**Instytut Informatyki  
Wydział Nauk Ścisłych i Technicznych  
Uniwersytet Rzeszowski**

**Przedmiot:**

**Nierelacyjne Bazy Danych**

**Dokumentacja projektu:**

**ZODB-Film-Manager**

**Wykonali: Mateusz Bocak i Gabriela Bieniek**

**Prowadzący: dr inż. Dariusz Bober**

**Rzeszów 2025**

Spis treści

[Temat i cel projektu 3](#_Toc197283895)

[Techniczne aspekty projektu 3](#_Toc197283896)

[Funkcjonalności aplikacji 3](#_Toc197283897)

[Wykorzystane technologie 3](#_Toc197283898)

[Działanie ZODB 4](#_Toc197283899)

[Instrukcja instalacji 5](#_Toc197283900)

[Wymagania wstępne 5](#_Toc197283901)

[Instalacja ręczna 5](#_Toc197283902)

[Uruchamianie aplikacji 5](#_Toc197283903)

[Uruchamianie za pomocą Dockera 6](#_Toc197283904)

[Użytkowanie 6](#_Toc197283905)

[Zalety ZODB w projekcie 6](#_Toc197283906)

[Rozwiązywanie problemów 7](#_Toc197283907)

## Temat i cel projektu

ZODB Film Manager to aplikacja demonstracyjna przezentująca możliwości ZODB (Zope Object Database) – obiektowej bazy danych dla Pythona.

Aplikacja umożliwia zarządzanie kolekcją filmów, reżyserów, aktorów i gatunków, wykorzystując przezroczystą trwałość i relacje obiektowe oferowane przez ZODB.

## Techniczne aspekty projektu

### Funkcjonalności aplikacji

* Dodawanie, edytowanie i usuwanie filmów, reżyserów, aktorów i gatunków.
* Ocenianie i recenzowanie filmów.
* Organizowanie filmów według gatunków z zachowaniem dwukierunkowych relacji,
* Wyszukiwanie filmów po tytule, reżyserze, aktorze lub gatunku.
* Wydajne indeksowanie za pomocą OOBTree w ZODB, zapewniające szybkie wyszukiwanie.
* Przezroczysta trwałość obiektów bez potrzebny korzystania z ORM lub SQL.

### Wykorzystane technologie

1. Python: Główny język programowania używany do implementacji aplikacji.
2. ZODB: Obiektowa baza danych, która umożliwia przechowywanie obiektów Pythona bez potrzeby serializacji lub mapowania na tabele relacyjne.
3. Flask: Lekki framework webowy używany do budowy interfejsu użytkownika aplikacji.
4. OOBTree: Struktura danych z ZODB używana do wydajnego indeksowania i wyszukiwania.
5. Docker: Narzędzie do konteneryzacji, które umożliwia łatwe uruchamianie aplikacji w izolowanym środowisku.
6. Virtual Environment (venv): Moduł Pythona używany do tworzenia izolowanego środowiska dla zależności projektu.

## 

## Działanie ZODB

ZODB to obiektowa baza danych dla języka Python, która umożliwia przechowywanie obiektów Pythona bez konieczności ich serializacji lub mapowania na tabele relacyjne. Jest to baza typu NoSQL, która działa w pełni w oparciu o obiekty – dane są przechowywane w takiej samej formie, w jakiej są reprezentowane w kodzie.

Kluczowe zalety ZODB:

* Przezroczysta trwałość: Obiekty Pythona są automatycznie zapisywane i odczytywane z bazy danych bez potrzeby jawnego zarządzania tym procesem.
* Relacje obiektowe: Obiekty w ZODB mogą bezpośrednio odnosić się do siebie za pomocą referencji.
* Brak schematu: ZODB nie wymaga z góry zdefiniowanego schematu, co czyni ją elastyczną i łatwą do rozbudowy. Można dynamicznie dodawać nowe atrybuty do obiektów lub zmieniać ich strukturę bez konieczności migracji bazy danych.
* Wydajne indeksowanie: ZODB wykorzystuje strukturę danych OOBTree do indeksowania, co umożliwia szybkie wyszukiwanie i sortowanie.
* Transakcje ACID: ZODB obsługuje transakcje zgodne z zasadami ACID (Atomicity, Consistency, Isolation, Durability). Wszystkie zmiany w bazie danych są stosowane atomowo – albo wszystkie zmiany są zatwierdzane, albo żadna z nich. Dzięki temu dane są zawsze spójne.
* Obsługa wersjonowania: ZODB umożliwia wersjonowanie danych, co pozwala na przechowywanie różnych wersji obiektów i ich późniejsze przywracanie.

## 

## Instrukcja instalacji

### Wymagania wstępne

* Python 3.8 lub nowszy
* Moduł wirtualnego środowiska (venv)
* Wymagane zależności Pythona (wymienione w pliku requirements.txt)

### Instalacja ręczna

Ręczna instalacja (Linux i Windows)

1. Sklonowanie repozytorium

git clone <https://github.com/JustFiesta/ZODB-film-manager>  
cd project

1. Utworzenie i aktywowanie wirtualnego środowiska

Na Linuxie:

python3 -m venv venv  
source venv/bin/activate

Na Windowsie:

python -m venv venv  
venv\Scripts\activate

1. Instalacja zależności

pip install -r requirements.txt

1. Zainicjowanie bazy danych

python -m app.database

## Uruchamianie aplikacji

Należy się upewnić, że środowisko wirtualne jest aktywowane. Uruchomienie aplikacji następuje za pomocą pliku run.py

Na Linuxie:

python3 run.py

Na Windowsie:

python run.py

Aplikacja będzie dostępna pod adresem <http://127.0.0.1:5000>

### Uruchamianie za pomocą Dockera

1. Zbudowanie obrazu Dockera

docker build -t zodb-film-manager .

1. Uruchomienie konteneru Dockera

docker run -p 5000:5000 zodb-film-manager

1. Uzyskanie dostępu do aplikacji

Aplikacja będzie dostępna pod adresem <http://localhost:5000>

## Użytkowanie

1. Otwórz aplikację w przeglądarce internetowej
2. Użyj paska nawigacyjnego, aby przeglądać filmu, reżyserów, aktorów i gatunki.
3. Dodawaj nowe filmy, edytuj istniejące lub usuwaj wpisy.
4. Wyszukuj filmy za pomocą paszka wyszukiwania.
5. Przeglądaj szczegółowe relacje między obiektami, takimi jak filmy i ich gatunki lub aktorzy.

## ZODB w projekcie

W projekcie ZODB-film-manager ZODB jest używana do:

* Przechowyania obiektów filmów, reżyserów, aktorów i gatunków.
* Utrzymywania relacji między obiektami (np. film → aktorzy, film → gatunki).
* Wydajnego wyszukiwania danych za pomocą OOBTree.
* Zarządzania transakcjami, co zapewnia spójność danych nawet w przypadku błędów.

### Zalety ZODB w projekcie

1. Uproszczone zarządzanie danymi: Brak potrzeby pisania zapytań SQL lub zarządzania złożonymi mapowaniami ORM.
2. Naturalne relacje obiektowe: Obiekty, takie jak reżyserzy i gatunki, utrzymują bezpośrednie odniesienia do siebie.
3. Skalowalność: Wydajne indeksowanie za pomocą OOBTree zapewnie szybkie wyszukiwanie nawet przy dużych zbiorach danych.
4. Elastyczność: Brak schematu pozwala na łatwe modyfikacje i dodawanie nowych elementów do modelu danych.
5. Spójność: Transakcje ACID zapewniają, że wszystkie zmiany są stosowane atomowo.

## 

## Rozwiązywanie problemów

Jeśli aplikacja nie uruchamia się, należy się upewnić że plik bazy danych (movies.fs) został zainicjowany i jest dostępny.

W przypadku problemów z zależnościami należy sprawdzić, czy wszystkie pakiety wymienione w requirements.txt zostały zainstalowane.