

Dokumentáció

Szótár

Kliensoldali technológiák házi feladat

Feladat

A feladat egy egyszerű szótár program implementálása. Az alkalmazás lehetőséget ad az API által ismert nyelvek közötti fordításra és az ismert nyelvekhez való szinoníma keresésre. A program bekér egy szót, majd a felhasználó kiválaszthatja a megfelelő nyelveket és, hogy milyen műveletet szeretne végrehajtani. Sikeres lekérdezés esetén kilistázza a találatokat, amik magából a szóból és a szófajból állnak.

Dictionary

Word

Translate from to

Synonym in

Result:

language unit (generic term)
(noun)

linguistic unit (generic term)
(noun)

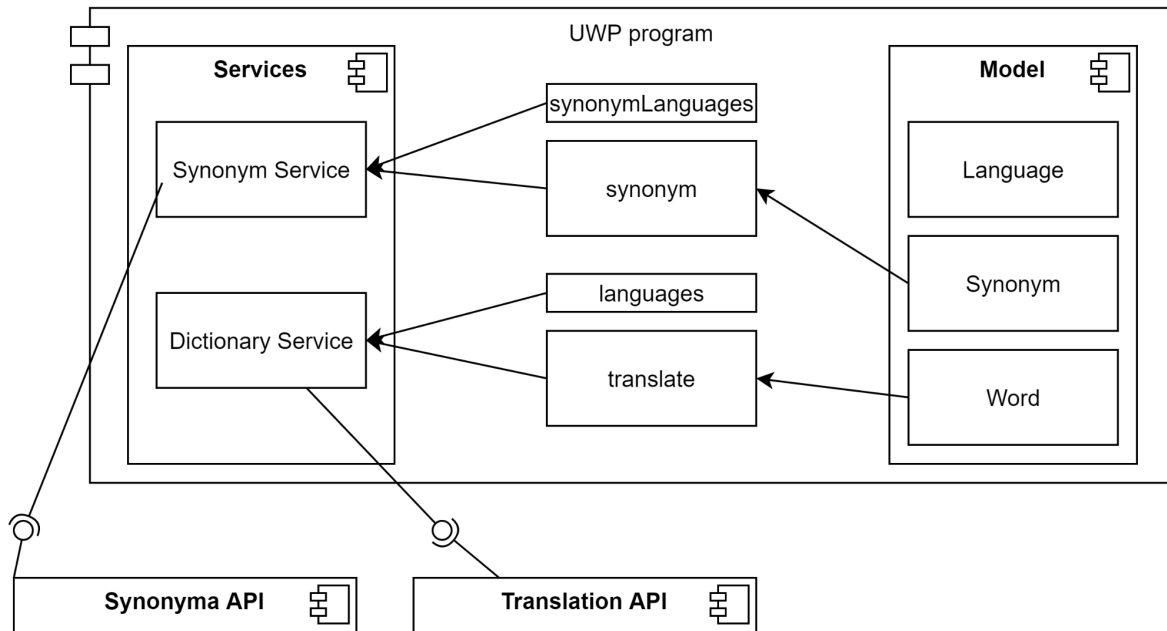
statement (generic term)
(noun)

news
(noun)

intelligence
(noun)

Architektúra

A frontend Universal Windows Platform-ot használ. A [szinonima kereséshez](http://thesaurus.altervista.org/service)¹ és a [fordításhoz](https://yandex.com/dev/dictionary/)² két különböző API-t használtam. A project felépítését ábrázoló diagramm:



¹ <http://thesaurus.altervista.org/service>

² <https://yandex.com/dev/dictionary/>

APIs

- A Synonyma API a szinonímák keresésére szolgál, amiket egy http get kérés segítségével ad meg.
- A Translation API a fordításokhoz szükséges, amitől az ismert nyelveket és egy szó adott nyelvű megfelelőjét egy-egy http get kérés végzi el.

Model

- A Language osztály nyelv párokat tárol el amikből megkapható az a formátum, amit az API-nak kell átadni. Sajnos nem egyezik a szinonímák és a szótár API formátumai, így ez az osztály csupán, a szótárhoz alkalmazható.
- A Synonym osztály formátuma a Synonym API által visszaküldött JSON törzsből határozódott meg. Számunkra fontos adat a szinonímák és a hozzájuk tartozó szófajok.
- A Word osztály a Translation API által visszaküldött JSON törzsből lett megalkotva. Számunkra fontos adata a lefordított szavak és azok szófajai. Lebutított változata a ResultWord, ami csak a számunkra fontos adatokat tartalmazza, és használható a szinonimáknál is.

Services

- A SynonymService aszinkron http get kéréseket küld az API felé, aminek eredménye a keresett szinonímák.
- A DictionaryService aszinkron http get kéréseket küld az API felé. Lekérdezi a szótár által ismert nyelvpárokat, valamint az adott szóhoz és nyelvpárhoz tartozó jelentéseket is.

Egyéb segéd függvények és változók

- A currentLanguage változó eltárolja az aktuális fordítási nyelvpárokat.
- A languages eltárolja az API által támogatott fordítási nyelvpárokat.
- A synonymLanguage eltárolja az aktuális szinonima nyelvét.
- A synonymLanguages eltárolja az API által támogatott szinonima nyelveket.
- A from és to listák arra kellene, hogy csak olyan fordítási nyelvpárok legyenek amiket támogat az API.
- A result lista tárolja a legutóbbi sikeres művelet eredményeit.
- Vannak függvények amik a XAML leírás való kommunikációért és a user által végzett változások kezeléséért felelnek.
- A synonym és translate függvény kommunikál a megfelelő service-szel.

API

Szinonima API

A szinonima API működéséhez szükség van egy egyedi kulcsra, egy szóra és a nyelvre amin keressük a szinonimákat. Ezek segítségével kérdezi le egy http get kérés a keresett eredményt.

Szótár API

A szótár API működéséhez szükség van egy egyedi kulcsra, egy szóra és egy nyelvpárra amik között akarunk fordítani. A lehetséges nyelvpárokat le lehet kérdezni az API-tól. Ezek segítségével kérdezi le egy http get kérés a keresett eredményt.

A szótár által támogatott nyelvek lekérése

A támogatott nyelvek lekérése megtörténik és visszaad egy string listát, amit aztán felhasználunk a későbbi műveletek során.

```
var serviceDic = new DictionaryService();  
List<string> l = await serviceDic.getLanguages();
```

A műveletet a DictionaryService végzi el. Ezen belül van egy getAsync template, ami a paraméterül kapott url segítségével végrehajtja a kérést. A válasz JSON törzsét ezek után átkonvertálja a kívánt formátumba, ebben az esetben egy string listává.

```
private async Task<T> getAsync<T>(Uri uri)  
{  
    using (var client = new HttpClient())  
    {  
        var response = await client.GetAsync(uri);  
        var json = await response.Content.ReadAsStringAsync();  
        T result = JsonConvert.DeserializeObject<T>(json);  
        return result;  
    }  
}
```

Ezt a függvényt használja fel a getLanguages függvény, ami aszinkron módon visszaadja a támogatott nyelvek string listáját. Ebben a függvényben lényegében csak az url összeállítása történik, a többit a már előbb ismertettet getAsync függvény hívása intézi.

```
public async Task<List<string>> getLanguages()  
{  
    string url = APIUrl + "/getLangs?key=" + key;  
    return await getAsync<List<string>>(new Uri(url));  
}
```

Az url összeállításához az API elérési címét, a getLangs parancsot és a kapott egyedi kulcsot használjuk.

```
readonly Uri APIUrl = new Uri("https://dictionary.yandex.net/api/v1/dic-service.json");  
readonly string key = "dict.1.1.20210430T075226Z.4c52d1197591553a.140645815d9a9917fc293be3af7d089413da206b";
```

Források

- <https://app.diagrams.net/>
- <https://yandex.com/dev/dictionary/>
- <http://thesaurus.altervista.org/service>