

# Raport tehnic: YoMovie

Munteanu Rares Costin, Guraliuc Iulian Bogdan, Cretu Marian Codrin, Manoleasa Tudor

Aprilie 2019

## 1 Abstract

YoMovie este o aplicatie ce va oferi utilizatorului posibilitatea de a urmari filme in mod interactiv.

## 2 Introducere

In cadrul proiectului YoMovie, se va construi o aplicatie ce va implementa un sistem client-server multithread, ce va fi folosita pentru a afisa sub forma unui site web informatii despre filme, actori si pentru a reda filme intr-un mod interactiv.

Interactiunea provine din posibilitatea de a alege in timpul filmului vizionat, diverse scenarii de continuare, astfel deznodamantul fiind diferit in functie de scenariul ales.

## 3 Module

### 3.1 Login/Register

#### 4.1.1

Creeaza noi utilizatori sau permite utilizatorilor existenti sa isi acceseze contul.

Validarea se va face cu ajutorul bazei de date, unde se vor afla atat username-ul, cat si parola utilizatorului daca doreste sa se autentifice.

In cazul in care doreste sa se inregistreze, noile credentiale vor fi inserate in baza de date.

### 3.2 Filme

Pe pagina principala se vor afla filme, ce vor putea fi filtrate dupa gen, actori participanti, etc.

### 3.3 Utilizator

Acesta va putea fi utilizator guest, ce nu va avea acces la toate functionalitatile aplicatiei sau utilizator logat, ce avea acces la toate functionalitatile.

## 4 Tipul Utilizatorului

### 4.1 Utilizator guest

Utilizatorul neautentificat va putea accesa doar fatada publica a aplicatiei.

Paginile incluse:

- Pagina de start
- Pagina de login
- Pagina de register

Implicit, utilizatorul neautentificat va putea doar accesa informatii despre filme, se va putea loga sau inregistra si de asemenea va putea viziona filme.

### 4.2 Utilizator autentificat

### 4.3

Utilizatorul autentificat va putea accesa orice pagina a aplicatiei.

Paginile incluse:

- Pagina de start
- Pagina de login
- Pagina de register
- Pagina de upload filme

Implicit, utilizatorul autentificat va putea accesa atat informatii despre filme, atat si viziona sau adauga filme.

## 5 Actori

Actor	Descriere
Guest	un utilizator care nu este logat pe site
User	un utilizator care este logat pe site
Film	Informatii despre acesta si posibilitatea de vizionare

## **6 Use Cases**

### **6.1 Actor Guest**

#### **6.1.1 Ca si Guest as dori sa imi creez un cont**

Un utilizator nou ar trebui sa aiba posibilitatea sa isi creeze un cont nou.

#### **6.1.2 Ca si utilizator as dori sa imi resetez parola**

Daca un utilizator si-a pierdut parola, ar trebui sa poata sa si-o reseteze printr-un formular public.

#### **6.1.3 Ca si utilizator as dori sa ma loghez**

Ar trebui sa existe un formular prin care utilizatorul sa poate sa se autentifice.

### **6.2 Actor User**

#### **6.2.1 Ca si utilizator as dori sa ma deconectez**

Un utilizator ar trebui sa se poata deconecta.

#### **6.2.2 Ca si utilizator as dori sa aflu informatii despre filme**

Ar trebui sa existe un sumar pentru fiecare film.

#### **6.2.3 Ca si utilizator as dori sa caut un film**

Ar trebui sa existe o bara de cautare.

#### **6.2.4 Ca si utilizator as dori sa vizionez un film**

Ar trebui sa existe posibilitatea de a viziona un film la alegere.

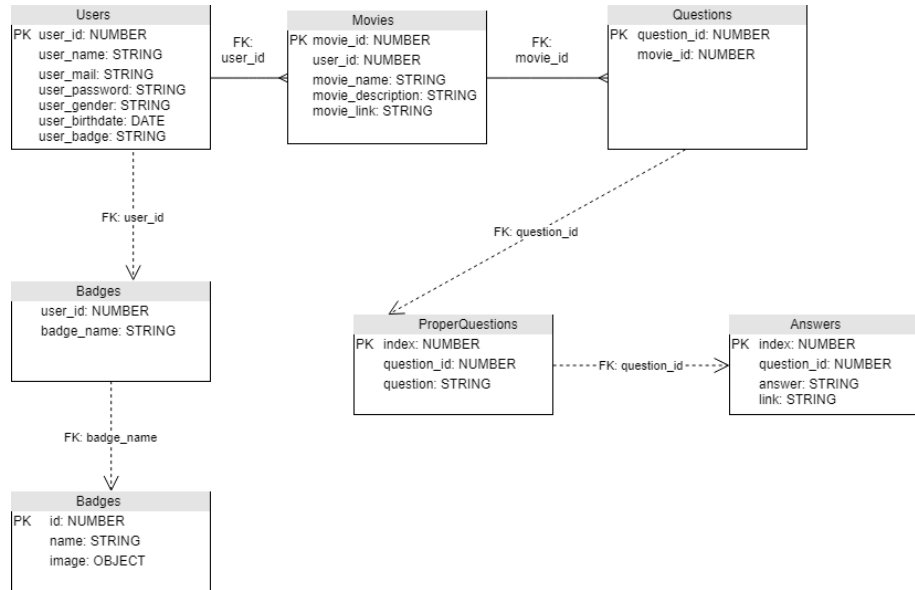
#### **6.2.5 Ca si utilizator as dori sa incarc un film**

Ar trebui sa existe posibilitatea de a incarca un film printr-un formular.

#### **6.2.6 Ca si utilizator as dori sa filtrez filmele**

Ar trebui sa existe posibilitatea de a sorta filmele dupa diverse criterii.

## 7 Diagrama bazei de date



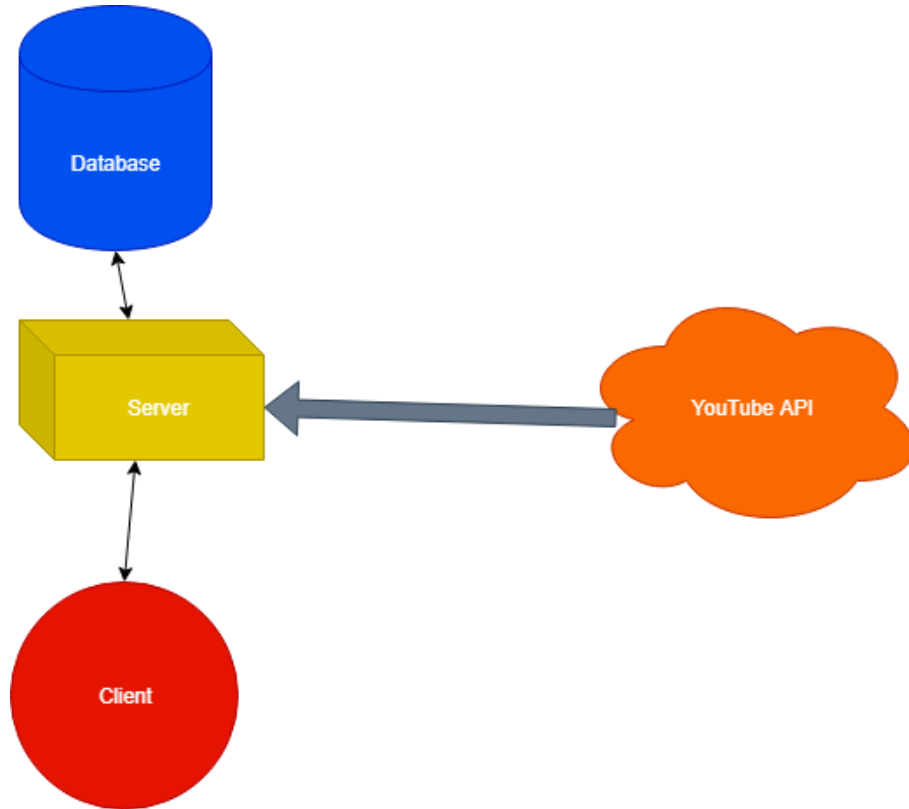
### 7.1 Descriere

Baza de Date va fi realizata in MySql conform imaginii de mai sus.

Tabelele din poza sunt normalizate, neexistand posibilitatea de a avea duplicate. Tipurile de date vor fi String-uri, Integer-uri care vor memora link-urile specifice filmelor/informatiilor despre utilizator, etc.

Tabelele principale vor fi reprezentate de Movies, Questions si Answers care vor deservi functionalitatii principale a aplicatiei.

## 8 Diagrama Master

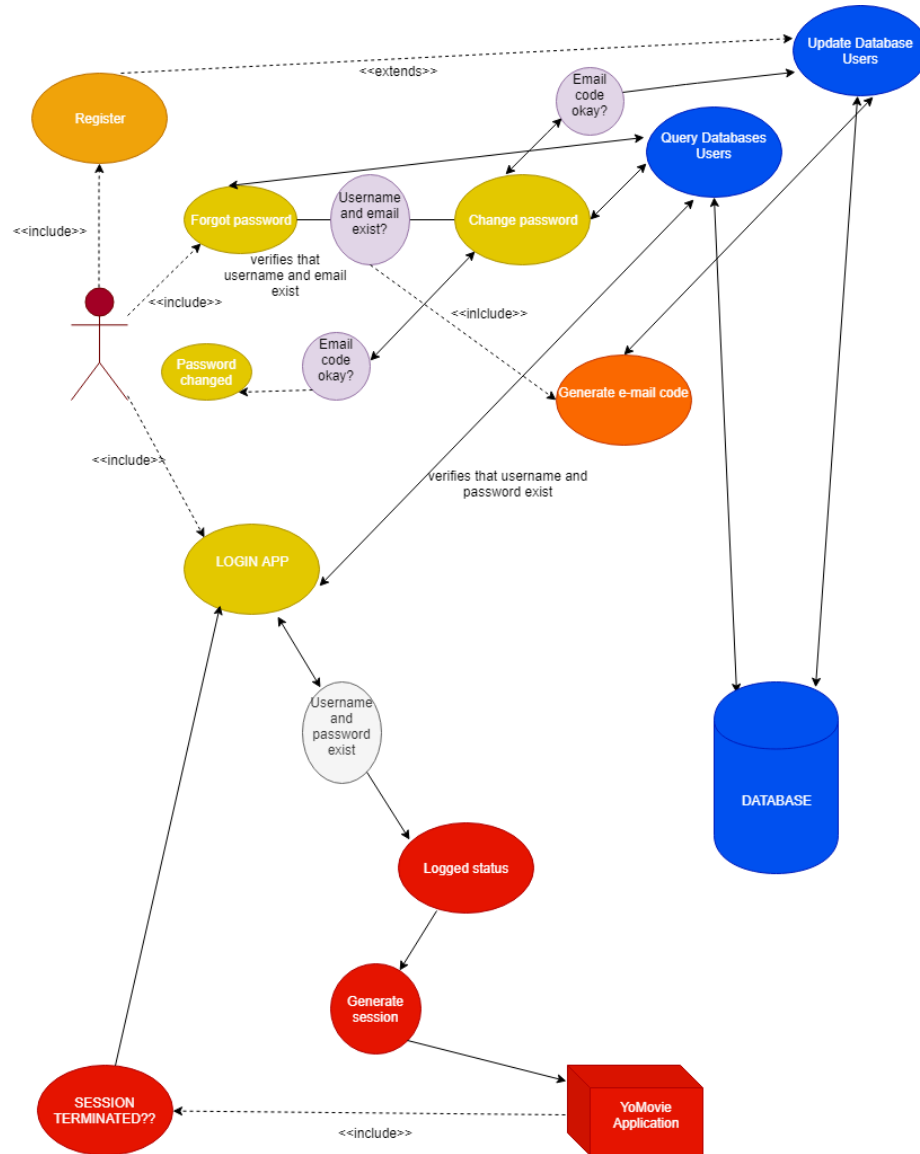


Din punctul de vedere al utilizatorului, acesta interactioneaza doar prin partea de view cu aplicatia. Din punctual de vedere al dezvoltatorilor, Ilustreaza modelul MVC al aplicatiei, care face apel si la API extern, cum ar fi YouTube API prin paradigma de programare REST.

Application Programming Interfaces (APIs) reprezinta un set de proceduri si tooluri care sunt folosite pentru dezvoltarea de aplicatii care fac apel la servicii externe.

## 9 Diagrame Use Case

### 9.1 Guest sau User - login/register



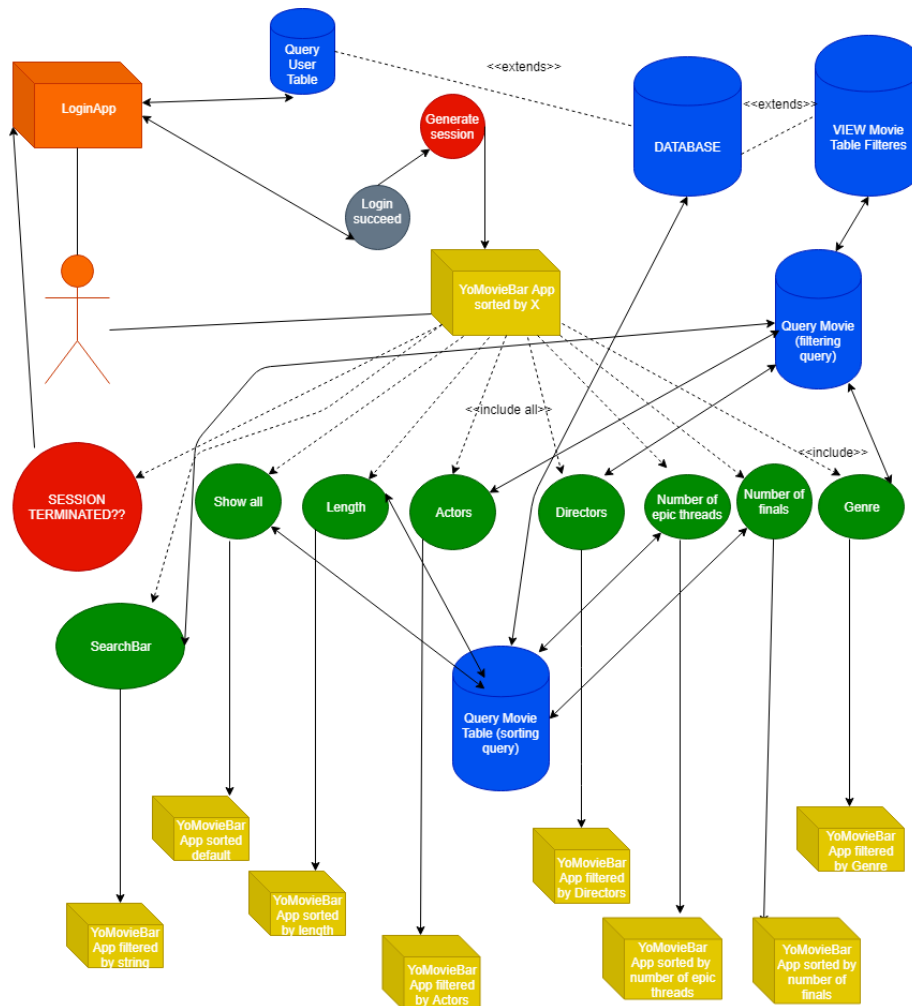
Utilizatorul, in aceasta faza, are la dispozitie trei aplicatii. Aplicatia de login permite introducerea unui nume de utilizator si a unei parole. Modulul interogheaza baza de date utilizatori pentru a afla daca username-ul si parola date exista. In cazul in care logarea reuseste, utilizatorul va primi un ID de

sesiune, folosit pentru persistent paginii dupa logare pentru un timp bine determinat(sa spunem de ordinal catorva ore).

In cazul in care sesiunea expira, utilizatorul va fi dus inapoi la pagina de logare, pentru resolicitarea procesului de logare.

Utilizatorul are optiunea de a solicita schimbarea parolei, in cazul in care acesta o uita. Isi introduce username-ul si email-ul, iar interogarea bazei de date user va valida aceste date. Aplicatia va trimite la adresa indicata de email un cod generat, si va actualiza baza de date cu acel cod generat. La pasul de schimbare al parolei, utilizatorul va introduce codul trimis prin email (prin SMTP), parola noua si reconfirmarea parolei. Dupa apasarea butonului de submit, BD va da update la noile date, iar schimbarea parolei va avea loc cu success. Utilizatorul are optiunea sa se inregistreze cu un username, parola si un email. In aceste conditii, se va face update la tabela de Users.

## 9.2 Functionalitatile barei de navigare



Pentru aceasta parte, utilizatorul poate sa fie logat sau nu.

In cazul in care utilizatorul opteaza pentru logare, se valideaza logarea (in cazul in care logarea nu reuseste, se va afisa un mesaj de eroare, si se reafiseaza pagina de logare), se genereaza sesiunea. Prin folosirea navbar, utilizatorul poate filtra dupa diverse criterii: Actors, Directors, Genres, sau dupa stringul transmis prin bara de search. Dupa criteriile din urma, se va crea un view intermediary, in care se va retine rezultatul acestei filtrari (din motivul ca filtrarea trebuie sa ramana persistent pe tot parcursul sesiunii: spre exemplu daca apas butonul filter Actors datele trebuie filtrate dupa criteriul de actori prestabilit).

Se va apela interogarea de filtrare dupa cum am scris mai sus, si se va afisa utilizatorului acele filme ordonate dupa criteriile de filtrare.



### 9.3 Film



Pentru acest use-case, este necesara autentificarea utilizatorului. Se reiau procedurile in ceea ce priveste validarea utilizatorului, generarea sesiunii si verificarea validitatii sesiunii.

Dupa logare, utilizatorul va putea accesa optiunile de filtrare din NavBar si va putea vizualiza o lista de filme(in cazul reprezentarii noastre FrontEnd, se vor afisa mai multe carduri cu imagini corespunzatoare; acest front-end va fi

generat dinamic in functie de sortari). Utilizatorul va da click pe acel card si va avea la dispozitie doua optiuni: de a evalua filmul ( de la pana la stele) sau de a viziona filmul.

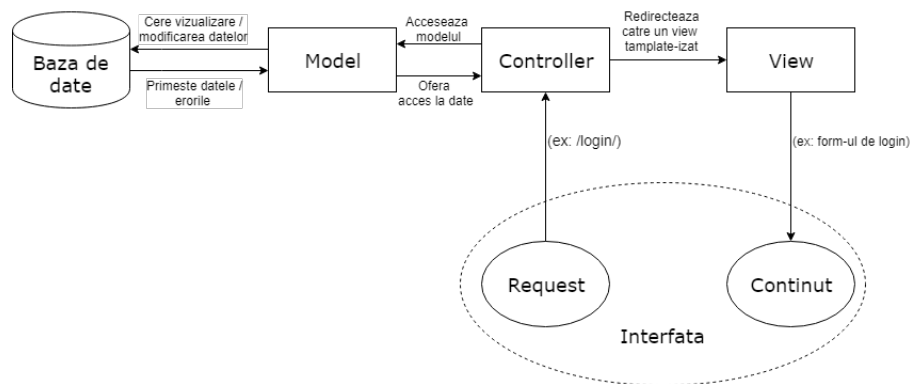
In cazul in care utilizatorul doreste sa dea rate la film, acesta va selecta reprezentarea SVG corespunzatoare, se va face update in baza de date Movies cu ratingul respective(se adauga la medie si se calculeaza media). In cazul in care utilizatorul doreste sa vizioneze un film, va apasa pe iconita de Play aferenta fiecarui card, se va face apel la YouTubeAPI (la extractiile specifice) si, corespunzator linkului oferit ca raspuns din baza de date, se va face apel la acest API si se va rula un film in format embedded. In cazul in care timpul vizionarii primei parti a expirat, utilizatorul este intrebat prin fereastra de Choose Scenario ce scenariu doreste sa aleaga. Se va interoga baza de date Scenarios pentru a afla bucată de film corespunzătoare filmului curent si a scenariului ales si se va rula urmatorul film. In cazul in care nu mai exista scenario aferente, se va reveni pe pagina principala.

De asemenea, utilizatorul va mai avea optiunea de a adauga filme. Acesta dispune de un formular in care poate adauga diverse linkuri de pe YouTube, iar baza de date Movies va fi adaptata in acest sens. Pentru imaginea filmului s-a folosit un api extern: imgur api, pe care noi il apelam dandu-i o imagine din calculator si acesta returnandu-ne un link ce este inserat in baza de date.

## 10 Descriere implementare

Am folosit MVC pe front-end și pe back-end. Ca și în cazul altor modele de design software, MVC exprimă "miezul soluției" unei probleme, permițându-i să fie adaptată pentru fiecare sistem.

- Modelul este componenta centrală a patternului. Exprimă comportamentul aplicației în ceea ce privește domeniul problemei, independent de interfața cu utilizatorul. Administrează direct datele, logica și regulile aplicației.
- O vizualizare poate fi orice reprezentare de ieșire a informațiilor, cum ar fi o diagramă sau o diagramă. Sunt posibile mai multe vizualizări care prezintă aceleași informații, cum ar fi o diagramă de bare pentru gestionare și o vizualizare tabelă pentru contabili.
- A treia parte sau secțiune, controlerul, acceptă intrarea și o transformă în comenzi pentru model sau vizualizare.



## 10.1 Front End

Tehnologii folosite:

- html
- css
- javascript
- ajax

HTML pentru structura semantică a informațiilor;

CSS pentru prezentarea informațiilor;

De ce utilizăm javascript?

JavaScript (JS) este un limbaj de programare orientat obiect bazat pe conceptul prototipurilor.[5] Este folosit mai ales pentru introducerea unor funcționalități în paginile web, codul JavaScript din aceste pagini fiind rulat de către browser. Limbajul este binecunoscut pentru folosirea sa în construirea siturilor web, dar este folosit și pentru accesul la obiecte încastrate (embedded objects) în alte aplicații.

Acesta a fost folosit pentru majoritatea generărilor dinamice de elemente (Add more parts), filtrarea filmelor după diverse categorii, etc.

Ajax (sau AJAX), prescurtare pentru Asynchronous JavaScript and XML, este o tehnică de programare pentru crearea de aplicații web interactive. Intenția este să facă paginile web să devină mai rapide și deci mai acceptate, prin schimbul în fundal al unor cantități mici de date cu serverul, astfel încât să nu fie nevoie ca pagina să fie încărcată la fiecare acțiune a utilizatorului. Aceasta are ca scop creșterea interactivității, vitezei și ușurinței în utilizare a aplicațiilor web.

Aplicația este o SPA clasică (Application Single Page), în care este afișată doar strictul necesar. Am hotărât să utilizăm toate componentele reutilizabile și să folosim API-ul REST pe Back-End prin intermediul serviciilor oferite de AJAX.

## 10.2 Back End

Tehnologii folosite:

- php
- mysql

PHP (acronim recursiv pentru PHP: Hypertext Preprocessor) este un limbaj de scripting de uz general, cu cod-sursă deschis (open source), utilizat pe scară largă, și care este potrivit în special pentru dezvoltarea aplicațiilor web și poate fi integrat în HTML.

MySQL este un sistem de gestiune a bazelor de date relaționale.

Am decis să dezvoltăm cel puțin o aplicație web în PHP, care va utiliza API-ul REST pentru interfața web. Decizia noastră de a alege PHP se bazează pe faptul că se predă la seminar.

Transferul de stat reprezentativ (REST) este un stil arhitectural care definește un set de constrângeri și proprietăți bazate pe HTTP. Serviciile Web care respectă stilul arhitectural REST sau cele mai performante servicii web asigură interoperabilitatea între sistemele informatice de pe Internet. Serviciile web compatibile cu REST permit sistemelor solicitante să acceseze și să manipuleze reprezentările textuale ale resurselor web utilizând un set uniform și predefinit de operații fără stat.

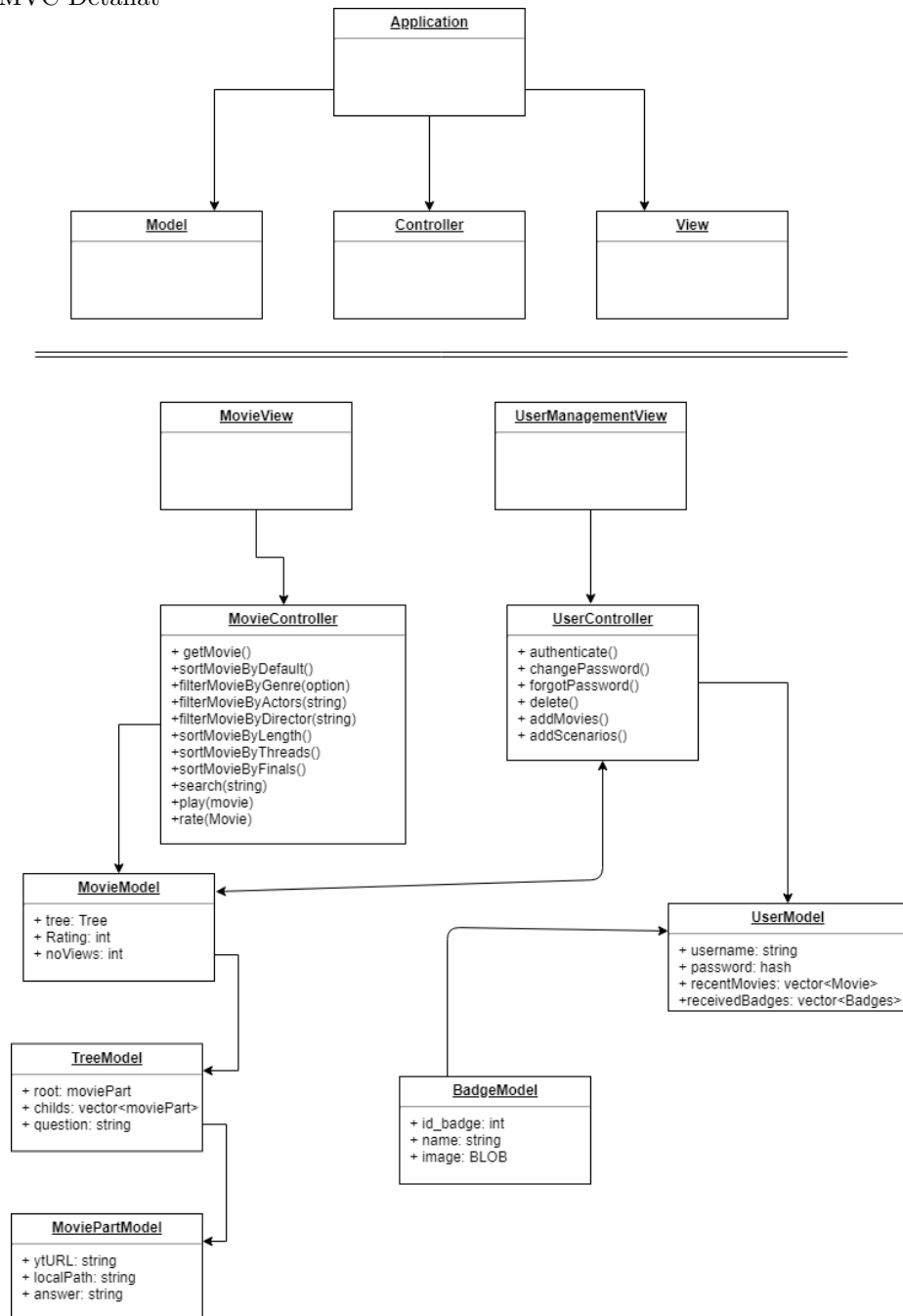
Decizia de a utiliza API-ul REST sa bazat pe capacitatea de a dezvolta independent Front-End și Back-End, practică care este în prezent un standard.

De ce am folosit MYSQL?

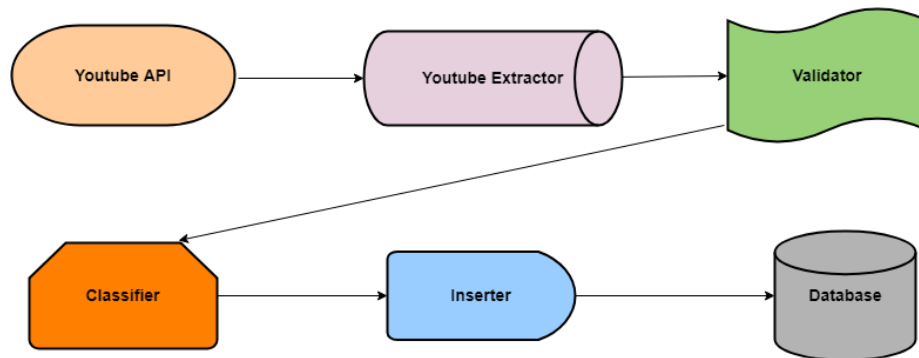
MySQL este un sistem gratuit de gestionare a bazelor de date open source (DBMS pe scurt). Un DBMS este un sistem care gestionează bazele de date și le conectează la software. De exemplu, o bază de date MySQL poate fi utilizată pentru a rula un site web, pentru a rula baza de date a unui ERP sau a oricărui alt software.

MySQL este un sistem puternic, gratuit, de gestionare a bazelor de date open-source, care a fost în jur de ani de zile. Este foarte stabil și are o comunitate mare care îi ajută să o întrețină, să o depaneze și să o actualizeze.

## MVC Detaliat



## 11 Extractor



Pentru furnizorul pe care îl analizăm pentru aplicația noastră (Youtube), vor exista date extrase bazate pe clasificarea ACM.

Sistemul de clasificare ACM pentru anul 2012 a fost dezvoltat ca o ontologie poli ierarhică care poate fi utilizată în aplicațiile web semantice. Acesta înlocuiește versiunea tradițională din 1998 a Sistemului de clasificare a informațiilor ACM (CCS). Acesta este integrat în capabilitățile de căutare și afișarea vizuală a subiectului Bibliotecii Digitale ACM.

Extractorul extrage datele care urmează a fi convertite în filme, care vor fi trimise la Validator. Validatorul verifică calitatea și procesează filmul. Clasificatorul clasifică datele în funcție de gen, etc.

Ulterior, datele despre filme sunt gata pentru a fi inserate în baza de date, după ce au completat toate datele necesare. Inserterul introduce în baza de date informațiile, actualizând atât tabelul Movies, cât și tabelele ce depind de acesta.

## 12 Pasi Intermediari

Task	Sprint
Crearea schemei bazei de date	sprint 1
Documentat despre youtube si imgur api	sprint 1
research pentru fiecare limbaj folosit	sprint 1
crearea mvc-ului	sprint 2
integrarea login-ului	sprint 2
integrarea register-ului	sprint 2
integrarea reset password	sprint 2
integrarea paginei principale	sprint 2
integrarea upload-ului	sprint 3
crearea filtrelor	sprint 3
integrarea functionalitatilor youtube api	sprint 3
integrarea functionalitatilor imgur api	sprint 3
posibilitatea de a viziona filme	sprint 3
testing toate functionalitatile site-ului	sprint 3

## 13 Referinte

1. <https://www.restapitutorial.com/>
2. [https://ro.wikipedia.org/wiki/Ajax\\_\(programare\)](https://ro.wikipedia.org/wiki/Ajax_(programare))
3. <https://www.quora.com/What-is-MySQL-used-for-and-what-are-its-advantages-and-disadvantages>
4. <https://www.php.net/manual/ro/intro-what-is.php>
5. <https://ro.wikipedia.org/wiki/MySQL>
6. <https://developers.google.com/youtube/v3/>
7. <https://www.dacast.com/blog/video-streaming-protocol/>
8. <https://eu.udacity.com/course/intro-to-javascript-ud803>
9. <https://eu.udacity.com/course/intro-to-ajax-ud110>