Modelare orientată obiect - de la limbă la limbaj

Coordonator:

Conf. Dr. Sabin - Corneliu Buraga

Absolvent:

Gordîn Ștefan

Cuprins

Cerințe

O aplicație care

La primirea unui text scris într-un limbaj controlat și a unui limbaj de programare

Să extragă modelul (conform paradigmei de programare orientată obiect)

Și să întoarcă codul (în limbajul de programare specificat) ce este asociat cu modelul extras

Cerințe

O aplicație care

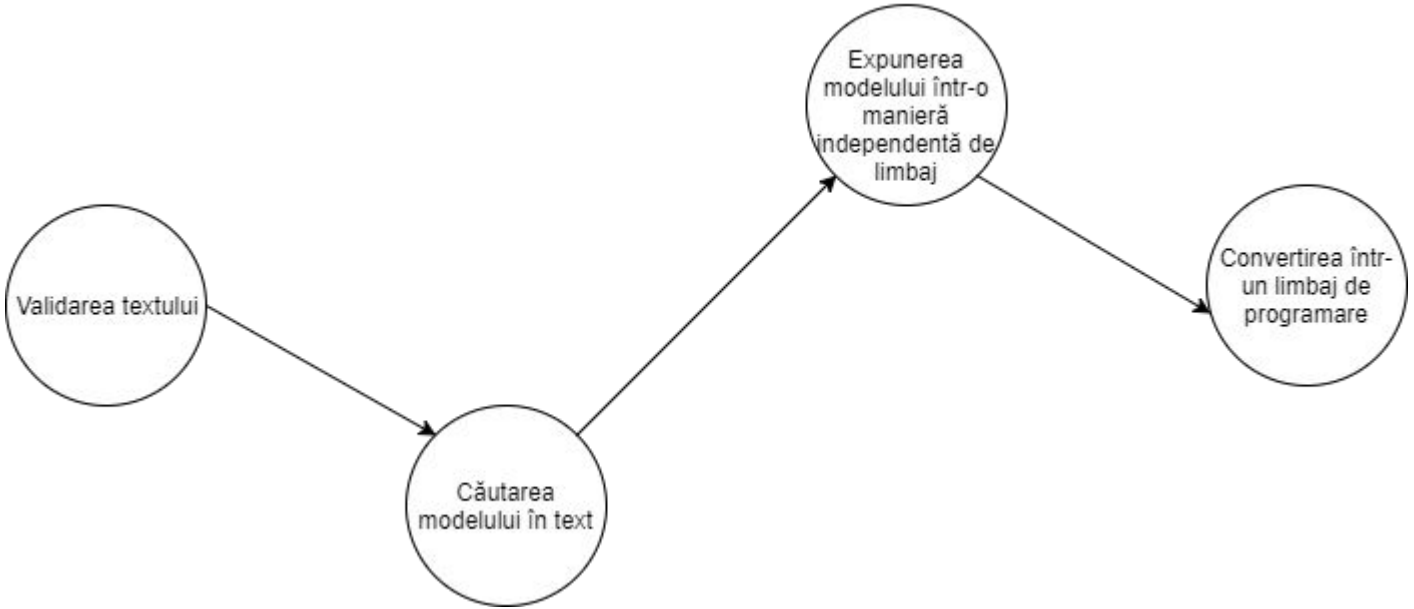
- Să fie extensibilă relativ la conceptele pe care le caută în textul controlat
- Să fie extensibilă relativ la limbajul de programare în care generează cod

Direcții asemănătoare

NLTK

Blockly

Fluxul aplicației



Soluția propusă

O soluție bazată pe trei servicii web. Acestea sunt:

- Serivicul de procesare de text
- Serivicul de generare de cod
- Serviciul de integrare

Serviciul de procesare de text

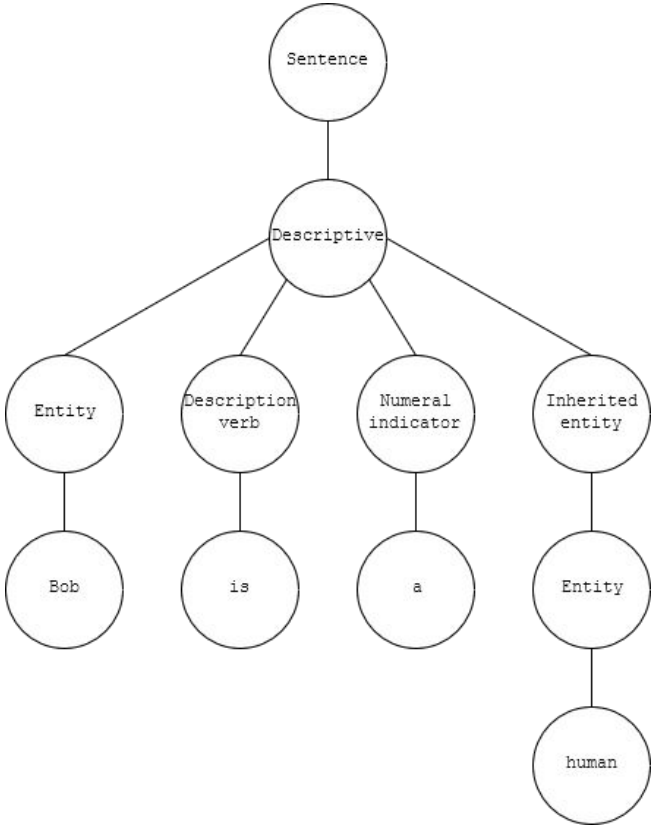
Primește textul din care se va extrage modelul. Propozițiile sunt validate de o gramatică independentă de context

```
sentence      ->   descriptive_sentence
                |  possessive_sentence
                |  behavioral_sentence
                |  singleton_sentence
```

Serviciul de procesare de text

Pe baza textului este creat arborele de propoziție.

"Bob is a human."



Serviciul de procesare de text

Aborele este parcurs și din acesta este extras "modelul dicționar”.

"Bob has a car. Bob's car is a Toyota. Bob can drive."

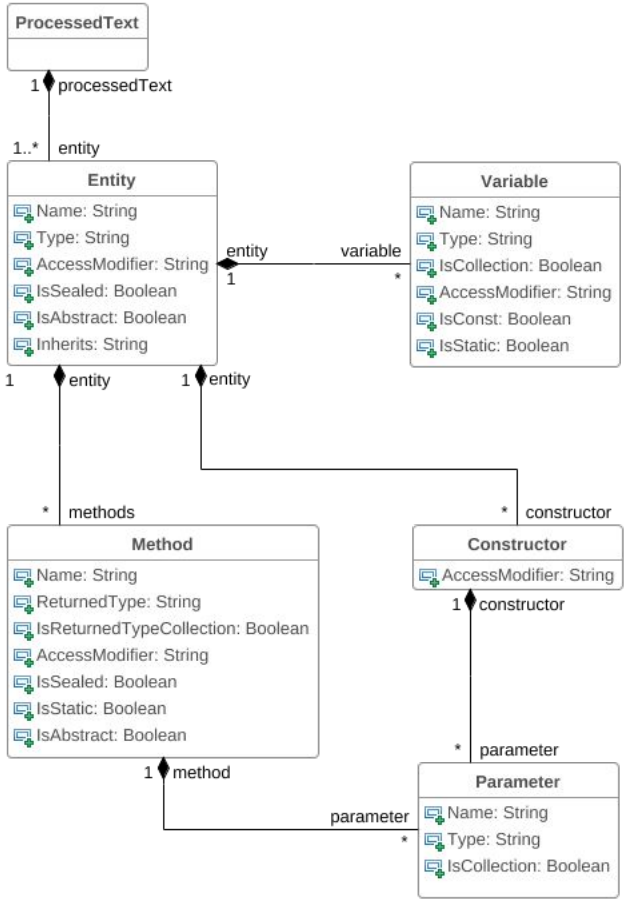
```
[{
    "name": "Bob",
    "methods": [{
        "accessModifier": "public",
        "name": "drive"
    }],
    "variables": [{
        "isCollection": false,
        "name": "car",
        "type": "Toyota",
    }]
},
{
    "name": "Toyota"
}]
```

Modelul-dicționar

Independent de limbaj

Expresiv

Ușor de procesat



Serviciul de generare de cod

Primește un limbaj de programare și o structură de tip model-dicționar. Acesta este validat:

Sintactic

```
public class String {

}
```

Semantic

```
public class Example extends Example {

}
```

Serviciul de generare de cod

Dacă structura primită este validă, se generează codul asociat cu modelul primit în limbajul de programare specificat. Spre exemplu, dacă limbajul solicitat este ***Java***:

"Bob has a car. Bob's car is a Toyota. Bob can drive."

```
public class Bob {
        public Toyota car;
        public void drive() {
                throw new UnsupportedOperationException();
        }
}
```

Serviciul de integrare

Acest serviciu concatenează fluxurile celor două servicii. Aici putem da la intrare un text și un limbaj de programare și vom primi clasele corespunzătoare modelului.

În plus, acesta oferă un URL către o arhivă în care se găsesc clasele generate.

Demo

Direcții de viitor

Generarea de diagrame

O multitudine de limbaje de programare

O multitudine de limbaje naturale

Fluxul invers (de la cod la limbaj natural)

Arhitectura bazată pe microservicii

Concluzii

Aplicația descrisă este o unealtă de programare. Aceasta extrage un model bazat pe paradigma de programare orientată obiect dintr-un text natural controlat și generează într-un limbaj de programare cod orientat obiect ce descrie modelul extras.

Vă mulțumesc!