

Tema 2

SmartTranslator – Serviciu de traducere texte

Responsabil: Irina Toma

Introducere

Scopul acestei teme este de a crea o aplicație similară cu Google Translate care poate să traducă cuvinte sau propoziții dintr-o limbă în alta.

Descrierea problemei

Firma la care lucrați are o colecție substanțială de dicționare explicative pentru mai multe limbi și vrea să creeze o aplicație care să rivalizeze cu Google Translate. Prin urmare, principală funcționalitate a aplicației este traducerea de cuvinte și propoziții.

În urma unui studiu de piață întreprins de firmă, aceștia au descoperit că persoanele care folosesc Google Translate ar vrea să aibă mai multe variante de traducere ale aceluiași text, precum și o listă de sinonime cu care să poată să fie înlocuite unele cuvinte.

Aplicația este dezvoltată de trei echipe separate, o echipă care se ocupă de scanarea dicționarelor și de procesul de OCR (Optical Character Recognition), o echipă care se ocupă de backend și una care se ocupă de frontend. Voi lucrați în echipa de backend care primește informațiile din dicționar de la prima echipă și trebuie să furnizeze date către echipa de frontend.

Prima echipă v-a pus la dispoziție două fișiere text ce conțin informații parțiale din dicționarele român și francez în format JSON (JavaScript Object Notation). Restul de dicționare vor fi livrate la un alt moment din viitor. Structura unui element din dicționar este următoarea:

Tabel 1. Forma cuvintelor din dicționar.

Nr. linie	Exemplu dicționar român	Descriere câmpuri
1	{	Un obiect JSON este cuprins între { }, o listă este cuprinsă între []
2	"word": "pisică",	Forma de dicționar în limba română
3	"word_en": "cat",	Forma de dicționar în limba engleză – disponibilă pentru orice cuvânt din orice dicționar
4	"type": "noun",	Partea de vorbire – noun, verb
5	"singular": ["pisică"],	Forma de singular a cuvântului
6	"plural": ["pisici"],	Forma de plural a cuvântului
7	"definitions": [Listă de definiții pentru cuvântul curent
8	{	
9	"dict": "Dicționar de sinonime",	Numele dicționarului
10	"dictType": "synonyms",	Tipul dicționarului – <i>synonyms</i> sau <i>definitions</i>
11	"year": 1998,	Anul publicării dicționarului

12	"text": ["mâță", "cotoroabă", "cătușă"]	Listă de <i>sinonime</i> sau de definiții ale cuvântului (în acest exemplu sunt disponibile 3 sinonime)
13	},	
14	{	
15	"dict": "Dicționarul explicativ al limbii române, ediția a II-a",	Numele dicționarului
16	"dictType": "definitions",	Tipul dicționarului – synonyms sau <i>definitions</i>
17	"year": 2002,	Anul publicării dicționarului
18	"text": ["Mamifer domestic carnivor din familia felinelor, cu corpul suplu, acoperit cu blană deasă și moale de diferite culori, cu capul rotund, cu botul foarte scurt, cu maxilarele puternice și cu ghearele retractile și ascuțite", "Dispozitiv de agățare și de desprindere a berbecului din capătul cablului de ridicare de la sonetele cu cădere liberă.", "Mănunchi de sârme de oțel, folosit pentru curățarea de noroi sau de pământ a utilajelor de foraj."]	Listă de sinonime sau de <i>definiții</i> ale cuvântului (în acest exemplu sunt disponibile 3 definiții)
19	}	
20]	
21	}	

Notă: În funcție de tipul părții de vorbire, câmpurile singular și plural pot avea următorul format:

Tabel 2. Formele singular și plural pentru diferite părși de vorbire.

Parte de vorbire	Singular	Plural
Noun	Listă cu forma de singular a cuvântului	Listă cu forma de plural a cuvântului
Verb	Listă de conjugări la indicativ prezent singular în ordine persoana I, a II-a și a III-a	Listă de conjugări la indicativ prezent plural în ordine persoana I, a II-a și a III-a

Aveți de implementat următoarele funcționalități:

- Citirea din dicționare și salvarea informațiilor într-o **singură** colecție de date. Pentru citirea din JSON se va folosi biblioteca **gson** (<https://github.com/google/gson>)
 - Se vor citi toate fișierele dintr-un folder de intrare care au extensia .json
 - Atenție** – nu știți de la început numărul lor
- Metodă pentru adăugarea unui cuvânt în dicționar – cuvântul are forma descrisă în Tabel 1
 - boolean addWord(Word word, String language)

- Întoarce true dacă s-a adăugat cuvântul în dicționar sau false dacă există deja cuvântul în dicționar
- 3. Metodă pentru ștergerea unui cuvânt din dicționar ce primește ca parametru cuvântul și limba
 - boolean removeWord(String word, String language)
 - Întoarce true dacă s-a șters cuvântul în dicționar sau false dacă nu există cuvântul în dicționar
- 4. Metodă pentru adăugarea unei noi definiții pentru un cuvânt dat ca parametru – o definiție are același format ca o intrare din lista *definitions* descrisă în Tabel 1 între liniile 8-13
 - boolean addDefinitionForWord(String word, String language, Definition definition)
 - Întoarce true dacă s-a adăugat definiția sau false dacă există o definiție din același dicționar (*dict*)
- 5. Metodă pentru ștergerea unei definiții a unui cuvânt dat ca parametru
 - boolean removeDefinition(String word, String language, String dictionary)
 - Întoarce true dacă s-a șters definiția sau false dacă nu există o definiție din dicționarul primit ca parametru
- 6. Metodă pentru traducerea unui cuvânt
 - String translateWord(String word, String fromLanguage, String toLanguage)
 - Întoarce traducerea cuvântului word din limba fromLanguage în limba toLanguage
- 7. Metodă pentru traducerea unei propoziții
 - String translateSentence(String sentence, String fromLanguage, String toLanguage)
 - Întoarce traducerea propoziției sentence din limba fromLanguage în limba toLanguage
- 8. Metodă pentru traducerea unei propoziții și furnizarea a 3 variante de traducere folosind sinonimele cuvintelor
 - ArrayList<String> translateSentences(String sentence, String fromLanguage, String toLanguage)
 - Întoarce traducerea propoziției sentence din limba fromLanguage în limba toLanguage
 - În cazul în care nu există 3 variante de traducere a propoziției se vor furniza doar variantele posibile
- 9. Metodă pentru întoarcerea definițiilor și sinonimelor unui cuvânt
 - ArrayList<Definition> getDefinitionsForWord(String word, String language)
 - Definițiile sunt sortate crescător după anul de apariție al dicționarului
- 10. Metodă pentru exportarea unui dicționar în format JSON
 - void exportDictionary(String language)
 - Se va exporta doar partea din structura de date ce ține de limba primită ca parametru și se vor scrie informațiile într-un fișier
 - Cuvintele din JSON sunt ordonate alfabetic, iar definițiile sunt ordonate după anul de apariție al dicționarului
 - Formatul JSON este la alegerea voastră dar trebuie să conțină toate informațiile despre un cuvânt

Atenție – trebuie să păstrați numele și semnătura metodelor. Clasele Word și Definition vor fi implementate pe baza modelului din Tabel 1.

Punctaj

- 20p – Citirea din fișierele dicționar și salvarea informațiilor într-o singură colecție de date
- 45p – Implementarea metodelor de la punctele 2-10 (5p fiecare metoda)
- 15p – testarea fiecărei metode cu câte 2 exemple care să reprezinte un caz normal de utilizare, și un caz de excepție (de exemplu, nu a fost adăugat un cuvânt în dicționar pentru că acesta deja există)
- 10p – Organizarea modulară a codului – organizarea claselor pe pachetelor, convenții de nume pentru clase/variabile, concepte POO
- 10p – Comentarii reprezentative în cod
- [Bonus] 10p – Folosire Maven (<https://maven.apache.org/>)

Depunctări suplimentare

- 100p – nefolosirea conceptelor POO
- 20p – salvarea informațiilor din dicționare în mai multe colecții de date (de exemplu, câte o listă pentru fiecare dicționar)
- 10p – specificarea numărului sau numelor dicționarelor care trebuie citite
- 2p - pentru fiecare metodă care nu păstrează numele sau semnătura specificată în lista de funcționalități

Observații suplimentare

Pentru crearea fișierelor de intrare puteți folosi <https://dexonline.ro/>