

Administrarea unei biblioteci

Cătălina Racolța

Tuns Andrei

Sima Alin

Vancea Paul

Ianuarie 2025

Contents

1	Introducere, motivația alegerii temei	2
2	Context, cerințe, ce dorim să obținem	2
2.1	Context	2
2.2	Persona angajat	2
2.3	Persona client	3
2.4	Cerințe funcționale	4
2.5	Cerințe non-funcționale	4
2.6	Schema MoSCoW	5
3	Aspecte teoretice relevante și state-of-the-art	6
4	Implementarea aspectelor teoretice	7
4.1	Funcția de filtrare a cărților	7
4.2	Arhitectura generală	8
4.3	Arhitectura detaliată	8
4.4	Diagrama de clase	10
4.5	Cazuri de utilizare diagramă	11
5	Testare și validare	12
6	Rezultate în funcție de cerințele funcționale	13
7	Concluzii	14

1 Introducere, motivația alegerii temei

Această temă a fost aleasă în principiu de către Cătălina, deoarece este o pasiune de citit. Alături de colegi ne-am gândit să implementăm o aplicație care să îi ajute atât pe cititori, cât și pe bibliotecari. Aceștia vor putea să își acceseze cărțile dorite, dar și de a le adăuga într-un mod cât mai rapid și eficient. Aceasta o să îi ajute să reducă timpul petrecut în bibliotecă, mai ales când nu au idee ce cărți își doresc să citească.

Închirierea unei cărți este o alternativă populară pentru cei care își doresc să citească o carte, fără a fi nevoiți să o achiziționeze. Opțiunile sunt variate și potrivite pentru dorințele fiecărui cititor. Unii cititori s-ar putea să nu știe ce își doresc să citească, dar aplicația o să le vină în ajutor printr-un simplu click pe site, ei au posibilitatea de a vizualiza toate cărțile din biblioteca disponibile, chiar și genurile lor.

Tot mai mulți cititori, mai ales cei tineri, își îndreaptă atenția spre librării, deoarece acestea oferă o abordare mai modernă asupra experienței cititorului cu lectura. Ideea noastră de proiect are scopul de a motiva cititorii de a apela mai mult la serviciile bibliotecii, deoarece i-ar ajuta să economisească bani și timp.

Prin realizarea acestui proiect ne dorim să facem cititul mai ușor și mai accesibil, astfel încurajând un număr cât mai mare de cititori, mai ales de tineri să își închirieze cărți și să pătrundă în minunata lume a cititului.

2 Context, cerințe, ce dorim să obținem

2.1 Context

Noi ne-am gândit ca aplicația să fie ușor de utilizat atât pentru bibliotecari cât și pentru utilizatori, astfel adăugând câteva secțiuni cheie. Atât cititorii cât și administratorii vor avea un formular de înregistrare cât și de logare unde vor fi redirecționați în funcție de accesul pe care îl au. După ce au intrat pe site-ul principal, vor avea o primă pagină de unde vor putea selecta diferite detalii legate de cărți. Vor exista anumite butoane de unde fiecare persoană va putea alege tipul cărții(ex. SF,Dramă,Dragoste,etc.), autorul preferat sau chiar anul de publicație.

2.2 Persona angajat

Nume: Roxana Dolanescu

Varsta: 48 de ani

Ocupatie: bibliotecara cu experienta

Descriere generala:

Roxana este o bibliotecara dedicata, cu peste 20 de ani de experienta in domeniu. A absolvit Facultatea de Litere si este pasionata de literatura. Este calma cu un bun simt al umorului. Are o abilitate naturala de a intelege nevoile diverse ale cititorilor. Pe langa activitatea obisnuita, se ocupa cu organizarea

de evenimente literare si colaboreaza cu colegii pentru a dezvolta noi programe care sa atraga toate categoriile de public.

Personalitate:

1. Introvertită
2. Vibrantă
3. Ambițioasă
4. Pozitivă
5. Realistă

Obiective:

1. Dezvoltare personală
2. Întreținerea familiei

2.3 Persona client

Nume: Racolta Catalina

Varsta: 21 ani

Ocupatie: studenta

Descriere generala:

Catalina este o iubitoare de carti. Citeste in timpul liber ca o forma de hobby, are o pasiune pentru gaming si indrageste animalele. Desi isi petrece majoritatea timpului in fata calculatorului, mereu isi face timp pentru lectura. Considera acest obicei ca un mijloc de relaxare si de evadare din lumea reala.

Personalitate:

1. Introvertită
2. Multi-tasking
3. Ambițioasă
4. Vulcanică
5. Copilăroasă

Obiective:

1. Finalizarea studiilor
2. Realizarea pe plan profesional
3. Întemeierea unei familii

2.4 Cerințe funcționale

Cerințele funcționale au rolul de a defini și modela modul în care aplicația rulează, cum se realizează procesele interne și cum este vizualizat de către cititor.

1. Utilizatorii o să poată să își caute cărțile după preferințe, cu ajutorul unui filtru implementat în funcție de: gen, autor, titlu, an de publicare, etc.
2. Adminii/bibliotecarii vor avea acces la un istoric detaliat al persoanelor unde vor putea vedea toți utilizatorii și ce au accesat pe site.
3. Angajații vor avea posibilitatea de a oferi sugestii de cărți cititorilor în baza împrumuturilor anterioare și preferințelor acestora.
4. Fiecare carte și anume datele acesteia va fi introdusă într-o bază de date unde se vor stoca toate informațiile necesare (autor, an de publicație).
5. Un ISBN-ISSN (numărul internațional standard al cărții, numărul internațional standard al revistelor) va fi pus pentru fiecare carte/revistă.
6. O bază de date va fi implementată pentru fiecare autor unde vor fi scrise o multitudine de titluri/publicații dar și pseundonime al acestuia.
7. Inventarierea cărților va fi necesară pentru o mai ușoară gestiune, implementând câteva criterii precum un bar-code, un RMF(registru de mișcare a fondurilor).
8. Angajații vor avea posibilitatea de a adăuga noi cărți și date despre autor, pentru a putea fi vizualizate de către utilizatori atunci când doresc să închirieze o nouă carte.

2.5 Cerințe non-funcționale

Aceste cerințe non-funcționale au rol la fel de important ca și cele funcționale, definind performanța dar și calitatea platformei. Fără a avea aceste cerințe aplicația poate să fie instabilă cât și nesigură pentru utilizatori, lipsindu-i eficiența.

1. Securitatea este unul dintre cele mai importante aspecte ale aplicației deoarece avem stocat în ea date cu caracter personal al utilizatorilor. Va fi nevoie să asigurăm protecția acestor date printr-un autentificator în doi factori dar și criptarea datelor.
2. Printr-o interfață interactivă și ușor de folosit, îi vom ajuta pe utilizatori să petreacă cât mai puțin timp și să găsească cartea sau revista dorită în cel mai scurt timp posibil.
3. Pentru a face accesarea de către utilizatori cât mai ușoară și rapidă, performanța va juca un rol foarte important în succesul acesteia. Cu cât persoanele vor avea de așteptat mai mult cu atât vor deveni mai iritați, acest lucru afectând negativ modul în care se percepe biblioteca online și totodată succesul ei.

2.6 Schema MoSCoW

Schema sau metoda MoSCoW are rolul de prioritizare a cerințelor, asigurând implementarea funcționalităților de bază, de care este obligatoriu nevoie, mai apoi în cazul în care este nevoie, având posibilitatea de a adăuga unele cerințe opționale.

Must have

1. Bază de date pentru cărți
2. Bază de date pentru utilizatori
3. Sistem de logare cu drepturi diferite
4. Sistem de inventar al cărților
5. ISBN/ISSN pentru fiecare carte/revistă
6. Bază de date detaliată pentru autori

Should have

1. Istoric persoane
2. Verificare stoc

Could have

1. Donare cărți
2. Sugestii cărți
3. Pseudonime autori
4. Prolungire termen predare carte
5. Notificare termen predare carte
6. Rezervare cărți

Won't have

1. Sistem de înregistrare online
2. Filme/muzică/CD/artă/etc

3 Aspecte teoretice relevante și state-of-the-art

Noi am făcut comparația aplicației noastre cu cea de la Biblioteca Județeană "Petre Dulfu" Baia Mare și am constatat că în aplicația noastră avem în partea de logare, o casuță de parolă uitată și logarea se poate face direct, fără a fi nevoie de un click suplimentar. Avem o interfață mai modernă și mai intuitivă care permite cititorului o utilizare mai ușoară și mai plăcută. Informațiile sunt prestate într-un mod ușor de înțeles.

Produsele software au devenit componentele esențiale ale multor sisteme, fiind necesare în orice domeniu. Astfel chiar și bibliotecile care cândva își făceau gestiunea manual, apelează la soluții moderne, fiind nevoite să se digitalizeze.[1]

Gestionarea informației are loc prin elaborarea unui sistem de informații care permite utilizatorilor și bibliotecarilor autentificarea acestora în baza de date, care ulterior le permite cititorilor să rezerve și să împrumute cărți.[2]

Bibliotecile moderne se adaptează unui domeniu informațional în continuă dezvoltare, răspunzând cerințelor persoanelor pentru un acces mai rapid la informații. Sistemele integrate oferă soluții mai bune pentru a susține modul de lucru, satisfăcând cerințele actuale și viitoare. [4]

Site-ul web al bibliotecii este un instrument important de marketing online, necesar pentru promovarea resurselor și serviciilor oferite de bibliotecă. Calitatea acestuia trebuie asigurată încă de la început, iar dezvoltarea sa trebuie să corespundă nevoilor utilizatorilor, asigurând astfel succesul bibliotecii.[3]

Interfața utilizator și experiența utilizator sunt importante în realizarea unui site web pentru bibliotecă, deoarece fac utilizatori mai interesați prin adăugarea de meniuri și design-uri mai plăcute pentru fiecare utilizator. [5]

Arhitectura software este un lucru esențial în ingineria programelor, vizând provocări noi cum ar fi creșterea complexității, schimbări tehnologice rapide și cereri pentru programe de calitate în cel mai scurt timp posibil. [6]

Dreptul de împrumut public este un drept economic acordat cetățenilor, finanțat din bugetul statului și calculat pe baza împrumuturilor, exemplarelor din bibliotecă sau numărului de utilizatori. Fondurile statului pot susține literatura națională sau include alte tipuri de creații intelectuale. [7]

Pentru a asigura o creștere continuă economică cât și ocuparea forței de muncă este necesar un nivel superior al învățământului. Astfel, bibliotecile au avut și au în continuare un rol foarte important în societatea curentă deoarece sunt niște elemente cheie pentru dezvoltare prin intermediul informației libere. [8]

Bibliotecile contemporane au ajuns să se dezvolte de la rolul tradițional de păstrare a cunoașterii înspre oferirea informațiilor către utilizatori, adaptându-se noilor cerințe tehnologice. Ele integrează colecții digitale și tehnologii moderne, devenind bibliotecă ce pot oferi informații utile pentru utilizator.[9]

Un sistem personalizat de recomandare a cărților va fi realizat folosind un algoritm hibrid combinând o filtrare colaborativă și o metodă bazată pe conținut. Acesta are rolul de a îmbunătăți relația dintre utilizator și produs. [10]

4 Implementarea aspectelor teoretice

4.1 Funcția de filtrare a cărților

Filtrarea cărților permite utilizatorilor o căutare mai ușoară după cartea dorită. Aceștia vor putea filtra cărțile după mai multe criterii cum ar fi: titlul cărții, numele autorului care a scris cartea, anul publicării, limba în care a fost scrisă (ex: română, engleză, etc.), categoria în care se încadrează cartea (ex: dramă, acțiune, etc.).

```
const [filters, setFilters] = useState({
  author: "",
  title: "",
  category: "",
  language: "",
  yearOfPublication: ""
});
```

De fiecare dată când se schimbă detaliile filtrului, acesta este actualizat și se revine la prima pagină a cărților afișate.

```
const handleFilterChange = (event) => {
  const { name, value } = event.target;
  setFilters({ ...filters, [name]: value });
  setCurrentPage(1); // Reset to the first page when filter changes
};
```

Se creează o copie superficială a cărților primite prin intermediul filtrului care respectă condițiile impuse mai sus. Copia este folosită mai departe pentru afișarea cărților structurate sub formă de pagini.

```
const filteredBooks = booksArray.filter((book) => {
  const author = authors.find((author) => author.id ===
    book.authorId);
  const authorFullName = author ? `${author.firstName}
    ${author.lastName}`.toLowerCase() : "";

  return (
    (filters.author === "" ||
      authorFullName.includes(filters.author.toLowerCase())) &&
    (filters.category === "" ||
      book.category.toLowerCase().includes(filters.category.toLowerCase()))
    &&
    (filters.language === "" ||
      book.language.toLowerCase().includes(filters.language.toLowerCase()))
    &&
    (filters.title === "" ||
      book.title.toLowerCase().includes(filters.title.toLowerCase()))
    &&
  )
});
```

```
        (filters.yearOfPublication === "" ||  
          book.yearOfPublication.includes(filters.yearOfPublication))  
      );  
    });
```

4.2 Arhitectura generală

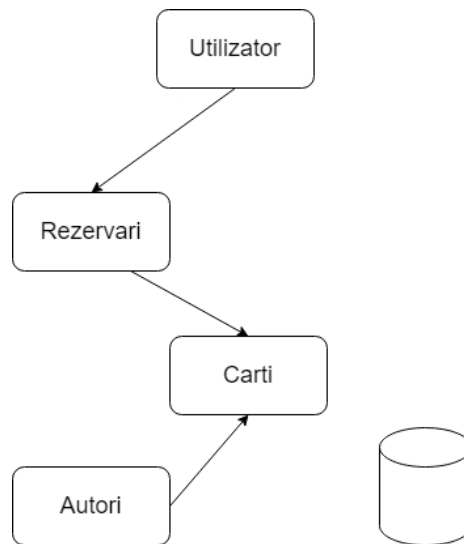


Figure 1: Arhitectura generală

4.3 Arhitectura detaliată

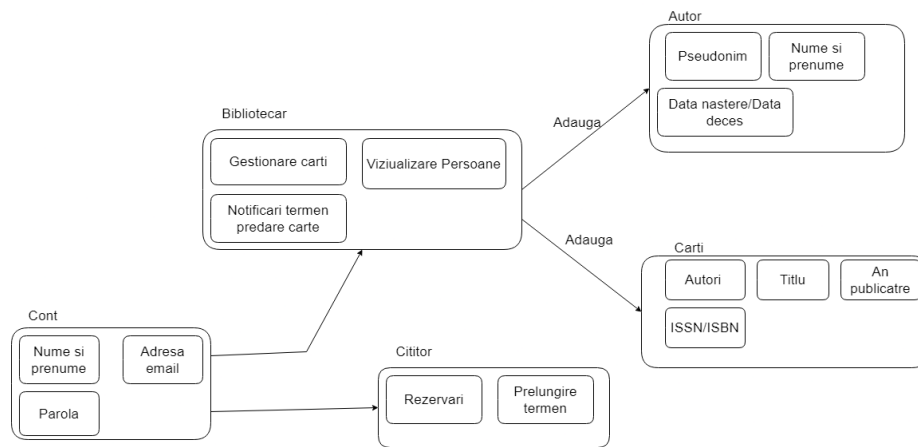


Figure 2: Arhitectura detaliată

4.4 Diagrama de clase

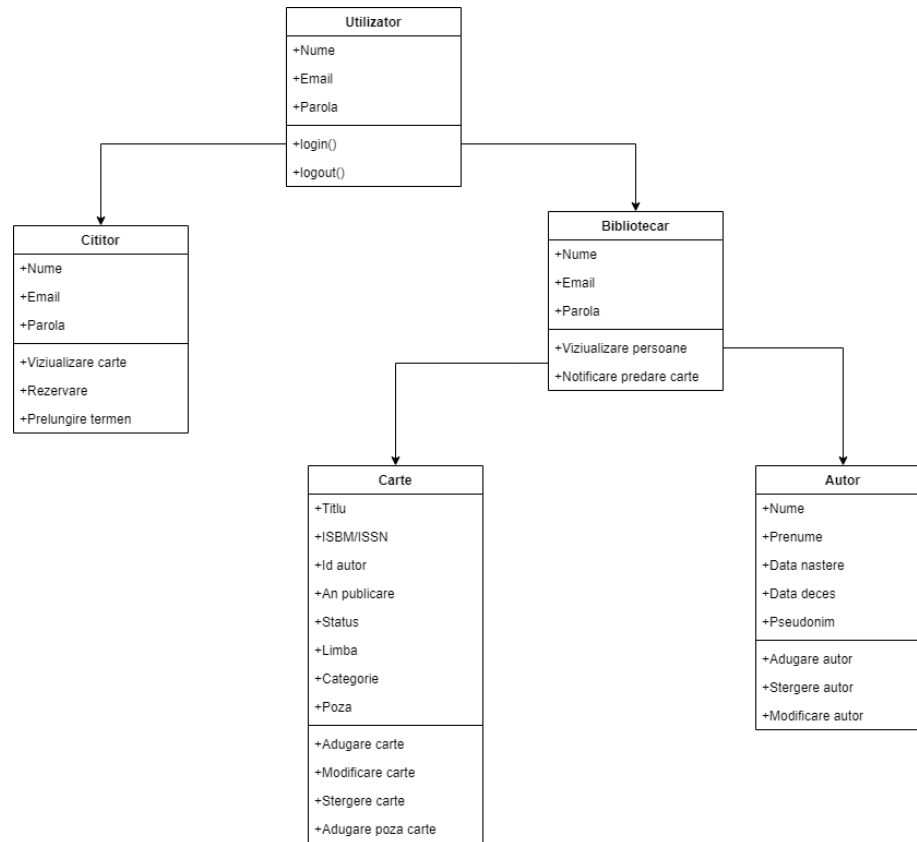


Figure 3: Diagrama de clase

4.5 Cazuri de utilizare diagramă

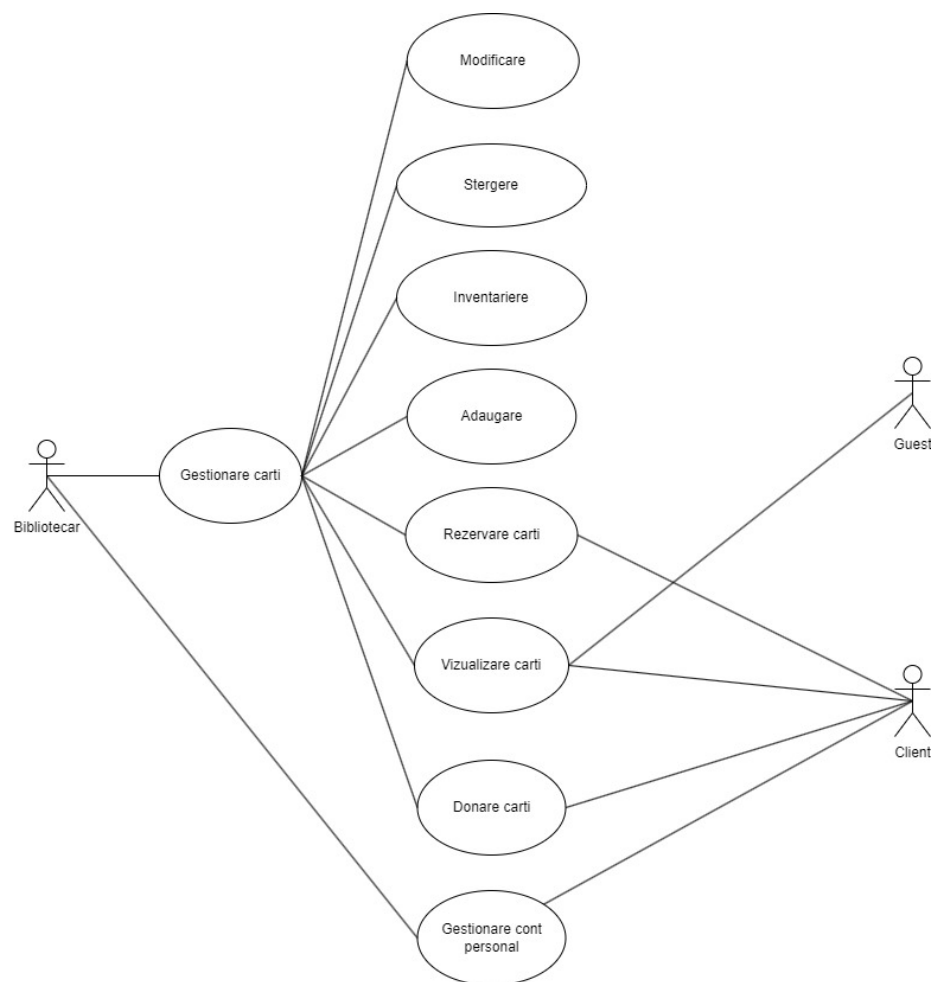
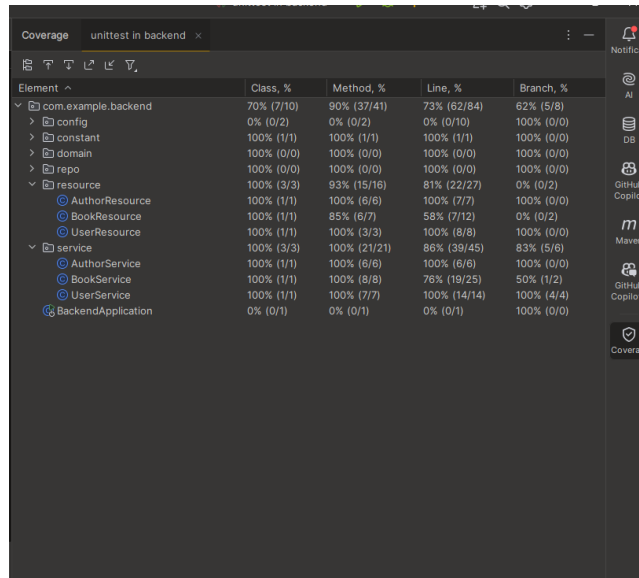


Figure 4: Cazuri de utilizare diagramă

5 Testare și validare

Ne-am folosit de Copilot AI pentru a evita subiectivitatea în rezolvarea erorilor. În mare parte testele au fost generate corect, fără a fi nevoie de modificari, dar am întâmpinat probleme la unele teste și le-am rezolvat cu ajutorul debugger-ului. Testele ne-au ajutat pentru a verifica corectitudinea codului și pentru a ne asigura că funcțiile implementate acoperă majoritatea cazurilor posibile de interacțiune a utilizatorului cu aplicația web.



Element	Class, %	Method, %	Line, %	Branch, %
com.example.backend	70% (7/10)	90% (37/41)	73% (62/84)	62% (5/8)
config	0% (0/2)	0% (0/2)	0% (0/10)	100% (0/0)
constant	100% (1/1)	100% (1/1)	100% (1/1)	100% (0/0)
domain	100% (0/0)	100% (0/0)	100% (0/0)	100% (0/0)
repo	100% (0/0)	100% (0/0)	100% (0/0)	100% (0/0)
resource	100% (3/3)	93% (15/16)	81% (22/27)	0% (0/2)
AuthorResource	100% (1/1)	100% (6/6)	100% (7/7)	100% (0/0)
BookResource	100% (1/1)	85% (6/7)	58% (7/12)	0% (0/2)
UserResource	100% (1/1)	100% (3/3)	100% (8/8)	100% (0/0)
service	100% (3/3)	100% (21/21)	86% (39/45)	83% (5/6)
AuthService	100% (1/1)	100% (6/6)	100% (6/6)	100% (0/0)
BookService	100% (1/1)	100% (8/8)	76% (19/25)	50% (1/2)
UserService	100% (1/1)	100% (7/7)	100% (14/14)	100% (4/4)
BackendApplication	0% (0/1)	0% (0/1)	0% (0/1)	100% (0/0)

Figure 5: Coverage

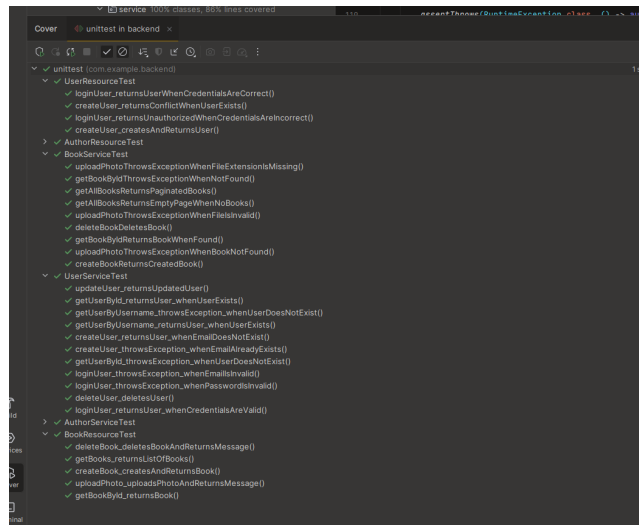


Figure 6: Rezultate unitesting

6 Rezultate în funcție de cerințele funcționale

În aplicația noastră am reușit să:

1. Implementăm funcția de filtrare a cărților în funcție de gen, autor, titlu, etc., care îi ajută pe utilizatori să își caute cărțile după preferințe.
2. Bibliotecarii pot introduce în baza de date informațiile necesare pentru fiecare carte (autor, an de publicație, etc.).
3. Fiecare carte are ISBN-ul ei, la fel și revistele au ISSN-ul lor.
4. Bibliotecarii pot introduce în baza de date detaliată, date despre autor mai amănunțite (pseudonime, etc.).
5. Bibliotecarii au posibilitatea de a introduce noi cărți și date despre autori, care pot fi vizualizate de către utilizatori atunci când doresc să închirieze cărți.

Pe viitor vor fi implementate și următoarele:

1. Bibliotecarii o să aibă posibilitatea de a vedea istoricul fiecărui utilizator și ce au acces aceștia pe site.
2. În funcție de cărțile pe care utilizatori și le-au împrumutat, bibliotecarii o să poată oferi sugestii de noi cărți.
3. Inventarierea cărților, implementând câteva criterii precum un bar-code, un RMF.

7 Concluzii

Partea de backend a fost realizată de Alin, dar și Andrei a încercat să implementeze în backend o funcționalitate pentru pagina de parolă uitată, care necesită trimiterea unui cod pe email pentru resetarea parolei, însă nu a reușit să îl facă funcțional.

Partea de frontend a fost făcută de către Cătălina, Andrei și Paul.

Documentația a fost scrisă de către Cătălina și Paul, dar și Andrei cu Alin au mai scris cate ceva în el.

Unit testing-ul a fost realizat de Paul și Cătălina, ajutați de Alin și Andrei.

Am avut bătaii de cap la selectarea cerințelor, deoarece erau multe idei din care a trebuit să alegem ce am putea face și ce nu, mai exact schema MoSCoW.

Am întâmpinat dificultăți la conectarea bazei de date cu programul, deoarece noi în aplicație la partea de backend am folosit un framework mai exact spring-boot care îți oferă posibilitatea de a defini tabelele și relațiile dintre acestea direct din cadrul aplicației.

Andrei a întâmpinat probleme la partea de frontend când a încercat să schimbe design-ul paginii, pe care Sima l-a structurat doar pentru verificarea funcționalității. Pozitionarea filtrelor a fost schimbată într-un sidebar pentru o estetică mai bună, dar având și o funcționalitate excelentă.

Cătălina a făcut pagina de înregistrare, dar după terminarea ei, a primit feedback de îmbunătățire a aspectului paginii, ceea ce a fost o mică problemă pentru ea, deoarece la început nu a știut cum să îl modifice, dar s-a ajutat de youtube și bootstrap, reușind să îi pună o imagine de fundal și să îl facă mai estetic.

Alin a făcut inițial să primească de la backend lista de cărți sub formă de pagini, dar când am dorit să realizăm filtrul, pe care l-am realizat în frontend, a avut nevoie de întreaga listă de cărți nepaginată, deoarece filtrarea a fost realizată în frontend. A trebuit să filtreze toate cărțile, nu doar cărțile disponibile pe pagina selectată, iar în urma acestei schimbări, a realizat paginarea în frontend.

Paul și Cătălina au întâmpinat probleme la realizarea unit testing-ului, deoarece nu aveau conștințele necesare care să îi ajute la o rezolvare cât mai bună și corectă a problemelor.

În mare parte, toți din grupă ne-am ajutat de vizualizarea unor videoclipuri pe YouTube, am folosit documentația de la bootstrap pentru diferite nevoi, am apelat și la Copilot, ne-am ajutat reciproc. De fiecare dată când am lucrat la proiect, am intrat cu toții pe discord, pentru a ne putea împărtăși părerile și pentru a ne sfătui.

Am mai întâmpinat toți probleme la Overleaf, deoarece a fost prima dată când am lucrat în el și ne-a luat ceva până am găsit comenzile necesare pentru documentația noastră.

Prezentarea în Canva a fost realizată de către Andrei, cu mici indicații de la Alin, Cătălina și Paul.

Procentaj de lucru în proiect: Alin 45 %, Cătălina 20 %, Andrei 20 %, Paul 15 %.

References

- [1] Lidia Băjenaru, Mihaela Tomescu, and Ion Alexandru Marinescu. Modernizarea sistemului bibliotecii naționale de programe prin utilizarea mediului de dezvoltare oracle apex. *Romanian Journal of Information Technology & Automatic Control/Revista Română de Informatică și Automatică*, 25(3), 2015.
- [2] Valentina BALUȘCA. *Sistem informațional de digitalizare a patrimoniului cultural al UTM-Biblioteca. Gestionarea și împrumutul cărților*. PhD thesis, Universitatea Tehnică a Moldovei, 2023.
- [3] Natalia Cheradi. Crearea și dezvoltarea site-ului web al bibliotecii universitare din perspectiva marketingului. *Magazin bibliologic*, (4):95–98, 2006.
- [4] Ionel Enache. Sisteme de management al bibliotecilor. *Studii de Biblioteconomie și Știința Informării*, (16):13–25, 2012.
- [5] Shivangi Gupta, N Meeramani, Sorabh Sharma, Prashant Raj, and Shikhar Gupta. Enhancing the user experience with ux/ui principles in library websites. *Library Progress International*, 44(3):9467–9470, 2024.
- [6] Christine Hofmeister, Robert Nord, and Dilip Soni. *Applied software architecture*. Addison-Wesley Professional, 2000.
- [7] Ana-Maria Marinescu. Dreptul de împrumut public. implementare națională și internațională. *Revista Română de Dreptul Proprietății Intelectuale*, (1):111–139, 2014.
- [8] Tana Oprea. Rolul bibliotecilor în societatea modernă. *Biblioteca*, 2019(3):4–31, 2019.
- [9] Diana Ioana Petre. Bibliotecile în mediul online. promovarea activităților și a proiectelor prin intermediul rețelelor de socializare. *Biblioteca*, 2013(6):191, 2013.
- [10] Yonghong Tian, Bing Zheng, Yanfang Wang, Yue Zhang, and Qi Wu. College library personalized recommendation system based on hybrid recommendation algorithm. *procedia cirp*, 83:490–494, 2019.