

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Žan Pižmoht

**Sistem za zagotavljanje zaupanja v  
dobavni verigi zdravil z uporabo  
verige blokov**

MAGISTRSKO DELO  
MAGISTRSKI ŠTUDIJSKI PROGRAM DRUGE STOPNJE  
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: prof. dr. Vlado Stankovski  
SOMENTOR: doc. dr. Petar Kochovski

Ljubljana, 2025



To delo je ponujeno pod licenco *Creative Commons Priznanje avtorstva-Deljenje pod enakimi pogoji 2.5 Slovenija* (ali novejšo različico). To pomeni, da se tako besedilo, slike, grafi in druge sestavine dela kot tudi rezultati zaključnega dela lahko prosto distribuirajo, reproducirajo, uporabljajo, priobčujejo javnosti in predelujejo, pod pogojem, da se jasno in vidno navede avtorja in naslov tega dela in da se v primeru spremembe, preoblikovanja ali uporabe tega dela v svojem delu, lahko distribuira predelava le pod licenco, ki je enaka tej. Podrobnosti licence so dostopne na spletni strani [creativecommons.si](http://creativecommons.si) ali na Inštitutu za intelektualno lastnino, Streliška 1, 1000 Ljubljana.



Izvorna koda zaključnega dela, njeni rezultati in v ta namen razvita programska oprema je ponujena pod licenco GNU General Public License, različica 3 (ali novejša). To pomeni, da se lahko prosto distribuira in/ali predeluje pod njenimi pogoji. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses/>.



## ZAHVALA

*Na tem mestu zapišite, komu se zahvaljujete za izdelavo magistrske naloge. V zahvali se poleg mentorja spodobi omeniti vse, ki so s svojo pomočjo prispevali k nastanku vašega izdelka.*

*Žan Pižmoht, 2025*



Vsem rožicam tega sveta.

*"The only reason for time is so that  
everything doesn't happen at once."*

— Albert Einstein





# Kazalo

**Povzetek**

**Abstract**

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Pregled literature in ozadje</b>	<b>3</b>
2.1	Zaupanje . . . . .	3
2.2	Upravljanje zaupanja v razpršenih sistemih . . . . .	7
2.3	Farmacevtske dobavne verige . . . . .	7
2.4	Tehnologija veriženja blokov v dobavnih verigah . . . . .	7
2.5	Semantični splet in ontološke metode . . . . .	7
2.6	Analiza vrzeli in prispevek te naloge . . . . .	7
2.7	Ontologija zaupanja . . . . .	7
2.8	Pametne pogodbe in pametni orakli . . . . .	7
<b>3</b>	<b>Implementacija sistema</b>	<b>9</b>
3.1	Arhitektura sistema . . . . .	9
3.2	Orodja in razvojno okolje . . . . .	15
3.3	Ontologija in orodja za sklepanje . . . . .	15
3.4	Razvoj pametnih pogodb in orakov . . . . .	17
3.5	Uporabniški vmesnik in API-ji . . . . .	17
<b>4</b>	<b>Evalvacija</b>	<b>19</b>
4.1	Varnostna analiza . . . . .	19

## KAZALO

4.2	Zmogljivost in razširljivost . . . . .	19
4.3	Natančnost ocene zaupanja . . . . .	19
4.4	Prednosti in omejitve . . . . .	19
4.5	Primerjava z obstoječimi rešitvami . . . . .	19
4.6	Pridobljena spoznanja . . . . .	19
<b>5</b>	<b>Zaključek in nadaljnje delo</b>	<b>21</b>
5.1	Povzetek prispevkov . . . . .	21
5.2	Nadaljnje raziskave . . . . .	21
<b>A</b>	<b>Shema ontologije in SWRL pravila</b>	<b>23</b>
<b>B</b>	<b>Izvilleki kode pametnih pogodb</b>	<b>25</b>

# Seznam uporabljenih kratic

kratica	angleško	slovensko
CA	classification accuracy	klasifikacijska točnost
DBMS	database management system	sistem za upravljanje podatkovnih baz
SVM	support vector machine	metoda podpornih vektorjev
...	...	...



# Povzetek

**Naslov:** Sistem za zagotavljanje zaupanja v dobavni verigi zdravil z uporabo verige blokov

V vzorcu je predstavljen postopek priprave magistrskega dela z uporabo okolja L<sup>A</sup>T<sub>E</sub>X. Vaš povzetek mora sicer vsebovati približno 100 besed, ta tukaj je odločno prekratek. Dober povzetek vključuje: (1) kratek opis obravnavanega problema, (2) kratek opis vašega pristopa za reševanje tega problema in (3) (najbolj uspešen) rezultat ali prispevek magistrske naloge.

## Ključne besede

*zaupanje, tehnologija veriženja blokov, dobavna veriga zdravil*



# Abstract

**Title:** Trust System in Pharmaceutical Supply Chain Using Blockchain

This sample document presents an approach to typesetting your BSc thesis using L<sup>A</sup>T<sub>E</sub>X. A proper abstract should contain around 100 words which makes this one way too short. A good abstract contains: (1) a short description of the tackled problem, (2) a short description of your approach to solving the problem, and (3) (the most successful) result or contribution in your thesis.

## Keywords

*trust, blockchain, pharmaceutical supply chain*





# Poglavje 1

## Uvod



## Poglavje 2

# Pregled literature in ozadje

### 2.1 Zaupanje

Pojem zaupanja je temeljni gradnik pri zasnovi našega sistema za farmacevtsko dobavno verigo. Zaupanje je kompleksen, večdimenzionalen in interdisciplinaren koncept, ki ga obravnavajo različne vede. Obravnava se v sociologiji, psihologiji, ekonomiji, računalništvu in še ostalih vedah. Prav zaradi te raznolikosti v literaturi ne obstaja enotna definicija, temveč več pristopov, ki poudarjajo različne vidike [1, 2].

#### 2.1.1 Definicije zaupanja

V družboslovnem kontekstu je zaupanje pogosto opredeljeno kot stanje pozitivnih pričakovanj glede dejanj druge osebe v okoliščinah, kjer obstaja določena stopnja tveganja [3]. Gambetta [4] zaupanje definira kot subjektivno verjetnost, da bo agent opravil določeno dejanje, še preden je to mogoče preveriti. Mayer, Davis in Schoorman [5] pa ga opredelijo kot pripravljenost ene stranke, da se izpostavi ranljivosti glede na dejanja druge stranke, ob pričakovanju, da bo ta delovala v skladu s pričakovanji.

V digitalnem okolju se koncept zaupanja prenaša iz družbenega v tehnični kontekst. Denning [6] poudarja, da je zaupanje v sistem lastnost, ki jo je mogoče formalno modelirati, specifikirati in verificirati. Sistem je torej zau-

panja vreden, če mu njegovi uporabniki zaupajo glede na opaženo skladnost z vnaprej določenimi standardi. Grandison in Sloman [2] zaupanje obravnavata kot kvalificirano prepričanje zaupnika o kompetentnosti, integriteti, varnosti in zanesljivosti zaupanja vrednega subjekta.

### 2.1.2 Lastnosti zaupanja

Skupni imenovalci različnih definicij so lastnosti, ki jih zaupanje vključuje:

- **Relacijska narava:** zaupanje vedno obstaja med dvema entitetama – zaupnika in zaupnika vrednim subjektom.
- **Tveganje in negotovost:** zaupanje vključuje sprejemanje ranljivosti, saj nikoli ni popolnega zagotovila, da bo entiteta izpolnila pričakovanja.
- **Subjektivnost:** stopnja zaupanja se razlikuje med akterji in je odvisna od konteksta, izkušenj in preferenc [7].
- **Dinamika:** zaupanje se sčasoma spreminja glede na nove informacije, interakcije in dokaze [8].
- **Asimetrija:** entiteta A lahko zaupa entiteti B, vendar to ne pomeni nujno recipročne relacije.
- **Nepopolna tranzitivnost:** če A zaupa B in B zaupa C, to še ne pomeni nujno, da A zaupa C.

### 2.1.3 Atributi zaupanja

Za digitalno okolje je pomembno, da zaupanje operacionaliziramo v obliki merljivih atributov. V literaturi se pogosto omenjajo naslednji dejavniki:

- **Kompetentnost** – sposobnost subjekta, da pravilno opravi zahtevano nalogo.
- **Integriteta** – skladnost z etičnimi načeli, standardi in pravili.

- **Dobronamernost** – predpostavka, da subjekt ne bo namerno povzročil škode [5].
- **Predvidljivost** – konsistentnost vedenja skozi čas.
- **Zanesljivost podatkov** – v kontekstu IoT in dobavnih verig se kot atributi zaupanja pogosto uporabljajo pravočasnost, točnost in celovitost podatkov [8].

#### 2.1.4 Zaupanje v digitalnem okolju

V okviru informacijskih sistemov in verig blokov je zaupanje ključno za sprejemanje odločitev o interakcijah med entitetami, ki se pogosto ne poznajo. Različni modeli zaupanje formalizirajo z uporabo zgodovine interakcij (direktno zaupanje), priporočil tretjih strani (indirektno zaupanje), kontekstualnih dejavnikov ali kombinacij teh pristopov [9]. Posebej v razpršenih in IoT sistemih je zaupanje ena od redkih poti za zagotavljanje varnosti in odpornosti pred notranjimi napadi, kjer tradicionalne metode (avtentikacija, kriptografija) ne zadostujejo.

S temi definicijami in atributi postavimo konceptualni okvir, ki omogoča gradnjo ontologije zaupanja ter kasnejše formalno sklepanje o zaupanju med akterji v farmacevtski dobavni verigi.



## 2.2 Upravljanje zaupanja v razpršenih sistemih

### 2.2.1 Modeli zaupanja

### 2.2.2 Zaupanja v IoT in pervasive okolju

## 2.3 Farmacevtske dobavne verige

### 2.3.1 Izzivi sledenja in boj proti ponarejanju

### 2.3.2 Regulatorni okvir

## 2.4 Tehnologija veriženja blokov v dobavnih verigah

### 2.4.1 Javne in dovoljene verige blokov

### 2.4.2 Pametne pogodbe za nadzor dostopa in sledljivost

## 2.5 Semantični splet in ontološke metode

### 2.5.1 RDF, OWL in orodja za ontologijo

### 2.5.2 Ontologija zaupanja

## 2.6 Analiza vrzeli in prispevek te naloge

## 2.7 Ontologija zaupanja

### 2.7.1 Koncepti in odnosi

### 2.7.2 SWRL pravila

### 2.7.3 Mehanizem razumevanja (angl. Reasoning Mechanism)

## 2.8 Pametne pogodbe in pametni orakli





## Poglavje 3

# Implementacija sistema

### 3.1 Arhitektura sistema

Arhitektura sistema za upravljanje zaupanja v farmacevtski dobavni verigi je sestavljena iz več plasti. Te omogočajo kombinacijo semantičnih podatkov, subjektivne ocene zaupanja in decentraliziranega shranjevanja rezultatov v verigo blokov. Z delitvijo sistema na plasti omogočamo modularnost, večjo preglednost in lažjo implementacijo nadaljnjih razširitev.

Arhitekturo razdelimo na tri plasti, katere bodo podrobneje opisane v naslednjih podpoglavjih:

#### 1. Plast znanja

V tej plasti se nahajajo vse ontološke definicije, primerki akterjev in njihove lastnosti. Sem spada tudi konceptualna zasnova lastnosti zaupanja, kot so licence, temperaturne nepravilnosti, točnost dostave ipd.

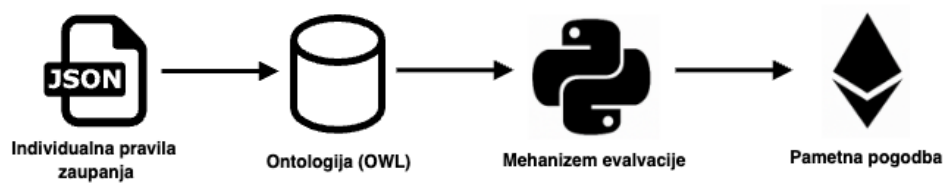
#### 2. Mehanizem za sklepanje zaupanja

Ta plast vključuje sistem za izvajanje ocenjevanja zaupanja. Združuje podatke iz ontologije in pravil zaupanja ter izvede sklepanje za vsak par entitet. Rezultat je preslikava razmerja zaupanja med akterji, ki je pripravljena za zapis v verigo blokov.

#### 3. Graf zaupanja na verigi blokov

Zadnja plast predstavlja pametno pogodbo, ki beleži razmerja zaupanja med entitetami. Shranjevanje rezultatov v verigo blokov omogoča transparentnost, trajnost in decentralizirano preverljivost zaupanja.

Na sliki 3.1 je prikazan arhitekturni pregled sistema.



**Slika 3.1:** Večplastna arhitektura sistema za upravljanje zaupanja

### 3.1.1 Plast znanja

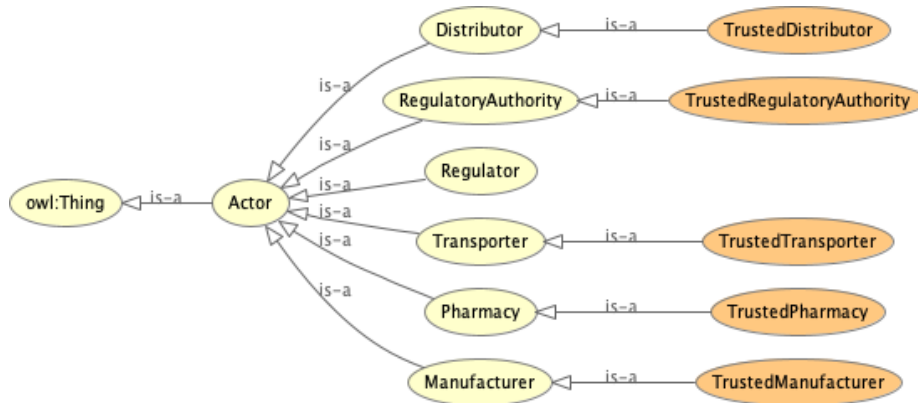
Plast znanja (angl. Knowledge Layer) definira semantično strukturo akterjev, njihove značilnosti in nabor pravil za individualno ocenjevanje zaupanja. V osnovi jo sestavljata dve ključni komponenti: ontologija in pravila zaupanja posameznih akterjev.

**Ontologija** Ontologija v jeziku OWL predstavlja formalno definiran model, ki vključuje vse vrste entitet, ki sodelujejo v farmacevtski dobavni verigi (npr. *Proizvajalec*, *Lekarna*, *Prevoznik*, *Regulator*). Vsaka vrsta entitete ima lahko specifične lastnosti, ki jih sistem uporablja pri sklepanju o zaupanju, kot so na primer:

- `hasDeliveryPunctuality` - točnost dostave,
- `hasTempViolationRate` - stopnja temperaturnih odstopanj,
- `hasLicense` - podatek o veljavni licenci,
- `hasGMP` - skladnost s proizvodnimi standardi GMP [10],
- `hasAuditScore` - rezultat presoje kakovosti.

Ontologija je zapisana v formatu RDF [11] zasnovan na označevalnem jeziku XML [12], ki omogoča enostavno obdelavo s knjižnico `rdflib` [13] v programskem jeziku Python. Na ta način lahko sklepanje temelji na aktualnih podatkih, zapisanih v ontološkem grafu, brez potrebe po relacijski bazi ali ročnem povezovanju.

Na sliki 3.2 vidimo vse akterje v ontologiji predstavljene v programu Protege.



**Slika 3.2:** Akterji ontologije v programu Protege

V ontologiji so definirani tudi posamezni primerki entitet (npr. *Pfizer*, *DHL*, *MediPlus*), skupaj z njihovimi lastnostmi. To omogoča sistemu, da na podlagi resničnih podatkov izvede evalvacijo zaupanja.

**Pravila zaupanja posameznih akterjev** Vsaka entiteta v sistemu lahko definira svoja lastna pravila zaupanja, ki opisujejo pogoje, pod katerimi določena entiteta zaupa drugi. Ta pravila so zapisana v berljivi in strukturirani obliki JSON, kar omogoča enostavno vključevanje v mehanizem sklepanja.

Primer politike pravil zaupanja za entiteto *Pfizer* je prikazan v spodnjem zapisu v formatu JSON:

```

{
  "actor": "http://example.org/trust#Pfizer",
  "trusts": {
    "Transporter": {
      "hasDeliveryPunctuality": { "gte": 0.9 },
      "hasTempViolationRate": { "lte": 0.05 }
    }
  }
}

```

Zapis pove, da entiteta *Pfizer* zaupa samo tistim prevoznikom, ki dosegajo vsaj 90 % pravočasnih dostav (`hasDeliveryPunctuality`  $\geq 0.9$ ) in katerih delež temperaturnih odstopanj pri transportu zdravil ne presega 5 % (`hasTempViolationRate`  $\leq 0.05$ ). Na ta način se pravila zaupanja natančno preslikajo v merljive pogoje, ki temeljijo na lastnostih, definiranih v ontologiji.

Mehanizem sklepanja nato uporabi ta pravila za vsako entiteto, s katero obravnavani akter vzpostavi zaupanje ter preveri, ali so pogoji izpolnjeni.

Ker so pravila zapisana v JSON obliki, je možno preprosto razviti uporabniški vmesnik, kjer lahko predstavnik podjetja (npr. administrator lekarne ali regulatorja) preko obrazca določil svoje kriterije za zaupanje. Vmesnik bi generiral JSON strukturo na podlagi definiranih lastnosti v ontologiji. To nam omogoča visoko stopnjo prilagodljivosti in enostavne uporabe brez tehničnega predznanja.

JSON pravila služijo kot most med semantičnim modelom in subjektivno presojo vsakega akterja. Tako omogočamo personalizirano ocenjevanje zaupanja.

### 3.1.2 Mehanizem za sklepanje zaupanja (angl. Trust Resolution Engine)

Mehanizem za sklepanje zaupanja predstavlja osrednjo logično plast sistema, kjer se združujejo podatki iz ontologije in pravila zaupanja posameznih akterjev. Tukaj se izvede proces ocenjevanja, ki določi, ali določena entiteta zaupa drugi glede na pogoje, zapisane v pravilih zaupanja posameznega akterja.

**Delovanje mehanizma** Python modul, razvit za potrebe sistema, izvaja naslednje korake:

1. Naloži ontologijo (OWL/RDF) in iz nje izlušči podatke o vseh akterjih ter njihovih lastnostih.

2. Prebere pravila zaupanja vsakega akterja, ki so zapisana v JSON formatu.
3. Za vsak par (ocenjevalec in tisti, ki je ocenjen) preveri, ali vrednosti atributov ocenjenega akterja ustrezajo pravilom ocenjevalca.
4. Rezultate ocenjevanja shrani v zbirko rezultatov (CSV datoteka) in jih pripravi za zapis v verigo blokov.

**Primer ocenjevanja** Če ima npr. akter *Pfizer* pravilo, da zaupa le prevoznikom z vsaj 90 % točnih dostav in manj kot 5 % temperaturnih nepravilnosti, bo mehanizem preveril te pogoje za vsakega prevoznika, zapisanega v ontologiji. Rezultat ocenjevanja je binarna vrednost (`true/false`), ki pove, ali akter izpolnjuje kriterije zaupanja.

### 3.1.3 Graf zaupanja na verigi blokov (angl. Blockchain Trust Registry)

Zadnja plast arhitekture je namenjena trajnemu in preverljivemu beleženju rezultatov ocenjevanja zaupanja. To dosežemo z uporabo pametne pogodbe na verigi blokov, ki deluje kot decentraliziran register zaupanja (angl. *trust registry*).

**Struktura pametne pogodbe** Pametna pogodba vsebuje podatkovno strukturo, ki beleži relacije zaupanja med akterji v obliki:

$$(akter1, akter2) \mapsto status\_zaupanja$$

kjer je `status_zaupanja` logična vrednost (`true/false`), ki označuje, ali akter 1 zaupa akterju 2.

**Funkcionalnosti** Pametna pogodba omogoča naslednje osnovne funkcije:

- `setTrustStatus()` — zapis oziroma posodobitev rezultata zaupanja na verigi blokov.

- `isTrusted()` — preverjanje, ali določen akter zaupa drugemu.
- `event TrustUpdated` — beleženje dogodkov o spremembi zaupanja za transparentnost in enostavno spremljanje.

## 3.2 Orodja in razvojno okolje

## 3.3 Ontologija in orodja za sklepanje

### 3.3.1 Ontologija zaupanja

Ontologija zaupanja je osnovni gradnik sistema, saj omogoča formaliziran opis vseh akterjev v farmacevtski dobavni verigi ter njihovih lastnosti. Zasnovana je v jeziku OWL. Glavni namen ontologije je, da lahko vse akterje opišemo na enoten način in jih obogatimo s podatki, ki so pomembni za ocenjevanje zaupanja.

#### Razredi akterjev

V ontologiji so definirani osnovni razredi, ki predstavljajo tipe akterjev. To so proizvajalec, distributer, lekarna, prevoznik in regulator. Vsi ti razredi so podrazredi splošnega razreda `Actor`, kar pomeni, da jih lahko obravnavamo skupaj, ko govorimo o celotni verigi. Vsak tip akterja ima svoje značilne lastnosti. Na primer, proizvajalci imajo lastnost `hasGMP`, ki pove, ali imajo veljavno GMP skladnost, prevozniki imajo lastnosti `hasDeliveryPunctuality` in `hasTempViolationRate`, lekarne pa lastnost `hasPrescriptionComplianceRate`.

#### Lastnosti zaupanja

Lastnosti opisujejo merljive kriterije, ki se uporabljajo pri presoji zaupanja. Nekaj primerov: - `hasGMP` opisuje ali je proizvajalec skladen z dobrimi proizvodnimi praksami, - `hasAuditScore` predstavlja oceno presoje kakovosti, - `hasLicense` označuje, ali ima akter veljavno licenco, - `hasDeliveryPunctuality`

izraža delež pravočasnih dostav, - `hasTempViolationRate` pove, kolikšen delež transportov je imel temperaturna odstopanja, - `hasPrescriptionComplianceRate` opisuje skladnost lekarne pri izdaji zdravil na recept, - `hasIssuedCertifications` je število certifikatov, ki jih je regulator izdal, - `hasJurisdictionLevel` pa določa raven pristojnosti regulatorja (npr. lokalna, nacionalna, globalna).

### Razredi zaupanja

Za vsak tip akterja so v ontologiji definirani tudi posebni razredi, ki predstavljajo zaupanja vredne entitete. To so na primer `TrustedManufacturer`, `TrustedDistributor`, `TrustedPharmacy`, `TrustedTransporter` in `TrustedRegulatoryAuthority`. Ti razredi so definirani z ekvivalenčnimi pogoji, kar pomeni, da reasoner lahko samodejno uvrsti entiteto v določen zaupanja vreden razred, če so pogoji izpolnjeni. Na primer, proizvajalec, ki ima GMP certifikat in dovolj visoko oceno presoje kakovosti, bo klasificiran kot `TrustedManufacturer`. Podobno velja za druge tipe akterjev.

### Primerki akterjev

V ontologijo so dodani tudi konkretni primerki, ki predstavljajo dejanske akterje v sistemu. Tako imamo na primer proizvajalca Pfizer in Novartis, prevoznika DHL, lekarno MediPlus, distributerja EuroLogistics ter regulatorja EMA. Vsak od teh primerkov ima določene lastnosti, kot so na primer ocena presoje, točnost dostave ali število izdanih certifikatov. Te vrednosti se potem uporabijo pri sklepanju o zaupanju.

#### 3.3.2 Primer toka podatkov in logike

Za ponazoritev delovanja si oglejmo primer, kjer proizvajalec Pfizer ocenjuje prevoznika DHL:

1. **Ontologija (OWL):** v ontologiji so zabeležene lastnosti prevoznika DHL, npr. `hasDeliveryPunctuality = 0.95` in `hasTempViolationRate = 0.02`.



2. **Pravila zaupanja:** Pfizerjeva politika zaupanja za razred *Transporter* zahteva `hasDeliveryPunctuality`  $\geq 0.90$  in `hasTempViolationRate`  $\leq 0.05$ .
3. **Sklepanje (Python):** mehanizem prebere lastnosti DHL iz ontologije in jih primerja s Pfizerjevimi pogoji. Ker sta oba pogoja izpolnjena, rezultat sklepanja je `true`.
4. **Zapis v verigo blokov:** rezultat (Pfizer zaupa DHL) se zapiše v pametno pogodbo. Funkcija `isTrusted(Pfizer, DHL)` nato vrne `true`.

Na ta način sistem poveže semantične podatke, subjektivna pravila in trajne zapise rezultatov v verigi blokov v enoten potek.

### 3.3.3 Način posodabljanja blockchain registra

Zapis rezultatov v pametno pogodbo se lahko sproži na tri načine:

- **Ročno:** prek CLI ali uporabniškega vmesnika, primerno za prezentacije in testne scenarije.
- **Dogodkovno:** ob spremembi entitete, pravil ali metrik v ontologiji (npr. nova licenca, posodobljen audit).
- **Periodično:** v rednih intervalih (npr. dnevno), pri čemer se zapis izvrši le ob spremembi rezultata glede na prejšnje stanje.

## 3.4 Razvoj pametnih pogodb in orakov

## 3.5 Uporabniški vmesnik in API-ji



## Poglavje 4

### Evalvacija

4.1 Varnostna analiza

4.2 Zmogljivost in razširljivost

4.3 Natančnost ocene zaupanja

4.4 Prednosti in omejitve

4.5 Primerjava z obstoječimi rešitvami

4.6 Pridobljena spoznanja



## Poglavje 5

### Zaključek in nadaljnje delo

#### 5.1 Povzetek prispevkov

#### 5.2 Nadaljnje raziskave



## Dodatek A

### Shema ontologije in SWRL pravila





## **Dodatek B**

### **Izvlečki kode pametnih pogodb**



# Literatura

- [1] S. Holtmanns, Trust modeling and management: From social trust to digital trust, in: Computer Security, Privacy and Politics: Current Issues, Challenges and Solutions, IGI Global, 2008. doi:10.4018/978-1-59904-804-8.ch013.
- [2] T. Grandison, M. Sloman, A survey of trust in internet applications, IEEE Communications Surveys & Tutorials 3 (4) (2000) 2–16.
- [3] S. D. Boon, J. G. Holmes, The dynamics of interpersonal trust: Resolving uncertainty in the face of risk, Cooperation and Prosocial Behavior (1991) 167–182.
- [4] D. Gambetta, Trust: Making and breaking cooperative relations, Basil Blackwell, 1988.
- [5] R. C. Mayer, J. H. Davis, F. D. Schoorman, An integrative model of organizational trust, Academy of Management Review 20 (3) (1995) 709–734.
- [6] D. E. Denning, A new paradigm for trusted systems, Proceedings of the 1993 New Security Paradigms Workshop (1993) 36–41.
- [7] D. H. McKnight, N. L. Chervany, Trust and distrust definitions: One bite at a time, Trust in Cyber-societies 2246 (2001) 27–54.
- [8] S. Karthik, R. Shankar, N. Arunkumar, Ontology-based trust model for pervasive computing environments, Journal of Ambient Intel-

- ligence and Humanized Computing 8 (2017) 557–568. doi:10.1007/s12652-016-0442-y.
- [9] A. H. Wardana, M. Syafrudin, G. Alfian, J.-W. Rhee, Lightweight trust-managing and privacy-preserving framework in iot environment, *Sensors* 24 (6) (2024) 1759. doi:10.3390/s24061759.
- [10] J. D. Nally, *Good manufacturing practices for pharmaceuticals*, CRC Press, 2016.
- [11] R. Cyganiak, D. Wood, M. Lanthaler, Rdf 1.2 concepts and abstract syntax, World Wide Web Consortium (W3C) Recommendation (2024). URL <https://www.w3.org/TR/rdf12-concepts/>
- [12] D. Beckett, B. McBride, Rdf/xml syntax specification (revised), World Wide Web Consortium (W3C) Recommendation (2004). URL <https://www.w3.org/TR/rdf-syntax-grammar/>
- [13] C. Boettiger, *rdflib: A high level wrapper around the redland package for common rdf applications* (2018). doi:10.5281/zenodo.1098478. URL <https://doi.org/10.5281/zenodo.1098478>