



FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

Documentație Proiect - Proiectare Software

Movie Website

Student: Marcu Ariana-Mălina

Grupa: 30232/2

CUPRINS

1. *Introducere*
2. *Analiză - prezentarea temei*
3. *Proiectare - detalii de implementare, funcționalități*
4. *Modelare diagrame*
5. *Prezentarea interfeței grafice utilizator*
6. *Concluzii și dezvoltări ulterioare*

1. Introducere

În era digitală în care trăim, accesul la divertisment a devenit mai ușor ca niciodată. Cu o creștere continuă a tehnologiei și a conexiunilor la internet, platformele online pentru streaming de filme au devenit tot mai populare. În acest context, implementarea unui site de filme reprezintă o oportunitate captivantă de a aduce în prim-plan conținutul cinematografic pentru o audiență vastă.

Această documentație are scopul de a oferi o perspectivă detaliată asupra procesului de implementare a unui site de filme, acoperind aspecte esențiale precum arhitectura sistemului, funcționalități cheie și experiența utilizatorului. Chiar dacă piața serviciilor de streaming este competitivă, există mereu spațiu pentru inovație și îmbunătățiri. O implementare reușită a unui site de filme nu constă doar în oferirea unui acces facil la conținutul video, ci și în furnizarea unei experiențe de utilizare plăcute și sigure.

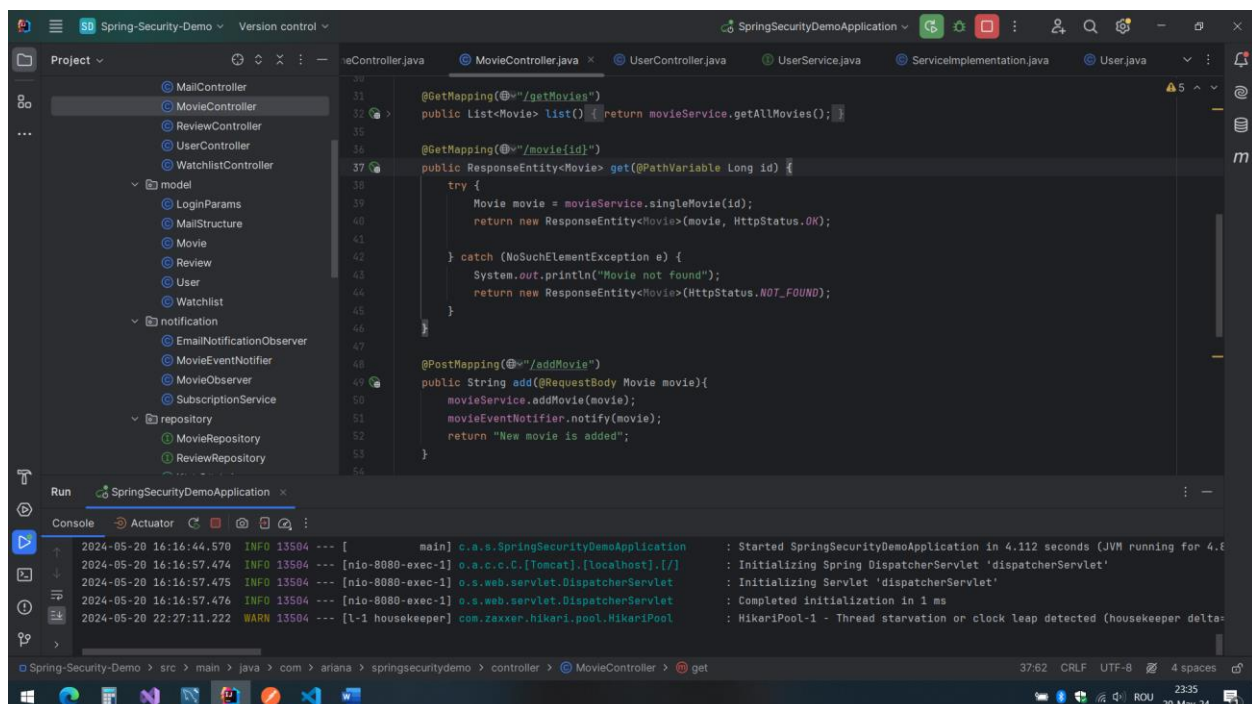
Astfel, din pasiune pentru filmele de toate genurile, am realizat acest site și cu această ocazie am învățat extrem de multe lucruri.

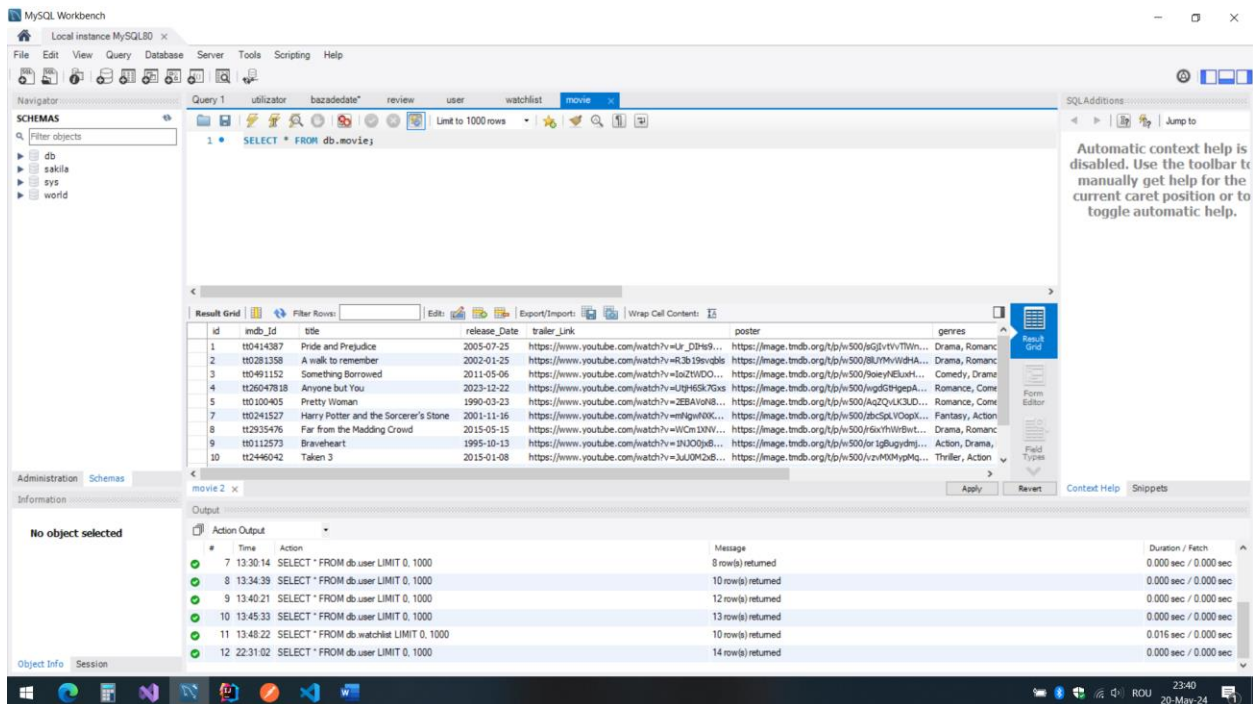
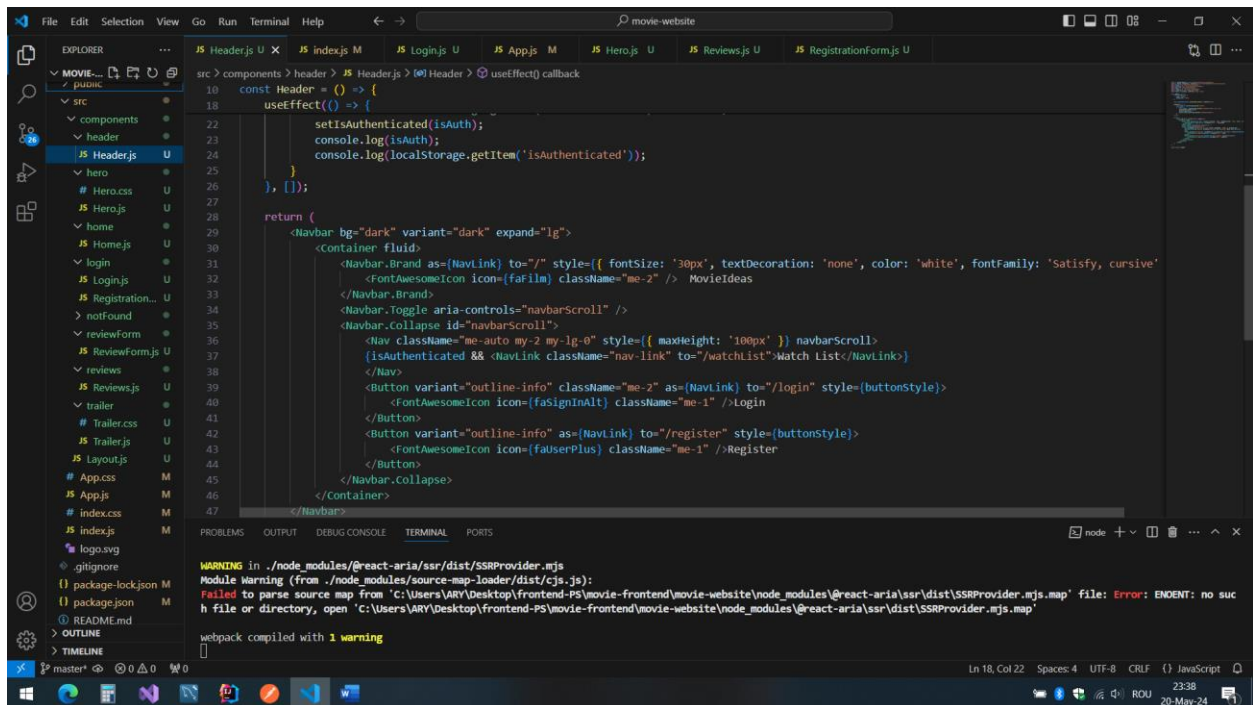
2. Analiză - prezentarea temei

Am avut de realizat o pagină web care să vină cu scopul de a oferi informații/idei cu privire la mai multe filme, de toate genurile, care să îi ofere ocazia utilizatorului să navigheze printre filme, să vadă imagini cu acestea, să poată viziona și trailerul și de ce nu, să poată să adauge review-uri pe care mai apoi toți ceilalți utilizatori să le poată urmări, pentru a-și da seama dacă vor sau nu să îl vadă în întregime. Totodată, vizitatorii site-ului se pot loga pentru a beneficia de funcționalități mai multe decât oferă deja pagina principală, și anume, a-și crea propriul watchlist. Dacă un user nu are încă un cont, el se poate înregistra, iar după logare, va avea și un buton de logout.

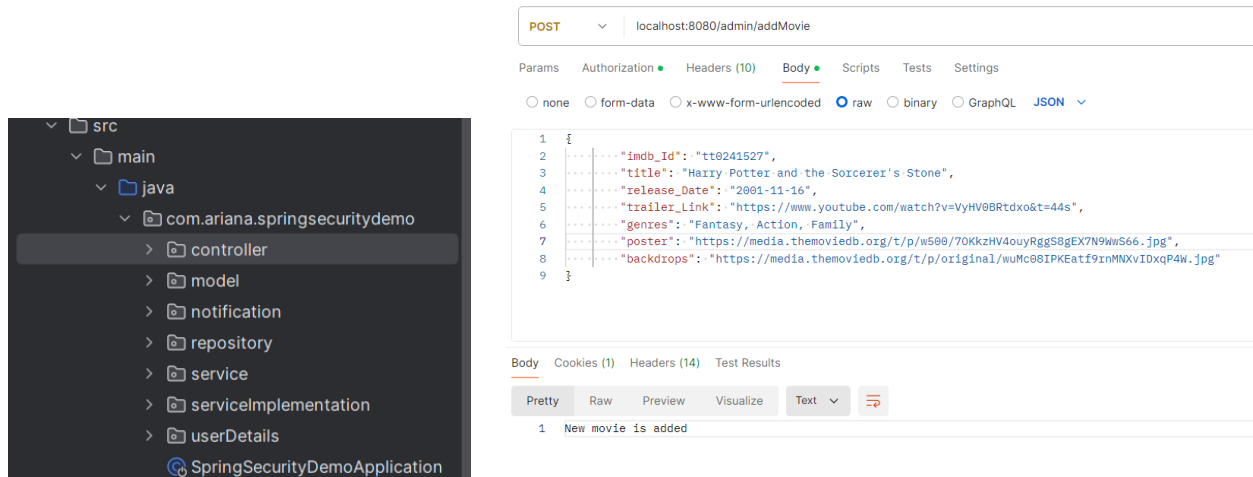
3. Proiectare - detalii de implementare, funcționalități

În realizarea acestui proiect, m-am folosit de parte de front-end și parte de back-end, pentru a-l putea organiza mai bine. Partea de back-end am realizat-o cu **Java Spring Boot** în IntelliJ, partea de front-end cu **React**(js, css) în Visual Studio Code, iar pentru baza de date m-am folosit de MySQL Workbench, după cum se poate vedea în imaginile de mai jos:





Ca să pot testa partea de **backend**, m-am folosit de Postman API, iar mai jos am atașat o imagine din timpul lucrării, unde adăugam un film în baza de date iar utilizatorii abonați primeau mail de notificare, ca să verific funcționalitatea. Ierarhia de clase din Java, poate fi observată în poza de mai jos. Atât pentru film, cât și pentru review, user, watchlist, mail, am câte o clasă **Service**, una **Repository**, **Controller** și una **ServiceImplmentation** în care mi-am făcut metodele necesare aplicației.



New Movie Added: Harry Potter and the Sorcerer's Stone



arianamarcu6@gmail.com

către sirbuiulia05 ▼

A new movie has been added to the website: Harry Potter and the Sorcerer's Stone

	id	username	password	role	email	subscribed
▶	1	ari	\$2a\$10\$2uGjUK/5deb4teCAwwx17.uCR5mCIB/...	ADMIN	arianamarcu6@gmail.com	0
	4	cristi	\$2a\$10\$C3408lyVLhREhQXh5IpWeZUmWFPC...	USER	52cristi25@gmail.com	0
	5	iulia	\$2a\$10\$zL65fX3AllwA9jZmFmNK.u9r1ZH2DwNV...	USER	sirbuiulia05@gmail.com	1
	6	camii	\$2a\$10\$juhrAfCVdPmh9dCmegdLcOGpKJPtomc...	USER	camii.groza.1606@gmail.com	0

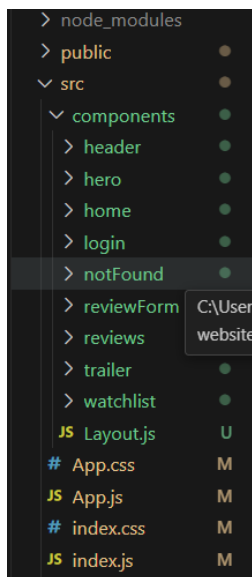
De asemenea, obiectivul acestui proiect a fost învățarea designului și implementării **MVC** (Model, View, Controller), așa că de aceea le-am integrat și în aplicația mea. Per total, ea conține două tipuri de useri, setați după rol în baza de date. Administratorul poate efectua **CRUD** (Create, Read, Update, Delete) pe toate entitățile existente, în timp ce user-ul poate efectua activitățile normale, destinate lui. Am obținut această funcționalitate în pachetul **userDetails**, unde în **WebSecurityConfiguration** autorizez request-urile și salvez parola criptată în baza de date.

Pentru trimiterea de mail-uri și notificarea abonaților la adăugarea unui film de către admin, m-am folosit de **Observable Design Pattern**. API-ul este **RESTful** și am folosit **Hibernate** pentru accesarea bazei de date. Am folosit **dependency injection** - un tipar de design care urmărește să decupleze obiectele de dependențele lor. În loc să își creeze propriile dependențe intern, obiectele le primesc dintr-o sursă externă.

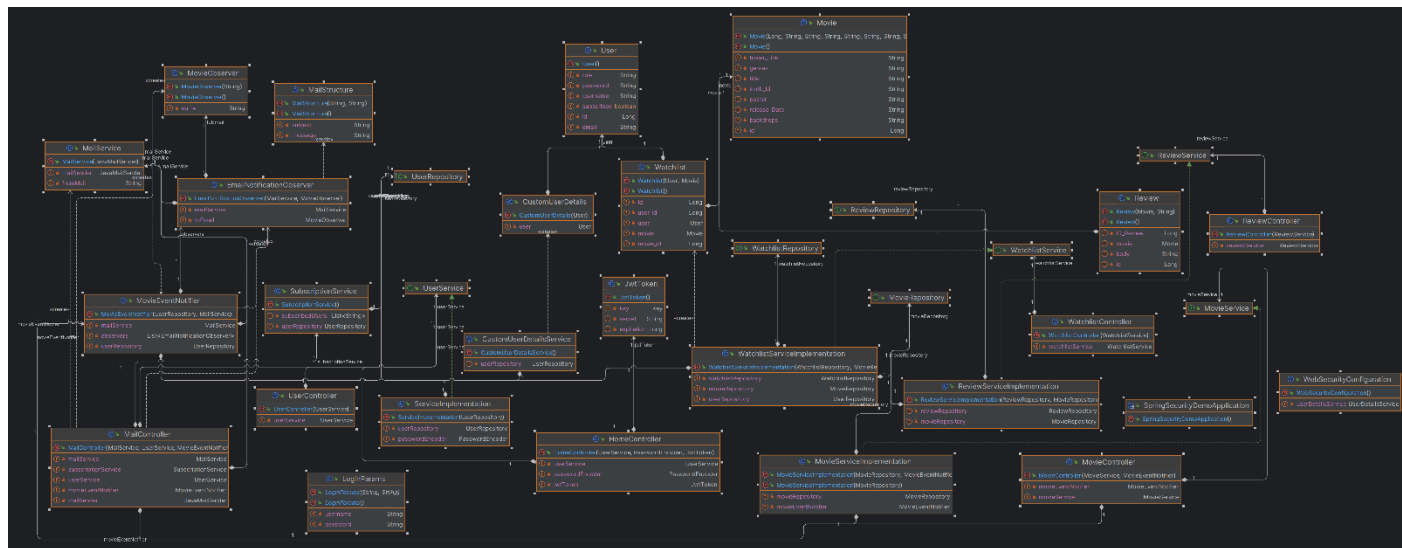
Pentru partea de **login** pe backend, mă folosesc de o clasă separată numită **JwtToken**, care generează token-ul specific utilizatorului autentificat, ce va ajuta în frontend să fie ținut minte și după refresh.

La partea de **frontend**, de asemenea se poate observa ierarhia de clase în imaginea de mai jos. Mi-am creat câte un folder pentru **header**, cel în care am pagina de intrare pe site cu bara de navigare din partea de sus a acestuia, **hero**, în care prezintă structura fiecărui film din caruselul cu care am ales să mă joc, **home**, care exportă un component React numit Home. Componenta Home primește o proprietate numită movies și returnează un component Hero cu această proprietate transmisă mai departe.

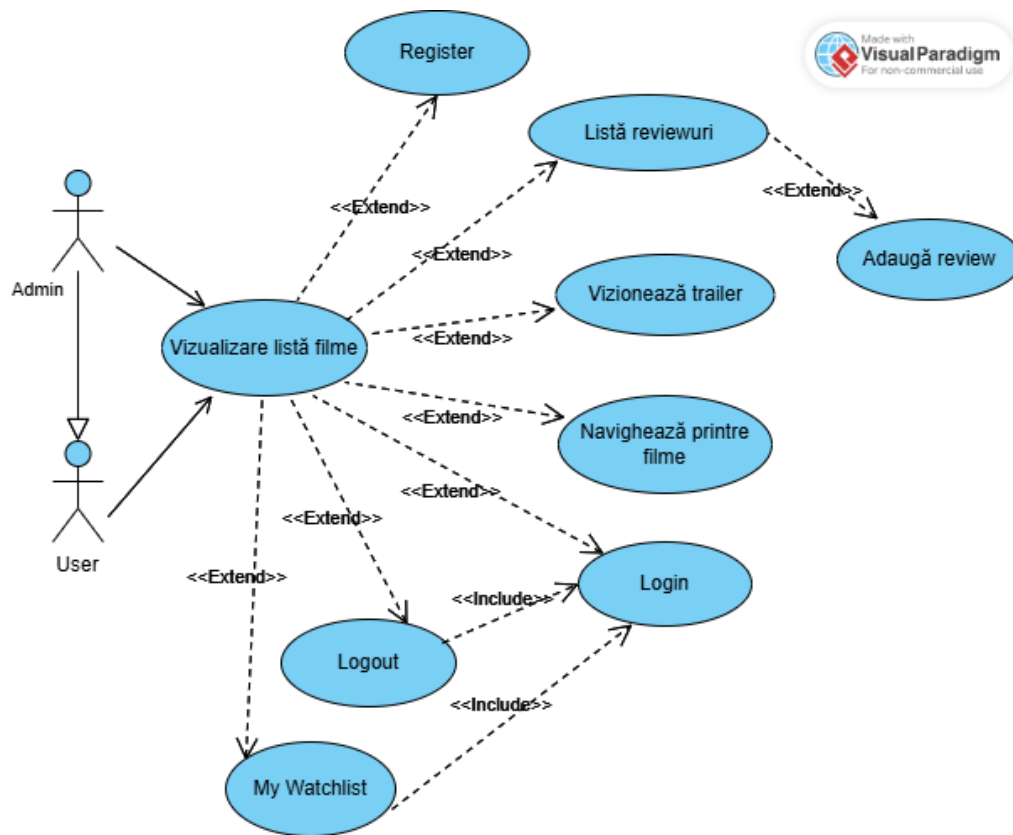
În continuare, pachetul login deține partea de login, logout și register, **notFound**, pentru a afișa mesajul în cazul în care pagina nu poate fi găsită, **reviewForm**, unde mă ocup de butonul de Submit de la review-uri, **reviews**, unde primesc de la tastatură un review și îl adaug în baza de date și pe ecran, salvându-l în istoricul filmului, **trailer**, unde mă ocup de afișarea trailerului fiecărui film și **watchlist**, unde mă ocup de lista filmelor pe care un user le are adăugate în lista lui. App.js este fișierul în care creez legătura între toate. Toate icon-urile care apar în frontendul aplicației au fost adăugate cu @forawesome și de asemenea am adăugat fonturi de pe google.apis. Nu m-am folosit de axios pentru configurare ci am ales metoda originală cu fetch, în realizarea codului meu.



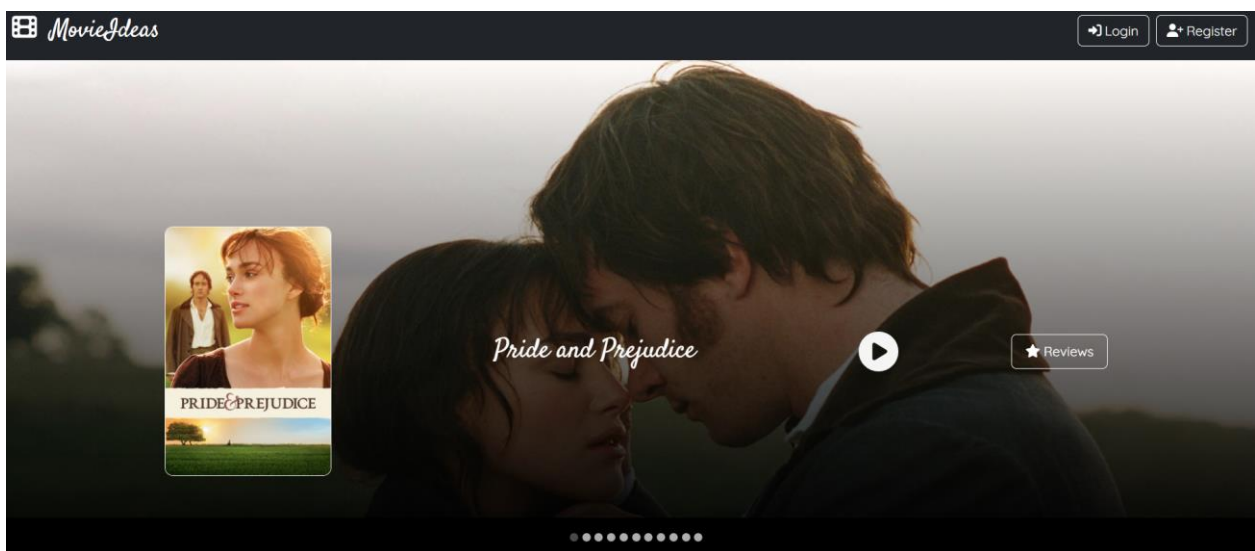
4. Modelare diagrame

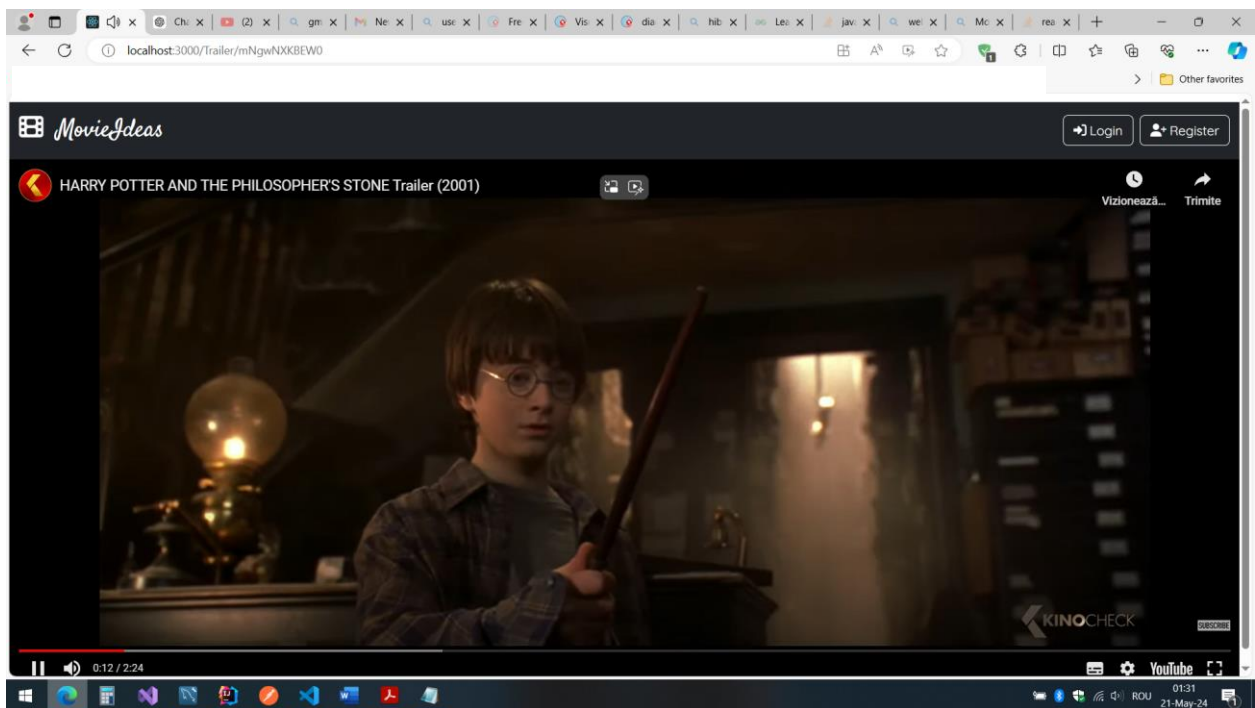
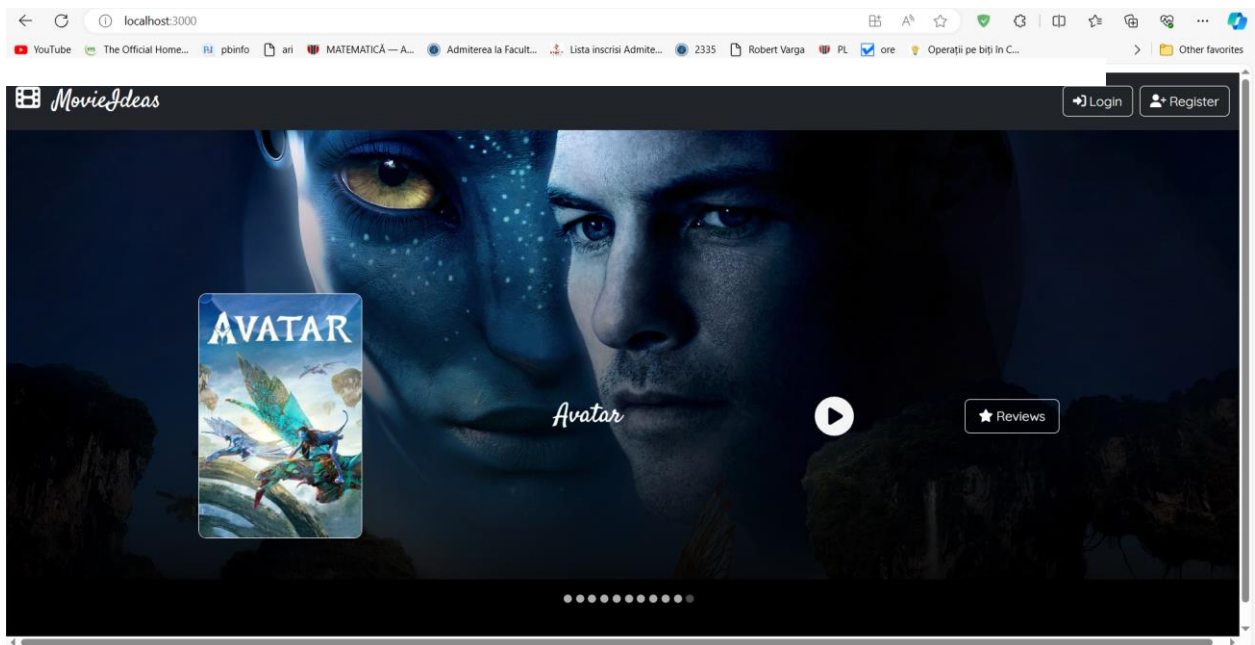


Aceasta a fost **diagrama de clase**. Ea poate fi văzută mai detaliat în png-ul atașat. În ceea ce privește **diagrama de use case**, o am pe cea de mai jos.



5. Prezentarea interfeței grafice utilizator





Adăugarea review-ului, care se păstrează pe site și după părăsirea acestuia:

★ Reviews

What did you think of this movie?


Stunning!!! =))))

Watched this recently in cinema, gorgeous!!! :***

★ Reviews

SYDNEY SWEENEY GLEN POWELL

IUBESC SĂ TE URĂSC



What did you think of this movie?

Submit

Stunning!!! =))))

Watched this recently in cinema, gorgeous!!! .***

GREAT MOVIE!

MovieIdeas

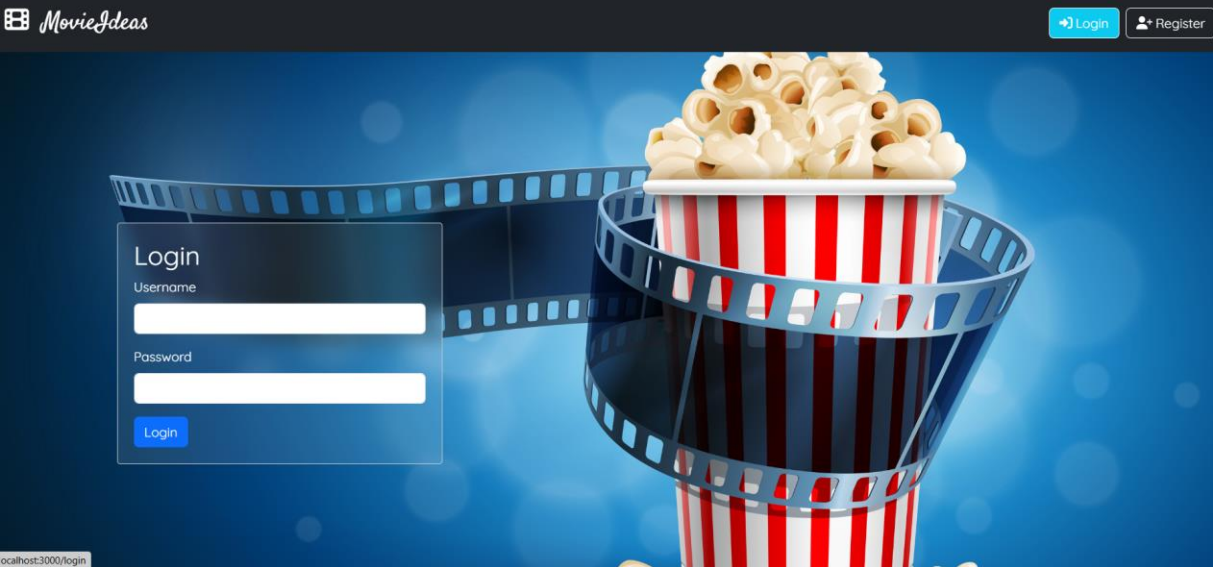
Login Register

Login

Username

Password

Login




localhost:3000/login

Login

Username

Password



Login

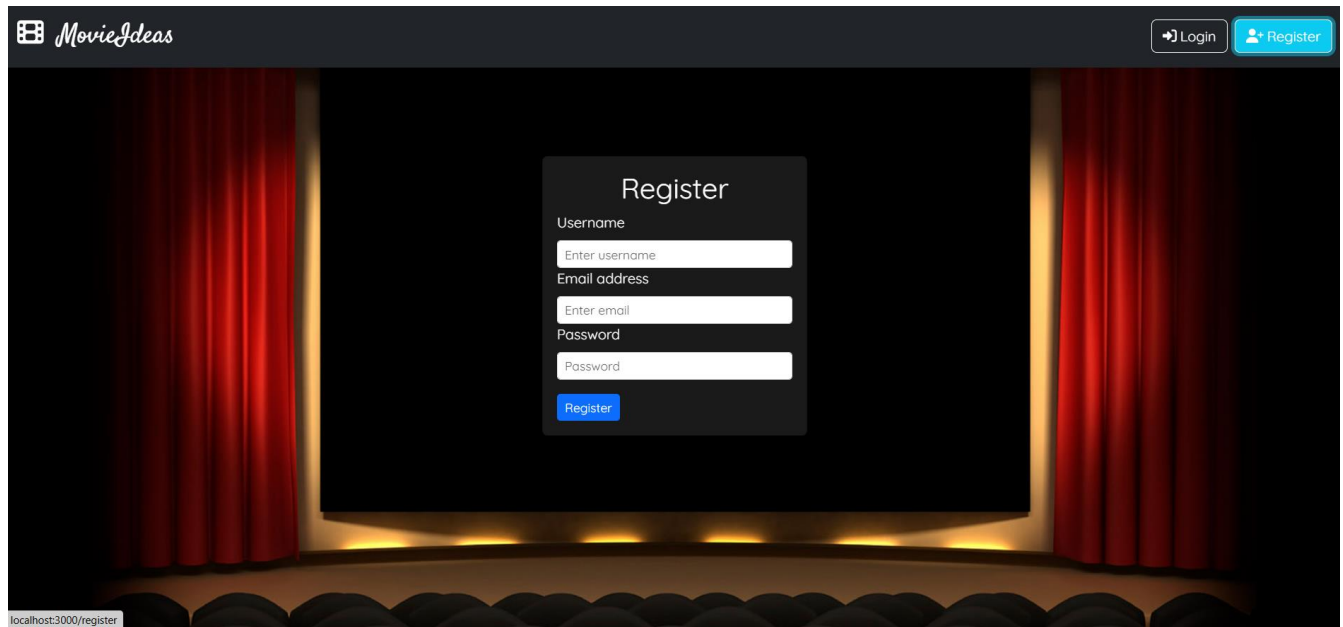
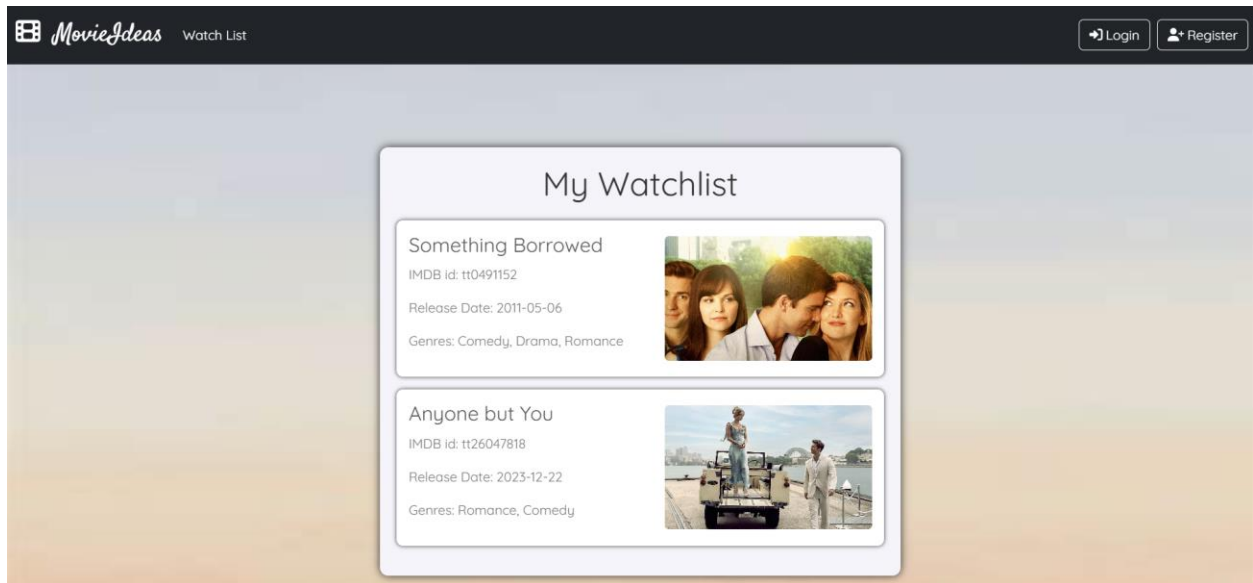
MovieIdeas

Watch List Login Register

Welcome, iulia!

Logout





6. Concluzii și dezvoltări ulterioare

Consider că acest proiect are posibilități ulterioare de dezvoltare nemărginite. Am putea adăuga rating cu stele pentru fiecare film, sortarea filmelor după genurile fiecăruia pentru a fi mai ușor utilizatorilor să caute după genul preferat, și în general aplicarea funcționalităților multiple din backend, să fie posibile și în front.

În principiu, acest proiect m-a ajutat să învăț să lucrez cu parte de back-end și front-end, să înțeleg React și Spring Boot mult mai bine și să-mi dezvolt capacitățile de programator. Cred că un astfel de site, dacă are toate elementele necesare, poate deveni de folos tuturor celor pasionați de lumea miraculoasă a filmelor/serialelor.