**Wiki Indexer**

Pentru ca am ramas fara subiecte de discutat in pauza de masa, am zis ca ar fi amuzant sa aflam care sunt cele mai folosite 10 cuvinte din fiecare articol Wikipedia☺ Cu ajutorul vostru o sa aflam si asta, si o sa ducem cu totii o viata mai fericita.

**Descrierea aplicatiei**

Aplicatia trebuie sa indeplineasca un rol foarte simplu. Primind un titlu al unui articol Wikipedia, trebuie sa afiseze un bar-chart, cu cele mai frecvente 10 cuvinte din acel articol. In cazul in care se doreste interogarea unui numar mai mare de articole, atunci utilizatorul poate uploada un fisier cu mai multe titluri, iar rezultatele afisate vor fi cele aggregate din toate articolele (Pe primul loc va fi cuvantul cu cele mai multe aparitii in toate articolele agregate). Pentru ca noi suntem oameni cumsecade si nu am vrea sa sacaim Wikipedia cu prea multe requesturi, in cazul in care un articol a mai fost deja cautat, atunci rezultatele ar trebui luate dintr-o baza de date locala, in locul unei noi interogari a API-ului Wikipedia.

**UI**

In interfata ar trebui sa aveti urmatoarele componente, afisate oricum doriti:

* Un text box in care sa se poata introduce titlul cautat
* Un bar-chart care sa afiseze top 10 cuvinte ordonate dupa frecventa aparitiei lor in articol
* Timpul necesar cautarii cuvintelor in articol (cat a durat cautarea cuvintelor si identificarea top 10)
* Sursa de date de unde au fost preluate rezultatele (BD sau Wikipedia)
* O componenta care sa permita upload de fisiere pentru modul de prelucrare batch
* Orice altceva mai considerati voi relevant (continutul articolului , stelute aurii si unicorni)

**Backend**

Partea de backend ar trebui sa sa contina logica aferenta care sa suporte toate functionalitatile din UI. Dupa orice cautare, rezultatele ar trebui salvate intr-o baza de date, pentru caching in cazul unei cautari identice, iar in cazul in care se uploadeaza un fisier, prelucrarea titlurilor de acolo ar trebui facuta in paralel, pe mai multe threaduri, si apoi agregata.

**Application Milestones**

Va puteti apuca de orice modul in orice ordine, dar aveti aici o lista cu milestone-urile pe care noi le vom lua in cosiderare in cadrul corectarii. Ficare milestone are toate functionalitatile din milestone-urile de dinaintea lui:

1. No UI, cautarea unui singur titlu si afisarea rezultatelor la consola
2. Salvarea rezultatelor in baza de date si caching de acolo
3. Cautarea a mai multe titluri, folosind fisier de pe local (single thread)
4. 3 + multiple threads
5. Integrare UI, cu afisarea rezultatelor in mod text si fisier de pe local in cazul modului batch
6. 5 + componenta upload fisier
7. Full UI, cu afisare bar chart
8. Executor pattern folosit pentru search bazat pe fisier de intrare (handling al threadurilor mai frumos)

**Stelute aurii**

* Folosirea cat mai multor componente de sintaxa din Java 8
* Teste unitare si de integrare care sa verifice functionalitatea codului
* UI demn de pus in rama

**Hints**

* You can use Wikipedia special export(<https://en.wikipedia.org/wiki/Special:Export>)

(Example URL for post call you can try in Postman <https://en.wikipedia.org/wiki/Special:Export?pages=Snail&curonly=1>)

* http://www.mkyong.com/java/how-to-read-xml-file-in-java-dom-parser/