

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

Roberto Šandro

**APLIKACIJA ZA UPRAVLJANJE
KORISNIČKIM RAČUNIMA U
OBJEKTNO-RELACIJSKOJ BAZI
(POSTGRESQL) S WEB SUČELJEM**

PROJEKT

TEORIJA BAZA PODATAKA

Varaždin, 2026.

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
V A R A Ž D I N

Roberto Šandro

Matični broj: 0016159806

Studij: Baze podataka i baze znanja

**APLIKACIJA ZA UPRAVLJANJE KORISNIČKIM RAČUNIMA U
OBJEKTNO-RELACIJSKOJ BAZI (POSTGRESQL) S WEB SUČELJEM**

PROJEKT

Mentor:

doc. dr. sc. Bogdan Okreša Đurić

Varaždin, siječanj 2026.

Roberto Šandro

Izjava o izvornosti

Izjavljujem da je ovaj projekt izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvatanjem odredbi u sustavu FOI Radovi

Sažetak

U radu je razvijen sustav za upravljanje korisničkim računima koji obuhvaća uloge, prava pristupa i evidenciju aktivnosti (audit). Baza podataka implementirana je u PostgreSQL-u kao objektno-relacijskom sustavu, pri čemu su korišteni pogledi (views) i okidači (triggers) za automatizaciju bilježenja događaja. Aplikacijski dio razvijen je u Pythonu korištenjem Flask okvira, a komunikacija s bazom ostvarena je putem psycopg2 adaptera. Implementiran je RBAC model (Role-Based Access Control) kroz relacije korisnik–uloga i uloga–dozvola, uz autentikaciju s hashiranjem lozinki. Rad prikazuje konceptualni i logički model, ključne dijelove implementacije te primjere korištenja i provjeru funkcionalnosti.

Ključne riječi: PostgreSQL; objektno-relacijske baze; RBAC; uloge i dozvole; okidači; pogledi; Flask; psycopg2; audit log

Sadržaj

1. Uvod	1
2. Opis aplikacijske domene	2
3. Teorijski uvod	3
3.1. Poglavlje druge razine	3
3.1.1. Poglavlje treće razine	3
3.1.2. Poglavlje četvrte razine	4
4. Tehničke upute	5
4.1. Upute za oblikovanje izgleda rada	5
4.1.1. Oblikovanje stranice	5
4.1.2. Tekst rada	5
4.1.3. Slike	6
4.1.4. Tablice	6
4.1.5. Programski kod	7
4.1.6. Formule	8
4.1.7. Kratice	8
4.1.8. Strano nazivlje	8
4.2. Navođenje literature	8
5. Zaključak	10
Popis literature	12
Popis slika	13
Popis tablica	14
Popis isječaka koda	15
1. Prilog 1	17
2. Prilog 2	18

1. Uvod

Upravljanje korisničkim računima i kontrola pristupa predstavljaju temeljni dio gotovo svake suvremene aplikacije. Sustavi koji obrađuju povjerljive podatke ili omogućuju rad većem broju korisnika moraju jasno definirati tko ima pravo pristupa pojedinim funkcionalnostima sustava. Nepravilno postavljena prava pristupa mogu dovesti do sigurnosnih problema, neovlaštenih izmjena podataka ili gubitka povjerenja u sustav.

U ovom projektu razvijena je web aplikacija za upravljanje korisničkim računima koja omogućuje definiranje korisnika, dodjelu uloga i pripadajućih dozvola te evidenciju aktivnosti korisnika kroz audit zapise. Aplikacija je implementirana korištenjem objektno-relacijske baze podataka PostgreSQL i web aplikacijskog okvira Flask. Poseban naglasak stavljen je na korištenje naprednih mogućnosti baze podataka, poput okidača (engl. *triggers*) i pogleda (engl. *views*), kako bi se dio logike sustava realizirao izravno na razini baze.

Motivacija za izradu ovog projekta proizlazi iz potrebe za jednostavnim, ali funkcionalnim sustavom za upravljanje korisnicima kakav se često susreće u manjim i srednjim aplikacijama. Umjesto korištenja složenih gotovih rješenja, cilj je bio izraditi sustav koji jasno demonstrira osnovne koncepte kontrole pristupa i objektno-relacijskog pristupa bazama podataka. PostgreSQL je odabran zbog svoje stabilnosti, podrške za napredne SQL značajke i široke primjene u praksi [1]. Flask je korišten kao lagani web okvir koji omogućuje brzu izradu pregledne aplikacije uz minimalnu količinu koda [2].

Cilj rada je prikazati cjelokupan postupak izrade ovakvog sustava, od teorijskih osnova i modeliranja baze podataka, preko implementacije, pa sve do primjera korištenja i evaluacije rješenja.

2. Opis aplikacijske domene

Aplikacijska domena ovog projekta odnosi se na sustave za upravljanje korisničkim računima i njihovim pravima pristupa. Takvi sustavi koriste se u gotovo svim informacijskim aplikacijama koje imaju više korisnika s različitim razinama ovlasti. Cilj ovakvih sustava je osigurati da svaki korisnik može izvršavati samo one akcije koje su mu dopuštene.

U razvijenoj aplikaciji definirane su tri osnovne vrste korisnika: administrator, obični korisnik i auditor. Administrator ima najvišu razinu ovlasti te može dodavati i brisati korisnike, dodjeljivati im uloge i pregledavati sve podatke u sustavu. Obični korisnik ima ograničen pristup, koji se svodi na pregled podataka za koje ima dozvolu. Auditor ima specifičnu ulogu – nema mogućnost izmjene podataka, ali ima pravo pregleda audit zapisa, čime se osigurava nadzor nad aktivnostima u sustavu.

Temeljni koncepti aplikacijske domene su korisnik, uloga i dozvola. Korisnik predstavlja entitet koji se autentificira u sustav, dok uloga definira skup dozvola koje korisnik posjeduje. Dozvole određuju koje akcije su dopuštene, poput pregleda korisnika, dodavanja novih korisnika ili pregleda audit zapisa. Između korisnika i uloga uspostavljena je veza tipa više-prema-više, čime je omogućena fleksibilna kontrola pristupa.

Posebna značajka domene je vođenje audit zapisa. Svaka važna promjena u sustavu, poput dodavanja ili brisanja korisnika, automatski se bilježi u posebnoj tablici. Time se omogućuje praćenje aktivnosti i naknadna analiza događaja, što je posebno važno u sustavima gdje je sigurnost bitan faktor [3].

3. Teorijski uvod

Objektno-relacijske baze podataka predstavljaju proširenje klasičnih relacijskih baza koje uz standardne relacijske koncepte podržavaju i naprednije strukture podataka. Takav pristup omogućuje jednostavnije modeliranje složenijih domena, uz zadržavanje prednosti relacijskog modela poput transakcijske sigurnosti i SQL jezika [4].

U okviru ovog projekta korištena je objektno-relacijska baza podataka PostgreSQL. PostgreSQL podržava napredne mehanizme poput pogleda, okidača i funkcija, koji omogućuju implementaciju dijela poslovne logike izravno u bazi podataka. Time se smanjuje opterećenje aplikacijskog sloja i osigurava konzistentnost podataka neovisno o načinu pristupa bazi.

U usporedbi s objektno-orijentiranim bazama podataka, objektno-relacijski pristup zadržava jednostavnost relacijskog modela i bolju kompatibilnost s postojećim alatima. S druge strane, u odnosu na čisto relacijske baze, objektno-relacijske baze nude veću fleksibilnost pri modeliranju kompleksnijih odnosa i pravila. Nedostatak ovog pristupa može biti veća složenost dizajna baze i potreba za boljim razumijevanjem naprednih SQL koncepata.

Za kontrolu pristupa u aplikaciji korišten je RBAC model (Role-Based Access Control). U tom modelu korisnicima se ne dodjeljuju izravno dozvole, već se one dodjeljuju ulogama koje korisnici posjeduju. Ovakav pristup pojednostavljuje upravljanje pravima i olakšava proširenje sustava novim ulogama [5].

3.1. Poglavlje druge razine

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

3.1.1. Poglavlje treće razine

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

3.1.2. Poglavlje četvrte razine

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

4. Tehničke upute

Tehničke upute u nastavku opisuju način tehničkog oblikovanja rada i navođenja literature.

4.1. Upute za oblikovanje izgleda rada

4.1.1. Oblikovanje stranice

Stranice su oblikovane korištenjem sljedećih parametara:

- veličina i oblik papira je A4, okomito usmjerenje, margine 2,5 cm na svakoj strani;
- naslovna stranica rada se ne numerira;
- nakon naslovne stranice, sve sljedeće stranice do 1. Poglavlja se numeriraju rimskim brojevima, počevši od i;
- od 1. poglavlja nadalje, stranice se numeriraju arapskim brojevima;
- broj stranice treba pozicionirati desno 1,25 cm od dna stranice, font Arial 9.

4.1.2. Tekst rada

Tekst rada je oblikovan na sljedeći način:

- u pisanju teksta koristite font Arial 11 pt, s proredom 1,5 te razmakom 0 pt prije i razmakom 6 pt poslije odlomka, pri čemu je prvi redak uvučen za 1,25 cm;
- u naslovima prve razine „3. Razrada teme“ koristite font Arial 18 pt, podebljano, prijelom stranice (svaki naslov prve razine treba biti na novoj stranici), s proredom 1,5 te razmakom 0 pt prije i razmakom 18 pt poslije odlomka;
- u naslovima druge razine „2.1. Naslov“ koristite font Arial 16 pt, podebljano, s proredom 1,5 te razmakom 18 pt prije i razmakom 12 pt poslije odlomka;
- u naslovima treće razine „2.1.1. Naslov“ koristite font Arial 14 pt, podebljano, s proredom 1,5 te razmakom 12 pt prije i razmakom 6 pt poslije odlomka;
- u naslovima četvrte razine „2.1.1.1. Naslov“ koristite font Arial 12 pt, podebljano, s proredom 1,5 te razmakom 6 pt prije i razmakom 6 pt poslije odlomka;
- ostalo značajno isticanje cjelina rada može biti istaknuto podebljanim i kurziv slovima, korištenjem fonta Arial 11 pt.



Slika 1: Osnovni koncepti modela umjetnog agenta; preuzeto iz [6]

4.1.3. Slike

Slike u radu je potrebno oblikovati na sljedeći način:

- opis slike navedite ispod slike uz numeraciju;
- za nazive slika koristite iste postavke fonta kao i za tekst, ali stavite opis slike u centrirani položaj;
- za oblikovanje same slike koristite font Arial 9 pt za tekst na slici; ispred same slike umetnite jedan prazan redak (osim ako je slika pozicionirana na početku stranice);
- kod prijeloma stranice treba obratiti posebnu pozornost da opis slike, izvor i sama slika moraju biti na istoj stranici;
- slike je potrebno numerirati redom pojavljivanja u tekstu;
- ako je slika preuzeta iz drugog izvora, nakon navođenja opisa slike je potrebno dodati i referencu izvornog djela;
- dozvoljeno je napraviti vlastitu preradu slika, grafikona ili tablica na način da se zadrži isti smisao sadržaja, ali promijeni izgled, pri čemu je obavezno u opisu slike navesti referencu izvornog djela ovako: <opis slike> ; prema [X]
- dozvoljeno je preuzeti samo jednu sliku, grafikon ili tablicu u izvornom obliku iz istog izvora. Za doslovno preuzimanje većeg dijela sadržaja potrebno je ishoditi dozvolu nositelja autorskih prava;
- primjer označavanja slike prikazan je kod slike 1.

4.1.4. Tablice

Tablice rada je potrebno oblikovati sukladno ovim uputama:

- opis tablice navedite iznad slike;

```

1 <head>
2   <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
3   <link rel="shortcut icon"
4     ↪ href="https://www.foi.unizg.hr/sites/default/files/favicon_0_1.ico"
5     ↪ type="image/vnd.microsoft.icon" />
6   <meta name="generator" content="Drupal 7 (http://drupal.org)" />
7   <link rel="canonical" href="https://www.foi.unizg.hr/hr" />
8   <link rel="shortlink" href="https://www.foi.unizg.hr/hr" />
9   <!-- Set the viewport width to device width for mobile -->
10  <meta name="viewport" content="width=device-width, initial-scale=1.0">
11  <title>Dobro %*došli*) na FOI | FOI</title>...
12 </head>

```

Isječak koda 1: Primjer isječka koda

```

1 print("Ovo je preuzeti dio koda")

```

Isječak koda 2: Ovo je primjer koda koji je preuzet iz [6, str. 23]

- za opise tablica koristite iste postavke fonta kao i za tekst, ali stavite opis tablice u centrirani položaj;
- za oblikovanje same tablice koristite font Arial 9 pt za tekst u tablici;
- tablice numerirajte redom pojavljivanja u tekstu;
- kod prijeloma stranice treba obratiti posebnu pozornost da opis tablice, izvor i sama tablica moraju biti na istoj stranici;
- ako je tablica preuzeta iz drugog izvora, nakon navođenja opisa tablice potrebno je navesti izvor, na isti način kako je opisano kod slika;
- primjer označavanja tablice možete vidjeti u nastavku (tablica 1).

Tablica 1: Prikaz podataka o učestalosti pojavljivanja objekta; prema [7]

4.1.5. Programski kod

Za oblikovanje teksta koji je programski kod korišten je font Courier, veličine 10 pt, jednostruki prored, 6 pt iza odlomka, npr. HTML kod dijela zaglavlja početne web stranice FOI weba je prikazan kao isječak koda 1.

U slučaju preuzetog programskog koda, za isti je nužno potrebno naznačiti izvor, kao u isječku koda 2.

Ponekad palatali mogu stvarati probleme u opisima isječaka koda, pa ih je potrebno zamijeniti \LaTeX kodovima: č/š/ž: `\v{c/s/z}`, ć: `\'c`, đ: `\dj`.

4.1.6. Formule

Za unos formula koristite editor za formule. Matematičke izraze moguće je pisati unutar teksta $E = mc^2$ ili izdvojeno:

$$a^2 + b^2 = c^2$$

4.1.7. Kratice

Želite li koristiti kratice pojmova u tekstu, kad prvi put spominjete pojam potrebno je navesti puni naziv, a kraticu navesti u zagradi: informacijske i komunikacijske tehnologije (IKT). Nakon toga možete koristiti kratice u tekstu. Poželjno je u naslovima koristiti pune nazive.

Kratice je moguće definirati i korištenjem naredbe `\acronym` paketa, prije početka dokumenta (naredba `\begin{document}`). Tako definirane kratice pozivaju se unutar dokumenta naredbom `\ac{}` ili `\Ac{}` za veliko početno slovo. Takva će naredba svoje prvo pojavljivanje u dokumentu zamijeniti potpuno ispisanim značenjem kratice i kraticom, dok će svako sljedeće pojavljivanje biti zamijenjeno samo postavljenom kraticom, npr.: Višeagentni sustav (VAS) je sustav koji se sastoji od većeg broja agenata. Prilikom ponovnog korištenja naredbe, ispisana je samo kratica: VAS.

4.1.8. Strano nazivlje

Strano nazivlje se u tekstu navodi u zagradi, napisano *kurzivom*, nakon hrvatskog izraza, npr. analiza društvene mreže (engl. *Social Network Analysis - SNA*).

4.2. Navođenje literature

Za navođenje literature u radu koristite **IEEE** stil. Važno je dosljedno primjenjivati odabrani stil u cijelom radu.

U popisu literature potrebno je navesti svu literaturu i samo literaturu koju ste koristili u tekstu.

Uz svaku preuzetu tvrdnju potrebno je navesti njezin izvor, tj. referencu. Reference se u tekstu navode tako da se uz citirani tekst navede izvor, sukladno načinu propisanom odabranim stilom i FOI preporukama za citiranje i referenciranje [8]. Prilikom referenciranja knjiga, uvijek je potrebno navoditi i broj ili brojeve stranica.

Za citate i njihovo referenciranje koristite naredbu `\blockquote` kako slijedi. Citat će se automatski oblikovati kako je potrebno, ovisno o duljini citata. U kratkom citatu je pojašnjeno da bi se hibridna platforma za orkestraciju mogla u domeni videoigara koristiti „za implementaciju većeg broja inteligentnijih, a samim time i interesantnijih, suparnika.” [9, str. 5] Ono što slijedi nakon obavezno navedenog broja stranice uz izvor citata je primjer duljeg citata.

„Iako je testiranje performansi MMO igara dobro implementirano na brojnim platformama, testiranje iskustva igranja (posebno u primjerice MMORPG igrama) velikog razmjera postaje iznimno složen zadatak, kao što je okvirno predstavljeno u (Schatten, Okreša Đurić, Tomičić i Ivković, 2017). Ne samo da je potrebno testirati razne logičke zagonetke i zadatke koje igrač mora riješiti kako bi napredovao u igri, već i pojavljujuću interakciju među igračima koja bi mogla u potpunosti promijeniti rezultat igre. Korištenjem distribuirane platforme za orkestraciju bilo bi moguće dodavanje automatiziranih igrača po volji instanciranjem novih agenata. Štoviše, rezultati testiranja mogli bi biti dodatno analizirani dodavanjem agenata za analizu ili izvještavanje.” [9, str. 6]

Podaci o svim bibliografskim jedinicama nalaze se u `lib.bib` datoteci u `BibLaTeX` formatu. Bibliografske jedinice korištene u ovom dokumentu su:

- knjige [6], [7], [10],
- članak s konferencije [11],
- članci iz časopisa [12], [13],
- web-stranica [14].

5. Zaključak

Ovdje treba sažeto rezimirati najvažnije rezultate razrade teme rada. Potrebno je sažeto opisati što je predmet rada [14], koje su metode, tehnike, programski alati ili aplikacije korištene u razradi rada te koje su pretpostavke dokazane, a koje opovrgnute. Sadržajno, ono što se u uvodu rada najavljuje i kasnije je obuhvaćeno u samom radu, moralo bi biti opisano u zaključnom dijelu kroz rezultate rada.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Popis literature

- [1] PostgreSQL Global Development Group, *PostgreSQL Documentation*, <https://www.postgresql.org/docs/>, Accessed: 2026-01-10, 2023.
- [2] M. Grinberg, *Flask Web Development*, 2. izdanje. O'Reilly Media, 2018.
- [3] A. Behl, *Cybersecurity and Cyberwar*. Oxford University Press, 2017.
- [4] R. Elmasri i S. B. Navathe, *Fundamentals of Database Systems*, 7. izdanje. Boston: Pearson, 2016.
- [5] R. S. Sandhu, E. J. Coyne, H. L. Feinstein i C. E. Youman, „Role-Based Access Control Models,” *IEEE Computer*, sv. 29, br. 2, str. 38–47, 1996.
- [6] S. J. Russell i P. Norvig, ur., *Artificial Intelligence: A Modern Approach* (Pearson Series in Artificial Intelligence), 4. izdanje, u sur. s M. Chang i dr. Harlow, UK: Pearson Education Limited, 2022., 1166 str., ISBN: 978-1-292-40113-3.
- [7] M. Wooldridge, *An Introduction to MultiAgent Systems*, 2. izdanje. Glasgow, UK: John Wiley & Sons Ltd., 2009., 484 str., ISBN: 978-0-470-51946-2.
- [8] Sveučilište u Zagrebu, Fakultet organizacije i informatike, *FOI: Preporuke citiranja i referenciranja primjenom stila referenciranja IEEE*, 11. 2017.
- [9] M. Schatten, B. Okreša Đurić i I. Tomičić, „Platforme za orkestraciju hibridne umjetne inteligencije u računalnim igrama – konceptualni model,” *Central European Conference on Information and Intelligent Systems*, V. Strahonja, W. Steingartner i V. Kirinić, ur., Varaždin, HR: Faculty of Organization and Informatics, University of Zagreb, 7.–9. 10. 2020., str. 3–8.
- [10] D. Oraić Tolić, *Akademsko pismo: Strategije i tehnike klasične retorike za suvremene studentice i studente*. Zagreb, HR: Naklada Ljevak, 2011., 681 str., ISBN: 978-953-303-357-0.
- [11] B. Okreša Đurić i I. Tomičić, „Modelling Forming of Temporary Coalitions in the Context of Massively Multiplayer Online Role-Playing Games,” *Proceedings of the Central European Conference on Information and Intelligent Systems*, V. Strahonja i V. Kirinić, ur., Varaždin, HR: Faculty of Organization and Informatics, University of Zagreb, 2.–4. 10. 2019., str. 7–12.
- [12] M. Schatten, J. Ševa i I. Tomičić, „A roadmap for scalable agent organizations in the Internet of Everything,” *Journal of Systems and Software*, sv. 115, str. 31–41, 2016.

- [13] J. Rincon, F. de la Prieta, D. Zanardini, V. Julian i C. Carrascosa, „Influencing Over People with a Social Emotional Model,” *Neurocomputing*, sv. 231, str. 47–54, 3.2017., ISSN: 09252312. DOI: 10.1016/j.neucom.2016.03.107
- [14] B. Copeland. „Artificial Intelligence,” Britannica, pogledano 15. rujna 2020. adresa: <https://www.britannica.com/technology/artificial-intelligence>

Popis slika

1. Osnovni koncepti modela umjetnog agenta; preuzeto iz [6] 6

Popis tablica

1. Prikaz podataka o učestalosti pojavljivanja objekta; prema [7] 7

Popis isječka koda

1.	Primjer isječka koda	7
2.	Ovo je primjer koda koji je preuzet	7

Prilozi

1. Prilog 1

2. Prilog 2