

# Movie Recommender

## Ce este aplicația **Movie Recommender**?

**Movie Recommender** este o aplicație creată în Java, cu ajutorul căreia utilizatorul primește recomandări de filme pe baza unor întrebări și răspunsuri. Atât întrebările, cât și răspunsurile sunt salvate într-un fișier XML, structurat într-un arbore. Opțional, utilizatorul poate folosi un fișier XML pentru prescurtările întrebărilor și răspunsurilor. De exemplu: **Ce tip de filme îți plac?**  
→ q1.

Întrebări noi sau răspunsuri noi, cât și prescurtări noi, pot fi adăugate direct în fișierele XML, respectând structura dată, iar aplicația va ști să pună întrebări noi, cu răspunsuri noi, nefiind nevoie de schimbare de cod. De aici rezultă caracterul dinamic al programului.

## Cum poți folosi aplicația?

Având în vedere că proiectul este făcut în Eclipse, iar utilizatorul interacționează cu aplicația prin consolă, este nevoie de **JDK**, **Eclipse**, și aplicația, care poate fi găsită pe: <https://github.com/andreimihut95/Movie-Recommender/>. Se pornește Eclipse, se deschide clasa **Main** din pachetul **com.furiapolitehnicii.main**. Rulăm programul, utilizatorul poate interacționa cu aplicația prin consolă. Trebuie specificate cele 2 fișiere XML, primul fișier fiind fișierul cu întrebări și răspunsuri, iar al doilea, opțional, fiind fișierul de mapări, pentru prescurtări. După care, utilizatorul primește o serie de întrebări la care trebuie să răspundă, prin tastarea cifrei aferente răspunsului. La final, utilizatorul primește un set de recomandări, afișate tot în consolă.

## Cum funcționează aplicația?

Există o clasă **MovieRecommender**, care primește ca și parametrii 2 fișiere (unul fiind opțional, existând doi constructori, unul cu un parametru și celălalt cu doi parametri), primul fișier fiind fișierul XML cu întrebări și răspunsuri, iar al doilea fișier fiind un fișier de mapări pentru prescurtări. Pe lângă cei doi membri ai clasei, mai există încă doi membri, unul reprezintă o listă

de filme(recomandări), pe care clasa o construiește la execuția metodei **run()**, iar celălalt membru o mapă(HashMap), pe care clasa o construiește la execuția metodei **loadMapFile()**. Fișierul XML pentru mapări trebuie să respecte următoare structură:

- numele elementul root al fișierului nu are relevanță(în exemple a fost pus **maps**).
- copiii elementului root trebuie sa fie elemente cu numele de **map**, ce are un atribut cu numele de **id**, având valoarea unei întrebări sau răspuns din celălalt fișier XML, iar ca și conținut text, prescurtarea dorită de creatorul fișierului.
- elementul **map** nu poate sa aibă copii.

Dacă aplicația o sa aibă ca parametru și un fișier de mapări, atunci fișierul XML pentru întrebări și răspunsuri trebuie să conțină prescurtarea pentru fiecare întrebare și pentru fiecare răspuns, din fișierul de mapări.

Metoda **run()** din cadrul clasei este cea care se ocupă de parsarea fișierului XML, și de interacțiunea cu utilizatorul. Fișierul XML, trebuie să respecte următoarea structură:

- numele elementul root al fișierului nu are relevanță(în exemple a fost pus **movies**).
- copilul elementului root trebuie sa fie neapărat un element cu numele **question** și atribut **question**, având valoare unei întrebări dată de creatorul fișierului.
- copilul elementului **question**, trebuie, pentru ca XML să fie valid, să aibă ca și copil elemente de tip **answer** sau **finalAnswer**, fiecare cu atributul **answer**, iar la valoare fiind un răspuns dat de creatorul fișierului.
- **answer** poate avea ca și copil un singur element de tip **question**.
- **finalAnswer** poate avea ca și copii elemente de tip **recommendation**.
- **recommendation** nu poate avea copii, are doar ca attribute **title**, **imdbURL**, **imdbRating**, **photo**, și are ca și conținut text o descriere a filmului.

Având fișierul cu structura de mai sus, și împreună cu răspunsurile date de utilizator prin consolă, metoda parcurge arborele din fișier și ajunge la un set de recomandări pe care le salvează într-o listă(membrul clasei). Tehnica utilizată pentru a parcurge arborele este "forward chaining". Aceasta constă în determinarea concluziilor (frunzele arborelui) pornind de la premise (răspunsurile date de utilizator). Pentru afișarea recomandărilor, clasa dispune de o metodă numită **viewRecommendations()**, care afișează recomandările salvate.

## Tehnologii, API-uri, Framework-uri, IDE-uri folosite

Următoarele au fost folosite pentru crearea aplicației: **Eclipse**, **Maven**, **Java**, **Git**, **XML**, **xPath**, **DOM4j** și **Jaxen**.