System Rekomendacji Książek

Dokumentacja Projektu

Przetwarzanie Danych w Chmurze Obliczeniowej

Julia Przeździk

Grudzień 2025

Spis treści

1 Informacje Ogólne

1.1 Cel Projektu

Celem projektu jest stworzenie systemu rekomendacji książek opartego o algorytm **Collaborative Filtering** wykorzystującego grafową bazę danych Neo4j. System analizuje preferencje użytkowników i na podstawie podobieństw proponuje książki.

1.2 Technologie

Stack Technologiczny

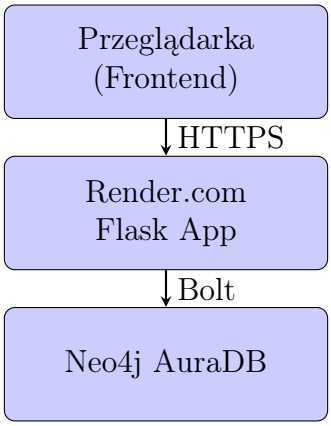
- **Backend:** Python 3.9 + Flask (RESTful API)
- **Baza danych:** Neo4j AuraDB Professional (DBaaS)
- **Frontend:** SPA - HTML5, CSS3, JavaScript ES6
- **Deployment:** Render.com (Cloud PaaS)

1.3 Linki

- **Live Demo:** https://pdwcho-project.onrender.com
- **GitHub:** https://github.com/juliaprzezdzik/pdwcho_project.git

2 Architektura Systemu

2.1 Diagram Architektury



Rysunek 1: Architektura systemu

2.2 Model Danych

Schemat Grafowy

Węzły:

- `User`: username, name
- `Book`: isbn, title, author

Relacje:

- `[:READS {rating: Float}]`

3 Implementacja

3.1 Backend - Flask API

Metoda	Endpoint	Opis
GET	/	Strona główna
GET	/hello	Healthcheck
GET	/recommendations/<user>	Rekomendacje

Tabela 1: Endpointy API

3.2 Algorytm Collaborative Filtering

1. **Znajdź podobnych użytkowników** - według wspólnych książek
2. **Oblicz similarity score** - liczba wspólnych książek
3. **Generuj rekomendacje** - książki podobnych użytkowników
4. **Ranking** - sortuj według oceny i popularności

3.3 Frontend - SPA

Funkcjonalności:

Responsywny interfejs

Animacje CSS3

Error handling

4 Deployment

5 Testy i Walidacja

5.1 Graph Database vs SQL

SQL (JOIN-heavy):

Test Case	Input	Output	Status
Rekomendacje użytkownika	AnnaK	5 książek	
Nieistniejący użytkownik	XYZ	Pusta lista	
Healthcheck	/hello	Status 200	
Responsiveness	375px	OK layout	

Tabela 2: Testy funkcjonalne

```sql
SELECT b.title FROM users u1
JOIN reads r1 ON u1.id = r1.user_id
JOIN reads r2 ON r1.book_id = r2.book_id
JOIN users u2 ON r2.user_id = u2.id
WHERE u1.username = 'AnnaK'
```

Neo4j Cypher:

```
MATCH (me:User {username: 'AnnaK'})-[:READS]->()
      <-[:READS]-(other)
MATCH (other)-[:READS]->(recBook)
RETURN recBook.title
```

6 Instrukcja Uruchomienia

```
# Clone repository
git clone https://github.com/username/PDCHO_PROJEKT.git
cd PDCHO_PROJEKT

# Virtual environment
python3 -m venv venv
source venv/bin/activate

# Instalacja
pip install -r requirements.txt

# Zmienne srodowiskowe
export NEO4J_URI="neo4j+s://..."
export NEO4J_USER="neo4j"
export NEO4J_PASSWORD="..."

# Uruchomienie
python app.py
```

Podsumowanie

Projekt prezentuje system rekomendacji wykorzystujący Neo4j i nowoczesne technologie chmurowe.