

Primjena grafova znanja temeljena na Wikidata repozitoriju

Novak, Vlatko

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:235:810750>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2025-01-30**

Repository / Repozitorij:

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Vlatko Novak

Zagreb, 2020. godina.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Mentori:

doc. dr. sc. Tomislav Stipančić, dipl. ing.

Student:

Vlatko Novak

Zagreb, 2020.godina.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem se svim kolegama i kolegicama na druženju i zajedničkom učenju tijekom prošlih 7 semestara, ekipa je bila odlična. Nadalje hvala na velikom razumijevanju prijateljima iz kvarta, osnovne i srednje škole koje sam puno puta odbio da bih ostao učiti. Veliko hvala mojoj obitelji bez koje bi ovo bilo puno teže ostvarivo, dali su mi vremena i mogućnosti da više-manje bezbrižno studiram. Hvala mentoru doc.dr.sc. Tomislavu Stipančiću na zadanom završnom radu i svojoj pomoći na putu do izrade ovoga rada. Hvala svim profesorima, profesoricama, asistentima i demonstratorima na prenesenom znanju. Kaže se da se najviše napreduje kad je najteže, a u prošle 3 i pol godine na ovom fakultetu znalo je biti jako zahtjevno. U tim trenucima nije lako vidjeti svrhu, ali sada sam na tome vrlo zahvalan jer sam napredovao kao čovjek i inženjer.

Vlatko Novak



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za završne ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala i mehatronika i robotika

Sveučilište u Zagrebu	
Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur.broj:	

ZAVRŠNI ZADATAK

Student: **Vlatko Novak**

Mat. br.: 0035210206

Naslov rada na hrvatskom jeziku: **Primjena grafova znanja temeljena na Wikidata repozitoriju**

Naslov rada na engleskom jeziku: **Application of knowledge graphs based on Wikidata repository**

Opis zadatka:

Grafovi znanja postaju neizostavne komponente sustava strojne inteligencije budući da se njima može opisati domena djelovanja razvijenog sustava. Često se izvode koristeći metodologiju semantičkog weba, uključujući formalno definiranu bazu znanja u vidu ontologije te posebno strukturirane računalne upite definirane nekim od upitnih jezika (npr. SPARQL). Grafovi znanja se kreiraju u stvarnom vremenu u trenutku kada do baze znanja (ontologije) stigne pojedini upit te stoga predstavljaju trenutno znanje koje tehnički sustav može iskoristiti prilikom zaključivanja.

U radu je potrebno:

- objasniti metodologiju grafova znanja u kontekstu semantičkog weba,
- odrediti tehnologiju za ostvarivanje grafova znanja temeljenu na Wikidata repozitoriju znanja,
- razviti modularnu aplikaciju koja koristi Wikidata repozitorij da bi prikupila ključne riječi koje karakteriziraju osnovne ljudske emocije,
- eksperimentalno evaluirati i objasniti rezultate.

Programska aplikacija mora biti izvedena koristeći programski jezik Python te metodologiju semantičkog weba.

Rad predati u pisanom i elektroničkom obliku uz prikladne upute za korištenje razvijene aplikacije.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:
28. studenog 2019.

Datum predaje rada:
1. rok: 21. veljače 2020.
2. rok (izvanredni): 1. srpnja 2020.
3. rok: 17. rujna 2020.

Predviđeni datumi obrane:
1. rok: 24.2. – 28.2.2020.
2. rok (izvanredni): 3.7.2020.
3. rok: 21.9. - 25.9.2020.

Zadatak zadao:

Doc. dr. sc. Tomislav Stipančić

Predsjednik Povjerenstva:

Prof. dr. sc. Branko Bauer

SADRŽAJ

SADRŽAJ	I
POPIS SLIKA	III
POPIS TABLICA.....	IV
SAŽETAK.....	V
SUMMARY	VI
1. UVOD.....	1
2. METODOLOGIJA GRAFOVA ZNANJA U KONTEKSTU SEMANTIČKOG WEBA . 2	
2.1. Semantički web	2
2.2. Grafovi znanja.....	3
3. TEHNOLOGIJA ZA OSTVARIVANJE GRAFOVA ZNANJA TEMELJENA NA WIKIDATA REPOZITORIJU ZNANJA.....	7
3.1. URI.....	7
3.1.1. IRI	9
3.2. XML.....	11
3.3. RDF.....	13
3.3.1. RDF, XML analogija	15
3.4. RDFS.....	16
3.5. OWL.....	17
3.5.1. Modeliranje znanja, osnovni pojmovi.....	17
3.5.1.1. Primjeri klase i hijerarhije.....	18
3.6. SPARQL	19
3.7. WIKIDATA	20
3.7.1. Kako radi Wikidata	21
3.7.2. Postavljanje podataka na Wikidata repozitorij znanja	21
4. MODULARNA APLIKACIJA KOJA KORISTI WIKIDATA REPOZITORIJ ZA PRETRAŽIVANJE O LJUDSKIM EMOCIJAMA	24
4.1. Komentar rezultata.....	28

5. KREIRANJE VLASTITE ONTOLOGIJE I PRETRAŽIVANJE LOKALNOG SPARQL SERVERA	29
5.1. Modeliranje podataka u programu Protege	29
5.2. Pokretanje i učitavanje podataka na Apache Jena Fuseki server	33
5.3. Pretraživanje kreirane ontologije na Apache Jena Fuseki serveru.....	36
6. ZAKLJUČAK	38
LITERATURA.....	39
PRILOZI.....	41

POPIS SLIKA

Slika 2.1	Primjer međusobnog povezivanja entiteta pomoću grafova znanja	4
Slika 2.2	Usporedba grafova znanja i RDBMS	5
Slika 2.3	Vizija semantičkog weba	6
Slika 3.1	Primjer trojke i njihovih URIja	7
Slika 3.2	URI RDF i HTML	8
Slika 3.3	URI, URL URN	8
Slika 3.4	Udio standarda za kodiranje na web stranicama	10
Slika 3.5	Razlika ASCII (URI) i IRI znakova	11
Slika 3.6	Razlike HTML i XML	12
Slika 3.7	Prikaz XML koda u strukturi drva	13
Slika 3.8	Primjer RDF trojke	14
Slika 3.9	RDF graf	15
Slika 3.10	RDF-RDFS	16
Slika 3.11	Wikidata	20
Slika 3.12	Primjer San Francisco na Wikidata repozitoriju	21
Slika 3.13	Početak stvaranja entiteta na Wikidata.....	22
Slika 3.14	Osnovni podaci koji se moraju unijeti na Wikidata	22
Slika 3.15	Početni izgled stranice napravljenog entiteta na Wikidata.....	23
Slika 3.16	Završni izgled entiteta Vlatko Novak na Wikidata	23
Slika 5.1	Primjer graf.....	30
Slika 5.2	IRI ontologije	30
Slika 5.3	Kreiranje entiteta	31
Slika 5.4	Klase i podklase.....	31
Slika 5.5	Entitet happiness.....	32
Slika 5.6	Mogućnosti u Protege.....	33
Slika 5.7	Command prompt naredbe za pokretanje Fuseki servera	33
Slika 5.8	Fuseki lokalni server	34
Slika 5.9	Kreiranje nove baze podataka	34
Slika 5.10	Ubacivanje OWL datoteke napravljene u Protege	35
Slika 5.11	Dodavanje prefixa	35

POPIS TABLICA

Tablica 4.1 Rezultat SPARQL upita o emocijama.....	25
Tablica 5.1 Rezultat SPARQL upita na Fuseki serveru	36

SAŽETAK

Grafovi znanja postaju neizostavne komponente sustava strojne inteligencije, a njihovu važnost i potencijal u budućnosti najbolje se očituju činjenicom da Microsoft, Google, Facebook i ostali tehnološki giganti ulažu sve više novca u njihovo korištenje i razvoj. Grafovi znanja predstavljaju skup međusobno povezanih objekata i događaja iz stvarnog svijeta ili apstraktnih koncepata uz formalnu strukturu koja omogućava i ljudima i strojevima razumijevanje. Grafovi znanja glavni su alat semantičkog weba koji je nastavak na World Wide Web. Na semantičkom webu podaci su međusobno povezani tako da su strojno razumljivi. Za ostvarivanje grafova znanja i semantičkog weba potrebne su tehnologije: URI, XML, RDF, RDFS, OWL i SPARQL koje daju potrebne opise entiteta i znanje kako bi računalo razumjelo podatke što bliže kao ljudi.

Kako su podaci na Wikipediji bili razumljivi samo za ljude, tako je na principu grafova znanja nastala Wikidata, baza znanja namijenjena da informacije koje sadrži budu čitljive i ljudima i strojevima. To omogućava da se pomoću strukturiranih računalnih upita definiranim nekim od upitnih jezika (npr. SPARQL) kreiraju odgovori (grafovi znanja) u stvarnom vremenu u trenutku kada do baze znanja stigne pojedini upit te stoga predstavljaju trenutno znanje koje tehnički sustav može iskoristiti prilikom zaključivanja.

Ključne riječi: grafovi znanja, semantički web, strojna inteligencija, wikidata

SUMMARY

Knowledge graphs are becoming an irreplaceable components of machine intelligence. Their importance and potential for the future is the most obvious as Microsoft, Google, Facebook and other tech giants are investing more and more money in their use and development. Knowledge graphs are a sets of interconnected objects and events from real world or abstract concepts with a formal structure that allows both humans and machines to understand. Knowledge graphs are the main tool of the semantic web, which is an extension of the World Wide Web. On the semantic web, data is interconnected so that is machine-readable. Knowledge graphs and semantic web require tehnologije like: URI, XML, RDF, RDFS, OWL and SPARQL that provide the necessary description of entities and knowledge to help the machine to understand the data as closely as humans.

As Wikipedia data was understandable to humans only, so on the principle of knowledge graphs Wikidata was created, a knowledge base designed to make information it contains legible to both humans and machines. This technology, with the help of structured computer queries defined by some of the query languages (eg SPARQL), allows Wikidata to create real-time answers (graphs of knowledge) at the very time when a query arrives at the knowledge base and therefore represent current knowledge that the technical system can use to conclude.

Key words: knowledge graphs, semantic web, machine intelligence, wikidata

1. UVOD

U današnje vrijeme često se može čuti kako količina podataka eksponencijalno raste iz godine u godinu. Iz tog razloga više nismo zadovoljni s time da te podatke samo negdje pohranimo, nego dolazi do potrebe što efikasnijeg spremanja i ponovnog korištenja tih podataka. Želimo model podataka kojim možemo zrcaliti kompleksnost razumijevanja podataka kao ljudi. Razvojem informacijske i komunikacijske tehnologije (engl. *Information and communications technology*, ICT) i integracijom analitike u računalom podržane procese, industrija se suočava s novim prilikama, ali i izazovima u potrebi da uravnoteži vlastite kompetencije i potrebe tržišta. Porast svijesti o mogućnostima primjene interneta stvari (eng. *internet of things*, IOT) i koncepta Industrije 4.0, kako na globalnoj, tako i na lokalnoj razini, omogućuje međusobno povezivanje sve većeg broja objekata te pristup podacima koji su povezani s tim objektima. Sam pristup velikoj količini podataka ne znači nužno i ostvarivanje koristi od samih podataka, ali otvara niz mogućnosti na koje se ti podaci mogu iskoristiti. Pametne baze znanja se ubrzano razvijaju posljednjih desetak godina, omogućujući korisnicima da izvuku maksimum iz znanja koje imaju. Tu dolazimo do grafova znanja koji postaju neizostavne komponente sustava strojne inteligencije i do semantičkog weba koji je neograničena baza podataka razumljivih ljudima i strojevima.

2. METODOLOGIJA GRAFOVA ZNANJA U KONTEKSTU SEMANTIČKOG WEBA

2.1. Semantički web

Semantički web [1] je web na kojem su podaci međusobno povezani i strojno razumljivi te je nastavak na World Wide Web. Širenje semantičkog weba i tehnologija povezanih s njim omogućuju strojevima bolji rad u proizvodnji sadržaja, upravljanju podacima, učenju, podršci, e-trgovini, znanstvenim istraživanjima, upravljanju znanjem itd. [1]

“Semantički Web je ekstenzija klasičnog weba gdje je informaciji dano dobro definirano značenje, omogućujući računalima i ljudima zajednički rad i suradnju.“

[Berners-Lee *et al*, 2001]

Na tradicionalnom webu informacije su bile tako strukturirane da su ih samo ljudi mogu razumjeti. Uzmimo za primjer jedan HTML kod:

```
<html>
```

```
    <head>
```

```
        <title> Kralj lavova </title>
```

```
    </head>
```

```
    <body>
```

Kralj lavova (eng. The Lion King) je američki animirani film iz 1994. Godine, kojeg je producirao Disneyjev studij. To je 32. Animirani film iz produkcije Disneyjevog studija.

```
    </body>
```

```
</html>
```

Na primjeru ovog koda ljudi lako razumiju što ovaj sadržaj znači, ali računalo samo zna da ima neki sadržaj u naslovu te neki sadržaju u tijelu, ali ga ne razumije i ne može ga povezati s ostalim podacima na internetu.

U pokušaju da strojevi razumiju podatke na webu što bliže kao ljudi dolazimo do semantičkog weba. Naziv je dao Tim Berners-Lee za mrežu podataka (podatke na webu) koje strojevi mogu razumjeti – to jest onaj u kojem je veći dio značenja strojno čitljiv.

Cilj semantičkog weba je da može napredovati, skupljati podatke, mijenjati početno znanje, pomoći ljudima da pronađu što znaju i pomoći ljudima da pronađu odgovore na svoja pitanja. Kako bi u tome uspjeli koristimo dobro poznate tehnologije kao što su RDF (eng. *Resource Description Framework*), OWL (eng. *Web Ontology Language*) itd.

Semantički web funkcionira tako da strukturiramo podatke na način koji će omogućiti strojevima razumijevanje sadržaja, to se radi tako da se dodaju oznake riječima (objektima) koje se neće vidjeti na samoj web stranici ali će omogućiti računalu da razumije podatke. To znači da će se web stranice na semantičkom webu činiti iste kao i na običnom webu, tek ako se pogleda u strukturu će se vidjeti velika razlika, te naravno u načinu na koji računalo razumije podatke. Ovo strukturiranje podataka nije tako strogo strukturirano kao shema (tablične) baze podataka kod kojih se za upite koristio SQL (eng. *Structured Query Language*), dok će se na semantičkom webu koristiti SPARQL koji je detaljnije objašnjen u poglavlju 3.6.

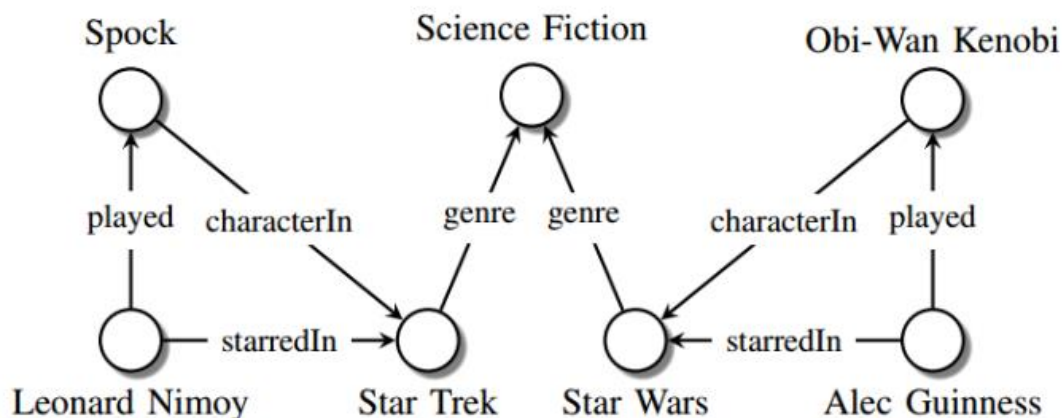
2.2. Grafovi znanja

Količina podataka eksponencijalno raste iz godine u godinu, ali danas više nije bitno samo upravljati s velikom količinom podataka, nego je kritična potreba za povezivanje i boljim shvaćanjem tim podataka. Kako je rekao Seth Godin, poduzeća bi trebala prestati samo prikupljati podatke te ih početi povezivati. Drugim riječima da bi dobili svu snagu iz povezanih podataka, poduzeća trebaju tehnologiju baze podataka koja pohranjuje informacije o odnosima između podataka. [2]

Većina weba je prilagođena za korištenje ljudima. Primjena weba je usmjerena ka pretrazi informacija, publikaciji i korištenju informacija, pretrazi stvarnih pojmova koji su razumljivi ljudima, a ne računalima. Tu dolazimo do semantičkog weba koji može napredovati,

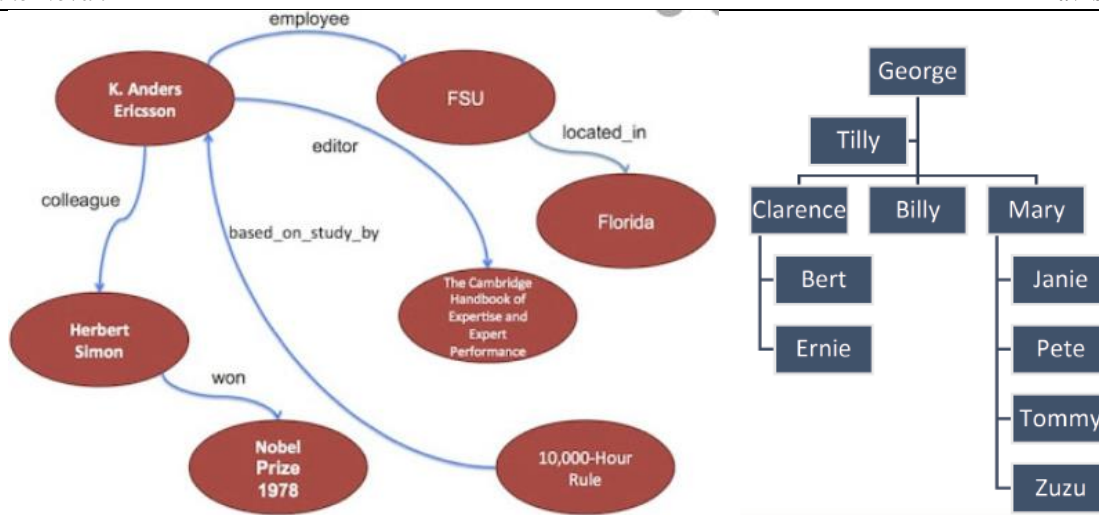
sakupljati podatke te mijenjati početno znanje i pomoći ljudima da pronađu odgovore na svoja pitanja. Ta tehnologija se naziva Grafovi znanja.

Grafovi znanja- glavni alat semantičkog weba, za sada nema jedne egzaktne definicije, pa možemo reći da su to baze znanja koje koriste strukturu grafova a koriste se od strane Googlea, ali i sličnih servisa poput Microsoft→Satori, Facebook→Grafovi entiteta itd. Grafovi znanja prezentiraju skup međusobno povezanih i opisanih entiteta, cilj je olakšati i poboljšati rezultate pretraživanja na webu. Oni omogućuju da dobijemo što bolje rezultate na naš upit (eng.Query) te je ovo bitan korak prema izgradnji nove generacije weba, gdje bi inteligencija na webu trebala razumjeti informacije kao što ih ljudi razumiju. Grafovi znanja uglavnom opisuju entitete iz stvarnog svijeta te definiraju odnose između njih i omogućuju potencijalno međusobno povezivanje entiteta kao što je prikazano na slici 2.1. [3]



Slika 2.1 Primjer međusobnog povezivanja entiteta pomoću grafova znanja

Do sada se koristio za pohranjivanje podataka uglavnom koristio RDBMS (eng. *Relational DataBase Management Systems*) kod kojeg su se teško mogli definirati odnosi između podataka (slika 2.2 desno)[4]. Zato su grafovi znanja budućnost. Ne samo da ove baze podataka efikasno pohranjuju informacije o odnosu između podataka, nego su fleksibilni u pogledu dodavanja novih odnosa između podataka (slika 2.2 lijevo) ili prilagođavanja modela podataka prema novim poslovnim zahtjevima.

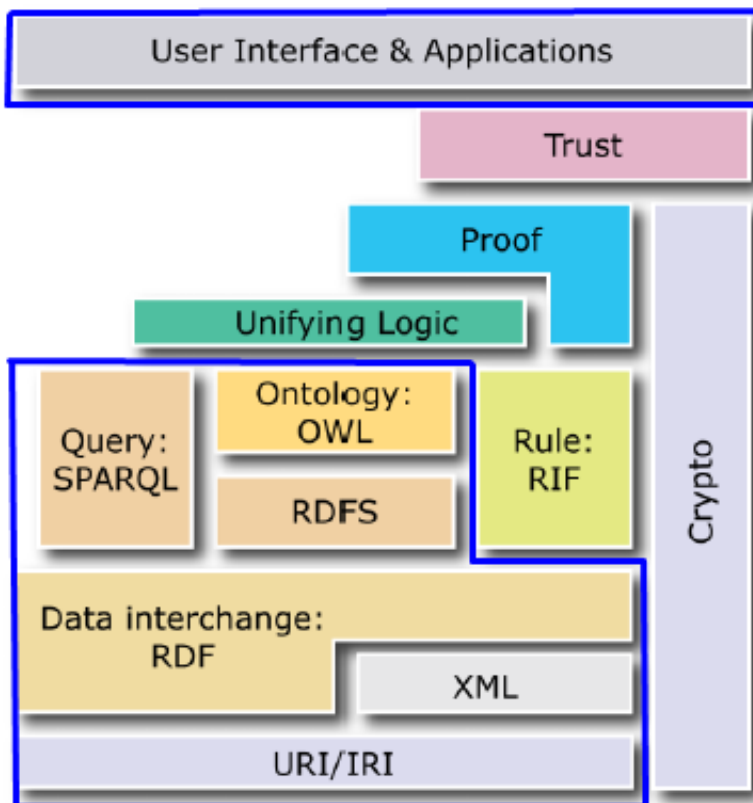


Slika 2.2 Usporedba grafovi znanja i RDBMS

Neke od primjena grafova znanja su: online detekcija prijevara, generiranje preporuka u stvarnom vremenu, upravljanje identitetom i pristupom, upravljanje velikom količinom podataka...

„What is the best database in the world? The human brain. The human brain is ultimately structured as a graph. Neurons connected to other neurons.“ – Emil Eifrem, CEO of Neo4j

Na slici 2.3 [5] su prikazani slojevi tehnologija potrebnih za ostvarivanje semantičkog weba. Označene tehnologije omogućuju ljudima da kreiraju i spremaju podatke na web, kreiraju vokabulare i pišu pravila za rukovanje podacima. U nastavku su tehnologije detaljnije objašnjene jer su potrebne za ostvarivanje grafova znanja na Wikidata repozitoriju.

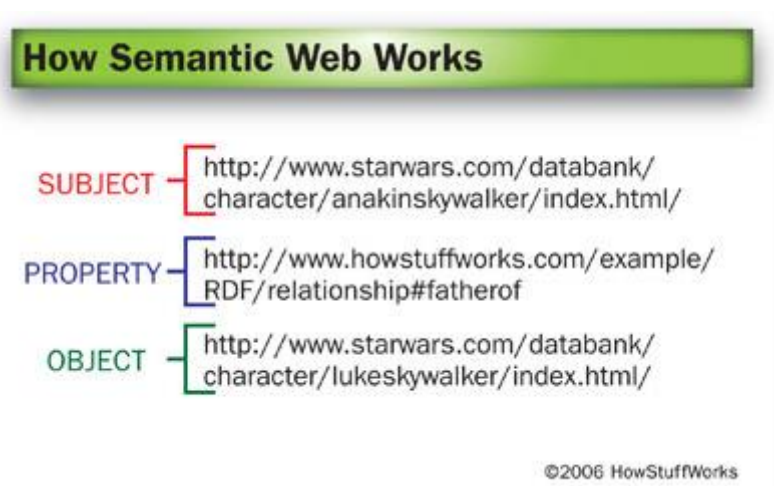


Slika 2.3 Vizija semantičkog weba

3. TEHNOLOGIJA ZA OSTVARIVANJE GRAFOVA ZNANJA TEMELJENA NA WIKIDATA REPOZITORIJU ZNANJA

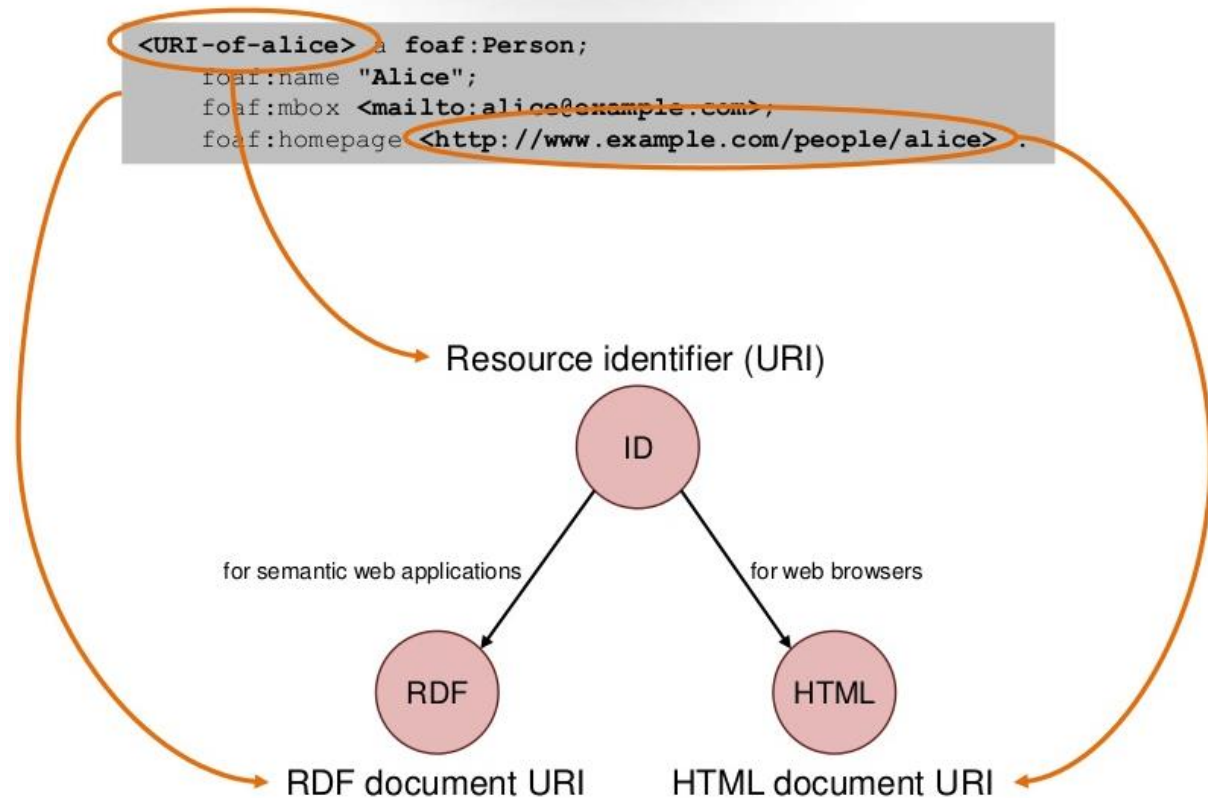
3.1. URI

URI (eng. *Uniform Resource Identifier*) [6] – je kompaktan niz znakova koji identificiraju neki apstraktni entitet ili entitet iz stvarnog svijeta. Svaka stranica na webu ima svoj jedinstveni URI. Na slici 3.1 [7] vidimo kako svaki dio trojke ima svoj URI. Identifikator pruža informacije potrebne da razlikovanje onoga što je identificirano od svega ostalog. Jedan resurs se razlikuje od svih drugih resursa. Sintaksa korištena kod URIja određena je pomoću globalne transkripte, tako da niz dopuštenih znakova je limitiran: osnovna slova Latinske abecede, brojevi i nekolicina specijalnih znakova. Taj limit donosi neke prednosti: URI će moći upisati ljudi sa svih tipkovnica i ljudi različitih jezika. URI često treba biti zapamćen, pa je ljudima jednostavnije zapamtiti već poznate znakove. Interpretacija URIja ovisi samo o korištenim znakovima a ne kako su ti znakovi prezentirani na mrežnom protokolu.



Slika 3.1 Primjer trojke i njihovih URIja

Bitno je naglasiti da postoje razlike između identifikatora za web dokumente i identifikatora za neke druge resurse. URI identificira samo jedno, nikad jedan URI ne može biti i za web dokument i za entitet iz stvarnog svijeta. [8] i [9]



Slika 3.2 URI RDF i HTML

Na semantičkom webu URI služi za dodjeljivanje nedvosmislenih imena entitetima. Ovaj korak je vrlo bitan za strojeve kako bi mogli jasno razlikovati entitete. URI može biti klasificiran kao lokator, ime ili oboje. Svaki zaseban URI ne mora biti klasificiran kao samo „ime“ ili samo „lokator“, nego ovisno o onome tko izrađuje URI, može biti i jedno i drugo odjednom. [10]



Slika 3.3 URI, URL, URN

Porastom broja podataka izraženih preko RDFa znatno se povećava i broj URIja korištenih za identificiranje entiteta. Do problema dolazi kada isti entitet ima više URIja kojima je opisan, do toga dolazi tako što različiti ljudi opisuju isti entitet. To je jedan od nedostataka otvorenog

semantičkog weba. Jedan od načina kako tome doskočiti je korištenje OWL relacije „istiKao“ (eng. *sameAs*) između entiteta koji se smatraju istima.

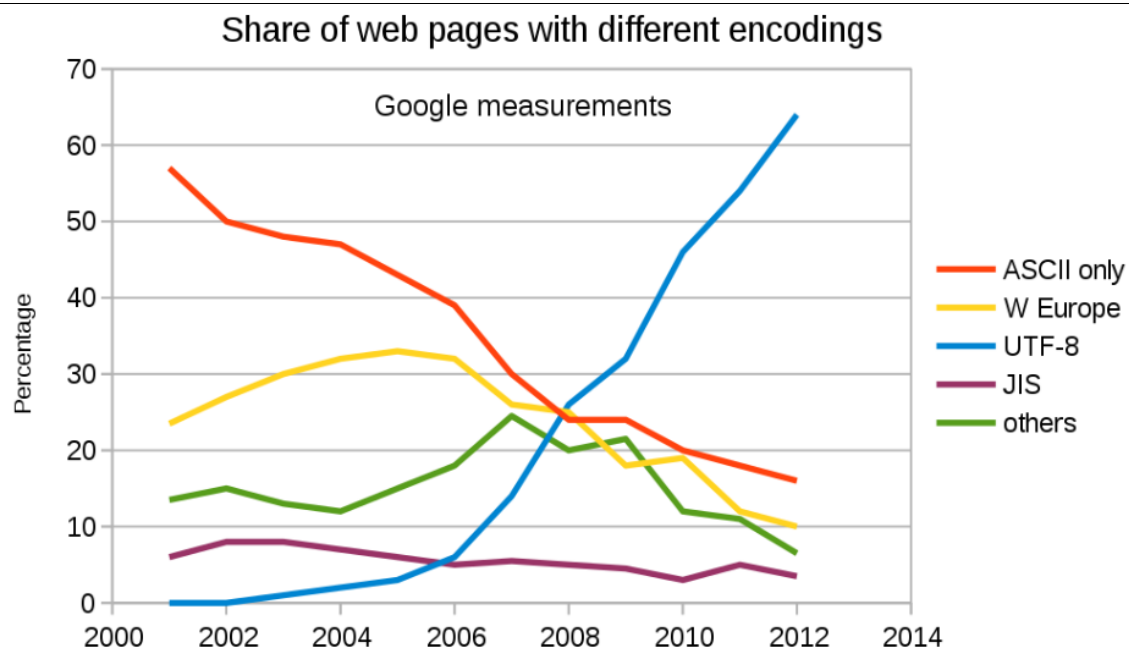
3.1.1. IRI

IRI (eng. *Internationalized Resource Identifier*) – dok URI podržava samo ASCII kod, IRI je kao produžetak na URI koji značajno povećava broj mogućih simbola. Prednost dolazi do izražaja jer je moguće pisati simbole iz drugih jezika poput Kineskog, Japanskog, Korejskog itd. Moglo bi se postaviti pitanje zašto onda jednostavno ne postoji samo IRI da samo njega koristimo? Stvar je u tome što korisnici weba često ne obraćaju pažnju na simbole (znakove) koji su slični. Primjer je što bi korisnici mogli pomisliti da su ove dvije web adrese iste, ali nisu:

<http://www.päypal.com> <http://www.paypal.com>

Netko može napraviti prvu stranicu i staviti na nju virus i korisnik slučajno ode na tu stranicu. Postoji čak i naziv za ovakve slučajeve *Internationalized Domain Name (IDN) homograph attack*. [11]

U praksi IRI najčešće koristi UTF-8 standard simbola za kodiranje. UTF-8 može prikazati podržati mnogo jezika i prilagoditi web stranicu različitim jezicima te se najviše koristi kod web stranica. Slika 3.4 [12] prikazuje najviše korištene. A slika 3.5 prikazuje razlike IRI i URI znakova [13]



Slika 3.4 Udio standarda za kodiranje na web stranicama

ASCII	Character
173	❖
168	❖
142	Ä
143	Å
132	ä
166	❖
	À
	Á
	Ã
133	à
160	á
	ã
134	â
146	Æ
145	æ
128	Ç
135	ç
	È
144	É
	Ê
	Ë
138	è

Slika 3.5 Razlika ASCII (URI) i IRI znakova

3.2. XML

XML (eng. *Extensible Markup Language*) – jezik koji se koristi za opisivanje podataka. XML standard je fleksibilan način za kreiranje informacijskih formata i elektroničko dijeljenje strukturiranih podataka putem interneta. XML kod je sličan HTML-u jer oboje koriste simbole za označavanje (eng. *markup symbols*) kako bi opisali što sadrži neka web stranica ili neka datoteka. Osnovni gradivni blok XML dokumenta je element koje je definiran pomoću tagova. Svaki tag ima svoj početak i kraj. XML oznake (eng. *tagovi*) definirani su od strane korisnika i specifične su u odnosu na domenu. Moć XML-a je u njegovoj jednostavnosti.

Razlika između XML i HTML-a je u tome što HTML kod opisuje sadržaj web stranica (uglavnom tekst i grafičke slike) samo u vidu kako ih prikazati na stranici i u kakvom su međusobnom odnosu. Dok XML je poznat kao samo-opisujući ili samo-definirajući, što znači da je struktura podataka ugrađena zajedno s podacima. Posljedica toga je da nema potrebe za unaprijed definiranom strukturom za pohranu podataka, jer XML ih razumije dinamički. Još jedna razlika je što XML nema definirani set oznaka kao HTML, nego korisnici koriste tagove koje sami napišu. [14]

HTML:

```
<H1>Internet and World Wide Web</H1>
  <UL>
    <LI>Code: G52IWW
    <LI>Students: Undergraduate
  </UL>
```

XML:

```
<module>
  <title>Internet and World Wide Web</title>
  <code>G52IWW</code>
  <students>Undergraduate</students>
</module>
```

Slika 3.6 Razlike HTML i XML

XML dokument formira strukturu drva (eng. *tree*). Sljedeći primjer prikazuje kod u XMLu i zatim prikaz toga koda u strukturi drva na slici 3.7

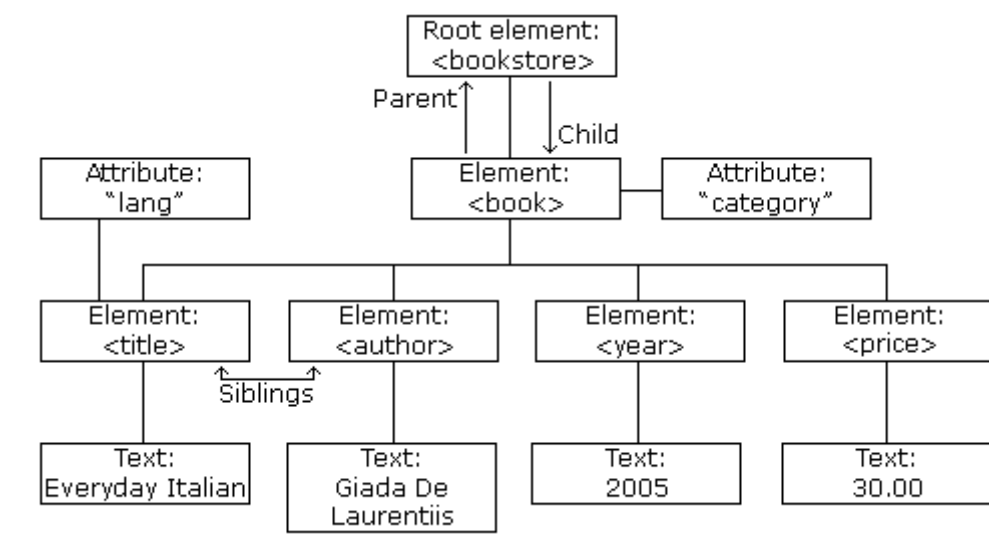
. [15]

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="cooking">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="children">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
```

```

<year>2005</year>
<price>29.99</price>
</book>
<book category="web">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>
</bookstore>

```



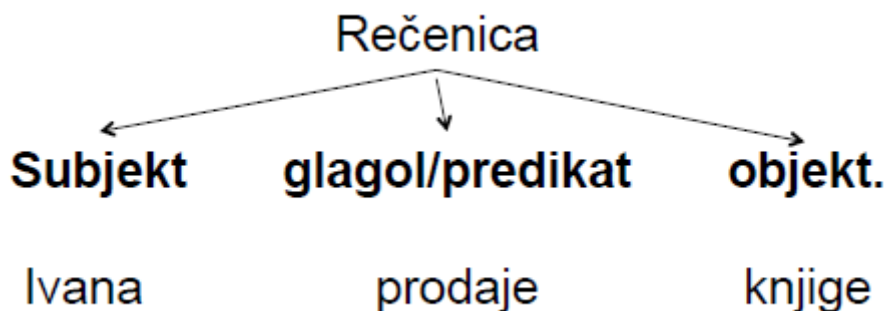
Slika 3.7 Prikaz XML koda u strukturi drva

XML je opisao sadržaj, ali još uvijek taj sadržaj nema značenje, tu dolazimo do RDFa.

3.3. RDF

RDF (eng. *Resource Description Framework*) – standardizirani model (nije jezik) za opisivanje podataka, uključujući i metadata, na internetu npr. opisivanje web stranice i njezinog sadržaja tako da ih strojevi mogu čitati i razumjeti. Svi podaci na semantičkom webu prikazani su pomoću RDFa. Često se kaže da su to „podaci o podacima“ te je bitno napomenuti da se ovi opisi informacije ne prikazuju na webu. Također je bitno da se svi podaci, neovisno o njihovom formatu, mogu prebaciti u RDF model. Ova tehnologija se

sastoji od takozvanih trojki a svaka trojka se sastoji od subjekta predikata i objekta. Svaki dio u trojkama ima svoj posebni identifikator (URI). [16]



Slika 3.8 Primjer RDF trojke

RDF je za razliku od XML-a prikazan kao struktura grafa. Sastoji se od čvorova u kojima je upisan entitet, iako može biti i praznih čvorova, i označenih linija sa strelicama koje povezuju čvorove. Kao što je ranije spomenuto svaki subjekt, relacija i objekt ima svoj URI odnosno IRI. Za prikaz grafa na Slici 3.9 [17] ovo su neki od korištenih URI/IRI:

Čvor s IRIjem: <http://www.w3.org/TR/rdf-syntax-grammar>

Predikat s IRIjem: <http://example.org/terms/editor>

Čvor bez IRIja

Predikat s IRIjem: <http://example.org/terms/homePage>

Čvor s IRIjem: <http://purl.org/net/dajobe/>

Ovo bi bio kod u RDFu za primjer na Slici 3.9

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dc="http://purl.org/dc/elements/1.1/"
        xmlns:ex="http://example.org/stuff/1.0/">

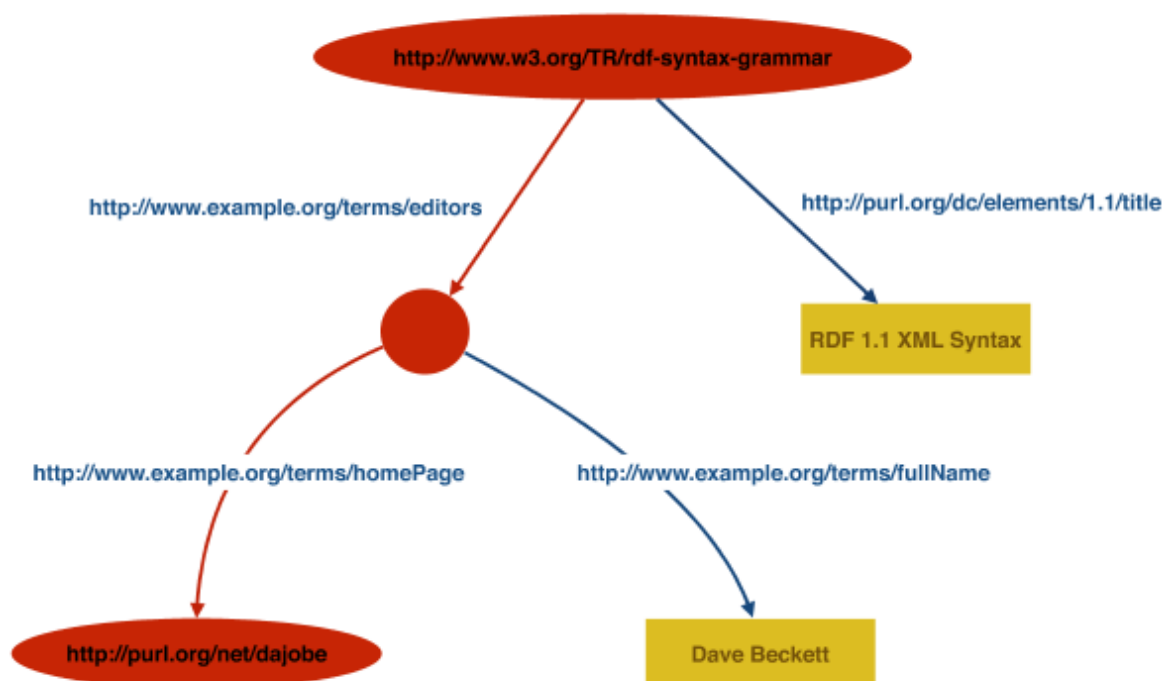
  <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar"

```

```

dc:title="RDF1.1 XML Syntax">
  <ex:editor>
    <rdf:Description ex:fullName="Dave Beckett">
      <ex:homePage rdf:resource="http://purl.org/net/dajobe/" />
    </rdf:Description>
  </ex:editor>
</rdf:Description>
</rdf:RDF>

```



Slika 3.9 RDF graf

3.3.1. RDF, XML analogija

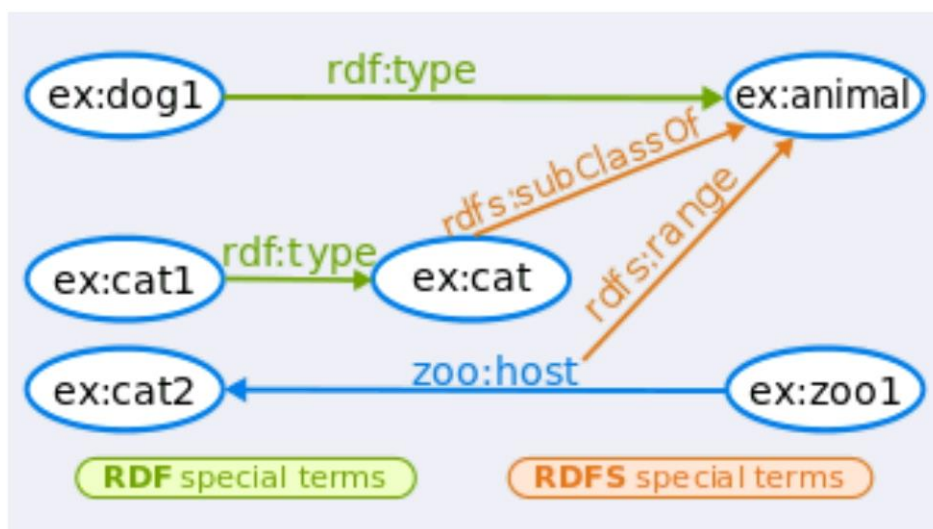
Uzmimo za primjer knjigu „Harry Potter“. Moguće ju je kupiti s mekanim koricama ili tvrdim koricama. Moguće ju je kupiti u kompletu s još dva nastavka ili samo jedan dio. Moguće ju je čitati tako da nam pošalju svaki dan dio knjige na email ili odmah kao cijela pdf datoteka da se čita na iPadu ili Kindleu. Jasno je da svaki od mogućih formata i dalje predstavlja istu knjigu, to što knjiga može biti u papirnatom ili električnom obliku ne mijenja samu knjigu. [18]

Iz pogleda RDFa to je ista knjiga jer ono što je bitno RDFu jest značenje sadržaja knjige a ono se ne mijenja bez obzira na oblik format knjige.

Ali iz pogleda XMLa je drugačije, jer XMLu je bitno na koji način je prezentiran sadržaj.

3.4. RDFS

RDFS (*RDF Shema*) [19] i [20]– semantička ekstenzija na RDF. To je ključna tehnologija (uz OWL) koja nam daje način da napišemo opise RDF podataka. Definira vokabular za RDF i organizira vokabular za pisanu hijerarhiju: *Class, subclassOf, type; Property, subPropertyOf; domain, range*. Za razliku od RDFa koji je baza podataka prikazana u strukturi grafa, RDFS je po svojoj prirodi objektno orijentiran, on opisuje klase objekata. Da ne bude zabune RDFS nije zasebna sintaksa nego dio RDF-a. Kako se RDF i RDFS nadopunjavaju vidimo na sljedećem primjeru, slika 3.10 prikazuje prikaz grafa, a ispod slike je riječima opisan graf.



Slika 3.10 RDF-RDFS

- Dog1 is an animal
- Cat1 is a cat
- Cats are animals
- Zoos host animals
- Zoo1 hosts the Cat2

3.5. OWL

OWL (eng. *Web Ontology Language*) [21]– semantički web jezik napravljen tako da bogato i kompleksno predstavi znanje o objektima, grupama objekata te odnose između njih. Implicitno znanje prebacuje u eksplicitno. To je jezik za izražavanje ontologija. Pojam ontologija ima kompleksnu prošlost kako u računalnim znanostima tako i općenito, ali možemo reći da je ontologija precizan opis nekog entiteta iz svijeta. Cilj je smanjiti mogućnost zabune ljudi i strojeva oko nekog entiteta, zato ga treba što je bolje moguće definirati.

„Ontologija je strojno čitljiv, eksplicitno definiran prikaz neke pojave, fenomena iz stvarnog svijeta.“ – Tom Gruber

OWL se temelji na deskriptivnoj logici za predstavljanje formalizma. Kako je OWL jezik za korištenje na semantičkom webu, imena u OWL su zapravo IRIji, a kako su oni uglavnom dugački koriste se kratice. Podcrtana slova na primjeru ispod su upravo te kratice.

Prefix (xsd :=<http://www.w3.org/2001/XMLSchema#>)

Prefix (owl :=<http://www.w3.org/2002/07/owl#>)

OWL nije programski jezik nego jezik koji tumači stanje stvari na logičan način. Također OWL nije ni shematski jezik kao što je XML, jer OWL ne propisuje način na koji dokument treba biti strukturiran.

3.5.1. Modeliranje znanja, osnovni pojmovi

OWL je jezik za predstavljanje znanja [21], a ovo su osnovni pojmovi koje treba objasniti kako bi se lakše razumio način kako je to znanje predstavljeno:

- Axiomi : osnovne izjave koje izražava OWL ontologija
- Entiteti : element korišten za opis nekog objekta iz stvarnog svijeta
- Izrazi : kombinacija osnovnih entiteta kako bi se formirao kompleksan opis

Iako svrha OWLa je da prikaže znanje, to znanje, naravno, nije odraz svih aspekata znanja koje posjeduje čovjek. OWL možemo smatrati kao snažan jezik za modeliranje određenih dijelova ljudskog znanja.

Kako bi formulirali eksplicitno znanje, korisno je pretpostaviti osnovne elemente a to su izjave. Izjave kao „pada kiša“ ili „svaki čovjek je smrtan“ su tipičan primjer jednostavne izjave. Zapravo svaka OWL ontologija je zapravo skup ovakvih jednostavnih izjava. Ove izjave nazivaju se aksiomi, a ontologija pretpostavlja da su aksiomi istiniti. Općenito, OWL izjave mogu biti istinite ili lažne. To ih razlikuje od entiteta i izraza koji će nadalje biti opisani.

Kad ljudi razmišljaju, oni izvlače zaključke iz svog znanja, mogu razumjeti da neki događaj ima svoju posljedicu. Bitna značajka OWLa je sličnost s tim aspektom ljudske inteligencije. Što to točno znači? U osnovi to znači da neka izjava je istinita kad god je neka druga izjava istinita. Izraženo u OWLu možemo reći da je skup izjava A istinit kad god je skup izjava B istinit.

3.5.1.1. Primjeri klase i hijerarhije

Navedeni su samo neki primjeri OWL sintakse. Za detaljnije pogledati referencu [21].

Sljedeća izjava govori o pojedincu s imenom Mary i kaže da je taj pojedinac osoba. Tehnički rečeno ovako kažemo da Mary pripada klasi osoba (eng. *is an instance of or class of all persons*).

```
ClassAssertion( :Person :Mary )
```

Sljedećim primjerom se pokazuje da to što Mary pripada klasi Person nije ekskluzivna povezanost. Zato što Mary možemo dodati i drugim klasama kao npr.:

```
ClassAssertion( :Women :Mary )
```

Do sada su spomenute dvije klase: klasa svih osoba(Person) i klasa svih žena (Women). Čovjeku je jasno da su te dvije klase u posebnom odnosu: osoba je puno općenitije od žene, drugim riječima ako je neki pojedinac žena odmah podrazumijevamo da je i osoba. Kako bi to OWL znao također zaključiti mora biti informiran o ovoj korespondenciji. U OWLu to se radi takozvanim subclass axiomom:

```
SubClassOf( :Woman :Person )
```

Ako bi još dodali:

```
SubClassOf( :Mother :Woman )
```

To bi značilo da svaki pojedinac koji je klasificiran kao majka ujedno je i žena i osoba.

U ljudskom rječniku postoji mnogo sinonima, a OWL pruža mehanizam kako ih napisati tako da ih i računalo smatra semantički ekvivalentnim:

```
EquivalentClasses( :Person :Human )
```

I naravno postoje slučajevi kada entitet pripada nekoj klasi, a ta pripadnost odmah isključuje mogućnost pripadanja nekoj drugoj klasi, za primjer možemo uzeti klasu muško i klasu žensko. Sintaksa za ovaj primjer glasi:

```
DisjointClasses( :Woman :Man )
```

Do sada je opisano kako opisivati pojedince, ali često ontologija također znači specificirati kako su pojedinci povezani s drugim pojedincima. Jednostavan primjer je obitelj. U ovom primjeru je indicirano da je Mary udana za Johna.

```
ObjectPropertyAssertion( :hasWife :John :Mary )
```

3.6. SPARQL

SPARQL (rekurzivna kratica za eng. *Protokol and RDF Query Language*) – semantički upitni jezik, najbolji način za pronaći željene informacije na Wikidata bazi znanja. Prema [22], „RQL“ dio imena znači RDF Query Language, jezik dizajniran za upite RDF podataka, ali SPARQL nije limitiran samo za upite podataka spremljenih kao RDF model. Komercijalne i *open source* usluge dostupne za upravljanje povezanim podacima kao XML, tablični prikaz podataka i ostali formati mogu se prikazati kao RDF i onda je moguće primijeniti SPARQL na njih. „Protocol“ dio SPARQL imena odnosi se na set pravila kako klijentov program i SPARQL server za procesuiranje izmjenjivati SPARQL upite i rezultate. SPARQL je upitni jezik koji ima mogućnost dohvaćanja i manipuliranja podacima opisanim RDF tehnologijom.

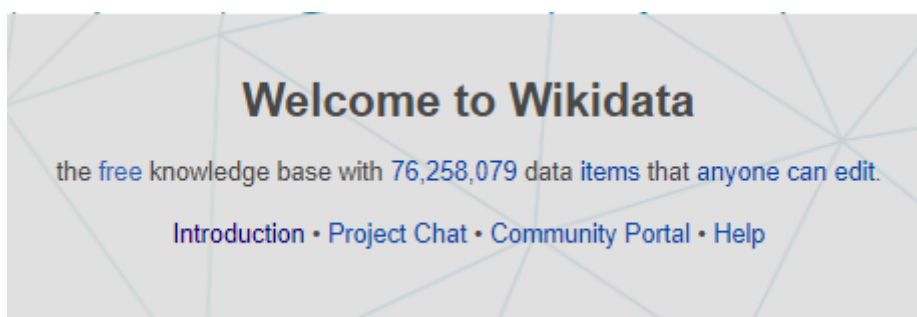
[23] Jedna je od ključnih tehnologija semantičkog weba. Većina SPARQL upita sastoji se od seta trojki što se naziva struktura osnovnog grafa (eng. *basic graph pattern*). Set trojki je kao u RDFu, s tom razlikom da svaki od subjekata, predikata i objekta može biti varijabla. [24] Detaljniji opis sintakse i korištenja SPARQLa objašnjen je u 4. poglavlju.

3.7. WIKIDATA

Sve dosad navedene tehnologije potrebne su kako bi se ostvarila Wikidata.

Wikidata je besplatna i otvorena baza znanja koja može biti čitana ili uređivana od strane i ljudi i strojeva (dok je Wikipediju moguće čitati samo do strane ljudi). Wikidata je baza strukturiranih podataka koji se mogu kopirati, modificirati, distribuirati i izvoziti, čak i u komercijalne svrhe, bez potrebnog odobrenja. Sadržaj na Wikidata može se izvoziti koristeći standardne formate (npr. JSON). Modificiranje, korištenje, pretraživanje i ponovo korištenje podataka je u potpunosti višezlično, u praksi to znači da možemo unijeti podatke na bilo kojem jeziku a oni su odmah dostupni u drugim jezicima. Ako mi modificiramo podatak na jednom jeziku on će biti odmah modificiran na svima (na Wikipediji to nije bio slučaj). Nametanje visokog stupnja strukturiranosti podataka omogućava jednostavno korištenje podataka i računala ih mogu razumjeti. [25] i [26]

Wikidata je od početka zamišljena da se svi ljudi ujedine i postavljaju što više informacija na Wikidata, tako će ta baza znanja rasti. Što više podataka čovječanstvo unese, to ćemo više odgovora moći dobiti na željena pitanja. Wikidata je pokrenuta 2012.godine, do 2018.g u bazi je bilo 44 milijuna informacija a sada Wikidata baza sadrži 76 milijuna informacija. To je pokazatelj kako sve više ljudi koristi mogućnosti koje ova tehnologija pruža.



Slika 3.11 Wikidata

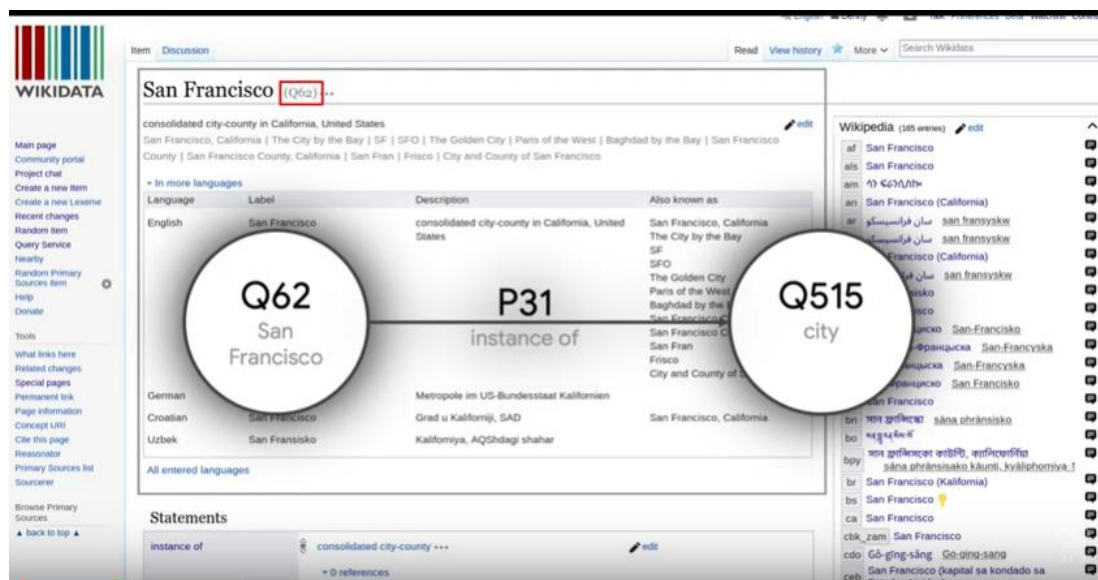
3.7.1. Kako radi Wikidata

Wikidata repozitorij sastoji se od entiteta. Svaki entitet ima svoju oznaku, opis i određeni broj pseudonima.

Entiteti su jedinstveno označeni sa slovom **Q** iza kojeg dolazi redni broj, primjer je **San Francisco** ima oznaku **Q62**.

Izjave (eng. *statements*) opisuju detaljne karakteristike entiteta i sastoje se od svojstva i vrijednosti. Svojstva imaju oznaku **P** iza koje dolazi redni broj npr. **primjerak od** (eng. *instance of*) ima oznaku **P31**.

Slika 3.12 Pokazuje kako je na Wikidata zapisano da je San Francisco grad.

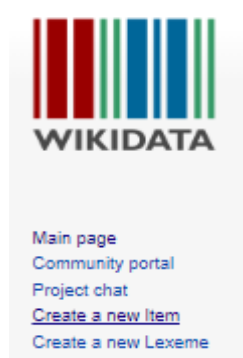


Slika 3.12 Primjer San Francisco na Wikidata repozitoriju

3.7.2. Postavljanje podataka na Wikidata repozitorij znanja

Postavljanje podataka na Wikidata je jednostavan proces. Za primjer napraviti ću sebe i opisati neke osnovne stvari o sebi na Wikidata bazi podataka. Na stranici [26] u lijevom gornjem kutu potrebno je kliknuti „*Create a new Item*“ kao što je pokazano na slici 3.13 Nakon toga nam se otvori prozor kao na slici 2. Gdje upišemo potrebne podatke. Nakon toga pokaže nam se prozor kao na slici 3.14 gdje možemo klikom na „*add statement*“ dodati još podataka o sebi i tako povezivati sebe s ostalim entitetima i relacijama u ovoj bazi podataka. Dodao sam svoje ime, svoj opis, svoj nadimak, stavio da sam čovjek, muško i stavio svoju sliku. Slika

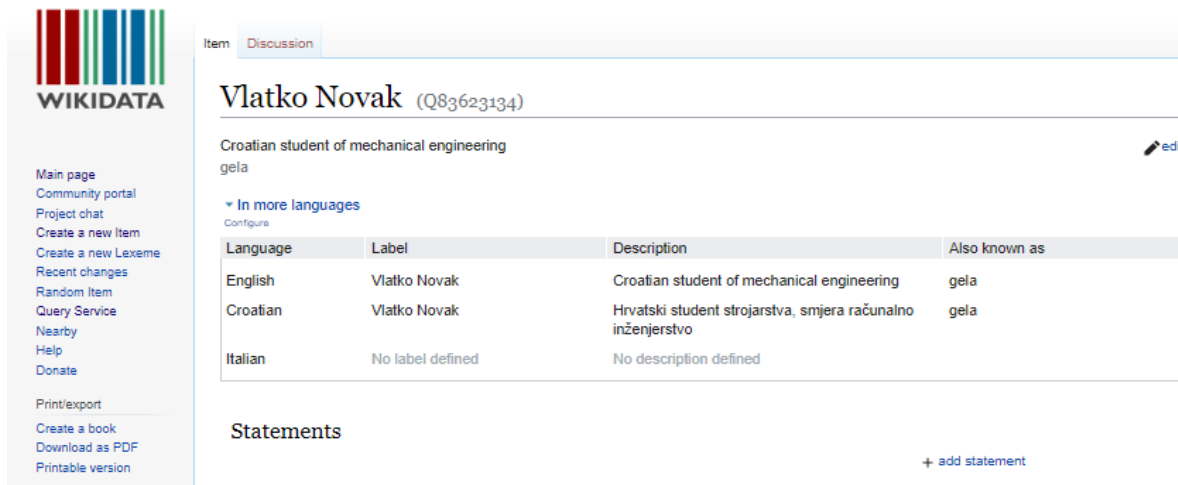
3.15 pokazuje kako izgleda završena stranica o meni na Wikidati. U bilo kojem trenutku bilo tko može izmjenjivati i dodavati podatke. Također, vidljivo je da je oznaka koja mi je pridružena na Wikidati **Q83623134**.



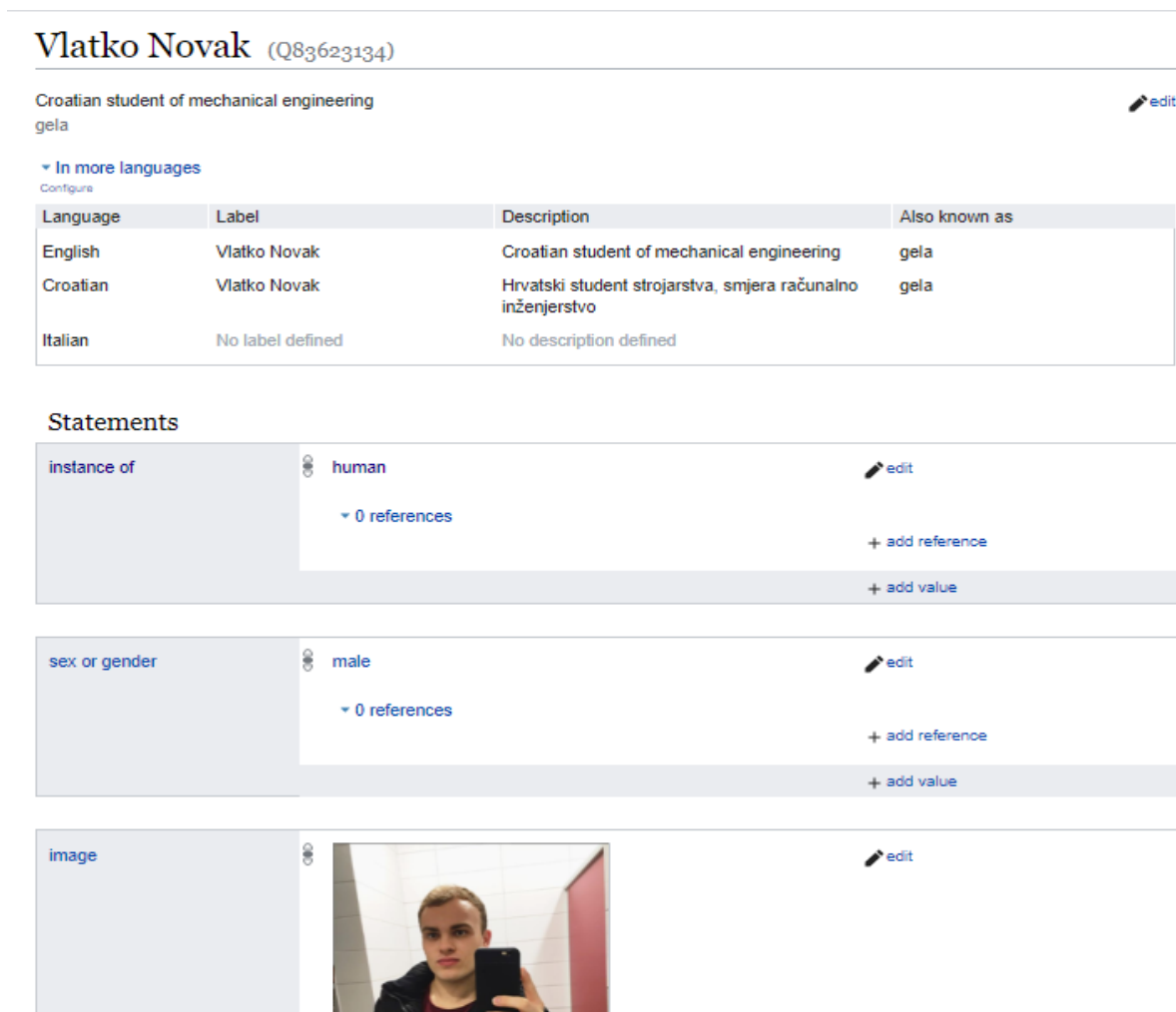
Slika 3.13 Početak stvaranja entiteta na Wikidata

A screenshot of the Wikidata "Create a new Item" form. The page title is "Create a new Item". Below the title, there is a warning message: "Please make sure that the item you want to create complies with our notability policy and that it doesn't already exist. If you want to create an item about a living person, be mindful of their privacy. We appreciate it if you create a label and a description for all of your new items. The first letter of your label should only be capitalized if it is a proper noun (Q147276), and your description should not To create a new lexeme, please use Special:NewLexeme." Below the warning, there is a "Create" button. The form fields are: "Language:" with a dropdown menu showing "hr"; "Label:" with a text input field containing "Vlatko Novak"; "Description:" with a text input field containing "Hrvatski student strojarstva, smjera računalno inženjerstvo"; and "Aliases, pipe-separated:" with a text input field containing "gela". There is a "Create" button at the bottom of the form.

Slika 3.14 Osnovni podaci koji se moraju unjeti na Wikidata



Slika 3.15 Početni izgled stranice napravljenog entiteta na Wikidata



Slika 3.16 Završni izgled entiteta Vlatko Novak na Wikidata

4. MODULARNA APLIKACIJA KOJA KORISTI WIKIDATA REPOZITORIJ ZA PRETRAŽIVANJE O LJUDSKIM EMOCIJAMA

Za izradu modularne aplikacije koristio se program Sublime Text 3. To je funkcionalan i brz uređivač koda s podrškom za mnoge dodatke. Dodatak koji je bio potreban u ovoj aplikaciji je *requests*, a to je zapravo standard koji omogućava slanje HTTP zahtjeva u Pythonu. Cilj je ovdje bio napraviti modularnu aplikaciju koja će pomoću Python i SPARQL koda dati rezultate o emocijama s Wikidata baze znanja.

Detaljni opis koda: u upitu imamo 3 varijable: Emocija, SuprotnoOd i partof. Dio **Label** je dodan kako bi dobili ispis riječima, bez toga bi dobili samo oznaku entiteta na Wikidata kao što je vidljivo u prvom stupcu tablice. U 9. Redu koda zadano je da u varijablu Emocija se ispišu svi entiteti koji imaju svojstvo da su **primjerak od (P31)** entiteta **emocija (Q9415)**. Red 10. ovoga koda traži da nam Wikidata ispiše u varijablu partof sve entitete u varijabli Emocija koji još imaju i svojstvo da su dio nekog drugog entiteta, ispisat će nam ime tog entiteta jer smo u prvoj liniji koda napisali partofLabel. Red 11. ovoga koda traži da nam Wikidata ispiše u varijablu SuprotnoOd sve entitete u varijabli Emocija koji još imaju i svojstvo da su suprotnog značenja od nekog drugog entiteta, ispisat će nam ime tog entiteta jer smo u prvoj liniji koda napisali SuprotnoOdLabel. Da nismo stavili osmu i devetu liniju koda pod **OPTIONAL** u tom slučaju dobili bi samo rezultate emocija koje imaju sve tražene karakteristike, a u ovako napisanom kodu jedino obavezna tražena karakteristika je da pronađeni entitet bude primjerak od entiteta emocija. Kao u mnogih programskim jezicima i ovdje se komentar označava simbolom #. Rezultati na Wikidati ispisuju ovako kao tablica, ali moguće je zapisati i na puno različit načina kao što su mape, razne vrste dijagrama, drvo itd.

```
1 import requests
2
3 url = 'https://query.wikidata.org/sparql'
4 query = """
5 SELECT ?Emocija ?EmocijaLabel ?SuprotnoOdLabel ?partofLabel #?pic
6 WHERE
7 {
8
```

```

9  ?Emocija wdt:P31 wd:Q9415 . #P31=instance of
10 OPTIONAL { ?Emocija wdt:P361 ?partof .} #P361=part of
11 OPTIONAL { ?Emocija wdt:P461 ?SuprotnoOd .} #P461=opposite of
12
13 SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
14 }
15
16 ""
17 r = requests.get(url, params = {'format': 'json', 'query': query})
18 data = r.json()
19 print (data)

```

Tablica 4.1 Rezultat SPARQL upita o emocijama

Emocija	EmocijaLabel	SuprotnoOdLabel	partofLabel
wd:Q2824657	admiration		psychology terminology
wd:Q851918	adoration		psychology terminology
wd:Q1522565	angst		psychology terminology
wd:Q256401	annoyance		psychology terminology
wd:Q255266	anticipation	surprise	theory of emotion
wd:Q255266	anticipation	surprise	primary emotion
wd:Q255266	anticipation	surprise	psychology terminology
wd:Q581459	antipathy	sympathy	psychology terminology
wd:Q154430	anxiety		psychology terminology
wd:Q379784	arousal		psychology terminology
wd:Q196668	artistic inspiration		
wd:Q1369657	broken heart		psychology terminology
wd:Q16301153	chance		
wd:Q1367353	comfort		psychology terminology
wd:Q1412160	compassion		psychology terminology

wd:Q4116378	confidence		psychology terminology
wd:Q366791	curiosity		psychology terminology
wd:Q1182783	defeatism		psychology terminology
wd:Q775842	desire		psychology terminology
wd:Q34302	doubt		theory of emotion
wd:Q34302	doubt		psychology terminology
wd:Q31883801	eco-anxiety		
wd:Q12313331	emotional vulnerability		
wd:Q1096367	emptiness		psychology terminology
wd:Q840407	enthusiasm		psychology terminology
wd:Q2312056	epiphany		psychology terminology
wd:Q24053870	first love		
wd:Q2728730	gratitude	ingratitude	psychology terminology
wd:Q187234	haughtiness	modesty	
wd:Q1320124	infatuation		psychology terminology
wd:Q16523851	ingratitude	gratitude	
wd:Q571571	insult		psychology terminology
wd:Q616314	interest	boredom	psychology terminology
wd:Q29299	irritability		psychology terminology
wd:Q488085	kindness	envy	Seven virtues
wd:Q488085	kindness	envy	theological virtues
wd:Q1210705	limerence		psychology terminology
wd:Q16513670	longing		
wd:Q1823911	Lovestruck		
wd:Q1132131	loyalty	betrayal	psychology terminology
wd:Q26132411	lucidity		
wd:Q2913981	lust for power		
wd:Q1689785	Mudita		psychology terminology

wd:Q83977449	nuclear anxiety		
wd:Q7112875	outrage		psychology terminology
wd:Q335046	passion		psychology terminology
wd:Q484954	pessimism	optimism	theory of emotion
wd:Q484954	pessimism	optimism	psychology terminology
wd:Q2724817	pity		theory of emotion
wd:Q2724817	pity		psychology terminology
wd:Q3071551	pride		psychology terminology
wd:Q3935810	relaxation		psychology terminology
wd:Q1338509	saudade		psychology terminology
wd:Q1892385	Sehnsucht		psychology terminology
wd:Q1422812	Self-pity		psychology terminology
wd:Q1726864	sentimentality	unsentimentality	psychology terminology
wd:Q913668	shyness		psychology terminology
wd:Q2143576	Spite		psychology terminology
wd:Q333173	surprise	anticipation	theory of emotion
wd:Q333173	surprise	anticipation	psychology terminology
wd:Q9503	suspense		psychology terminology
wd:Q516519	suspicion		psychology terminology
wd:Q752118	sympathy	antipathy	psychology terminology
wd:Q19816620	tenderness		
wd:Q2003212	unrequited love		
wd:Q48836991	vicarious embarrassment		
wd:Q20820117	wanderlust	homesickness	

4.1. Komentar rezultata

Rezultati pretragu su nisu konzistentni, vidljivo je da na Wikidata emocije nisu dobro i temeljito kategorizirane. Potrebno je da psiholog ili stručnjak koji se bavi emocijama detaljno razradi ovu temu na Wikidata repozitoriju znanja. Također lako se da zaključiti da manje od 50% emocija ima svoju suprotnost, dok bi ta broja trebala biti oko 100%. Nadalje zadnji stupac tablice rezultata bi trebala svaka emocija imati popunjen, vjerojatno bi svaka emocija trebala biti dio terminologije u psihologiji. Dobra strana ove pretrage je što se vrlo brzo dobije velik broj emocija, pa korisnik može lako krenuti u daljnje istraživanje emocija koje ga zanimaju.

5. KREIRANJE VLASTITE ONTOLOGIJE I PRETRAŽIVANJE LOKALNOG SPARQL SERVERA

Apache Jena Fuseki je besplatan SPARQL server za izgradnju semantičkog weba i povezanih podataka (eng. *Linked Data*). Može biti pokrenut kao servis operativnog sustava, Java web aplikacija (WAR datoteka) i kao samostalni server. Jedna od mnogih mogućih primjena ovog sustava bila bi u tvrtki koja želi pohranjivati podatke pomoću tehnologije grafova znanja. To znači da bi podatke morali modelirati pomoću RDFa i OWLa kako bi onda mogli koristiti SPARQL upite da dođu lakše i brže do boljih rezultata. Što više podataka tvrtka generira to je veća potreba za pametnijim pohranjivanjem tih podataka. Naravno, negativna strana ovoga je što upis podataka u bazu na ovaj način traje duže i zahtjeva edukaciju zaposlenika o ovome području. Za vjerovati je da tijekom vremena kako raste i količina podataka ova investicija bila bi sve isplativija. Često se zna desiti da je tvrtka radila neki projekt koji nije zaživio, to može biti i prije 20-30 godina, a sada bi baš ta tehnologija dobro došla u nekom novom projektu te s ovim načinom pretrage podataka omogućuje se mnogo lakše pronalaženje odgovora. [27]

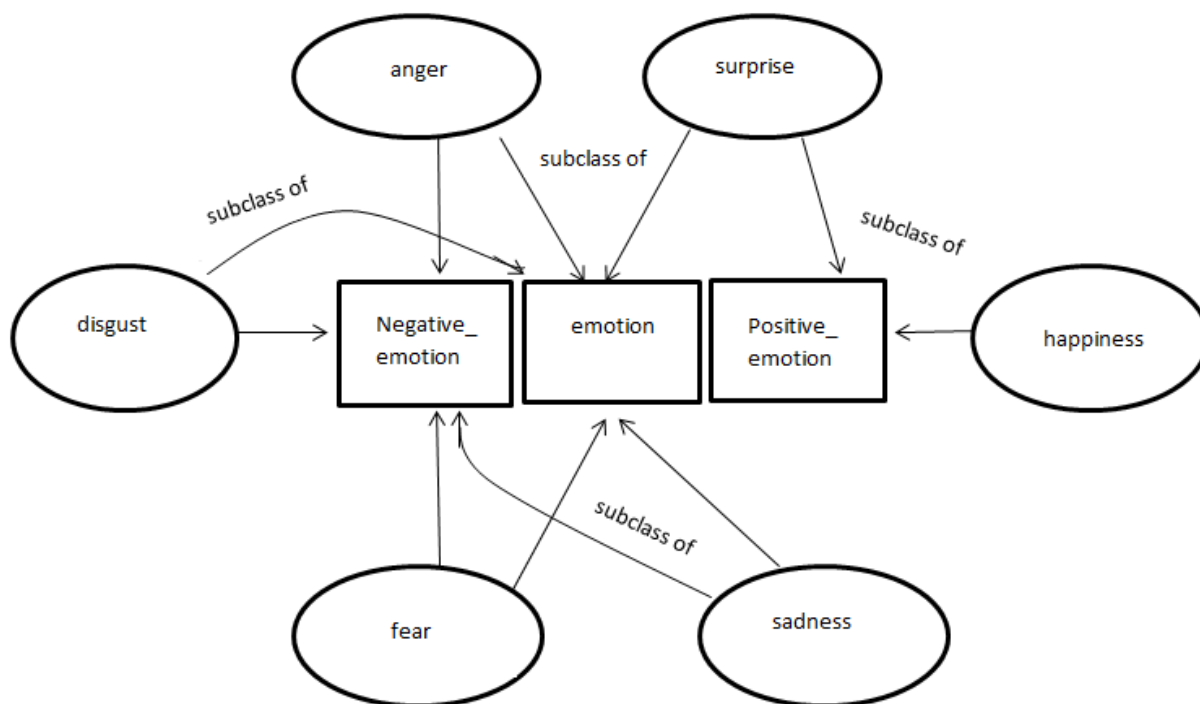
5.1. Modeliranje podataka u programu Protege

Protege je besplatna, otvorena (eng. *open-source*) platforma koja rastućoj zajednici korisnika pruža alate potrebne za izradu modela domena i aplikacija temeljenih na znanju s ontologijama. Sve mogućnosti Protege programa moguće je koristiti tako da se preuzme besplatna aplikacija, a postoji čak i WebProtege gdje je omogućeno korištenje ove tehnologije u internet pretraživaču. [28] Svrha ovog programa je modelirati u RDFu i OWLu podatke koje ćemo učitati (eng. *upload*) na Fuseki server, te ćemo onda moći izvršavati željene upite.

Za primjer sam odlučio pomoću RDFa i OWLa opisati osnovne ljudske emocije. Naravno, ovaj primjer bit će za ljudske emocije, ali na ovaj način se mogu opisivat bilo koji podaci. Osnovnih ljudskih emocija prema [29] ima 6 te će one biti potklase od klase *emotion*. Emocije koje se smatraju pozitivnima (*happiness* i *surprise*) će još biti potklase *positive_emotion*, a negativne (*sadness*, *fear*, *disgust*, *anger*) će biti potklase *negative_emotion*. Sve to je prikazano na slici 5.1 i slici 5.4. Svaka od 6 emocija imat će svoju RDF oznaku i opis, a kao za primjer na slici 5.5 to je prikazano za jedan entitet.

Class: emotion, positive_emotion, negative_emotion

subClassOf: happiness, sadness, fear, disgust, anger, surprise



Slika 5.1 Primjer graf

Prvi korak kod kreiranja naše nove ontologije je dati još njezin zaseban IRI. Ovo je potrebno jer svaki entitet koji izradimo u ovoj ontologiji mora imati svoj IRI, a taj IRI koji će dobiti entitet bit će ovaj početni koji tu napišemo + naziv toga entiteta. Npr. u ontologiji sa slike 5.1 izradi se entitet *emotion*, njegov iri će biti:

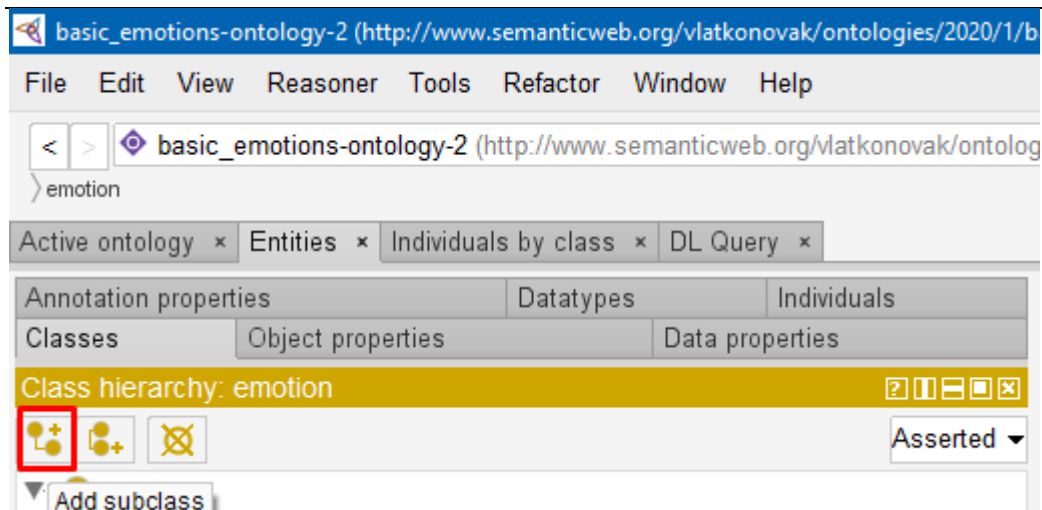
http://www.semanticweb.org/vlatkonovak/ontologies/2020/1/basic_emotions-ontology-2#emotion

Ontology header:

Ontology IRI http://www.semanticweb.org/vlatkonovak/ontologies/2020/1/basic_emotions-ontology-2

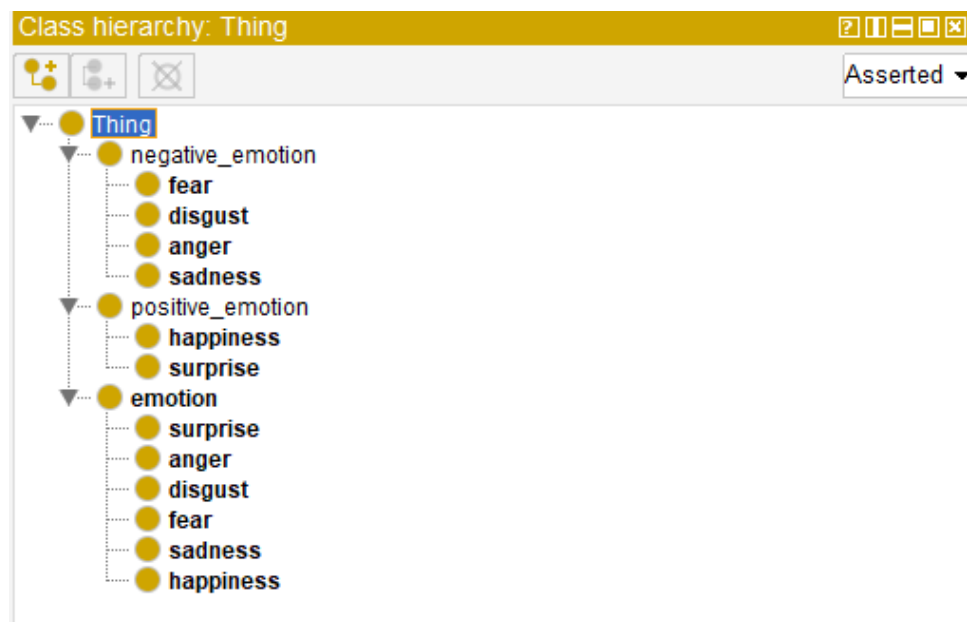
Slika 5.2 IRI ontologije

Kreiranje entiteta je relativno lako. Potrebno je označiti entitet emocija i kliknuti na gumb „Add subclass“ kako bi dodali potklasu.



Slika 5.3 Kreiranje entiteta

Kreirano je što je ranije u tekstu napisano i prikazano na slici 5.1, to je prikazano u programu Protege na slici 5.4.



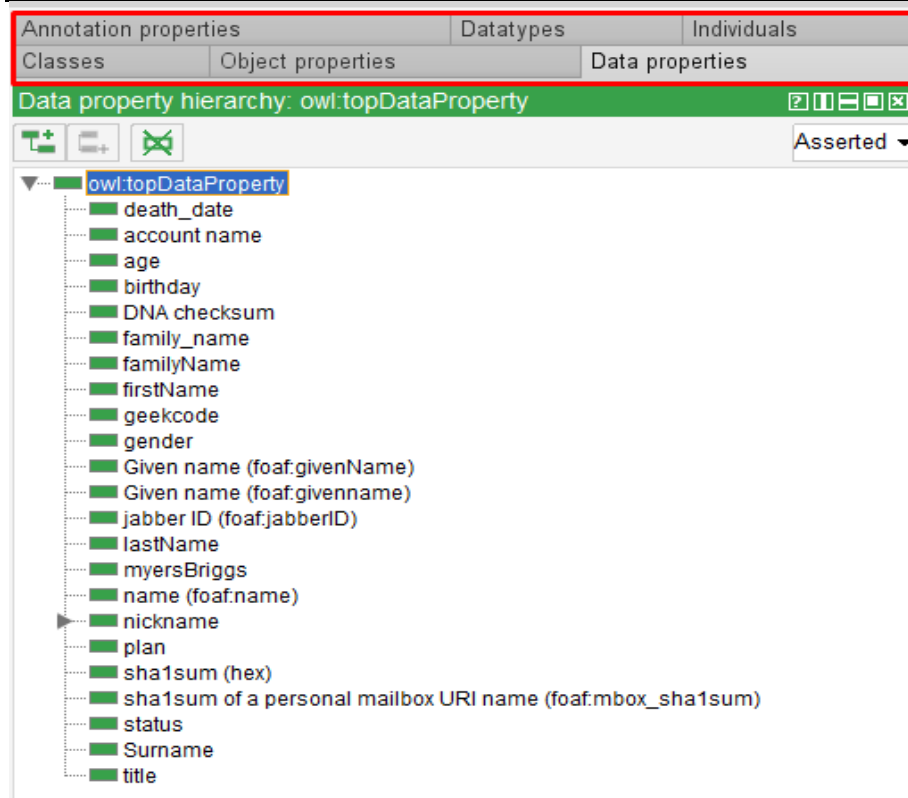
Slika 5.4 Klase i podklase

Dovoljno je kreirati entitet happiness, i onda klikom na taj entitet otvara se ovakav prozor u kojem se tada može definirati što treba, ovdje je definirana RDFS oznaka i opis, te je postavljeno da je ovaj entitet potklasa entiteta *emotion* i *positive_emotion*.



Slika 5.5 Entitet happiness

Protege pruža mnogo više mogućnosti od ovih do sada prikazanih, moguće je pridruživati svojstva objektima, svojstva relacijama i još mnogo toga. Slika 5.6.



Slika 5.6 Mogućnosti u Protege

5.2. Pokretanje i učitavanje podataka na Apache Jena Fuseki server

Pokretanje servera pokreće se iz command prompta. U command promptu potrebno je ući u datoteku u kojoj su raspakirane datoteke skinute sa stranice [30]. Zatim upisati naredbu za podizanje servera kao što je napravljeno na slici 5.6. Vrlo brzo lokalni server se pokrene i na ekranu nam je napisano na kojem portu je podignut.

```

C:\Windows\System32\cmd.exe - fuseki-server --update --mem /ds
Microsoft Windows [Version 10.0.18362.592]
(c) 2019. Microsoft Corporation. Sva prava pridržana.

C:\Windows\System32>cd..

C:\Windows>cd..

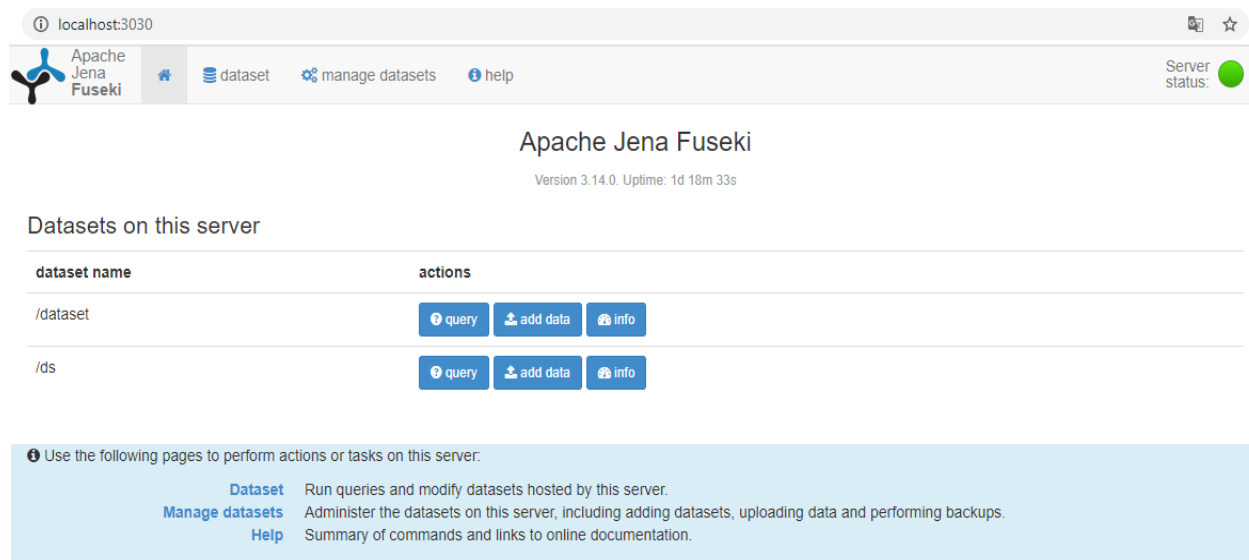
C:\>cd Fuseki

C:\Fuseki>fuseki-server --update --mem /ds
[2020-02-10 11:39:31] Server      INFO  Dataset: in-memory
[2020-02-10 11:39:32] Server      INFO  Apache Jena Fuseki 3.14.0
[2020-02-10 11:39:33] Config     INFO  FUSEKI_HOME=C:\Fuseki\.
[2020-02-10 11:39:33] Config     INFO  FUSEKI_BASE=C:\Fuseki\run
[2020-02-10 11:39:33] Config     INFO  Shiro file: file:///C:\Fuseki\run\shiro.ini
[2020-02-10 11:39:35] Config     INFO  Template file: templates/config-mem
[2020-02-10 11:39:37] Config     INFO  Register: /ds
[2020-02-10 11:39:37] Server      INFO  Started 2020/02/10 11:39:37 CET on port 3030

```

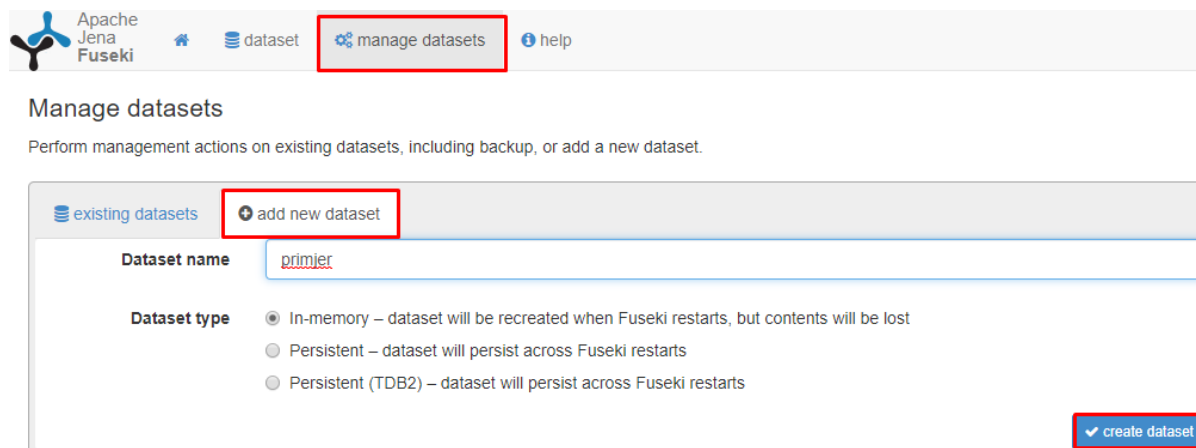
Slika 5.7 Command prompt naredbe za pokretanje Fuseki servera

Sljedeće je potrebno otići u bilo koji internet pretraživač i upisati adresu: <http://localhost:3030/> koja prikaže pokrenuti lokalni server.

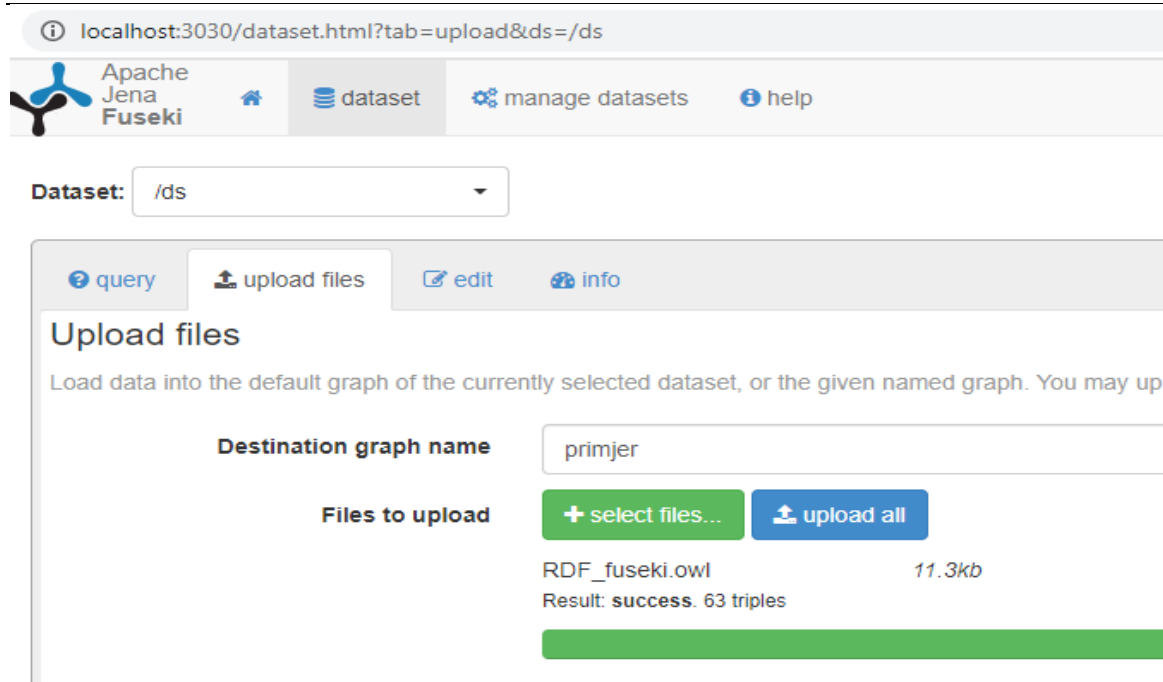


Slika 5.8 Fuseki lokalni server

Server za sada nema nikakvih podataka jer nisu dodane. Sada je potrebno kreirati dataset slika 5.8, u koji je potrebno dodati ontologiju napravljenu u programu Protege slika 5.9.



Slika 5.9 Kreiranje nove baze podataka

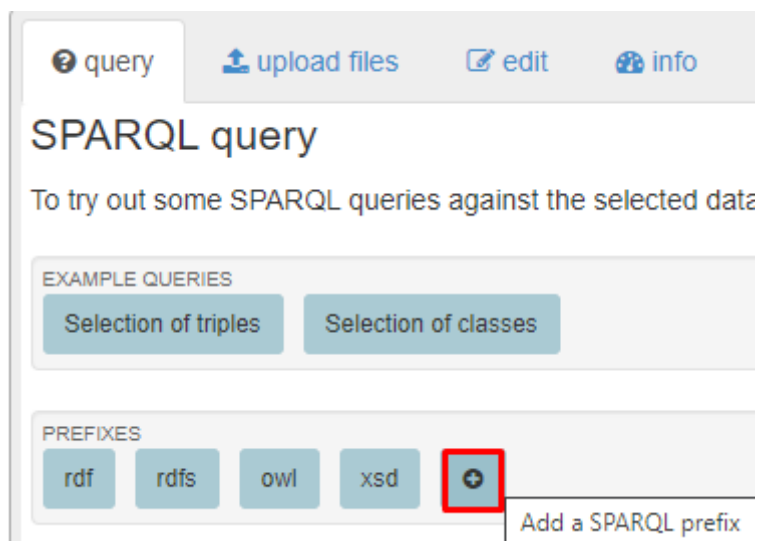


Slika 5.10 Ubacivanje OWL datoteke napravljene u Protege

Nadalje je potrebno dodati prefixe najviše radi preglednosti upita, slika 5.11. Dodan je prefix emo za link na izrađenu ontologiju:

http://www.semanticweb.org/vlatkonovak/ontologies/2020/1/basic_emotions-ontology-2#

Vidljivo je da su neki od unaprijed definiranih prefixa rdf, rdfs, owl, xsd.



Slika 5.11 Dodavanje prefixa

5.3. Pretraživanje kreirane ontologije na Fuseki serveru

Prvo je potrebno definirati prefixe, tj. kratice potrebnih IRIja. U ovome je jedina razlika u odnosu na slanje upita na Wikidata repozitoriju znanja. Tamo su prefiksi već biti upisani u wdt i wd, može se reći tamo je samo uljepšano.

Što se tiče rezultata pretrage, upitom je traženo da se ispišu svi entiteti koji su potklase entiteta *emotion*, i da se ispišu oznaka i opis svakog entiteta koji odgovara prvom uvjetu.

Npr. ako bi se samo promijenilo *emo:emotion* u *emo:postive_emotion* rezultat bi bili samo *happiness* i *surprise*.

```
PREFIX emo: <http://www.semanticweb.org/vlatkonovak/ontologies/2020/1/basic_emotions-ontology-2#>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
SELECT DISTINCT ?entity ?label ?description
```

```
WHERE {
```

```
  ?entity rdfs:subClassOf emo:emotion
```

```
  OPTIONAL { ?entity rdfs:label ?label }
```

```
  OPTIONAL { ?entity rdfs:comment ?description }
```

```
}
```

Tablica 5.1 Rezultat SPARQL upita na Fuseki serveru

entity	label	description
emo:sadness	sadness	Sadness is characterized by feelings od disappointment, grief, hopelessness, disinterest and dampened mood.
emo:surprise	surprise	Surprise is usually quite brief and is characterized by a physiological startle response following something

		unexpected.
emo:disgust	disgust	Disgust is typical reaction to blood, bad smell or unpleasant food.
emo:fear	fear	Fear is characterized by your muscles become tense, your hear rate and respiration increase and your mind becomes more alert.
emo:anger	anger	Anger is characterized by fellings of hostility, agitation, frustration and antagonism towards others.
emo:happiness	happiness	Happiness is often defined as a pleasant emotional state that is characterized by feelings of contentment, joy, gratification, satisfaction and well-being.

6. ZAKLJUČAK

Primjena semantičkog weba i grafova znanja započela je prije otprilike 10 godina, ali tek posljednjih nekoliko godina dolazi do značajnijeg širenja njihove primjene. Svake godine strojno učenje i predstavljanje znanja tako da ih računalo razumije temeljeno na metodologiji grafova znanja značajno napreduje. S jedne strane tehnike strojnog učenja postaju sve bolje u provođenju raznih zadataka, a s druge strane predstavljanje znanja sve više napreduje u pogledu povezivanja entiteta i relacija među njima. Do danas je upotreba grafova znanja zahvatila različite grane djelatnosti poput strojarstva, medicine, obrazovanja, ekonomije, a u bliskoj budućnosti se može očekivati daljnji još intenzivniji rast njihove primjene. Zanimljiv je primjer tvrtka Neo4j, koja se među prvima počela baviti grafovima znanja, da je u samo nekoliko godina dosegla bilijunsku vrijednost. Klijenti su im najveće svjetske tvrtke poput Adidasa, Volva, Walmarta, eBay, Microsoft itd. Semantički web sa svojom Wikidata bazom znanja ima velik potencijal izvlačenja odgovora na upite. Moguće je kreirati algoritme koji će slati upite i filtrirati rezultate, a to je moguće integrirati u robota koji tako može dolaziti do informacija i izvršavati željene radnje.

LITERATURA

- [1] what is the semantic web?: <https://simplea.com/>, Pristupljeno 25. siječnja, 2020.
- [2] Jim Webber & Ian Robinson, The top 5 use causes of graph knowledge, Neo4j, 2017.
- [3] Sudip Chowdhury, Knowledge Graph: The Perfect Complement to Machine Learning, 18. srpnja, 2019.
- [4] The Case for Knowledge Graphs: <http://pgroth.com/slides/savesd2015.html>, Pristupljeno: 25. siječnja, 2020.
- [5] Tomislav Stipančić, Semantički web servisi predavanje u sklopu kolegija Računalne Mreže
- [6] Uniform Resource Identifier (URI): Generic Syntax, T. Berners-Lee, R. Fielding, L. Masinter, 2005
- [7] Tracy V. Wilson, How semantic web works, howstuffworks
- [8] 2008., W3C, Danny Ayers, Max Volkel
- [9] URI for the Semantic Web: <https://www.slideshare.net/onlyjiny/the-semantic-web-3-uri>, Pristupljeno: 4. veljače, 2020.
- [10] What is URI?: <https://uri.to/learn.php>, Pristupljeno: 4. veljače, 2020.
- [11] IRI: https://en.wikipedia.org/wiki/Internationalized_Resource_Identifier, Pristupljeno: 4. veljače, 2020
- [12] UTF-8: <https://en.wikipedia.org/wiki/UTF-8>, Pristupljeno: 4. veljače, 2020.
- [13] ASCII: <https://en.wikipedia.org/wiki/ASCII>, Pristupljeno: 8. veljače, 2020.
- [14] Difference between XML and HTML: <https://www.guru99.com/xml-vs-html-difference.html>, Pristupljeno : 5. veljače, 2020.
- [15] XML tree: https://www.w3schools.com/xml/xml_tree.asp, Pristupljeno: 5. veljače, 2020.
- [16] Margaret Rouse, Resource Description Framework (RDF)
- [17] RDF 1.1 XML Syntax, Dave Beckett, 2014.
- [18] cambridgesemantics, RDF vs XML
- [19] RDF Schema: https://en.wikipedia.org/wiki/RDF_Schema, Pristupljeno: 8. veljače, 2020.
- [20] Brian McBride, RDF Schema 1.1, W3C Recommendation, 2014.
- [21] W3C, OWL 2 Web Ontology Language Primer (Second Edition), 2012.
- [22] O'Reilly, Learning SPARQL, Bob DuCharme

-
- [23] wikipedira.org, sparql
- [24] SPARQL 1.1 QUERY LANGUAGE, W3C Recommendation, 2013.
- [25] Wikidata: <https://en.wikipedia.org/wiki/Wikidata>, Pristupljeno: 2. veljače, 2020.
- [26] https://www.wikidata.org/wiki/Wikidata:Main_Page, Pristupljeno: 2. veljače, 2020.
- [27] Apache Jena Fuseki: <https://jena.apache.org/documentation/fuseki2/>, Pristupljeno: 10. veljače, 2020.
- [28] Protege: <https://protege.stanford.edu/products.php#desktop-protege>, Pristupljeno: 10. veljače, 2020.
- [29] The 6 types of Basic Emotions and their Effect on Human Behaviour: <https://www.verywellmind.com/an-overview-of-the-types-of-emotions-4163976>, Pristupljeno: 10. veljače, 2020
- [30] Apache Jena: <https://jena.apache.org/index.html>, Pristupljeno: 10. veljače, 2020

PRILOZI

I. CD-R disc