

SVEUČILIŠTE U ZAGREBU  
FAKULTET ORGANIZACIJE I INFORMATIKE  
VARAŽDIN

Roberto Šandro

**BAZAZNANJA O POSLOVNIM PRAVILIMA  
U SUSTAVU ZA IZNAJMLJIVANJE VOZILA  
PRIMJENOM JEZIKA FLORA-2**

PROJEKT

**DEKLARATIVNO PROGRAMIRANJE**

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**

**BAZA ZNANJA O POSLOVNIM PRAVILIMA U SUSTAVU ZA  
IZNAJMLJIVANJE VOZILA PRIMJENOM JEZIKA FLORA-2**

**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 potvrđio prihvaćanjem odredbi u sustavu FOI Radovi*

---

## **Sažetak**

U radu je razvijena baza znanja o poslovnim pravilima u domeni sustava za iznajmljivanje vozila, implementirana korištenjem deklarativnog programskog jezika Flora-2. Sustav modelira kupce, vozila i zahtjeve za najam te primjenjuje skup poslovnih pravila na temelju kojih se automatski donose odluke o prihvaćanju ili odbijanju zahtjeva. Implementirana pravila obuhvaćaju provjeru dobi i vozačkog iskustva kupca, status na crnoj listi, broj prometnih prekršaja te posebne uvjete najma, poput prelaska granice ili dodatnog vozača. Osim konačne odluke, sustav omogućuje dohvata razloga odbijanja i izračun potrebnog depozita. Rad demonstrira primjenu deklarativnog programiranja za izgradnju fleksibilnog i proširivog sustava poslovnih pravila.

**Ključne riječi:** baza znanja; poslovna pravila; deklarativno programiranje; Flora-2; logičko programiranje; ekspertni sustavi; rent-a-car sustav

# Sadržaj

<b>1. Uvod . . . . .</b>	<b>1</b>
<b>2. Teorijski temelj baze znanja i poslovnih pravila . . . . .</b>	<b>2</b>
2.1. Ekspertni sustavi i baze znanja . . . . .	2
2.2. Poslovna pravila i zaključivanje . . . . .	2
2.3. Logičko programiranje . . . . .	2
2.4. Flora-2 kao jezik za baze znanja . . . . .	3
<b>3. Kritički osvrt na praktičnu izvedivost i primjenu . . . . .</b>	<b>4</b>
3.1. Prednosti deklarativnog pristupa . . . . .	4
3.2. Ograničenja i nedostaci . . . . .	4
3.3. Primjenjivost u stvarnim sustavima . . . . .	4
<b>4. Opis implementacije sustava . . . . .</b>	<b>6</b>
4.1. Modeliranje osnovnih entiteta . . . . .	6
4.2. Unos činjenica u bazu znanja . . . . .	6
4.3. Implementacija poslovnih pravila . . . . .	7
4.4. Izračun depozita . . . . .	7
4.5. Prednosti implementacijskog pristupa . . . . .	7
<b>5. Prikaz rada sustava . . . . .</b>	<b>8</b>
5.1. Izvršavanje upita nad bazom znanja . . . . .	8
5.2. Primjer zaključivanja – razlog odbijanja . . . . .	8
5.3. Određivanje dodatnih uvjeta i depozita . . . . .	9
5.4. Zaključak prikaza rada . . . . .	9
<b>6. Zaključak . . . . .</b>	<b>10</b>
<b>Popis literature . . . . .</b>	<b>11</b>
<b>Popis isječaka koda . . . . .</b>	<b>12</b>

# 1. Uvod

U mnogim stvarnim sustavima odluke se ne donose proizvoljno, već na temelju jasno definiranih poslovnih pravila. Takva pravila često ovise o većem broju uvjeta i s vremenom se mijenjaju, što otežava njihovu implementaciju u klasičnim programskim jezicima. Zbog toga se u određenim domenama sve češće koriste deklarativni pristupi, koji omogućuju da se pravila opisuju na jasan i pregledan način, neovisno o konkretnoj implementaciji algoritama.

Jedna od domena u kojoj je primjena poslovnih pravila posebno izražena je sustav za iznajmljivanje vozila (rent-a-car). Prilikom obrade zahtjeva za najam potrebno je uzeti u obzir različite čimbenike, poput dobi i vozačkog iskustva kupca, vrste i vrijednosti vozila, trajanja najma te posebnih uvjeta kao što su prelazak granice ili dodatni vozač. Takvi sustavi moraju donositi konzistentne odluke, ali i omogućiti jasno objašnjenje zašto je određeni zahtjev prihvачen ili odbijen.

Deklarativno programiranje, a posebno logičko programiranje, pokazalo se prikladnim za izgradnju baza znanja i ekspertnih sustava koji se temelje na pravilima i zaključivanju. Naglasak je na definiranju činjenica i pravila koja opisuju domenu problema. Na temelju tih pravila sustav samostalno izvodi zaključke, što olakšava održavanje i proširivanje sustava.

U tom kontekstu razvijen je jezik Flora-2, koji proširuje klasično logičko programiranje objektno-orientiranim konceptima te omogućuje prirodno modeliranje stvarnih domena pomoću objekata, atributa i pravila [1]. Takav pristup posebno je pogodan za modeliranje poslovnih pravila, jer omogućuje jasno razdvajanje znanja o domeni od mehanizma zaključivanja.

U ovom radu razvijena je baza znanja o poslovnim pravilima u domeni rent-a-car sustava korištenjem jezika Flora-2. Sustav modelira kupce, vozila i zahtjeve za najam te primjenjuje skup pravila na temelju kojih se automatski određuje ishod zahtjeva. Osim donošenja konačne odluke, sustav omogućuje i dohvrat razloga odbijanja te dodatnih uvjeta najma, čime se postiže veća transparentnost procesa zaključivanja.

Cilj rada je pokazati kako se deklarativnim pristupom može izgraditi jednostavan, ali proširiv sustav poslovnih pravila, koji jasno odvaja znanje o domeni od implementacijskih detalja. Time se demonstrira praktična primjena baza znanja i logičkog zaključivanja u rješavanju stvarnog problema, u skladu s konceptima obrađenima u okviru kolegija Deklarativno programiranje.

## **2. Teorijski temelj baze znanja i poslovnih pravila**

Kako bi se bolje razumjela izrada baze znanja za sustav poslovnih pravila, potrebno je ukratko objasniti osnovne pojmove na kojima se ovakav sustav temelji. U ovom poglavlju dan je pregled ekspertnih sustava, baza znanja i logičkog programiranja, s posebnim naglaskom na jezik Flora-2 koji je korišten u praktičnom dijelu rada.

### **2.1. Ekspertni sustavi i baze znanja**

Ekspertni sustavi su računalni sustavi koji pokušavaju oponašati način razmišljanja ljudskog stručnjaka u određenoj domeni. Umjesto da se oslanjaju na unaprijed definirane algoritme, oni koriste znanje zapisano u obliku činjenica i pravila, na temelju kojih donose odluke [2].

Središnji dio svakog ekspertnog sustava je baza znanja. Ona sadrži informacije o domeni problema, kao i pravila koja opisuju kako se iz tih informacija mogu izvesti novi zaključci. Takav pristup omogućuje da se sustav lako prilagodi promjenama, jer se promjenom pravila može promijeniti ponašanje sustava bez izmjena programskog koda.

U praksi se baze znanja često koriste u domenama gdje postoji veći broj uvjeta koji utječu na donošenje odluka, kao što su medicinska dijagnostika, finansijski sustavi ili sustavi poslovnih pravila.

### **2.2. Poslovna pravila i zaključivanje**

Poslovna pravila predstavljaju formalizirani opis odluka koje se u nekoj organizaciji donose na temelju zadanih uvjeta. Tipičan primjer poslovnog pravila je odluka o tome ispunjava li korisnik uvjete za određenu uslugu ili ne.

U sustavima temeljenima na pravilima, poslovna pravila najčešće se zapisuju u obliku izraza tipa „ako–onda“. Na temelju zadanih činjenica i pravila sustav samostalno zaključuje nove informacije, poput razloga odbijanja zahtjeva ili dodatnih uvjeta koji se moraju ispuniti [3].

Ovakav način rada posebno je pogodan za domene u kojima je važno objasniti zašto je sustav donio određenu odluku. Budući da su pravila eksplicitno zapisana, moguće je jasno prikazati koji su uvjeti doveli do određenog zaključka.

### **2.3. Logičko programiranje**

Logičko programiranje je paradigma deklarativnog programiranja u kojoj se program sastoji od skupa činjenica i pravila, dok se izvođenje zaključaka prepušta mehanizmu zaključivanja. Temeljni koncepti logičkog programiranja, na kojima se temelje i suvremeni jezici za baze znanja, opisani su u klasičnoj literaturi [4]. Programer ne opisuje kako se problem rješava

korak po korak, već opisuje što vrijedi u određenoj domeni.

Jedna od glavnih prednosti logičkog programiranja je njegova preglednost. Pravila su često vrlo bliska prirodnom jeziku, što olakšava razumijevanje i održavanje sustava. Zbog toga se logičko programiranje često koristi za izgradnju baza znanja i ekspertnih sustava.

## 2.4. Flora-2 kao jezik za baze znanja

Flora-2 je jezik za izgradnju baza znanja koji se temelji na logičkom programiranju, ali uvodi i elemente objektno-orientiranog pristupa [1]. U Flora-2 jeziku moguće je definirati objekte, njihove atribute i odnose među njima, što olakšava modeliranje stvarnih sustava.

U kontekstu ovog rada, entiteti poput kupca, vozila i zahtjeva za najam mogu se prirodno opisati kao objekti s pripadajućim svojstvima. Poslovna pravila zatim se definiraju kao logičke implikacije koje povezuju te objekte i njihove atribute.

Prednost korištenja jezika Flora-2 je u tome što se poslovna pravila mogu jasno i pregleđeno zapisati, a sustav automatski zaključuje posljedice tih pravila. Time se postiže jasna razdvojenost između znanja o domeni i same implementacije sustava. Širi kontekst baza znanja i zaključivanja u sklopu umjetne inteligencije obrađen je i u klasičnoj literaturi iz tog područja [5].

### **3. Kritički osvrt na praktičnu izvedivost i primjenu**

U ovom poglavlju razmatra se praktična primjenjivost izrađenog sustava poslovnih pravila te se ističu njegove prednosti i ograničenja. Cilj poglavlja nije detaljna tehnička analiza, već realan osvrt na to koliko je ovakav pristup pogodan za rješavanje problema u stvarnom okruženju.

#### **3.1. Prednosti deklarativnog pristupa**

Jedna od glavnih prednosti korištenja deklarativnog pristupa i baze znanja je jasnoća zapisa poslovnih pravila. Pravila su zapisana na pregledan način i izravno opisuju uvjete pod kojima se donose odluke, bez potrebe za složenim algoritmima. Zbog toga je relativno jednostavno razumjeti kako sustav funkcioniра, čak i osobama koje nisu stručnjaci za programiranje. Takav pristup odgovara općim načelima upravljanja poslovnim pravilima, prema kojima se pravila trebaju jasno odvojiti od ostatka aplikacijske logike [6].

Dodatna prednost je mogućnost lakog proširivanja sustava. Dodavanje novog pravila ili izmjena postojećeg pravila ne zahtijeva promjene u cijeloj aplikaciji, već se svodi na izmjenu dijela baze znanja. Ovakav način razdvajanja znanja i mehanizma zaključivanja tipičan je za ekspertne sustave i baze znanja [2], [3]. Takav pristup posebno je koristan u domenama u kojima se poslovna pravila često mijenjaju, što je čest slučaj u stvarnim sustavima.

Također, sustav omogućuje objašnjenje donesenih odluka. Budući da su razlozi odbijanja i dodatni uvjeti eksplicitno modelirani pravilima, moguće je jasno prikazati zašto je određeni zahtjev prihvaćen ili odbijen, što povećava transparentnost sustava.

#### **3.2. Ograničenja i nedostaci**

Iako deklarativni pristup ima brojne prednosti, postoje i određena ograničenja. Prije svega, jezici poput Flora-2 nisu široko korišteni u komercijalnim sustavima, što može otežati integraciju s drugim tehnologijama ili pronalazak razvojnih alata i dokumentacije.

Osim toga, za složenije sustave s velikim brojem pravila može doći do problema s održavanjem baze znanja. Kako broj pravila raste, postaje teže pratiti njihove međusobne odnose i moguće konflikte, što zahtijeva dodatnu pažnju prilikom razvoja i testiranja sustava.

#### **3.3. Primjenjivost u stvarnim sustavima**

U kontekstu stvarnih rent-a-car sustava, ovakav pristup bio bi posebno koristan za dio sustava koji se bavi provjerom uvjeta i donošenjem odluka. Pravila vezana uz dob korisnika, vozačko iskustvo ili posebne uvjete najma mogu se jasno izraziti i lako prilagođavati.

Zaključno, izrađeni sustav predstavlja dobar primjer primjene baza znanja i deklara-

tivnog programiranja u rješavanju konkretnog problema. Iako postoje ograničenja, sustav uspješno demonstrira osnovne ideje i principe koji stoje iza ekspertnih sustava i poslovnih pravila.

## 4. Opis implementacije sustava

U ovom poglavlju opisana je implementacija baze znanja o poslovnim pravilima u domeni rent-a-car sustava. Implementacija je izrađena u jeziku Flora-2, pri čemu je naglasak stavljen na jasno modeliranje domenskih pojmove i definiranje poslovnih pravila koja se nad njima primjenjuju. Odabrani pristup implementaciji temelji se na mogućnostima jezika Flora-2 za modeliranje objekata, atributa i pravila u bazi znanja [1].

Sustav je koncipiran kao baza znanja koja se sastoji od objekata, njihovih atributa i skupa pravila na temelju kojih se izvode zaključci. Takav pristup omogućuje preglednu strukturu i jednostavno proširivanje sustava.

### 4.1. Modeliranje osnovnih entiteta

U prvom koraku definirani su osnovni entiteti sustava: kupac, vozilo i zahtjev za najam. Svaki od tih entiteta modeliran je kao objekt u jeziku Flora-2.

Entitet *kupac* sadrži osnovne podatke potrebne za donošenje odluka, kao što su dob, godine vozačkog iskustva, informacija o tome nalazi li se kupac na crnoj listi te broj prometnih prekršaja. Ovi atributi izravno utječu na pravila vezana uz prihvatanje ili odbijanje zahtjeva.

Entitet *vozilo* opisuje karakteristike vozila koje su važne za poslovna pravila, uključujući kategoriju vozila, informaciju o tome radi li se o luksuznom vozilu, njegovu vrijednost te dodatne tehničke značajke. Na temelju tih atributa definiraju se uvjeti najma i visina depozita.

Entitet *zahtjev\_najma* povezuje kupca i vozilo te sadrži podatke o samom najmu, poput trajanja najma, prelaska granice i dodatnog vozača. Osim ulaznih podataka, zahtjev za najam sadrži i attribute koji se određuju zaključivanjem, kao što su razlog odbijanja, posebni uvjeti i iznos depozita.

### 4.2. Unos činjenica u bazu znanja

Nakon definiranja strukture objekata, u bazu znanja uneseni su konkretni primjeri kupaca, vozila i zahtjeva za najam. Ti primjeri služe kao testni podaci na kojima se provjerava ispravnost definiranih poslovnih pravila.

Unosom različitih kombinacija podataka omogućeno je testiranje različitih scenarija, poput mladih vozača, luksuznih vozila ili zahtjeva koji uključuju prelazak granice. Time se osigurava da sustav može ispravno reagirati na različite situacije koje se mogu pojaviti u stvarnoj primjeni.

## **4.3. Implementacija poslovnih pravila**

Poslovna pravila implementirana su kao logička pravila u jeziku Flora-2. Pri implementaciji sustava korištene su i smjernice iz službene dokumentacije jezika Flora-2, osobito u dijelu definiranja objekata i pravila [7]. Pravila definiraju uvjete pod kojima se zahtjev za najam odbija ili prihvaca, kao i dodatne uvjete koji se primjenjuju na određeni zahtjev.

Primjeri takvih pravila uključuju odbijanje zahtjeva ako se kupac nalazi na crnoj listi, ako nema dovoljno vozačkog iskustva ili ako ne ispunjava minimalnu dob za određenu kategoriju vozila. Također su definirana pravila koja uvode dodatne uvjete, poput povećanog depozita u slučaju luksuznog vozila ili prelaska granice.

Pravila su zapisana na jasan i pregledan način, što omogućuje jednostavno razumijevanje razloga zbog kojih sustav donosi određenu odluku.

## **4.4. Izračun depozita**

Poseban dio implementacije odnosi se na izračun iznosa depozita. Depozit se sastoji od osnovnog iznosa koji ovisi o kategoriji vozila te dodatnih iznosa koji se primjenjuju u posebnim slučajevima.

Za izračun depozita definirana su pomoćna pravila koja određuju dodatne iznose za prelazak granice, luksuzno vozilo i dodatnog vozača. Konačni iznos depozita računa se kao zbroj osnovnog depozita i svih primjenjivih dodataka.

Ovakav način izračuna omogućuje jednostavno proširivanje sustava, primjerice davanjem novih dodatnih uvjeta bez promjene postojećih pravila.

## **4.5. Prednosti implementacijskog pristupa**

Implementacija u jeziku Flora-2 omogućila je jasno razdvajanje podataka i pravila. Poslovna pravila zapisana su deklarativno, bez potrebe za pisanjem složenih algoritama, što sustav čini preglednim i lakin za održavanje.

Osim toga, sustav omogućuje da se za svaki zahtjev jasno utvrdi razlog odbijanja ili posebni uvjeti najma, čime se povećava transparentnost rada sustava. Takav pristup posebno je pogodan kao temelj za daljnji razvoj složenijih sustava.

## 5. Prikaz rada sustava

U ovom poglavlju prikazan je rad izrađene baze znanja kroz primjere upita i odgovaraćih zaključaka. Cilj poglavlja je pokazati kako sustav, na temelju zadanih činjenica i poslovnih pravila, donosi odluke o prihvaćanju ili odbijanju zahtjeva za najam vozila te određuje dodatne uvjete i iznos depozita.

### 5.1. Izvršavanje upita nad bazom znanja

Rad sustava ispituje se postavljanjem upita nad bazom znanja u okruženju Flora-2. Upiti omogućuju dohvat svih zahtjeva za najam, kao i provjeru pojedinačnih svojstava zahtjeva, poput razloga odbijanja ili iznosa depozita.

Primjer osnovnog upita kojim se dohvaćaju svi zahtjevi za najam prikazan je u nastavku:

---

1 `?z : zahtjev_najma.`

Isječak koda 1: Osnovni upit za dohvat svih zahtjeva za najam

Izvršavanjem ovog upita sustav vraća sve instance objekta `zahtjev_najma` koje su une-sene u bazu znanja.

### 5.2. Primjer zaključivanja – razlog odbijanja

Jedna od ključnih funkcionalnosti sustava je automatsko određivanje razloga odbijanja zahtjeva. Razlozi odbijanja definirani su poslovnim pravilima koja se primjenjuju na temelju atributa kupca i zahtjeva za najam.

Primjer pravila kojim se zahtjev odbija ako se kupac nalazi na crnoj listi prikazan je u nastavku:

---

1 `?z [ razlog_odbijanja -> crna_lista ] :-`  
2   `?z : zahtjev_najma[ kupac -> ?K ],`  
3   `?K : kupac[ na_crnoj_listi -> true ].`

Isječak koda 2: Pravilo za odbijanje zahtjeva zbog crne liste

Kada se nad bazom znanja postavi upit za dohvat razloga odbijanja, sustav automatski primjenjuje definirana pravila i vraća odgovarajuće zaključke:

---

1 `?z : zahtjev_najma[ razlog_odbijanja -> ?razlog].`

Isječak koda 3: Upit za dohvat razloga odbijanja zahtjeva

Na temelju rezultata upita moguće je jasno utvrditi koji su zahtjevi odbijeni i iz kojeg razloga, što povećava transparentnost rada sustava.

### **5.3. Određivanje dodatnih uvjeta i depozita**

Osim razloga odbijanja, sustav određuje i dodatne uvjete najma te iznos depozita. Depozit se izračunava na temelju kategorije vozila i dodatnih uvjeta, poput prelaska granice ili dodatnog vozača.

Primjer upita kojim se dohvaća iznos depozita za određeni zahtjev prikazan je u nastavku:

---

1 zht\_horvat\_a8\_2026\_01\_17:zahtjev\_najma[depozit -> ?d].

---

Isječak koda 4: Upit za dohvrat iznosa depozita za konkretan zahtjev

Sustav na temelju definiranih pravila automatski izračunava konačni iznos depozita, uzimajući u obzir sve primjenjive dodatke. Time se pokazuje kako se složenija poslovna logika može izraziti na pregledan i razumljiv način pomoću baze znanja.

### **5.4. Zaključak prikaza rada**

Prikazani primjeri potvrđuju da sustav ispravno primjenjuje poslovna pravila definirana u bazi znanja. Na temelju unesenih podataka i pravila sustav može automatski donositi odluke, određivati razloge odbijanja i izračunavati dodatne uvjete najma.

Ovakav način rada pokazuje prednosti deklarativnog pristupa, jer se ponašanje sustava jasno može pratiti kroz definirana pravila i rezultate upita, bez potrebe za dodatnom aplikacijskom logikom.

## 6. Zaključak

U ovom radu izrađena je baza znanja o poslovnim pravilima u domeni rent-a-car sustava korištenjem deklarativnog i logičkog programiranja u jeziku Flora-2. Cilj rada bio je pokazati kako se stvarni problem, koji uključuje veći broj uvjeta i pravila, može modelirati na pregledan i razumljiv način primjenom baza znanja i ekspertnih sustava [2].

U praktičnom dijelu rada definirani su osnovni entiteti sustava, poput kupaca, vozila i zahtjeva za najam, te su nad njima implementirana poslovna pravila koja određuju ishod zahtjeva. Sustav omogućuje automatsko zaključivanje o prihvatljivosti zahtjeva, određivanje razloga odbijanja te izračun dodatnih uvjeta i iznosa depozita. Time je demonstrirana glavna prednost deklarativnog pristupa, odnosno jasno razdvajanje znanja o domeni od same implementacije.

Rezultati prikazani kroz primjere upita potvrđuju da sustav ispravno primjenjuje definirana pravila i donosi konzistentne odluke. Pravila su zapisana na način koji je razumljiv i lako proširiv, što omogućuje jednostavne izmjene i nadogradnje sustava bez većih promjena postojećeg koda, što je u skladu s osnovnim idejama baza znanja i logičkog programiranja [1].

Iako izrađeni sustav predstavlja pojednostavljeni model stvarnog rent-a-car sustava, on jasno pokazuje prednosti korištenja deklarativnog pristupa u domenama temeljenima na pravilima. U stvarnoj primjeni ovakav bi sustav mogao biti nadograđen dodatnim pravilima, većim skupom podataka ili integracijom s drugim dijelovima informacijskog sustava.

Zaključno, rad uspješno demonstrira praktičnu primjenu koncepata obrađenih u okviru kolegija Deklarativno programiranje te potvrđuje da se jezik Flora-2 može učinkovito koristiti za izgradnju funkcionalnih i proširivih baza znanja u konkretnim aplikacijskim domenama.

# Popis literature

- [1] M. Kifer i G. Yang, „Flora-2: An Object-Oriented Knowledge Base Language,” *Theoretical Computer Science*, sv. 396, br. 1–3, str. 243–260, 2008., Preuzeto: 16.01.2026. adresa: <https://www.sciencedirect.com/science/article/pii/S0304397508001449>
- [2] P. Jackson, *Introduction to Expert Systems*, 3. izdanje. Addison-Wesley, 1999.
- [3] F. van Harmelen, V. Lifschitz i B. Porter, „Handbook of Knowledge Representation,” *Foundations of Artificial Intelligence*, sv. 3, str. 1–28, 2008., Preuzeto: 16.01.2026. adresa: <https://www.sciencedirect.com/bookseries/foundations-of-artificial-intelligence/vol/3>
- [4] J. W. Lloyd, *Foundations of Logic Programming*. Springer, 1987., Preuzeto: 16.01.2026. adresa: <https://books.google.ba/books?id=TeKpCAAAQBAJ>
- [5] S. Russell i P. Norvig, *Artificial Intelligence: A Modern Approach*, 4. izdanje. Pearson, 2021., Preuzeto: 16.01.2026. adresa: <https://people.engr.tamu.edu/guni/csce625/slides/AI.pdf>
- [6] R. G. Ross, *Principles of the Business Rule Approach*. Addison-Wesley, 2003., Pogledano: 16.01.2026. adresa: <https://books.google.ba/books?id=Uyv9hM0t6BsC>
- [7] M. Kifer, *Flora-2 User's Manual*, <http://flora.sourceforge.net>, Preuzeto: 16.01.2026., 2023.

# **Popis isječaka koda**

1.	Osnovni upit za dohvat svih zahtjeva za najam . . . . .	8
2.	Pravilo za odbijanje zahtjeva zbog crne liste . . . . .	8
3.	Upit za dohvat razloga odbijanja zahtjeva . . . . .	8
4.	Upit za dohvat iznosa depozita za konkretan zahtjev . . . . .	9