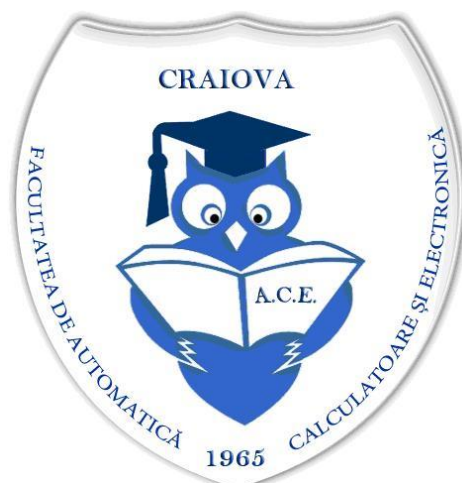


# RAPORT 1 – PROPUNERE DE PROIECT

---

Proiect III-Instrumente pentru dezvoltarea programelor



**Student:** Draghici Andreea-Maria

**Grupa:** CR4.S1 A

**Anul de studiu:** IV

**Specializarea:** Calculatoare Romana

---

**LIBRARY MANAGEMENT**

---

# Cuprins

1. Introducere
2. Rolul si obiectivele aplicatiei
3. Functionalitatile aplicatiei
4. Diagrame (Use-cases, Activity, etc.)
5. Tool-uri utilizate
6. Arhitectura sistemului si structura bazei de date
7. Concluzii

## Introducere

---

Library Management este o aplicatie folosita pentru gestionarea activitatii unei biblioteci / librarii in format electronic. Aceasta aplicatie este usor de utilizat, sigura si eficienta. Acest sistem este format din doua parti, o aplicatie web, care va fi utilizata de bibliotecar, dar si de catre cititori. Aplicatia comunica constant cu baza sa de date.

## Rolul si obiectivele aplicatiei

---

Aplicatia are doua tipuri de utilizatori, iar obiectivele acestora vor fi enumerate mai jos:

1. Utilizatorii care au rolul de user si pot vizualiza lista de carti disponibile in librerie de unde user-ul isi poate alege ce carte sa imprumute, pot sa trimita un ticket catre asistenta pentru utilizatori cu privire la orice eroare/ problema intalnita in timpul aplicatiei, pot avea acces la lista de carti care au fost accesate de catre alti utilizatori, astfel incat noii utilizatori sa poata vedea care sunt cele mai recente carti si sa ii ajute in alegerea unei carti potrivite.
2. Utilizatorii care au rolul de administrator si vor avea aceleasi functionalitati ca si utilizatori normali, dar pe langa asta vor mai avea si altele in plus:
  - Administratorul poate adauga, crea, sterge sau edita carti din lista de carti sau date despre acestea.
  - Administratorul poate vizualiza lista de carti rezervate.

- Administratorul poate modifica informatia starii unei carti anume din baza de date. ( Ex: Daca cartea nu se mai afla pe stoc in biblioteca).
- Administratorul poate actualiza informatii despre politica de confidentialitatea sau despre cartile rezervate.

## Functionalitatile aplicatiei

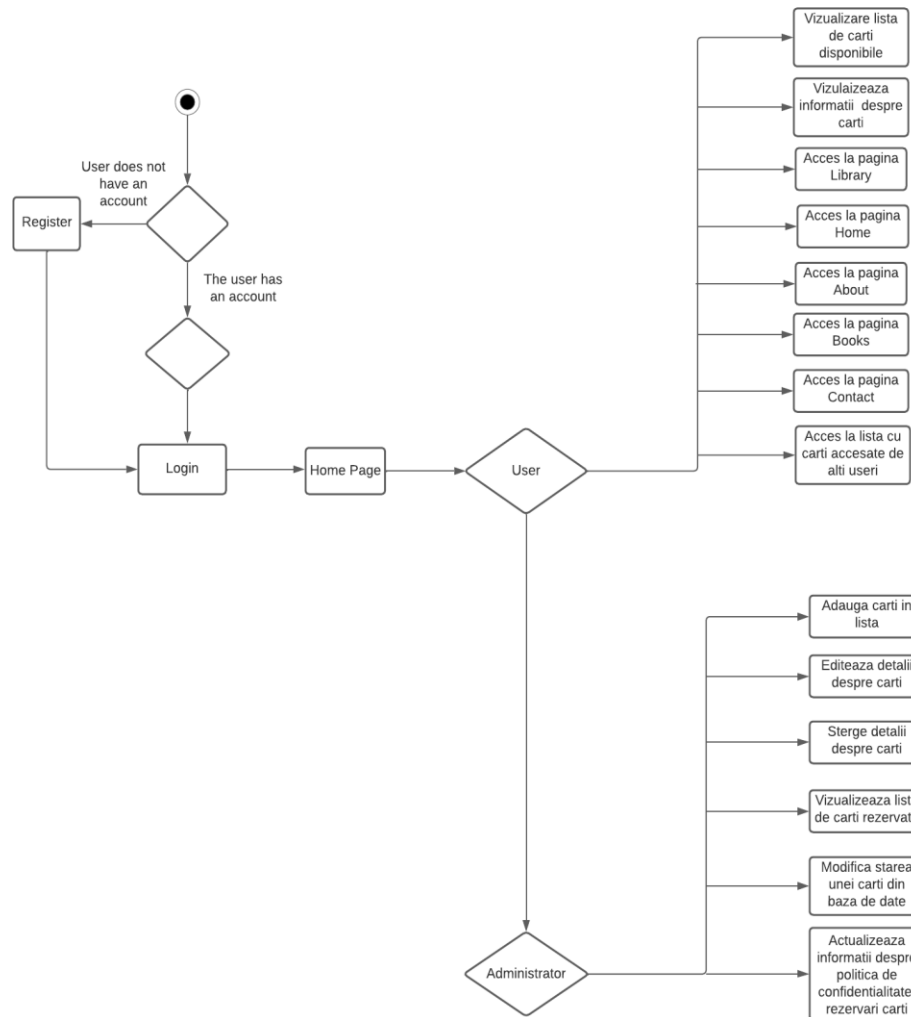
---

Aplicatia va fi alcatuita din doua parti si anume, o parte pentru user/client si o parte pentru administrarea aplicatiei, functionalitatile aplicatiei sunt urmatoarele:

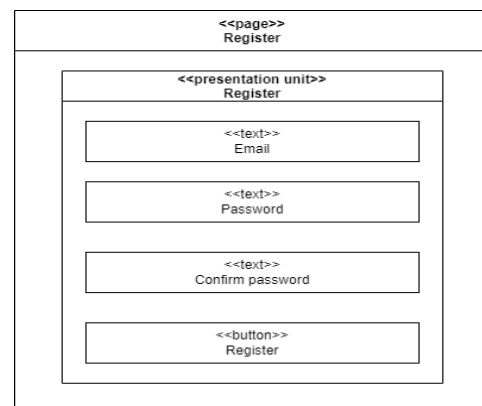
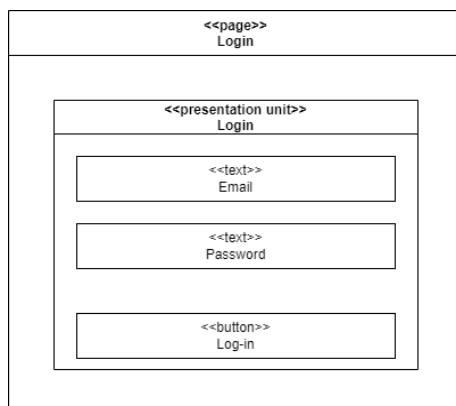
- In primul rand, aplicatia va avea un formular de inregistrare, unde user-ul isi introduce datele pentru a-si crea contul, iar utilizatorii care isi fac cont vor primi automat rolul de user. Acestia isi pot seta email-ul, parola, etc.
- Aplicatia va dispune si de o pagina de login, unde utilizatori isi vor introduce email-ul si parola, iar daca sunt corecte, ei vor fi redirectionati catre pagina principala de Home.
- Aplicatia va avea o pagina de contact.
- Aplicatia va avea o pagina de About care sa ofere informatii despre site si activitatea acestuia.
- Un utilizator ( indiferent de rol ) poate accesa functionalitatea aplicatiei, numai dupa ce s-a logat.
- Aplicatia va dispune si de o pagina numita Books, unde se gaseste lista cu carti disponibile in librerie de unde user-ul isi poate alege ce carte sa imprumute.
- Aplicatia va contine si o pagina de Privacy Policy, care include confidentialitatea si politica aplicatiei raportata reglementarilor si datelor stocate in baza de date.
- Aplicatia va mai contine si o pagina numita Library, unde utilizatorul poate vizualiza lista cu numarul de carti rezervate, cat si numarul total de carti din librerie.
- Aplicatia va contine si pagina principala de Home, aceasta oferind posibilitatea utilizatorului de a naviga catre lista de carti din librerie, aceasta pagina mai ofera si informatii si fotografii despre cartile care au fost accesate de catre alti utilizatori, astfel incat noii utilizatori sa poata vedea care sunt cele mai recente carti.

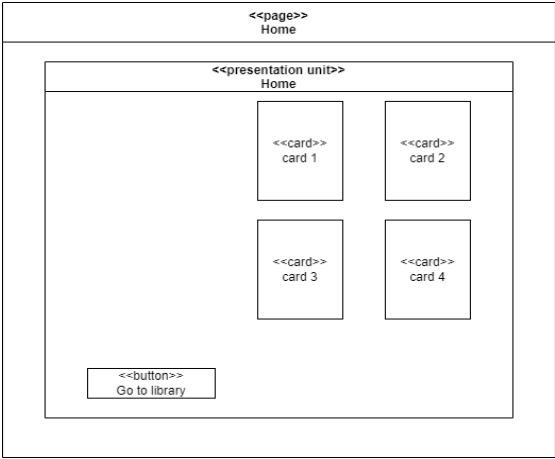
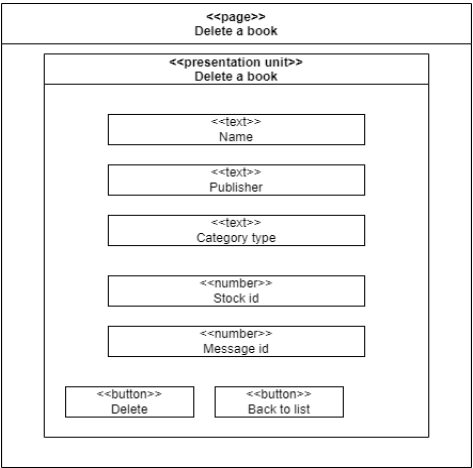
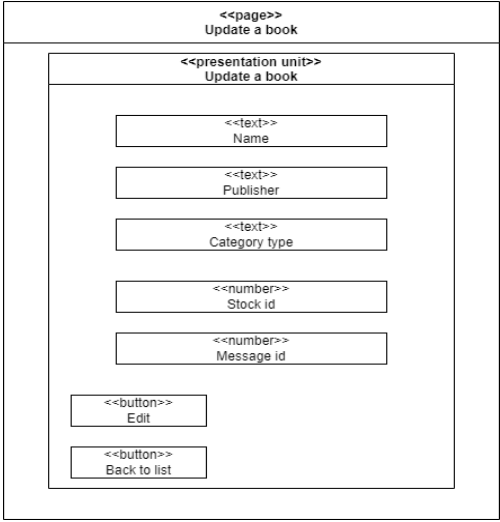
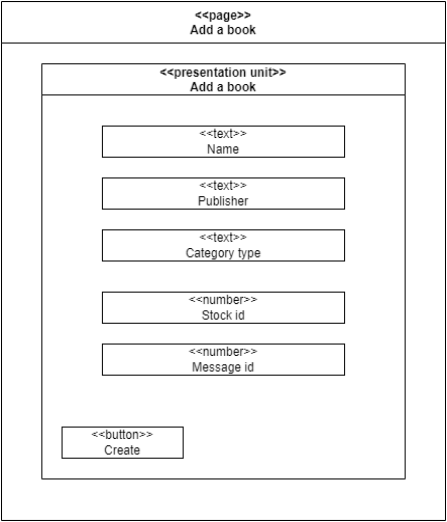
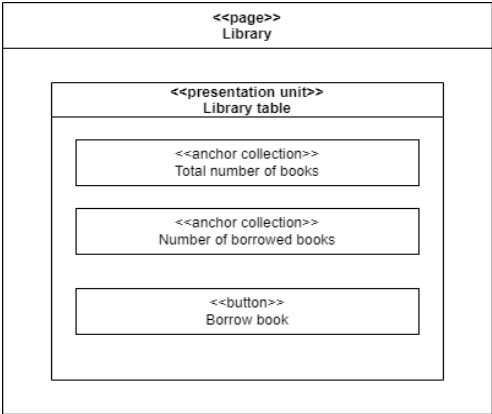
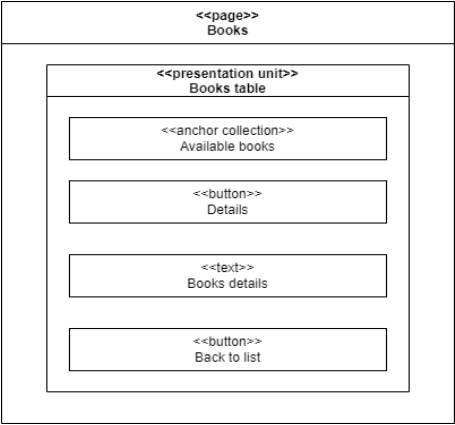


➤ Diagrama de activitatii (Activity Diagram):



➤ Paginile de prezentare:





## Tool-uri utilizate

---

Instrumentele pe care le voi folosi pentru a dezvolta aplicatia se regasesc detaliat in raportul preliminar, doar le voi enumera acum:

- Visual Studio
- Visual Studio Code
- Bootstrap
- GitHub

In cadrul acestei aplicatii se vor mai folosi si alte framework-uri, librarii care nu au fost detaliate in raportul preliminar:

- **HTML** -> este un limbaj de marcare utilizat pentru crearea paginilor web ce pot fi afisate intr-un browser. Scopul HTML este mai degraba prezentarea informatiilor – paragrafe, fonturi, tabele, etc. – decat descrierea semanticii documentului.
- **CSS** -> este un standard pentru formatarea elementelor unui document HTML. Stilurile se pot atasa elementelor HTML prin intermediul unor fisiere externe sau in cadrul documentului, prin elementul <style> sau atributul style.



- **JavaScript** -> este un limbaj de programare orientat pe obiecte bazat pe conceptul prototipurilor. Este folosit mai ales pentru introducerea unor functionalitati in paginile web, codul JavaScript din aceste pagini fiind rulat de catre browser.



- **ASP.NET MVC** -> este un framework de aplicatii web dezvoltat de Microsoft care implementeaza sablonul architectural MVC.



- **LocalDB** -> este o instant a SQL Server Express care poate crea si deschide baze de date SQL Server.



- **Entity Framework Core** -> este un framework open-source a tehnologiei de acces la date . Accesul la date se realizeaza folosind un model, fiind alcatuit din clasa de entitati si un obiect context care reprezinta o sesiune cu baza de date, acest obiect permitand interogarea si salvarea datelor.

## Arhitectura sistemului si structura bazei de date

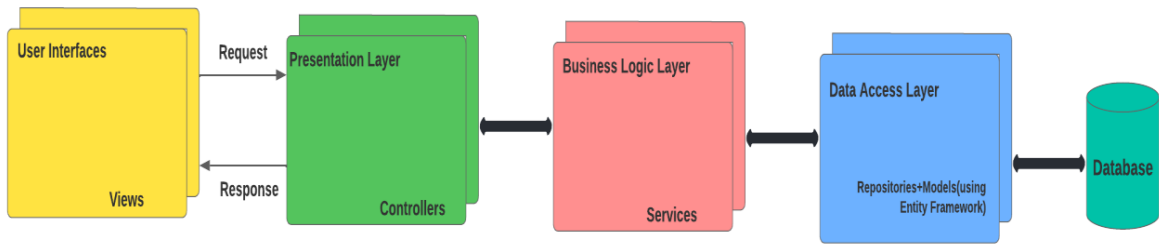
---

Aplicatia va fi dezvoltata folosind ASP.NET Core 6, Entity Framework Core versiunea 6.0.5 si ASP.NET Identity Core versiunea 6.0.5.

Va fi utilizat MVC (Model View Controller) si Repository Pattern. Controllerele aplicatiei fac parte din Presentation Layer, in Business Logic Layer sunt incluse serviciile. In Data Access Layer o sa incadrez Repository, unde fiecare repository va implementa o interfata specifica lui si va mosteni clasa de baza RepositoryBase, care are metode generice ca getAll, getByCondition, Update, Delete, Create, Save, etc si care implementeaza interfata IRepositoryBase.



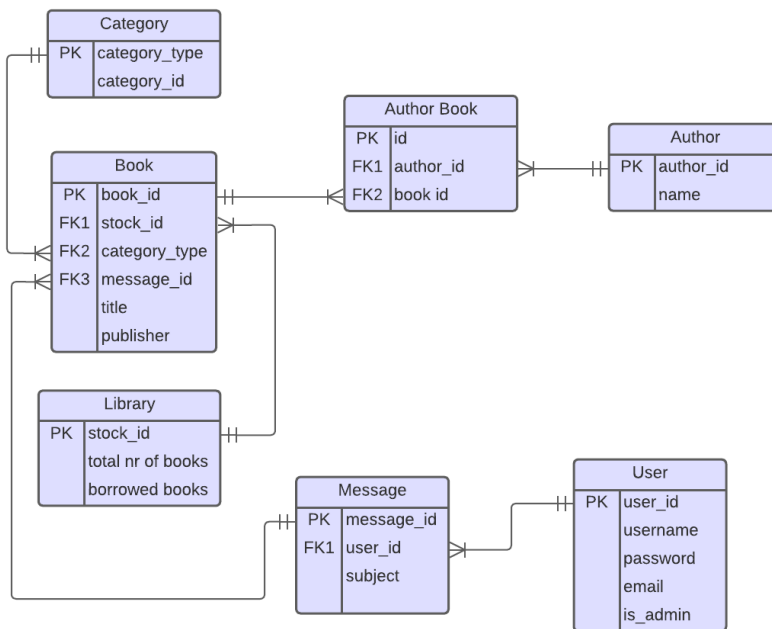
➤ **Arhitectura sistemului:**



Astfel se va respecta principiul Don't repeat yourself. Serviciile vor fi definite in clasa Program.cs si voi realiza injectia lor pe constructorul controllerelor in care vom sa folosesc serviciile respective (Dependency Injection).

Pentru autentificare si autorizare o sa folosesc ASP.NET Core Identity. O sa utilizez SQL Server localDB in Visual Studio 2022, unde baza de date este relationala, iar pentru utilizatori o sa folosesc IdentityUser.

➤ **Modelul Entitate-Legatura:**



## Concluzii

---



**Aplicatia Library Management va fi usor de utilizat deoarece toate paginile pot fi accesate din bara de navigatie, ele fiind impartite pe sectiuni.**

**Aplicatia va fi accesibila indiferent de echipamentul hardware sau denumirea browser-ului pe care utilizatorul le are la indemana, astfel daca tehnologia evolueaza, continutul aplicatiei ramane accesibil.**