

Semantički veb – ACM Curriculum & Bibo Ontologies

ANA GRAHOVAC, TARA POGANČEV

Uvod

Projekat predstavlja primenu tehnologija semantičkog veba u sklopu ontologija kreiranih na osnovu **ACM¹ Curriculum**-a i **bibliografskih** podatakaza dokumente.

Semantički veb, poznat i kao **Web 3.0** predstavlja ekstenziju World Wide Web-a kroz seriju standarda kako bi podaci bili mašinski čitljivi nezavisno od njihovog izvora. Kako bismo jasnije razumeli ideju semantičkog veba, potrebno je da pogledamo istoriju podataka i pre nastanka veba. Kao što nam je poznato, u svom izvornom obliku, podaci su čuvani putem dokumenata. Oni jesu bili sintetički uvezani, referencirali su se jedni na druge, ali suštinski kako bismo pogledali podatke referenciranog rada bilo ga je fizički potrebno pronaći, a potom i u okviru istog locirati relevantni deo.

Web 1.0 – inicijalna pojava interneta dovodi nam rešenje za uvezivanje udaljenih dokumenata, iako su oni čuvani na različitim uređajima. Nama danas uveliko poznato spajanje putem URL-a pomaže da se iz jednog dokumenta uredno lociraju i pronađu referentni dokumenti, nezavisno od toga koji fizički uređaj ih poseduje. Ovde i dalje govorimo i malom nivou interabilnosti sa korisnicima Interneta, ali je veliki značaj uvezanosti dokumenata.

Web 2.0 uvodi danas široko rasprostranjene veb aplikacije. Interakcija korisnika sa podacima postaje krucijalna stvar, sada ne samo u pogledu da korisnik istima pristupa, već i da može da ih i uređuje. Facebook, LinkedIn i druge aplikacije komforno su postale deo naše svakodnevnice.

Međutim, uprkos savremenim načinima upravljanja podacima putem aplikacija, oni i dalje nisu povezani. Podaci kao informacije postoje u sklopu aplikacije koja o njima vodi računa, ali nisu dostupni drugim aplikacijama koje bi se ticale istog domena. Na primer, ukoliko promenite vašu poslovnu poziciju na Facebook nalogu, dodatno je potrebno promeniti istu na LinkedIn-u, iako je suštinski u pitanju ista informacija.

Ovo je problem koji semantički veb pokušava da popravi. Fokus se svodi sa aplikacije na perzistirane podatke. Njihov oblik i korelacije detajno su opisani putem **ontologija**.

Motivacija

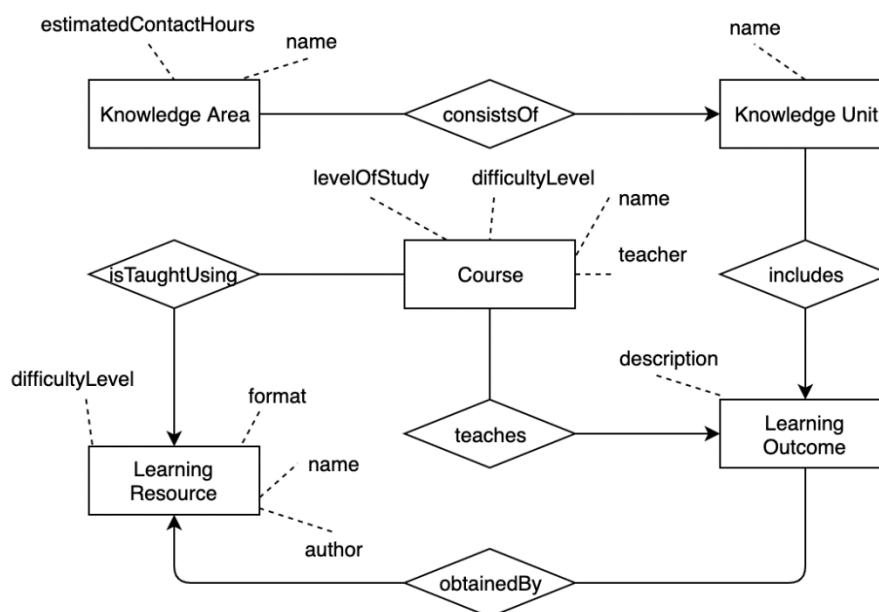
Osnovni cilj projekta jeste da pomoću tehnologija semantičkog veba profesorima olakša rad sa nastavnim materijalima, tako što ih međusobno uvezuje, kao i dodatno daje odgovore na to koji materijali su prikladni za koji kurs ili naučnu oblast. Ovako uvezani podaci bili bi lako iskoristivi na široj skali, te bi mogli biti deljeni od strane više profesora ili naučnih ustanova za veću i kompletniju bazu informacija.

¹ The Association for Computing Machinery (ACM)

Opis ontologija

ACM CURRICULUM ONTOLOGY

ACM Curriculum (u daljem tekstu ACM) ontologija kreirana je od strane Saše Boroša, na osnovu knjige „Computer Science Curricular Guidance for Associate-Degree Transfer Programs“. Sastoji se od entiteta *Knowledge Area*, *Knowledge Unit*, *Course*, *Learning Outcome*, i nama najbitnijeg *Learning Resource*. Relacije navedenih se mogu videti na priloženom grafu.



Ilustracija 1 - Šema ACM Ontologije

Na osnovu podataka iz knjige, ontologija je popunjena oblastima i ishodima znanja, kao i sekcijama učenja. Radi testiranja je dodato nekoliko kurseva i resursa za učenje kako bi bilo moguće testirati SPARQL upite. U okviru projekta naš najveći fokus je bio na manipulisanju resursima za učenje, kako bismo istražili načine na koje oni koreliraju sa drugim entitetima ontologije.

BIBO ONTOLOGY

BIBO ontology, odnosno bibliografska ontologija poprilično je širokog obima, te smo je u naše svrhe koristili samo za citate. Profesor prilikom dodavanja novog nastavnog materijala ima opciju za dodavanje detaljnih citata bilo kog dokumenta ili knjige. Ovo podržava entitet *Document*.

Arhitektura aplikacije

Aplikacija se sastoji od jedne serverske (backend) **Spring Boot** aplikacije pisane u **Javi**. Sva testiranja rađena su putem **Postman**-a, kako su mogućnosti sistema izložene preko kontrolera. Ontologije su definisane u **.owl** fajlovima koji se nalaze u *resources* folderu projekta. Dokument *data.rdf* na istoj lokaciji predstavlja popunjene ontologije. Dokument *sec_ontology_individuals.xlsx* predstavlja Excel bazu popunjenu informacijama o ACM ontologiji, kojima se ontologija popunjava prilikom inicijalnog generisanja *data.rdf* fajla.

Iako aplikacija podržava funkcionalnosti logovanja i registracije, kao i autentikacije putem JWT-a, u trenutnoj implementaciji one nisu iskorišćene, te su ostavljene za buduća proširivanja.

Za rad sa ontologijama uvezena je biblioteka **Jena** – prikladna za kreiranje modela, kao i za pisanje i čitanje rdf ili owl fajlova.

```
<dependency>
  <groupId>org.apache.jena</groupId>
  <artifactId>apache-jena-libs</artifactId>
  <version>3.16.0</version>
  <type>pom</type>
</dependency>
```

ONTOLOGY SERVICE

OntologyService zadužen je za osnovne funkcije nad samim modelima ontologija. Kako su iste definisane u svojim respektivnim *.owl* dokumentima, zadatak funkcija servisa jeste da ih iz istih očita kao kompletne modele ontologija, sa svim osobinama koje bi po opisu trebali imati. Ovi modeli opisani su u zasebnim klasama, *BiboOntologyModel* i *AcmOntologyModel*.

```
public AcmOntologyModel getAcmOntologyModel(OntModel ontModel) throws
IOException;

public BiboOntologyModel getBiboOntologyModel(OntModel ontModel) throws
IOException;
```

DATA SERVICE

Servis zadužen za popunjavanje ontologije jeste *DataService*. Njegove osnovne dve funkcije imaju ulogu dodavanja novih nastavnih materijala, kao i generisanje inicijalnog *.rdf* fajla popunjene ontologije.

```
public String uploadFile(FileUploadDTO dto) throws IOException;
public String generateStarterRdf() throws IOException;
```

Kako bi korisnik dodao novi nastavni materijal, potrebno je da prilikom poziva funkcije kontrolera prosledi JSON objekat koji ga opisuje.

```
{
  "format": "PDF",
  "name": "Introduction to Mathematical Analysis",
  "author": "Tara Pogancev",
  "difficultyLevel": 1,
  "cites": [{
    "content": "Analysis 1",
    "section": 5,
    "number": 3},
    {
    "content": "Algebra Basics",
    "section": 15,
    "number": 2}
  ],
  "obtainedBy": [
    "AR-4-1",
    "AR-4-2"
  ],
  "teachesCourses": [
    "C-1",
    "C-3"
  ]
}
```

SPARQL SERVICE

Servis zadužen za rad sa SPARQL upitima jeste SparqlService. Analogno svakoj funkciji jeste jedan poziv Sparql kontrolera zaduženog za razne upite nad populisanom ontologijom.

Primer funkcije sa Sparql upitom

Sledeća funkcija dobavlja pogodne nastavne resurse za učenje definisane nastavne oblasti (Knowledge unit). Najpre se putem DTO objekta proslede potrebne informacije za upit, što je u ovom slučaju samo potrebna nastavna oblast. Nakon toga se kreira startni model ontologije koji se potom popunjava podacima iz data.rdf fajla.

Nad ovim modelom ontologije se potom izvršava definisani upit, a rezultati istog se korisniku vraćaju kao String vrednost.

```
@Override
public String getResourceForKnowledgeUnit(RequestDTO dto) {
    OntModel om = ontologyService.getStarterModel();
    om.read(DATA_FILE);

    String queryString =
        "PREFIX acm:
<http://www.semanticweb.org/sasaboros/ontologies/2020/11/sec_ontology#>\n" +
        "\n" +
        "SELECT ?name\n" +
        "WHERE {\n" +
        "    ?unit acm:name \"" + dto.knowledgeUnit + "\" .\n"
    +
```

```

        "    ?unit acm:includes ?lo .\n" +
        "    ?lo acm:obtainedBy ?lr . \n" +
        "    ?lr acm:name ?name" +
        "}";

Query query = QueryFactory.create(queryString);
QueryExecution qe = QueryExecutionFactory.create(query, om);
ResultSet results = qe.execSelect();

String retVal = "Resources for course: " + dto.course + "\n\n";
retVal += ResultSetFormatter.asText(results);
qe.close();
return retVal;
}

```

FUNKCIONALNOSTI

U nastavku teksta biće opisani pozivi funkcija kontrolera sa korespondentnim JSON objektima koji se šalju prilikom istih.

GENERISANJE RDF FAJLA

1. Generate starter RDF

<http://localhost:8080/data/generate>

2. Upload File Metadata

<http://localhost:8080/data/upload>

```

{
  "format": "PDF",
  "name": "Introduction to Mathematical Analysis",
  "author": "Tara Pogancev",
  "difficultyLevel": 1,
  "cites": [{
    "content": "Analysis 1",
    "section": 5,
    "number": 3},
    {
    "content": "Algebra Basics",
    "section": 15,
    "number": 2}
  ],
  "obtainedBy": [
    "AR-4-1",
    "AR-4-2"
  ],
  "teachesCourses": [

```

```

        "C-1",
        "C-3"
    ]
}

```

SPARQL UPITI

1. Which resources are the best for learning a specific Course?

<http://localhost:8080/sparql/resources-for-course>

```

{
    "course": "Semantic Web"
}

```

2. Which resources are the best for learning a specific Knowledge Unit?

<http://localhost:8080/sparql/resources-for-knowledge-unit>

```

{
    "knowledgeUnit": "Basic Analysis"
}

```

3. For which courses do we need resources by a specific Author?

<http://localhost:8080/sparql/courses-with-resources-by-author>

```

{
    "author": "Jelena Jelic"
}

```

4. Which courses should i enroll in order to get a specific Learning Outcome?

Which authors are the creators for those courses' Learning Resources?

<http://localhost:8080/sparql/courses-and-authors-for-learning-outcome>

```

{
    "learningOutcome": "Describe security concerns in designing applications
for use over wireless networks"
}

```

5. Which documents are cited by a specific Author in a specific Resource Format?

<http://localhost:8080/sparql/documents-cited-by-author-in-format>

```

{
    "author": "Pera Peric",
    "resourceFormat": "csv"
}

```

RESURSI

<https://github.com/tara-pogancev/semantic-web>

Projekat

<https://github.com/sasa-boros/acm-curriculum-explorer/>

Startni projekat sa ACM Curriculum ontologijom

<https://youtu.be/V6BR9DrmUQA>

Dobro objašnjenje Web 3.0