

Przetwarzanie danych w chmurach obliczeniowych

Dokumentacja projektu CityResidents
- Grafowe Bazy Danych

Tomasz Kozieł

Wydział Fizyki i Informatyki Stosowanej
Akademia Górniczo Hutnicza im. Stanisława Staszica w Krakowie

1 Założenia początkowe

Welcome to City Residents App

Choose an option:

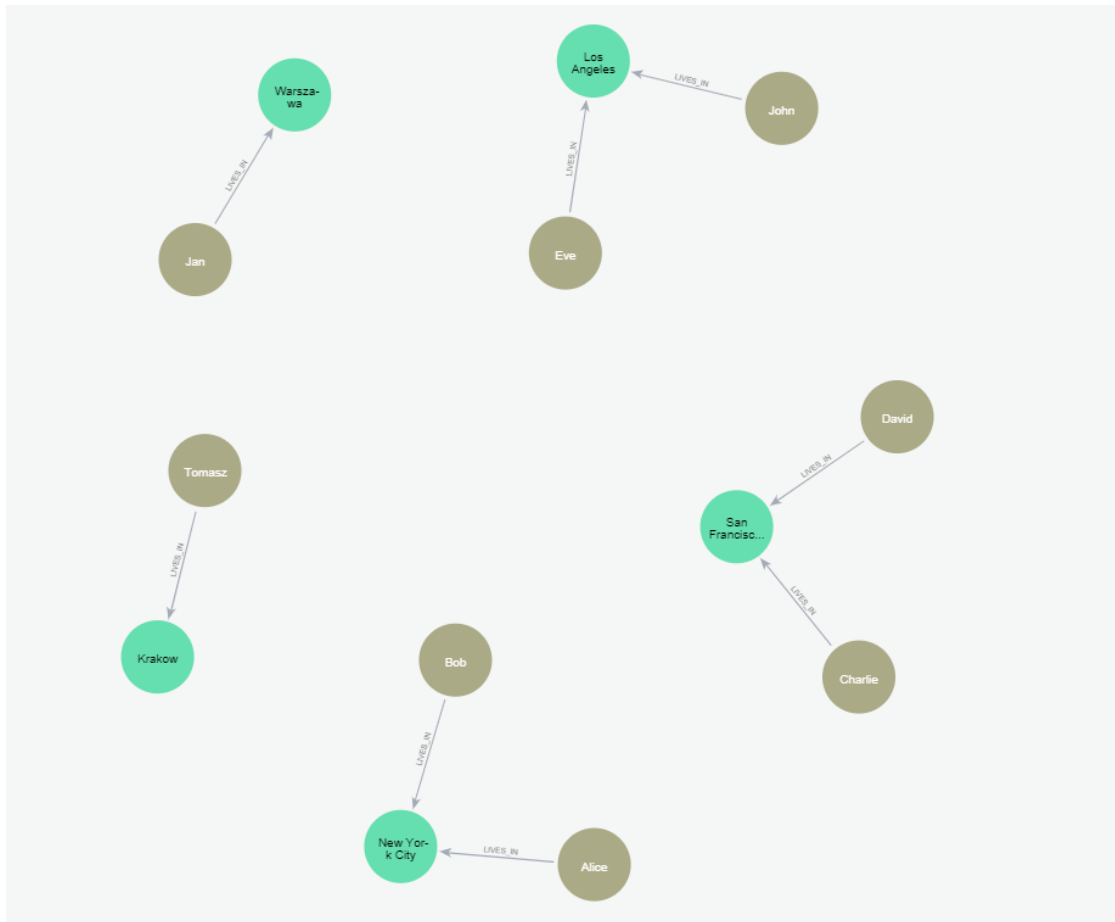
[City List](#)

[Person List](#)

[Person City List](#)

Rysunek 1: Początkowy widok strony.

Stworzona została prosta baza danych składająca się z mieszkańców (imię oraz wiek) oraz miast (nazwa miasta) połączonych relacją określającą kto, gdzie mieszka:



Rysunek 2: Graficzna prezentacja grafu.

Do edycji oraz wyświetlania danych zostały wykorzystane polecenia języka Cypher. Kod źródłowy możliwy jest do obejrzenia na platformie github, gdzie przekierowuje poniższy link:

<https://github.com/tomekkoziel/CityResidents>

2 Wykorzystane technologie

W celu przechowywania danych wykorzystana została baza danych Neo4J, natomiast projekt został napisany w technologii Django i umieszczony w chmurze Heroku, projekt jest dostępny pod poniższym linkiem:

<https://city-residents-d296f243fb43.herokuapp.com/>

3 Udostępnione funkcjonalności

Interfejs umożliwia użytkownikowi następujące akcje:

- Wyświetlanie użytkowników:

```
MATCH (person:Person) RETURN person
```

- Wyświetlanie miast:

```
MATCH (city:City) RETURN city
```

- Wyświetlanie miast z mieszkańcami:

```
MATCH (person:Person)-[:LIVES_IN]->(city:City)
      RETURN city.name AS cityName,
COLLECT(name: person.name, age: person.age) AS residents;
```

- Dodawanie osób:

```
CREATE (:Person name: $name, age: $age)
```

- Dodawanie miast:

```
CREATE (:City name: $name)
```

- Dodawanie relacji pomiędzy osobami a miastami:

```
MATCH (p:Person name: $person_name), (c:City name: $city_name)
      CREATE (p)-[:LIVES_IN]→(c)
```