

Aplikacja wystawia API dla aplikacji z bazą filmów, typu Filmweb.

Pakiet: DOMAIN

1. **interfejsy:** IService - zapewnia obsługę "bazy danych" aplikacji typowe metody CRUD, przyjmuje generyczny typ danych,
2. **klasy:**
 - Actor - reprezentacja Aktorów w aplikacji, każde pole posiada gettery i settery, wszystkie pola są prywatne

Pole	Typ danych
Id	Int
Name	String
Surname	String
ActorMovies	ArrayList<Movie>

- Movie - reprezentacja Filmów w aplikacji, implementuje interfejs IService, każde pole posiada gettery i settery, wszystkie pola są prywatne

Pole	Typ danych
Name	String
Year	String
Rate	Rate
RateCounter	Static int; początkowa wartość=1
Id	Int
Description	String
SumofRates	Static double
Comments	ArrayList<Comment>
Actors	ArrayList<Actor>

- Rate – reprezentacja oceny w aplikacji, można do niej dodać walidację wartości

Pole	Typ danych
rate	double

- Comment – reprezentacja komentarzy w bazie, każde pole posiada gettery I settery, wszystkie pola są prywatne

Pole	Typ danych
Id	Int
Author	String
Text	String

- ActorsService – klasa operująca na “bazie danych” którą na potrzeby projektu jest ArrayLista Posiada wszystkie metody CRUD, wszystkie pola są prywatne

Pole	Typ danych
Actors	Static ArrayList<Actor>
CurrentId	Static int, początkowa wartość = 1

- MoviesService – klasa operująca na “bazie danych” którą na potrzeby projektu jest ArrayLista Posiada wszystkie metody CRUD, wszystkie pola są prywatne

Pole	Typ danych
Movies	Static ArrayList<Movie>
CurrentId	Static int, początkowa wartość = 1

Pakiet: REST

Pakiet posiada klasy RESTowe, deklaracja każdej klasy posiada adnotacje dotycząca ścieżki pod jaką dostępne jest jej API

1. Klasy:

- *ApplicationConfig*

Klasa konfiguracyjna, dodaje do adresu bazowego /rest tak, że pełna ścieżka do aplikacji wygląda:

localhost:8080/RESTfilms/rest/~

Ścieżka bazowa: /movies

- *Movies*

klasa zawiera w sobie reference do klasy CRUDowej, MociesService

Metoda HTTP	Ścieżka	MediaType	Zwracany typ danych	Nazwa metody	Parametry	Opis
GET	/id}	APPLICATION_JSON	Response	Get()	Int id	Metoda zwraca film o zadanym id
POST	brak	APPLICATION_JSON	Response	Add()	Movie m	Metoda dodaje film do "bazy" format JSON: <pre>{ "name": "nazwa", "year": "rok_ prod", "description": "op is" }</pre>
PUT	/id}	APPLICATION_JSON	Response	Update()	Int id, Movie m,	Metoda aktualizuje film o podanym id, format JSON: jak w add();
DELETE	/id}	APPLICATION_JSON	Response	Delete	Int id	Metoda usuwa film o podanym id, jeżeli nie ma takiego filmu zwraca błąd http 404
GET	/movie} /comments	APPLICATION_JSON	Response	GetMovies	Int movieId	Metoda zwraca listę komentarzy danego filmu

POST	/{"id"}/comments	APPLICATION_JSON	Response	AddComment	Int movieId, Comment comment	Metoda dodaje komentarz do filmu o zadanym id <pre>{ "name": "<u>name</u>", "text": "<u>tresc</u>" }</pre>
POST	/{"id"}/rate	APPLICATION_JSON	Response	AddRate	Int id, Rate raate,	Metoda dodaje ocene do filmu o zadanym id, wyswietla tez ilosc głosów <pre>{ "rate": "<u>ocena</u>" }</pre>
GET	/{"movieId"}/actors	APPLICATION_JSON	ArrayList<Actor>	GetMovieActors()	Int movieId	Metoda zwraca liste aktorów danego filmu

Ścieżka bazowa: /actors
Klasa: Actors

klasa zawiera w sobie reference do klasy CRUDowej, ActorsService

Metoda HTTP	Ścieżka	MediaType	Zwracany typ danych	Nazwa metody	Parametry	Opis
GET		APPLICATION_JSON	Response	Getall()		Metoda zwraca liste aktorow
GET	/{"id"}	APPLICATION_JSON	Response	Get()	Int id	Metoda zwraca aktora z danym id
POST		APPLICATION_JSON	Response	Add()	Acotr actor	Metoda dodaje aktora do bazy <pre>{ "name": "<u>nazwa</u>", "surname" : "<u>nazwisko</u>" }</pre>
PUT	/{"id"}	APPLICATION_JSON	Repsonse	Update()	Int id, Actor actor	Metoda aktualizuje aktora, json jak w POST na

						ścieżce bazowej
DELETE	/ {id}	APPLICATION_JSON	Response	Delete	Int id	Metoda usuwa aktora o zadanym id
GET	/ {id} /movies	APPLICATION_JSON	ArrayList<Movie>	GetMovieActors	Int actorid	Metoda zwraca filmy danego aktora
POST	/ {actorId} / {movieId}	APPLICATION_JSON	Response	SetMovie	Int actorId, int movieId	Metoda przypisuje aktora do filmu, operuje na id jednego I drugiego

Metody GET I DELETE nie przyjmują żadnych danych