**Wymagana dokumentacja   
dla projektów realizowanych w ramach zajęć   
Inżynierii oprogramowania   
w semestrze zimowym roku akademickiego 2022/2023**

## Charakterystyka oprogramowania

1. MovieClass
2. System rekomendacji filmów
3. Aplikacja ma za zadanie proponować filmy podobne do wprowadzanego tytułu przez użytkownika. Zawiera zdecydowana większość filmów, które wydane zostały od 1980 do 2022 roku. Na stronie widoczne są obiekty, w których znajduje się okno do wpisywania tytułu filmu, dla którego byśmy chcieli zarekomendować tytuły. Dodatkowo została zaimplementowana lista rozwijana podpowiadająca tytuł filmu na podstawie bazy danych. Obok znajdują się kolejne okienka, w których wczytane są tytuły, opisy i plik z obrazkiem plakatu filmu.

## Prawa autorskie

1. Autorzy:

* Piotr Kapusta
* Eryk Kulengowski
* Jakub Kułak

1. Kod źródłowy aplikacji jest otwarty pod warunkiem wskazania twórców.

[Creative Commons — Attribution 4.0 International — CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)

## Specyfikacja wymagań

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identyfikator | Nazwa | Opis | Priorytet | Kategoria |
| PF | Porpozycje Filmów | System musi pokazywać minimum 3 podobne propozycje | 1 | Funkcjonalne |
| OF | Opisy do filmów | Pod propozycjami widoczne powinny być krótkie opisy filmów | 2 | Funkcjonalne |
| ZF | Zdjęcia do filmów | Do tytułów filmów dołączone powinny być plakaty | 2 | Funkcjonalne |
| LR | List rozwijana podczas wpisywania | Wpisując tytuł strona powinna proponować  dostępne filmy | 2 | Funkcjonalne |
| MN | Menu z wyborem podstron | Dostępna jest zakładka główna związana z działaniem rekomendacji filmu i podstrona z informacjami o projekcie. | 3 | Funkcjonalne |
| PO | Propozycje na podstawie opisu | Opis filmu dodany jako cecha filmu, na podstawie której proponowane będą kolejne filmy – **blokada mocy obliczeniowej** | 3 | Funkcjonalne |

## Architektura oprogramowania

1. Technologie:

* Python
* MongoDB
* HTML
* CSS
* JavaScript
* Pycharm Community Edition
* Spyder 5

1. Biblioteki:

* Requests
* Numpy
* Pandas
* Sckikit-learn
* Bootstrap 5
* Datetime
* Flask

## Testowanie aplikacji

### Scenariusze testów

Strona została przetestowana pod kątem kontroli błędów i podstawowego działania. Struktura strony została zbudowana aby była intuicyjna i działała na każdym urządzeniu. Wszystkie elementy strony zostały sprawdzone w działaniu w każdej możliwej konfiguracji.

#### Kontrola błędów

Strona działa na podstawie bazy danych tytułów filmów wcześniej pozyskanych za pomocą webscrapingu ze strony themoviedb.org. W przypadku wpisania tytułu, który nie istnieje w bazie danych, została zastosowana kontrola błędów, wyświetlająca odpowiedni obrazek w miejsce plakatu filmu oraz brak opisu.

Application, table

Description automatically generated

#### Lista podpowiadająca

W celu uniknięcia wpisywania tytułów niedostępnych w bazie danych została zaimplementowana lista rozwijana, która na podstawie listy wszystkich tytułów podpowiada tytuł i rok filmu na podstawie tekstu, który wpiszemy.

Graphical user interface

Description automatically generated

#### Link do obrazka plakatu filmu

W bazie danych znajduję się zmienna, w której zaszyty jest link do obrazka plakatu. W przypadku, gdy dany film nie ma przypisanego linku do pliku jpg, zostaje wyświetlany obrazek zastępczy.

Application

Description automatically generated with medium confidence

## Github

https://github.com/WzrProjekt/RecommendationSystem