




Video: A Movie Catalog Web Application

Explore. Discover. Experience the World of Movies

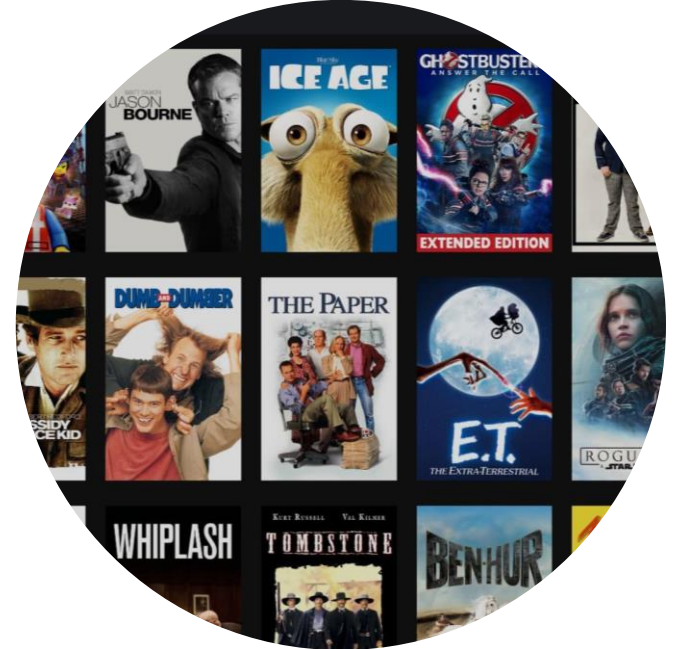


Despre ce o
sa vorbim
astazi?

Obiective
Features
Tehnologiile folosite
Viitoare imbunatatiri

Ce este Video?

- Video este o aplicatie web destinata pasionatilor de film. Scopul principal al aplicatiei este de a oferi utilizatorilor acces la o vasta selectie de filme si seriale din diverse genuri cinematografice, cum ar fi drama, actiune, comedie si altele. In plus, aplicatia isi propune sa introduca si sa aduca noi pasionati in lumea cinefililor, oferindu-le o experienta captivanta si diversificata in universul filmelor.





De ce Video?

De cand ma stiu, am fost complet prins de magia cinematografiei, mereu in cautarea unei povesti in care sa ma pierd timp de cateva ore. Dar ce-am observant este ca mereu mi-a fost greu sa gasesc titluri noi si captivante pe platformele de streaming.



Features



- Action Adventure Animation Comedy **Crime** Documentary **Drama** Family Fantasy History Horror Music **Mystery** Romance Science Fiction TV Movie Thriller War Western **Clear x**



Agatha Christie's Poirot 8.2



Elementary 7.6



Dexter 8.2



Monk 7.9



Vera 7.4

- Action Adventure Animation Comedy Crime Documentary Drama Family Fantasy History Horror Music Mystery Romance Science Fiction TV Movie Thriller War Western



Harry Potter and the Philosopher's Stone 7.815



Harry Potter and the Deathly Hallows: Part 2 8.11



Harry Potter and the Chamber of Secrets 7.723



Harry Potter and the Half-Blood Prince 7.599



Harry Potter and the Deathly Hallows: Part 1 7.787

- Action Adventure Animation Comedy Crime Documentary Drama Family Fantasy History Horror Music Mystery Romance Science Fiction TV Movie Thriller War Western



John Wick: Chapter 4 7.9



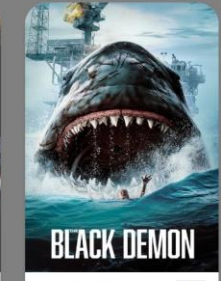
The Super Mario Bros. Movie 7.8



Spider-Man: Across the Spider-Verse 8.8

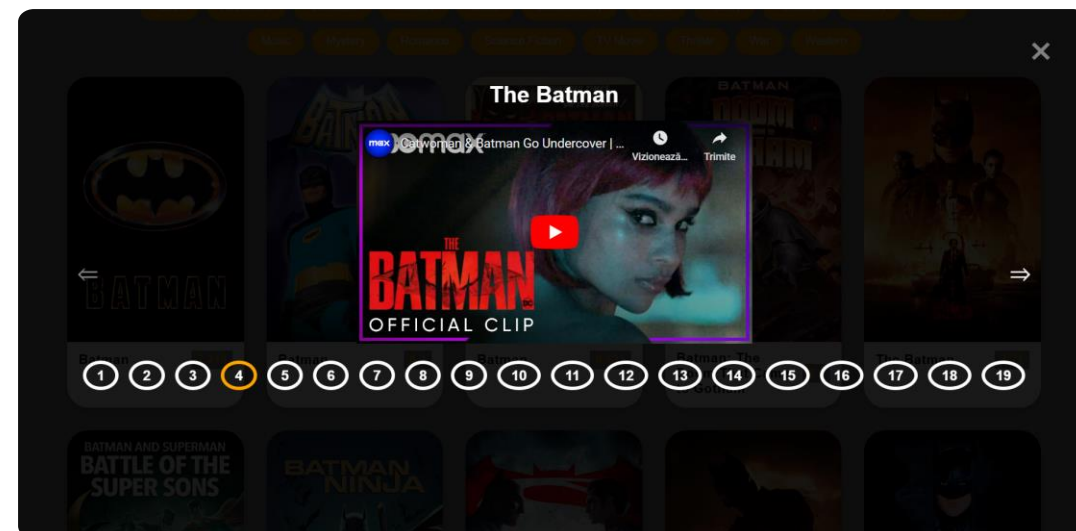
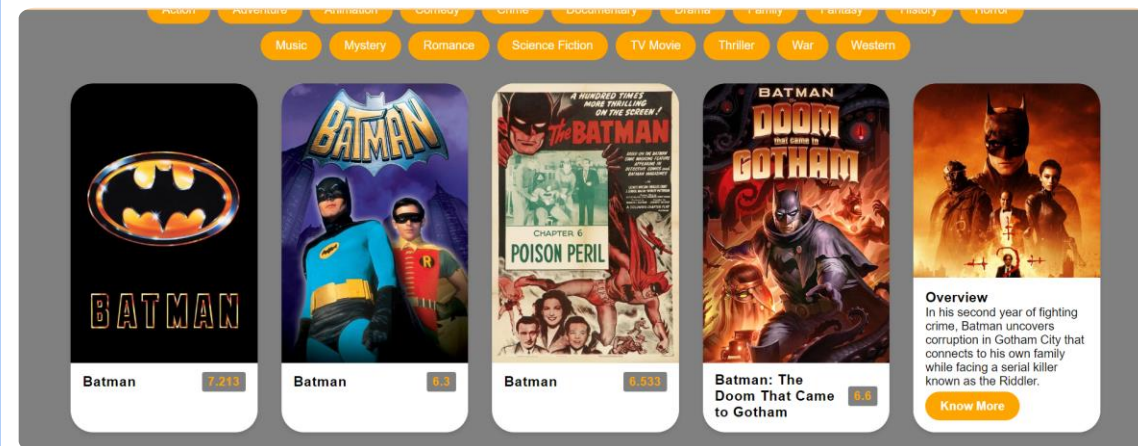


Fast X 7.2



The Black Demon 6.4

Rezumat si Referinte





Medii de dezvoltare si Tehnologii

Tehnologii

HTML, CSS si Javascript

- Site-ul a fost create folosind limbaje precum HTML pentru a creea template-ul fiecărei paginii, CSS pentru a crea un design mai atractiv si interactiv, si JavaScript pentru diferite functionalitati.

MySql

- Folosind MySql Server 8.0 am putut creea o baza de date eficienta pentru a stoca si gestiona informatii referitoare la cont-urile userilor.

The MovieDatabase API

- Am ales sa folosesc The Movie Database API pentru a beneficia de o sursa vasta si actualizata de informatii despre filme, inclusiv titluri, descrieri, si referinte. Aceasta integrare imi permite sa ofer utilizatorilor acces la cele mai recente si relevante detalii despre diferite titluri.

Au mai fost
folosite si:



HTML CSS JavaScript (exemple cod)

```
240 }
241 .nav_links li{
242     display:inline-block;
243     padding:0px 60px;
244     margin-top:25px;
245     margin-right:50px;
246 }
247 .nav_links li a{
248     transition:all 0.3s ease 0s;
249 }
250 .nav_links li a:hover{
251     color: #950000;
252 }
253 button1{
254     padding: 9px 25px;
255     background-color:#000000;
256     border : none;
257     border-radius:50px;
258     cursor:pointer;
259     transition:all 0.3s ease 0s;
260     color:white;
261 }
262 button1:hover{
263     background-color:#950000;
264 }
265 .film{
266     position:relative;
267     overflow:hidden;
268     padding:20px;
269     top:20px;
270     margin-bottom:-100px;
271 }
```

```
</style>
<script>
    var slideIndex = 0;
    var slides = document.getElementsByClassName("slider")[0].getElementsByTagName("div");

    function showSlides() {
        // ascunde slide-urile
        for (var i = 0; i < slides.length; i++) {
            slides[i].style.opacity = 0;
        }

        // incrementeaza slide indexul
        slideIndex++;
        if (slideIndex > slides.length) {
            slideIndex = 1;
        }

        // arata slide-ul recent
        slides[slideIndex-1].style.opacity = 1;

        // acesta este codul pentru a merge inapoi in slide
        function showSlides1() {
            // ascunde slide-urile
            for (var i = 0; i < slides.length; i++) {
                slides[i].style.opacity = 0;
            }

            // decrementeaza slide indexul
            slideIndex--;
        }
    }
}
```

```
21 </head>
22 <body>
23     <input type="hidden" id="status" value="<%= request.getAttribute("status")%>" />
24     <div class="login" style="align="center">
25         <h1>Login</h1>
26     </div>
27     <div class="container">
28         <form method="post" action="login" class="register-form" id="login-form">
29             <h2>Do you have an account?</h2>
30             <label for="email">Email</label>
31             <input type="email" name="email" id="email" placeholder="Enter Email">
32
33             <label for="password">Password</label>
34             <input type="password" name="password" id="password" placeholder="Enter Password">
35
36             <input type="hidden" name="signin" id="signin" class="form-submit" value="Log in">
37             <button type="submit" class="sign">Sign In</button>
38         </form>
39
40         <h3>Dont have an account?</h3>
41         <form action="Controller" method="post">
42             <input type="hidden" name="destinatie1" value="password.jsp">
43             <a class="locatie" href="signup.jsp"><button3 type="submit">Create one!</button3></a>
44         </input>
45         </form>
46     </div>
47     <script src="vendor/jquery/jquery.min.js"></script>
48     <script src="js/main.js"></script>
49     <script src="https://unpkg.com/sweetalert/dist/sweetalert.min.js"></script>
50     <link rel="stylesheet" href="alert/dist/sweetalert.css">
51
```

MySQL si Servlet-urile

	id	username	upassw	uemail
▶	1	x	xxx	x@yahoo.com
	2	Geo	tpiproject	funda_geo@yahoo.com
•	NULL	NULL	NULL	NULL

- Controller.java
- Login.java
- Logout.java

```
public class Controller extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        // input-urile din formularul de inregistrare
        String username = request.getParameter(string: "name");
        String uemail = request.getParameter(string: "email");
        String upassw = request.getParameter(string: "pass");
        RequestDispatcher dispatcher = null;
        Connection con = null;

        if (username.isEmpty() || uemail.isEmpty() || upassw.isEmpty()) {
            // Handle the case of empty inputs
            request.setAttribute(string: "status", o: "empty");
            dispatcher = request.getRequestDispatcher(string: "signup.jsp");
            dispatcher.include(sr: request, srl: response);
            return; // Exit the method to prevent further execution
        }

        try {
            Class.forName(className: "com.mysql.cj.jdbc.Driver");
            con = DriverManager.getConnection(url: "jdbc:mysql://localhost:3306/tpiproject?useSSL=false", user: "root",
                password: "Eminescul218");
            PreparedStatement pst = con.prepareStatement(string: "insert into users1(username,upassw,uemail) values(?,?,?)");
            pst.setString(i: 1, string: username);
            pst.setString(i: 2, string: upassw);
            pst.setString(i: 3, string: uemail);

            int randuri = pst.executeUpdate();
            dispatcher = request.getRequestDispatcher(string: "signup.jsp");
            if (randuri > 0) { // verifica ca am primit datele
                request.setAttribute(string: "status", o: "success");
            }
        }
    }
}
```

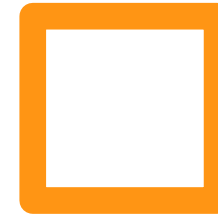
The Movie Database API


```
function showMovies(data) {
  main.innerHTML = '';

  data.forEach(movie => {
    const {title, poster_path, vote_average, overview, id} = movie;
    const movieEl = document.createElement('div');
    movieEl.classList.add('movie');
    movieEl.innerHTML = `
      
      <div class="movie-info">
        <h3>${title}</h3>
        <span class="${getColor(vote_average)}">${vote_average}</span>
      </div>
      <div class="overview">
        <h3>Overview</h3>
        ${overview}
        <br/>
        <button class="know-more" id="${id}">Know More</button>
      </div>
    `;

    main.appendChild(movieEl);

    document.getElementById(id).addEventListener('click', () => {
      console.log(id);
      openNav(movie);
    });
  });
}
```





The way to get started
is to quit talking and
begin doing.

Walt Disney

Imbunatatiri pe viitor

Pagina de User

- Una din implementarile dorite este adaugarea unei pagini de utilizator. Aceasta pagina ofera userilor posibilitatea de a vedea titlurile pe care le-au vizionat de-a lungul timpului si le permite sa-si exprime opinia personala despre acestea, oferind un spatiu ca un mini-blog.

Streaming

- Folosirea platformei "Plex" pentru a distribui continut utilizatorilor. Cu Plex pot configura propriul server media, adauga filme, seriale, muzica si alte fisiere media.

Recomandari

- O functie prin care putem recomanda utilizatorilor titluri bazate pe istoricul lor de vizionare, evaluari si preferinte de gen.



Goal-uri atinse

1. Inregistrare si Autentificare Utilizatori
2. Cautare si filtrare
3. Detalii despre filme si diferite surse promotionale
4. Design Receptiv
5. Integrare cu The MovieDatabase API



Va
Multumesc!

Student: Funda Gheorghe

Grupa: 433D