

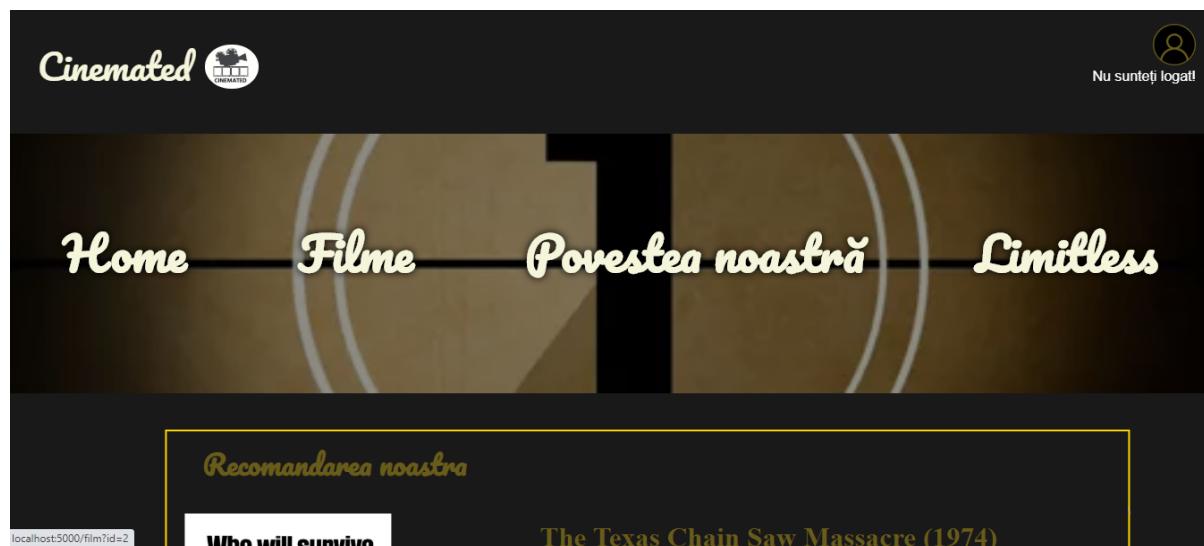
CINEMATED

- documentatie -

1. Descrierea proiectului

Scopul acestui proiect a fost de a realiza prezenta online a unui cinematograf, in acest caz, unul retro. Site-ul a fost construit utilizand tehnologii web consacrate (**HTML, CSS, JS**), alaturi de framework-ul **Jquery**, back-end-ul fiind bazat pe **Node.js si OracleDB**.

Site-ul ofera bazele unei experiente continue a utilizatorului, incepand de la explorarea galeriei de filme disponibile, pana la rezervarea unui loc in cinema pentru filmul dorit.



Utilizatorul poate gasi filmul potrivit filtrand prin galeria de filme dupa gen, intr-o anumita ordine sau dupa titlu. De asemenea, un administrator poate introduce sau elimina filme din baza de date.

A screenshot of the Cinemated website showing a search interface and movie filters. On the left, there's a circular profile picture placeholder and a '+' button. In the center, there's a search bar with the placeholder 'Introduceti titlul filmului'. Below the search bar is a section for 'Genul filmului:' with a list of checkboxes for various genres: Actiune, Aventura, Crima, Comedie, Drama, Familie, Horror, Razboi, Romantic, SF, and Thriller. The 'Comedie' checkbox is checked. To the right of the genre list are two movie cards: 'The Hangover (2009)' and '21 Jump Street (2012)'. Both cards show the movie title, genre ('Comedie'), runtime, distribution, and a small movie poster. At the bottom, there are buttons for 'Filtreaza' and 'Selecteaza optiunea'.

Fiecare pagina dedicata de film include un trailer, informații generale, cat si recenziile altor membri. De asemenea, utilizatorul are acces la o lista de ecranizari; odata ce alege, se deschide o interfata specializata pentru rezervare.



The screenshot shows a movie details page for "Rosemary's Baby". At the top, there are navigation tabs: Home, Filme, Povestea noastră, and Limitless. Below the tabs is a large movie poster showing a woman's face in profile against a dark, smoky background. To the right of the poster is a summary of the movie: "Dorindu-sa sa aiba o familie, tanara catolica Rosemary Woodhouse si Guy, sotul ei, se muta in Bramford, o cladire din New York despre care se spun numai lucruri rele. In scurt timp, cei doi se imprietenesc cu vecinii lor, Roman si Minnie Castevet, doi batrani excentrici, iar apoi Rosemary afla ca este incarcinata. Incet-incet, viitoarea mama este separata de prietenii si familia, iar semne alarmante ale unei conspiratii simestre incep sa apară, iar Rosemary este cuprinsa de o agonie mentala. De ce este tota lumea atat de dormica sa ajute? Si de ce Guy permite sa se intampla asta?" Below the summary are several movie details: Durata (137 min), Genul filmului (Horror), Distribuție (Mia Farrow, John Cassavetes, Ruth Gordon, Sidney Blackmer, Maurice Evans), Regizor (Roman Polanski), Anul aparitiei (1968), and Limba originala (Engleza). At the bottom of this section is a button labeled "Alege ecranizarea:" followed by a dropdown menu showing "21 iunie 2020, ora 16:00, sala 1" and a "Rezerva" button. A small logo for "Mia Farrow - Rosemary's Baby" is visible at the bottom left of the poster area.

Se poate alege numarul de bilete pentru fiecare categorie de pret, urmand apoi alegerea locurilor dorite in sala prin intermediul unei harti interactive.



The screenshot shows a ticket selection interface for the movie "Rosemary's Baby". The title "Alegeti bilete pentru: Rosemary's Baby" is at the top, followed by the text "Aveti un bilet gratis datorita cardului Limitless! Selectati restul biletelor dorite." Below this are four sections for selecting ticket types:

- Alegeti numarul de bilete pentru adulti: pret: 15lei
- Alegeti numarul de bilete pentru copii: pret: 7lei
- Alegeti numarul de bilete pentru elevi: pret: 10lei
- Alegeti numarul de bilete pentru studenti: pret: 10lei

Total: 70

At the bottom right is a "Next" button.

ECRAN

1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
3									
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	
1	2	3	4	6	7	8	9	10	

Film: Rosemary's Baby
 Data: 21 iunie 2020, 16:00
 Locuri:

R7 S2 • **R7 S3**

R7 S1

Bilete: 3
 Total: 70 lei

• Liber • Ocupat

[Next](#)

Site-ul dispune de un sistem de management al conturilor, atat pentru utilizatori obisnuiti, cat si pentru administratori. Se pot crea si sterge conturi, se pot schimba datele precum username, parola si alte date personale, iar administratorii pot controla accesul si privilegiile altor utilizatori.

Date personale

- Setări securitate
- Biletele mele
- Setări cont

Pagina principală

Date personale

Informații generale

Nume:

Adresă de e-mail:

[Modifică datele](#)

Cardul Limitless reprezinta un sistem de loialitate, in care orice utilizator logat se poate inscrie, pentru a putea profita de oferte speciale.

Informatii

Detaliile clientului

Nume...
Prenume...
Adresa de email...
Numar de telefon...

Tipul cardului

Standard
 Premium

Perioada de valabilitate

Iunie 2020

D	L	M	M	J	V	S
1	2	3	4	5	6	
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

Perioada aleasă:

Continuă

2. Organizare

Pentru o coordonare si organizare cat mai buna ca echipa, am folosit **Github** si **Trello**. Ca principale aplicatii pentru a ne intalni si discuta despre task-uri si functionalitati, opinii legate de utilizare sau cod, bug-uri, etc am folosit **Discord** si **Whatsapp**.

Am organizat implementarea functionalitatilor proiectului in mai multe sprint-uri. Astfel, la inceputul fiecarui sprint ne intalneam pe Discord si imparteam task-urile intre noi. Cream mai multe **user stories** si le notam pe Trello, pentru a putea tine evidenta progresului de-a lungul fiecarui sprint. La final, ne intalneam din nou pentru a prezenta actualizările facute, pentru a reporta diferite probleme/bug-uri si pentru a decide la ce idei renuntam si pe care urmeaza sa le implementam.

Trello

Starred Team Boards

Your Team Boards

Create new board

1 remaining

DISCORD

Cinemated

- TEXT CHANNELS
 - # general **LIVE**
- VOICE CHANNELS
 - General
 - tdnick
 - Andreea
 - Crisu
 - Mihala
 - Vero Chirut

Project MDS - Word

File Home Insert Design Layout References Mailings Review View Help

Clipboard

I want <some feature> — this is the WHAT. What are we building? What is the so that <some reason> — this is the WHY. Why are we building it? What is the customer?

Ex:
As an administrator
I want to be able to create other administrators for certain projects
So that I can delegate tasks more efficiently

Git config --global user.name "mihala"
Git config --global user.email mihala.antalburlacu@gmail.com
Git clone https://github.com/tdnick/cinemated

Silviu-Laurentiu Vasile prezintă

Cristina Vultur și încă 3

Tu

Nu avem "sala" in schema bazei de date.

Cocluzie: poate ramane asa schema, cu mentionea ca end date-ul cardului sa corespunda cu data Nu ne creeaza probleme de implementare ce avem pana acum.
Cauta: FN1, FN2, FN3.
Fiecare atribut din tabel care nu e cheie primara trebuie sa depinda de cheie si doar de cheie

Ce avem noi acum: FN1.
Atributele unui tabel trebuie sa fie date atomicel!
Daca vrem sa vedem filmele in care joaca un actor, nu putem face o singura coloana pentru act

Concluzia concluziei: ESTE IN REGULĂ!

<https://github.com/mateuszmarkowski/jQuery-Seat-Charts>

MDS - CINEMATED

S Silviu-Laurentiu Vasile
N Nicolas Iliescu
V Veronica Chirut

3. Source control

Aşa cum am mentionat mai sus, pentru a ne organiza cat mai eficient si pentru a avea mereu functionalitatile site-ului actualizate, am lucrat cu **Github**.

The screenshot shows a GitHub commit history for a repository. It is organized by date: Jun 14, 2020, Jun 13, 2020, and Jun 11, 2020. Each commit is shown with a small profile picture, the author's name, the commit message, the date it was committed, and a copy icon. The commit messages include: "Stergere cont utilizator, rez buguri", "Refactoring, reparat stilizare filtrare, mici buguri", "Index", "Home + Povestea noastră", "Merge branch 'master' of https://github.com/tdnick/cinemated", "Adaugare/stergere filme ca admin", "locuri ocupate plus nr rezervare", "Inserare bilete in bd si modificare aspect", and "veronicachiruit Stilizare buton modal". A "Clone or download" button is visible at the bottom right.

Fiecare dintre noi a dat commit cand a implementat ceva nou, cand a actualizat sau imbunatatit codul sau cand a modificat diferite bug-uri.

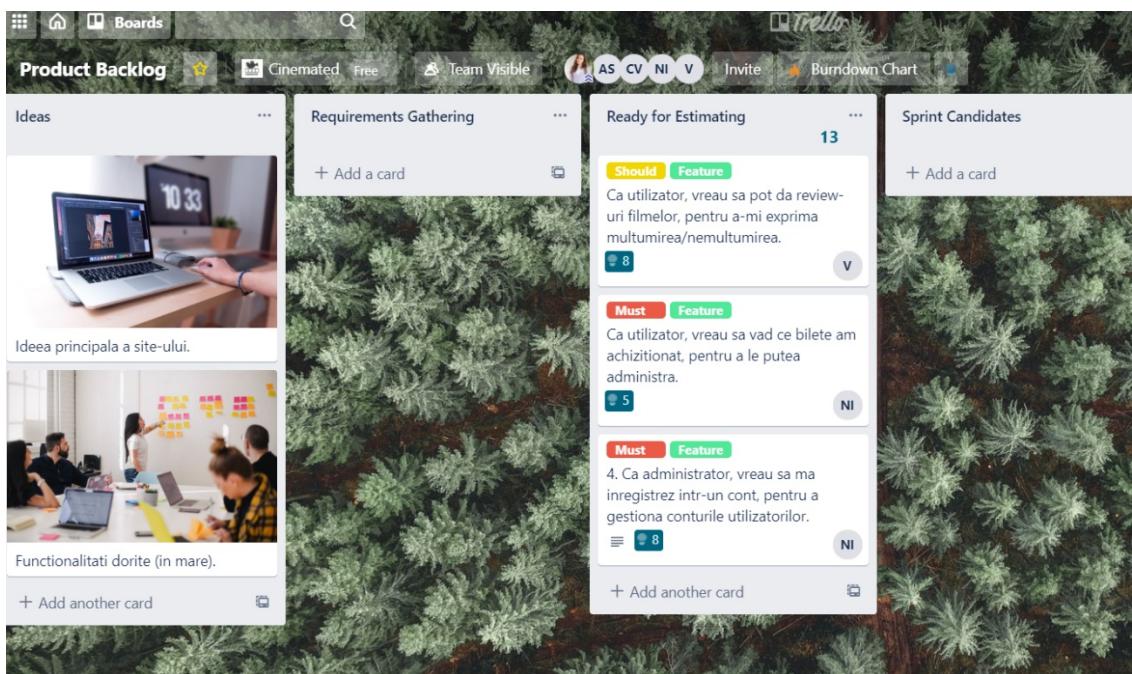
Astfel, in total avem peste 60 de commit-uri.

Full Stack Cinema Website Project

The screenshot shows the GitHub repository page for "Full Stack Cinema Website Project". At the top, it displays statistics: 64 commits, 2 branches, 0 packages, 0 releases, and 1 environment. Below this is a navigation bar with buttons for "Branch: master", "New pull request", "Create new file", "Upload files", "Find file", and "Clone or download". The main area lists 14 commits from various authors, each with a small profile picture, the file or directory name, the commit message, and the time it was made. The commits are: ".idea locuri ocupate plus nr rezervare 2 days ago", "node_modules Mici modificari la formularul de post 8 days ago", "public Stilizare buton modal 26 minutes ago", "views Stilizare buton modal 26 minutes ago", ".DS_Store Tabel useri, tabel bilete, functii adiacente 17 hours ago", "README.md Baza initiala 5 mai last month", "app.js Update stilizare, bugfix 1 hour ago", "package-lock.json Mici modificari la formularul de post 8 days ago", and "package.json Modificari recenzii 2 hours ago". A "Clone or download" button is also present at the bottom right.

4. User stories

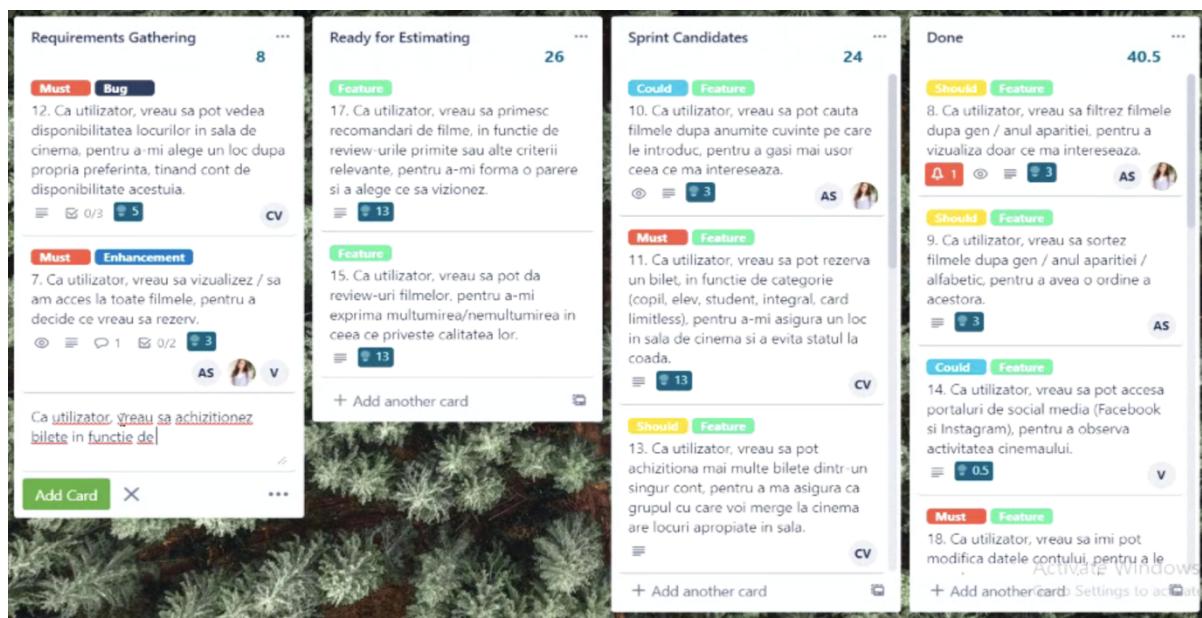
User stories au reprezentat unitatea de munca in proiectul nostru, bazat pe metodologiile **Agile**. Acestea au descris diverse functionalitati ale proiectului (Ca utilizator, as vrea sa..., pentru...), fiind implementate de-a lungul iteratiilor (sprinturilor). Am tinut cont de caracteristicile unui user story (independent, valoros pentru utilizator, estimabil etc.) si, pentru a estima dificultatea lor si gradul de necesitate in cadrul site-ului, am utilizat metoda **planning poker**, prin care am atribuit diverse punctaje (pe baza sirului lui Fibonacci), la care am ajuns de comun acord. Pentru a memora punctajele stabilite, am folosit urmatorul tool pus la dispozitie de platforma Trello: <http://scrumfortrello.com/>



User stories:

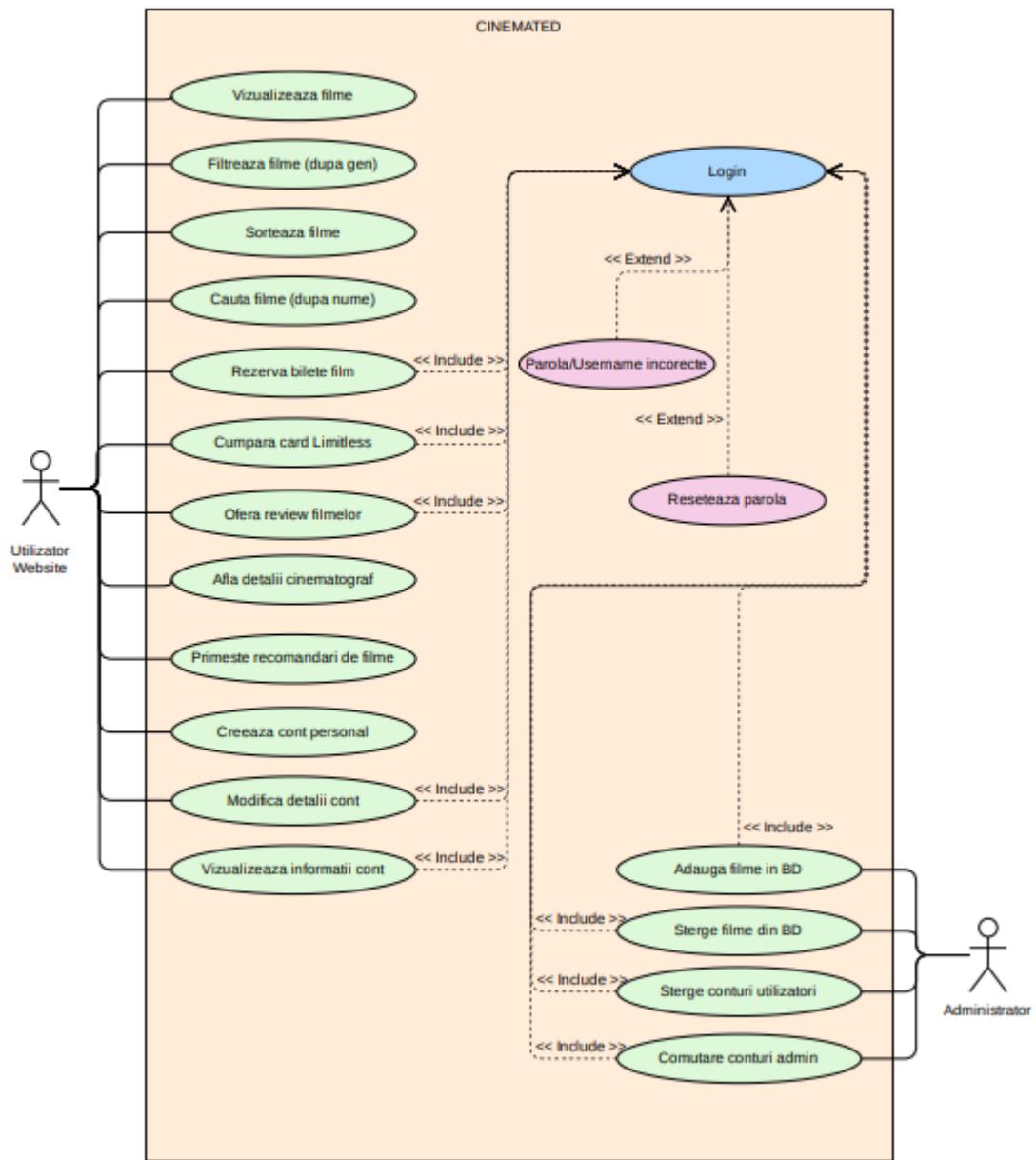
1. Ca utilizator, vreau sa ma inregistrez intr-un cont personal, pentru a putea avea acces la toate functionalitatile site-ului.
2. Ca utilizator, vreau sa ma pot loga in contul pe care mi l-am creat, pentru a-mi pastra evidenta activitatii si a nu fi necesara crearea repetata a altor conturi.
3. Ca utilizator, vreau sa imi pot modifica datele contului, pentru a le avea in permanenta actualizate.
4. Ca utilizator, vreau sa ma pot delega din cont, pentru a nu-l tine in permanenta vizibil sau pentru a putea intra in alt cont.
5. Ca utilizator, vreau ca meniul site-ului sa fie permanent vizibil, pentru a nu da scroll pana la inceputul paginii curente pentru a accesa o alta pagina.
6. Ca utilizator, vreau sa pot accesa portaluri de social media (Facebook si Instagram), pentru a observa activitatea cinematografului.
7. Ca utilizator, vreau sa aflu detalii despre cinema, pentru a fi sigur de autenticitate.

8. Ca utilizator, vreau sa am posibilitatea de a-mi achizitiona un abonament nelimitat (card limitless), pentru a avea acces la toate ecranizarile, pe durata unei luni.
9. Ca utilizator, vreau sa vizualizez / sa am acces la toate filmele, pentru a decide ce vreau sa rezerv.
10. Ca utilizator, vreau sa filtroz filmele dupa gen, pentru a vizualiza doar ce ma intereseaza.
11. Ca utilizator, vreau sa sorteze filmele alfabetic / dupa anul aparitiei / durata, pentru a avea o ordine a acestora.
12. Ca utilizator, vreau sa pot cauta filmele dupa anumite cuvinte pe care le introduc, pentru a gasi mai usor ceea ce ma intereseaza.
13. Ca utilizator, vreau sa pot obtine mai multe detalii despre un film, dand click pe acesta, pentru a avea o vizuire mai clara asupra filmului.
14. Ca utilizator, vreau sa pot vedea o lista a ecranizarilor pentru a putea alege mai usor biletul pe care il doresc.
15. Ca utilizator, vreau sa pot rezerva un bilet, in functie de categorie (copil, elev, student, integral, card limitless), pentru a-mi asigura un loc in sala de cinema si a evita statul la coada.
16. Ca utilizator, vreau sa pot achizitiona mai multe bilete dintr-un singur cont, pentru a ma asigura ca grupul cu care voi merge la cinema are locuri apropiate in sala.
17. Ca utilizator, vreau sa pot vedea disponibilitatea locurilor in sala de cinema, pentru a-mi alege un loc dupa propria preferinta, tinand cont de disponibilitate acestuia.
18. Ca utilizator, vreau sa verific datele rezervarii, pentru confirmare.
19. Ca utilizator, vreau sa pot da review-uri filmelor, pentru a-mi exprima multumirea/nemultumirea in ceea ce priveste calitatea lor.
20. Ca utilizator, vreau sa primesc recomandari de filme, in functie de review-urile primite sau alte criterii relevante, pentru a-mi forma o parere si a alege ce sa vizionez.
21. Ca administrator, vreau sa ma inregistrez intr-un cont, pentru a gestiona conturile utilizatorilor.
22. Ca administrator, vreau sa am acces la baza de date a utilizatorilor care si-au creat cont, pentru asigurarea integritatii ei.
23. Ca administrator, vreau sa am acces la baza de date a filmelor, pentru a adauga/sterge/modifica filme.



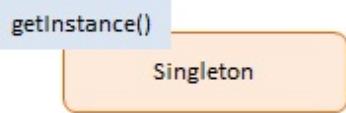
5. Diagrama UML

Am folosit diagrama de tip „**Use Case Diagram**” pentru a schita aspectele sistemului, prezentand comportamentul acestuia din punctul de vedere al utilizatorului.



6. Design patterns

Ca design pattern, am folosit sablonul creational **Singleton**, prin intermediul caruia am implementat clasa **dbHandler**. Intrucat ne-am dorit sa existe un singur manager al lucrului cu baza de date, utilizarea unei clase de tip Singleton a reprezentat varianta cea mai buna in realizarea acestui aspect. Clasa implementata a fost ulterior instantiata, iar natura ei ne-a asigurat ca orice incercare de a crea un nou obiect de tip **dbHandler** va duce catre acelasi obiect, creat anterior.



```
JS app.js JS locuri.js limitless.ejs # limitless.css JS dbHandler.js X
public > scripts > JS dbHandler.js > dbHandler > getInstance()
1 // singleton class
2
3 class dbHandler {
4   constructor() {
5     if (!dbHandler.instance) {
6       return dbHandler.instance;
7     }
8
9     dbHandler.instance = this;
10    return this;
11  }
12
13  getUser() {
14    return "SELECT * FROM users WHERE username = :name";
15  }
16
17  getLimitless() {
18    return "SELECT l.limitless_id, l.user_id, l.last_name, l.first_name, l.email, l.type, l.phone_number, l.start_date, l.end_date " +
19    "FROM users u JOIN limitless l ON (u.user_id = l.user_id) WHERE l.limitless_id = (SELECT MAX(limitless_id) FROM limitless WHERE user_id IN " +
20    "(SELECT user_id FROM users WHERE username = :name))";
21  }
22
23  insertIntoLimitless() {
24    return "INSERT INTO limitless (limitless_id, user_id, last_name, first_name, email, type, phone_number, start_date, end_date)" +
```

7. Build tool

Folosirea unui bulid tool a reprezentat o componenta esentiala in realizarea proiectului nostru. **Node Package Manager** (npm) ne-a facilitat lucrul cu pachetele specifice **JavaScript**, permitandu-ne incorporarea lor in proiect. De asemenea, un aspect important l-au reprezentat modulule, a caror gestionare a fost posibila prin intermediul npm, recunoscut pentru modularitatea sa.

```
215 app.use(express.static('static'));
216 app.use('/', router);
217 app.listen(port, function(error) {
218   if (error) {
219     console.log("Something went wrong!", error);
220   } else {
221     console.log("Server is on port " + port);
222   }
223 });
224

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Mihaela\Desktop\cinemated\commits\git1\cinemated> npm install express
```

8. Code standards + Code refactoring

Ca si **code standards**, am ales urmatoarele:

- camelCase (aceastaEsteVariabilaX, NU aceasta_este...)
 - fiecare variabila incepe cu o litera
 - spatii intre operatori, dupa punct si virgula, dupa virgula etc.
 - punem ";" dupa fiecare statement
 - punem "{}" chiar si unde avem o singura linie de cod (if-uri, for-uri etc.)
 - definirea unui obiect se face pe mai multe randuri
- ex: var person = {
 firstName: "John",
 lastName: "Doe",
 age: 50,
 eyeColor: "blue"
};
- maximum 80 de caractere/linie (pentru un cod lizibil)
 - constante scrise **UPPERCASE**
 - NU avem CSS sau JS inline; trebuie sa fie incluse in fisiere externe
 - folosim fisiere ejs, nu html
 - toate denumirile in engleza
 - comentarii in cod
 - pentru baza de date: **UPPERCASE** pentru comenzi (reserved words) si lowercase pentru valori

A fost nevoie de **code refactoring** atunci cazurile in care standardele amintite mai sus nu au fost respectate sau atunci cand anumite parti de cod necesitau sa fie modificate.

```
52 -     var buton_filtrare = document.getElementById("buton_filtrare");
53 -     buton_filtrare.onclick = function() {
54 -         // ...
55 -         var neselectat = false;
56 +     var filterButton = document.getElementById("filterButton");
57 +     filterButton.onclick = function() {
58 +         var selected = false;
59         for(let i = 0; i < checkboxes.length; i++){
60             if(checkboxes[i].checked == true){
61                 neselectat = true;
62                 selected = true;
63                 break;
64             }
65             if(neselectat == true){
66                 if(selected == true){
67                     for(i = 0; i < checkboxes.length; i++){
68                         var valoare = checkboxes[i].value;
69                         var sectiune = document.getElementById(valoare);
70                         var checkboxValue = checkboxes[i].value;
71                         var movieSection = document.getElementById(checkboxValue);
72                         if(checkboxes[i].checked == false)
73                             sectiune.style.display = "none";
74                         else
75                             sectiune.style.display = "block";
76                         movieSection.style.display = "block";
77                     }
78                 }
79             }
80         }
81     }
82 }
```

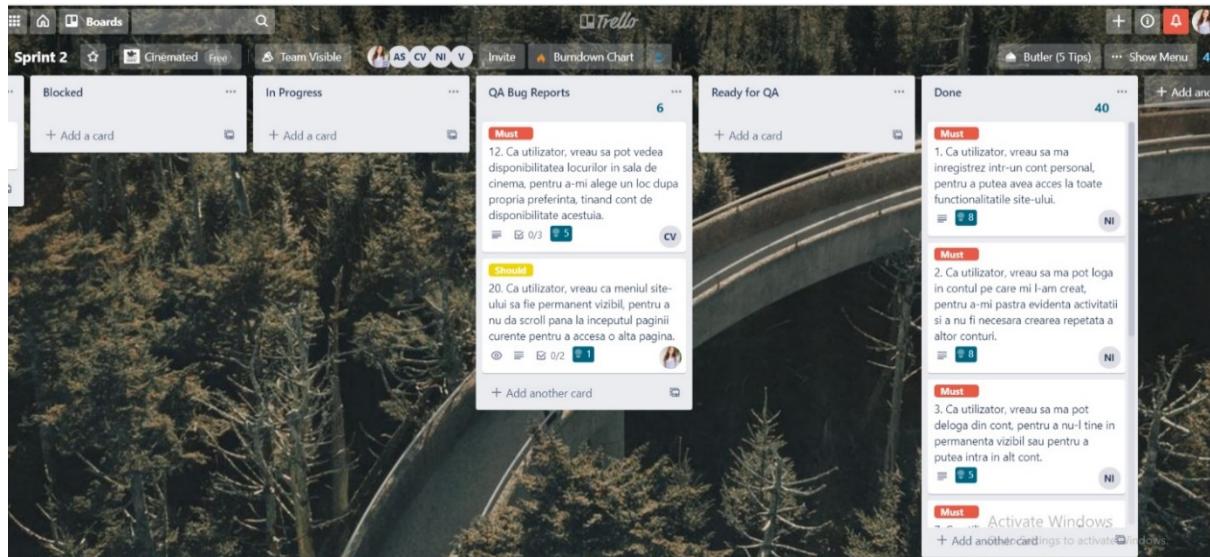
9. Bug reporting

Pe parcursul dezvoltarii proiectului nostru, ne-am confruntat cu numeroase buguri, unele a caror rezolvare a fost imediata si altele care ne-au ingreunat considerabil munca. In urma fiecarui sprint, ne-am rezervat un timp tocmai pentru discutarea lor, care a presupus incercarea de a gasi solutii si reetichetarea ca "bug" a componentelor cu erori neprevazute. Ulterior, acestea au fost transferate si rezolvate in sprinturile urmatoare.

Ca exemplu, am putea lua retrospectiva sprintului 2, in care am surprins urmatoarele:

Bugs:

- *Film cu titlu mai lung: imaginea e decalata fata de celelalte titluri*
- *Numar impar de filme: centrare pe mijloc a ultimului film, nu in partea stanga. Chiar daca rezolvam cu repozitionarea ultimului fiu in caz de numar impar, apare aceeasi problema la filtrare.*
- *Navbar: cand devine sticky (isi schimba stilizarea la scroll), sare cu inaltimea navbarului in pagina REZOLVAT*
- *Rezervare: apar duplicate: dau click pe un loc de mai multe ori, dar apar atatea bilete identice de cate ori dau click pe locul respectiv REZOLVAT*
- *Rezervare: daca am selectat un bilet, nu il pot sterge apoi REZOLVAT*
- *Cand esueaza inregistrarea unui user, nu-mi da eroare, ci ma redirectioneaza catre pagina principala. REZOLVAT*



```
c:\Users\Mihaela\Desktop\cinemated\commits\git13\cinemated\app.js:730
    req.session.limitlessData = ret;
^

ReferenceError: ret is not defined
    at C:\Users\Mihaela\Desktop\cinemated\commits\git13\cinemated\app.js:730:49
    at C:\Users\Mihaela\Desktop\cinemated\commits\git13\cinemated\node_modules\or
    at ResultSet.fetchRowsCb (C:\Users\Mihaela\Desktop\cinemated\commits\git13\ci
[nodemon] app crashed - waiting for file changes before starting...
```

Pentru rezolvarea bugurilor aparute pe parcurs, am folosit atat cea mai simpla metoda de debugging (diverse afisari, pentru a ne oferi un indicu asupra problemei existente), cat si varianta mai profesionala a debuggerului din browser.

10. Link-uri utile

Atunci cand am avut nevoie de inspiratie sau informatii suplimentare, am folosit mai multe site-uri, cum ar fi:

Pentru inspiratie:

- <https://www.cinemagia.ro/>
- <https://www.odeon.co.uk/>
- <https://www.cinemacity.ro/#/>

Pentru diagrama UML:

- <https://www.smartdraw.com/use-case-diagram/>
- <https://tallyfy.com/uml-diagram/#use-case-diagram>
- <https://www.visual-paradigm.com/features/uml-tool/>
- <https://online.visual-paradigm.com/drive/#diagramlist:proj=0&new=UseCaseDiagram>

Organizare user stories:

- https://www.youtube.com/watch?v=_jYiuYtrpp8

Baza de date:

- <https://www.youtube.com/watch?v=l23XTkUMrtM&feature=youtu.be>
- <https://github.com/oracle/node-oracledb/blob/master/INSTALL.md#instwino>

Rezervare bilete:

- <https://github.com/mateuszmarkowski/jQuery-Seat-Charts>
- <https://javascript.info/popup-windows>