL

---

**ECDL** (European Computer Driving License), ki ga v Sloveniji imenujemo evropsko računalniško spričevalo, je standardni program usposabljanja uporabnikov, ki da zaposlenim potrebno znanje za delo s standardnimi računalniškimi programi na informatiziranem delovnem mestu, delodajalcem pa pomeni dokazilo o usposobljenosti. V Evropi je za uvajanje, usposabljanje in nadzor izvajanja ECDL pooblaščen ustanova ECDL Fundation, v Sloveniji pa je kot član CEPIS (Council of European Professional Informatics) to pravico pridobilo Slovensko društvo INFORMATIKA. V državah Evropske unije so pri uvajanju ECDL močno angažirane srednje in visoke šole, aktivni pa so tudi različni vladni resorji. Posebno pomembno je, da velja spričevalo v 148 državah, ki so vključene v program ECDL. Doslej je bilo v svetu v program certificiranja ECDL vključenih že preko 16 milijonov oseb, ki so uspešno opravile preko 80 milijonov izpitov in pridobile ustrezne certificate. V Sloveniji je bilo doslej v program certificiranja ECDL vključenih več kot 18.000 oseb in opravljenih več kot 92.000 izpitov. V Sloveniji sta akreditirana dva izpitna centra ECDL, ki imata izpostave po vsej državi.

---





# Znanstveni prispevki

Miha Frangež, Matevž Pesek  
NAČRTOVALNIK POTI ZA JAVNI POTNIŠKI PROMET  
IN STORITVE MIKROMOBILNOSTI

Živa Hudobivnik, Maja Savinek, Marina Trkman  
UPORABA POGOVORNEGA ROBOTA ZA PODORO INTRANETU:  
PRIMER ELEKTRO LJUBLJANA

# Kratki znanstveni prispevki

Lidija Stanovnik, Miha Mraz  
NASLAVLJANJE DNA-POMNILNIKOV

# Strokovni prispevki

Timotej Knez  
NORA: NAPREDNO ORODJE ZA POUČEVANJE RELACIJSKE ALGEBRE

# Informacije

IZ ISLOVARJA

VABILO NA KONFERENCO DSI

ISSN 1318-1882



9 771318 188001