

KONTROLNA TAČKA 2

(UPRAVLJANJE DIGITALNIM DOKUMENTIMA)

AUTOR RADA: DAVID VULETAŠ , R1 4-2018

1. KONFIGURACIJA ELASTICSEARCH I POVEZIVANJE SA SERBIAN PLUGIN-OM

Elasticsearch koji se koristi za izradu projekta je najnovije 6.5.4 verzije. Server se pokreće kao zasebna instanca preuzeta sa oficijalne stranice *Elasticsearch-a*. Iskorišćena je predefinisana konfiguracija sa nazivima node-ova i svim ostalim opcijama. Razlog zbog čega je pomenuti server posebna instanca predstavlja postojanje plugin-a koji je potrebno dodati da bi ga server koristio. Reč je o *SerbianPluginAnalyzer-u*, mana ovog plugin-a je što je on izrađen u vreme kada je aktuelna verzija *Elasticsearch-a* bila 1.3.2 dok je verzija *Lucene-a* bila 4.9.

Konfiguracija sa serverom ostvarena je preko transportnog konfiguracionog fajla koji je definisan u okviru Spring Boot aplikacije (Slika 1).

```
@Configuration
@EnableElasticsearchRepositories(basePackages = "com.scientificcenter.repository.elastic")
public class ESConfig {

    @Value("localhost")
    private String EsHost;
    @Value("9300")
    private int EsPort;
    @Value("elasticsearch")
    private String EsClusterName;

    @Bean
    public TransportClient client() throws Exception {
        return new PreBuiltTransportClient(Settings.EMPTY)
            .addTransportAddress(new TransportAddress(InetAddress.getByName(EsHost), EsPort));
    }

    @Bean
    public ElasticsearchOperations elasticsearchTemplate() throws Exception {
        return new ElasticsearchTemplate(client());
    }
}
```

Slika 1 Konfiguracija Spring Boot aplikacije i ES servera

Kao glavni zadatak za ovu kontrolnu tačku je bilo prilagoditi pomenuti plugin da podržava novije verzije *Elasticsearch-a*. Glavnu prepreku prilikom izmene predstavljalo je to što se većina klasa iz verzije 1.3.2 izbrisano iz biblioteke te su ubačene nove klase,

pa je trebalo to sve uskladiti da radi. Kao primer refaktorisanja koda u cilju prebacivanja na noviju verziju korišćen je *analysis-kuromoji* (plugin za rad sa Japanskim jezikom) koji se može videti na linku (<https://github.com/elastic/elasticsearch/tree/master/plugins/analysis-kuromoji>). Da bi se podesila da se koristi druga verzija potrebno je u *pom.xml* fajlu podesiti da se koristi verzija 6.5.4 Elasticsearch-a.

Prva stvar koja je izmenjena je izbacivanje nasleđivanja klase *AbstractPlugin* zbog toga što ona više ne postoji, te je izvršeno nasleđivanje klase *Plugin* i proširivanje klase *AnalysisPlugin* samim tim je došlo do izmena i takozvanih override metoda. Primer izmene dat je na slici (Slika 2), gornji deo slike predstavlja kod iz verzije 1.3.2 dok je donji usklađen u skladu sa verzijom 6.5.4.

```
public class SerbianPlugin extends AbstractPlugin {  
    public String name() {  
        return "serbian-plugin";  
    }  
    public String description() {  
        return "Analyzer that converts cyrillic words into lowercase latinic";  
    }  
    public void onModule(AnalysisModule analysisModule) {  
        analysisModule.addProcessor(new SerbianBinderProcessor());  
    }  
}  
  
public class SerbianPlugin extends Plugin implements AnalysisPlugin {  
    @Override  
    public Map<String, AnalysisModule.AnalysisProvider  
        <AnalyzerProvider<? extends Analyzer>>> getAnalyzers() {  
        return Collections.singletonMap("serbian_analyzer",  
            SerbianAnalyzerProvider::new);  
    }  
}
```

Slika 2 Pre i posle izmene verzije

Nakon izvršene izmene bilo je potrebno dalje izbaciti procesor klasu *SerbianBinderProcessor* zbog toga što u novijim verzijama nije podržana klasa *AnalysisBinderProcessor* koju prethodno pomenuti procesor nasleđuje.

Takođe još par grešaka se vremenom pojavilo, koje su najviše problema zadale prilikom rešavanja.

U klasi *SerbianAnalyzerProvider* u konstruktoru je promenjen poziv metode roditeljskog konstruktora, odnosno jedan parametar je bilo potrebno izostaviti.

Greška koja se pojavila u klasi *SerbianAnalyzer* je bila vezana za metodu *makeStopSet*, ranije je ova metoda primala dva parametra a to su verzija *Lucene* biblioteke kao i lista reči, dok je u novijoj verziji izbačen parametar koji predstavlja verziju biblioteke. Razlog zbog čega je ovaj parametar izbačen je u tome što ranije *Lucene* biblioteka nije bila uključena u Elasticsearch, nego je predstavljala odvojenu biblioteku. U novijim verzijama Elasticsearch-a, biblioteka sadrži u sebi već pomenutu *Lucene* biblioteku pa shodno tome nije potrebno navoditi koja verzija će se koristiti.

I kao poslednja greška koja je dovela do dužeg proučavanja koda pa samim tim i kontaktiranja Vas, bila je promena parametra konstruktora klase *Among* u klasi *SerbianStemmer* zbog tog što rešenje nisam mogao pronaći nigde jer je vezano za

biblioteku *Snowball*. Kao rešenje je urađeno da umesto *SnowballProgram* parametra u konstruktoru je potrebno proslediti parametar koji je klase *MethodHandles.Lookup*.

Nakon što je izvršeno uspešno refaktorisanje plugin-a, instalacija u okviru servera izvršena je uz pomoć komande:

```
./bin/elasticsearch-plugin install file:///path_to_folder/SerbianPlugin-6.5.4.zip
```

Izvršavanjem prethodno navedene komande, svaka naredna pretraga u okviru našeg sistema vršila je filtriranje korišćenjem *SerbianPlugin analyzer-a*.

2. INDEKSIRANJE DOKUMENTA, MORE LIKE THIS, GEO PRETRAGA

Bitno je naglasiti da je u trenutku izrade korišćen *ElasticsearchRepository* kao olakšanje za pisanje osnovnih upita kroz potpise metoda. Veoma je jednostavno koristiti *Elasticsearch* biblioteku putem *Spring Boot* aplikacije kao klijenta, ubacivanjem jedne zavisnosti u *pom.xml* fajlu vezane za *Elasticsearch Spring Data* (3.1.3.RELEASE).

Uz pomoć prethodno pomenute biblioteke na jednostavan način uz korišćenje anotacija moguće je izvršiti kreiranje dokumenta (klase) u svrhu upisivanja u *Elasticsearch* bazu podataka.

Indeksiranje dokumenta se vrši tako što je potrebno dodati anotaciju *@Document* iznad klase modela za koju želimo da vršimo pretragu u okviru *Elasticsearch* servera. Nakon dodavanja potrebno je odrediti naziv parametra *index* kao i *type* u okviru prethodno pomenute anotacije. Za *index* sam se odlučio da to uvek bude indeks pod nazivom "scientificcenter" dok će *type* imati vrednost naziva entiteta za koji je vezan. Primer indeksiranja pomoću anotacija prikazana je na slici (Slika 3)

```
@Data
@Document(indexName = "scientificcenter", type = "journal", shards = 1, replicas = 0)
public class ScientificJournalElastic {

    @Id
    private String id;

    @Field(type = FieldType.Text)
    private String text;

    @Pattern(regexp = "\\d{4}-\\d{3}[\\dXX]")
    @Field(type = FieldType.Text)
    private String issnNumber;

    @Field(type = FieldType.Nested, includeInParent = true)
    private List<ScientificAreaCodeBook> area;

    @Field(type = FieldType.Boolean)
    private boolean published;

    //.....
}
```

Slika 3 Anotiranje pomoću ES anotacija

Što se tiče *More like this* pretrage, ona je namenjena za pretraživanje dokumenata koji su međusobno povezani po sličnosti. Na primer ukoliko korisnik gleda neki dokument I potrebno je da mu se prikažu na primer slični dokumenti, tada je idealno koristiti ovakav tip pretrage. Da bi se ovakva pretraga izvršila iz Spring Boot aplikacije potrebno je koristiti API Elasticsearch-a gde će se pozivati metoda *moreLikeThisQuery* koja će kao parametre primati *polja za pretragu*, *text za pretragu* kao i opcione parametre koji su *minimalno pojavljivanje rezultata*, *maksimalno pronalazanih rezultata*.

Da bi bilo omogućeno pretraživanje dokumenata po geografskoj lokaciji, potrebno je omogućiti Geo pretragu u okviru Elasticsearch-a. Geo pretraga omogućava pretraživanje po više različitih tipova geo-lokacija kao što su:

- Point (pretraga po jednoj geografskoj koordinati - tački)
- Linestring (pretraga po dva ili više povezana Point-a)
- Polygon (zatvoren oblik, prva i poslednja tačka se moraju poklopiti)
- Multipoint (lista više nepovezanih ali sličnih tačaka)
- MultiLineString (lista više nepovezanih linestring)
- MultiPolygon (lista više nepovezanih polygon-a)
- GeometryCollection
- Envelope (zatvorena kontura kod koje se pretražuju gornje,leve,donje,desne korditane)
- Circle (krug specifikiran po centralnoj tački, gde je moguće podesiti radius pripadnosti u raznim jedinicama)

Da bi se slali upiti ka Elasticsearch serveru potrebno je kreirati upite u okviru Spring Boot aplikacije korišćenjem klase *GeoShapeQueryBuilder*, a pritom je prethodno obavezno da je sačuvana lokacija nekog dokumenta korišćenjem *GeoJsonPoint* klase u odgovarajućem formatu. Takođe da bi se ovo sve moglo koristiti potrebno je dodati zavisnost u pom.xml

```
<dependency>
  <groupId>org.locationtech.spatial4j</groupId>
  <artifactId>spatial4j</artifactId>
  <version>0.7</version>
</dependency>
<dependency>
  <groupId>org.locationtech.jts</groupId>
  <artifactId>jts-core</artifactId>
  <version>1.15.0</version>
</dependency>
```