Upute za 1. projekt iz predmeta
Napredni modeli i baze podataka

Pretraživanje teksta u relacijskim bazama podataka i napredni SQL

akademska godina 2016/2017

Potrebno je izraditi aplikaciju (desktop/web/mobilnu/…) koja će omogućiti:

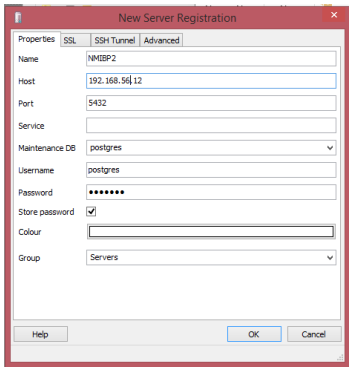
1. Unos tekstualnog sadržaja u PostgreSQL bazu podataka proizvoljne sheme (minimalne ali prikladne za demonstraciju svih zadataka iz projekta)
2. Pretragu u bazi podataka pohranjenog tekstualnog sadržaja
3. Analizu postavljanih upita u zadanom vremenskom periodu

Izgled aplikacije je proizvoljan kao i izbor programskih tehnologija pomoću kojih ćete ju izraditi.

Korištenje PostgreSQL SBP pri izradi projekta

Preporučujemo (ali nije obavezno) da kao repozitorij za tekstove/dokumente koristite PostgreSQL sustav instaliran na virtualnom računalu namijenjenom uvježbavanju gradiva iz predmeta.

Nakon što, prema uputama objavljenima na stranici predmeta, pokrenete virtualno računalo, SQL naredbe nad PostgreSQL sustavom možete obavljati pomoću ***psql*** naredbe na linux-u. Međutim, jednostavnije i lakše je koristiti neki od SQL editora prilagođenih za rad s PostrgreSQL-om npr. PgAdmin čiju instalaciju možete preuzeti sa http://www.pgadmin.org/download/.



U *pgAdmin-u* definirajte postavke (kao na slici desno) za spajanje na PostgreSQL server instaliran na virtualnom računalu pomoću opcija: *File – Add Server*

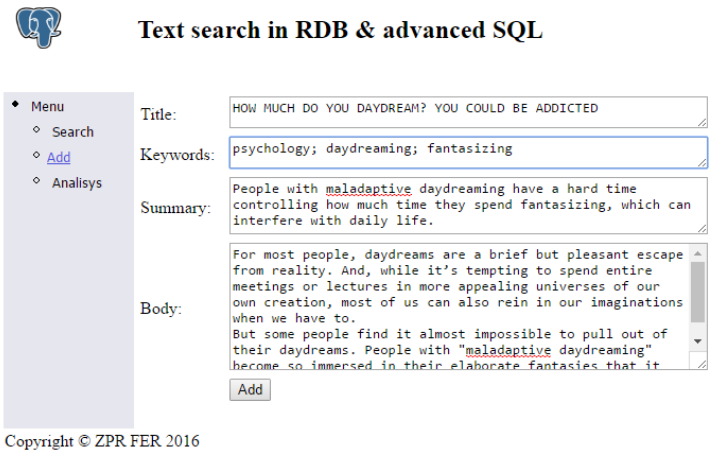
Spojite se na PostgreSQL SUBP. Kreirajte novu bazu podataka (desni klik mišem na Databases – New Database; Name: *poŽelji*, Owner: *postgres*). Osvježite sadržaj podstabla Databases (desni klik mišem na Databases - Refresh), odaberite bazu s kojom želite raditi i otvorite prozor za izvođenje upita (Tools – Query Tool).

Za podešavanje formata prikaza datumskog tipa podataka na *dd.mm.yyyy* izvedite sljedeću naredbu:

```
SET DateStyle ='German, DMY';
```

1. Unos tekstualnog sadržaja

Zbog nepostojanja cjelovite podrške za potpunu pretragu teksta za hrvatski jezik koristite tekstove na engleskom jeziku.



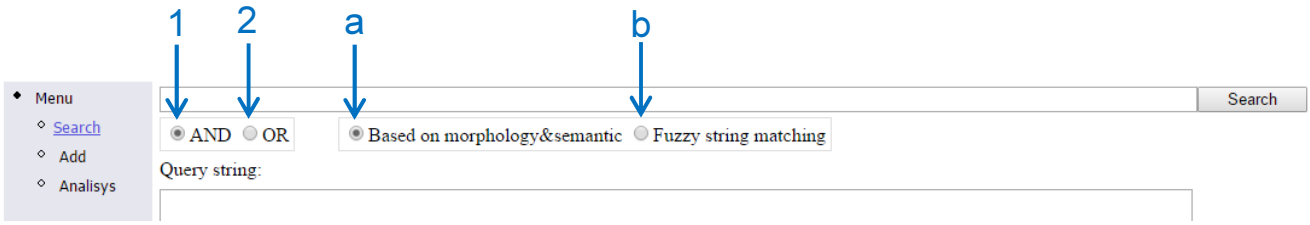
2. **Pretraga u bazi podataka pohranjenog tekstualnog sadržaja**

Potrebno je realizirati dvije vrste pretrage:

a. Dohvat dokumenata koji sadrže traženi uzorak (uzorke) u normaliziranom obliku - **Based on morphology&semantic**
   Ovdje treba dohvatiti i dokumente koji sadrže bilo koji oblik riječi iz teksta pretrage tj. potrebno je normalizirati riječi, ukloniti stop riječi i sl.
b. Dohvat dokumenata koji sadrže približno jednak uzorak (uzorke) - **Fuzzy string matching**
   Ovdje nas zadovoljavaju i dokumenti koji sadrže u manjoj mjeri izmijenjen uvjet pretrage – npr. za uvjet pretrage 'Haven from' i dokument 'A fast Trip from Heaven' treba biti vraćen kao rezultat (primijetite da Haven i Heaven nemaju jednak korijen/leksem i da to ovdje nije važno).

Podržati povezivanje zadanih uzoraka pretrage logičkim operatorom

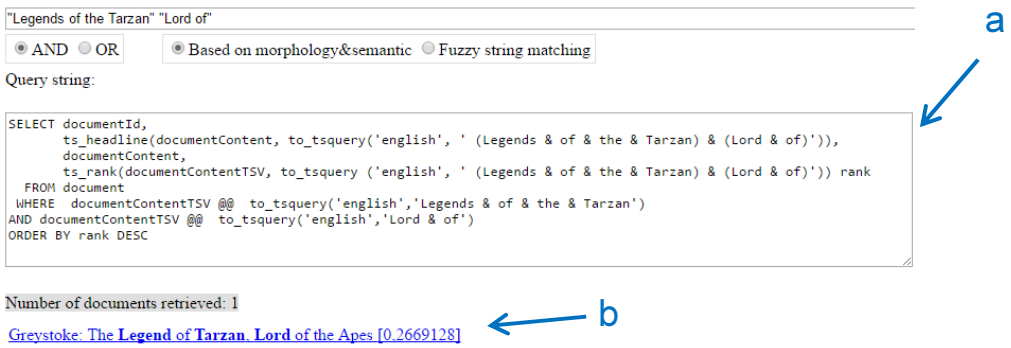
1. AND
2. OR



Modelirajte bazu podataka tako da koliko god možete ubrzate pretragu - npr. dodatna pohrana normaliziranog teksta, kreiranje specijalnih indeksa (koliko god i koji god mogu ubrzati bilo koju vrstu pretrage).
Za realizaciju pretraga pod a) i b) odaberite operatore odnosno funkcije PostgreSQL-a koje smatrate najprikladnijima. Izbor morate biti u stanju argumentirati.

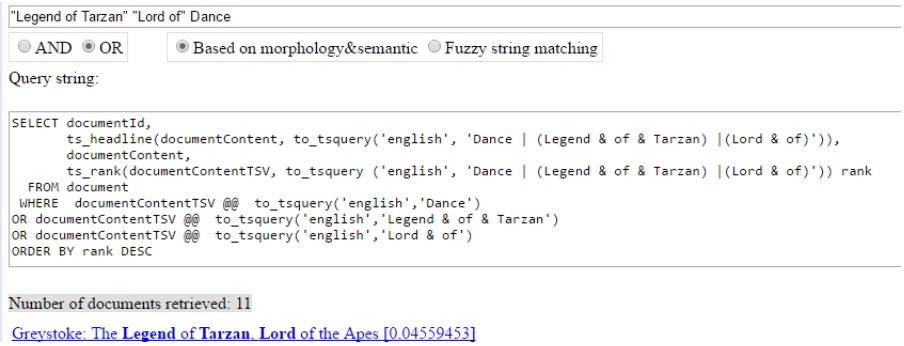
U aplikaciji je potrebno:

a. prikazati SQL upit kojim su temeljem korisnikovog uvjeta pretrage dohvaćeni relevantni dokumenti
b. prikazati informativne podatke o dohvaćenim dokumentima s podebljanim riječima temeljem kojih je dokument kvalificiran kao rezultat te **rang** dokumenta. Primijetite da su riječi Legend Tarzan i Lord podebljane u dokumentu na slici.



Proučite kako rade funkcije za rangiranje dokumenata implementirane u PostgreSQL-u (preporučujemo sadržaj na http://shisaa.jp/postset/postgresql-full-text-search-part-3.html) te u aplikaciji upotrijebite funkciju za rangiranje koju smatrate prikladnom s parametrima koje smatrate prikladnima. Nemojte slijediti primjer sa slike u kojem je funkcija ts_rank pozvana s "defaultnim" parametrima.

c. podržati traženje fraza (znakovni niz naveden unutar navodnika) i "jednostavnih" riječi kombiniranih logičkim operatorima AND i OR.
   Donja slika prikazuje jedan od načina kako bi se moglo pronaći dokumente koji sadrže fraze "Legend of Tarzan" ili "Lord of" ili riječ Dance



3. Analiza postavljanih upita u zadanom vremenskom periodu

Da bi analiza bila moguća potrebno je bilježiti upite koje korisnici postavljaju. To treba uzeti u obzir pri dizajniranju baze podataka i izradi aplikacije. Procijenite što bi sve o postavljanim upitima bilo dobro bilježiti.

Potrebno je omogućiti korisniku:

1. zadavanje vremenskog perioda u kojem treba provesti analizu u obliku:
   datumOd – datum do (npr. 10.10.2016 -13.10.2016)
2. Odabir granulacije analize:
   a. dan ili
   b. sat

Korištenjem naprednih mogućnosti SQL-a (pivotiranje) izraditi izvještaj s pregledom broja postavljanja konkretnog upita za zadani period (npr. 10.10.2016 -13.10.2016) po danima ili po satima ovisno o tome što je korisnik odabrao.

Po danima:

	querystring character(200)	d10102016 integer	d11102016 integer	d12102016 integer	d13102016 integer
1	'Dance' & 'Legend of Tarzan' & 'Lord'	4	3	2	
2	'Lord' & 'Dance'	3	2	2	

ili po satima:

	querystring character(200)	s00_01 integer	s01_02 integer	s02_03 integer	s03_04 integer	s04_05 integer	s05_06 integer	s06_07 integer	s07_08 integer	s08_09 integer	s09_10 integer	s10_11 integer	s11_12 integer	s12 inte
1	'Dance' & 'Legend of Tarzan' & 'Lord'								1		3	3	1	
2	'Lord' & 'Dance'										3	3	1	

Pomoć: Da biste mogli koristiti funkcije za Full Text Search i Fuzzy Text Serach morate uključiti module fuzzystrmatch i pg_trgm, odnosno tablefunc za korištenje funkcija za pivotiranje. Module je potrebno uključiti u svakoj bazi podataka u kojoj ih namjeravate koristiti. „Registriraju“ se izvođenjem sljedećih naredbi:

```
CREATE EXTENSION fuzzystrmatch;     -- (soundex, levenshtein, metaphone)
CREATE EXTENSION pg_trgm;           -- (similarity , show_Trgm,..., %, <->)
CREATE EXTENSION tablefunc;         -- (crosstab)
```

Za prikazivanje sažetih informacija o dohvaćenim dokumentima s podebljanim ključnim riječima možete koristiti funkciju *ts_headline.*

Rješenje projekta treba postaviti u vlastiti direktorij NMiBP\P1. U korijenski direktorij postaviti readme.txt datoteku koja sadrži:

- shemu korištene baze podataka zajedno sa svim kreiranim indeksima
- kratki opis korištenih tehnologija (rečenica-dvije)

- naputak za pokretanje rješenja

Razmatrat će se i polovična rješenja.

Studente se kod prezentiranja projekta može pitati da:
- objasne segment vlastitog rješenja
- objasne neki koncept iz područje pretrage teksta ili pivotiranja (npr. kako radi operator/funkcija koju su upotrijebili za neku od vrsta pretrage)
- objasne zbog čega su primijenili konkretni operator/funkciju, a ne neki drugi
- pokrenu tj. demonstriraju rješenje (i npr. izvedu upit)
- itd.

Rok za dostavu rješenja: četvrtak 27.10.2016 do 8:00