



Universitatea Babeș-Bolyai, Facultatea de Matematică și Informatică

PROIECT – INGINERIA SISTEMELOR SOFTWARE

AN UNIVERSITAR 2022-2023

SEMESTRUL 2

Aplicație ISS.Biblioteca

Student:

Neacșu-Miclea Liviu-Ștefan

Grupa: 225/1

Specializarea: Informatică

Cadru didactic îndrumător:

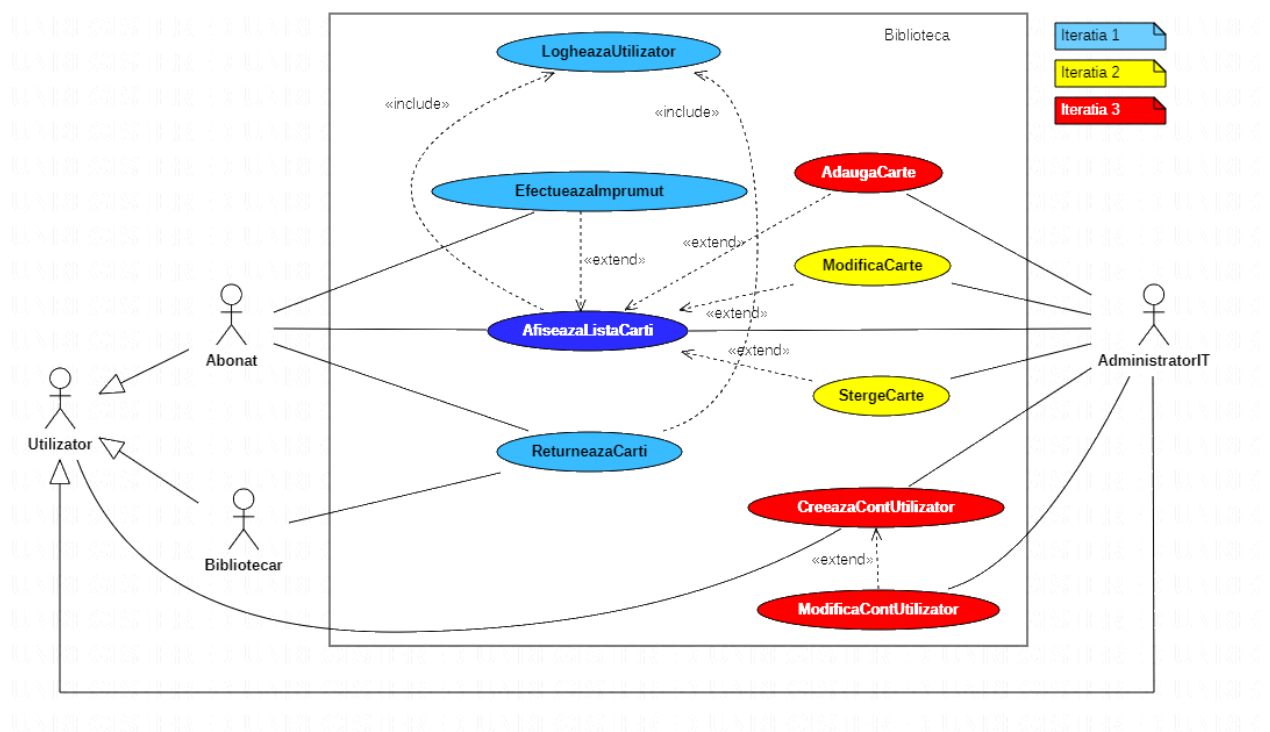
Drd. SIMA Ioan

1. Cerințele aplicației

O bibliotecă oferă abonaților săi o listă de cărți ce pot fi împrumutate. Pentru un abonat, se rețin în sistem (cel puțin) informații legate de cnp, nume, adresa, telefon și un cod unic de identificare a acestuia în cadrul bibliotecii. Fiecare carte poate exista în unul sau mai multe exemplare, identificate prin coduri unice. Abonații pot să împrumute cărți prin intermediul unei aplicații client online. Pentru a putea folosi o aplicație, un abonat trebuie să se autentifice. După autentificare, acesta vede lista exemplarelor disponibile în acel moment și poate împrumuta unul sau mai multe. Pentru restituirea cărților, există un singur punct de lucru, deservit de un bibliotecar. După fiecare împrumut/restituire, toți abonații conectați ai bibliotecii văd lista actualizată a cărților disponibile.

2. Modelul functional

2.1. Diagrama cazurilor de utilizare





2.2 Descrierea Cazurilor de Utilizare și a scenariilor alternative

ID and name	UC01 AfiseazaListaCarti		
Primary actor	Abonat/ AdministratorIT	Secondary actors	
Description	Un Abonat acceseaza sistemul bibliotecii printr-unul dintre terminalele disponibile, si vizualizeaza lista cartilor disponibile.		
Trigger	Un Abonat se conecteaza la terminalul bibliotecii.		
Preconditions	PRE-1. Abonatul este logat in sistem.		
Postconditions	POST-1. Terminalul afiseaza utilizatorului lista de carti.		
Normal flow	1.0 Afiseaza lista cartilor 1. Biblioteca afiseaza lista cartilor disponibile.		
Alternative flows	1.1 Cauta carte specifica 1. Biblioteca afiseaza lista cartilor disponibile. 2. Abonatul introduce date in meniul de filtrare. 3. Biblioteca afiseaza lista cartilor care se potrivesc cheilor de cautare.		
Exceptions	2a. Daca Abonatul anuleaza imprumutul, Biblioteca termina CU. 2b. Altfel daca Abonatul elimina o parte din cartile imprumutate din wishlist, Biblioteca reinitializeaza CU.		

ID and name	UC02 EfectueazaImprumut		
Primary actor	Abonat	Secondary actors	
Description	Un Abonat alege din lista cartilor disponibile pe acelea pe care doreste sa le imprumute. Biblioteca confirma si inregistreaza imprumutul, apoi actualizeaza lista cartilor disponibile		
Trigger	Un Abonat se conecteaza la terminalul bibliotecii.		
Preconditions	PRE-1. Abonatul este logat in sistem. PRE-2. Abonatul se afla in meniul de afisare a listei de carti. PRE-3. Abonatul nu are mai mult de 5 carti imprumutate si nereturnate.		
Postconditions	POST-1. Exemplarele imprumutate sunt asociate cu contul Abonatului si eliminate din lista publica.		
Normal flow	2.0 Imprumuta o singura carte 1. Abonatul selecteaza o carte pe care doreste sa o imprumute. 2. Biblioteca adauga cartea selectata intr-un wishlist temporar al Abonatului 3. Abonatul cere efectuarea imprumutului. 4. Biblioteca confirma imprumutul 5. Biblioteca inregistreaza imprumutul si actualizeaza lista cartilor disponibile.		



Alternative flows	<p>1.1 Imprumuta mai multe exemplare ale aceleiasi carti</p> <ol style="list-style-type: none"> Biblioteca afiseaza lista cartilor disponibile. Abonatul selecteaza o carte pe care doreste sa o imprumute si precizeaza numarul de exemplare. Se trece la pasul 3 normal flow. <p>1.2 Imprumuta mai multe carti</p> <ol style="list-style-type: none"> Biblioteca afiseaza lista cartilor disponibile. Abonatul selecteaza o carte din lista Se trece la pasul 1 normal flow
Exceptions	<p>1.0.E1. Wishlist-ul Abonatului este gol</p> <ol style="list-style-type: none"> In momentul confirmarii, Biblioteca anunta Abonatul ca nu are nicio carte in lista de imprumut a. Daca Abonatul anuleaza imprumutul, Biblioteca termina CU. b. Altfel daca Abonatul continua imprumutul, Biblioteca reinitiaza CU. <p>1.0.E2. Abonatul a depasit limita de carti nereturnate</p> <ol style="list-style-type: none"> Biblioteca anunta Abonatul ca nu mai poate imprumuta carti pana cand nu le returneaza pe cele deja imprumutate. Biblioteca termina CU <p>1.1.E1;</p> <p>1.2.E1. Abonatul a depasit limita de carti care pot fi imprumutate</p> <ol style="list-style-type: none"> Biblioteca anunta Abonatul ca doreste sa imprumute prea multe carti. a. Daca Abonatul anuleaza imprumutul, Biblioteca termina CU. b. Altfel daca Abonatul elimina o parte din cartile imprumutate din wishlist, Biblioteca reinitializeaza CU.

ID and name	UC03 ReturneazaCarti		
Primary actor	Bibliotecar	Secondary actors	Abonat
Description	Bibliotecarul preia cartile Abonatului		
Trigger	Abonatul aduce fizic cartile la unul dintre punctele de retur.		
Preconditions	PRE-1. Bibliotecarul este logat in sistem PRE-2. Abonatul ofera datele sale de identificare in sistem PRE-3. Cartile aduse sunt marcate ca imprumutate in contul Abonatului		
Postconditions	POST-1. Exemplarele imprumutate sunt marcate ca returnate in contul clientului POST-2. Cartile returnate sunt afisate in listele publice ale cartilor disponibile		
Normal flow	<p>2.0. Returneaza o singura carte</p> <ol style="list-style-type: none"> Abonatul prezinta cartea Bibliotecarului pentru returnare. Bibliotecarul acceseaza contul Abonatului pe baza datelor sale de identificare. Biblioteca afiseaza lista cartilor imprumutate de Abonat, inca nereturnate. 		



	4. Bibliotecarul inregistreaza cartea adusa ca fiind returnata. 5. Biblioteca actualizeaza lista cartilor disponibile.
Alternative flows	2.1. Returneaza mai multe carti 1. Bibliotecarul inregistreaza o carte adusa ca fiind returnata. 2. Se repeta pasul 3, normal flow.
Exceptions	2.0.E1. Abonatul nu poate fi identificat 1. Se termina CU 2.0.E2. Abonatul nu are carti nereturnate 1. Se termina CU 2.0.E3. Abonatul prezinta o carte care pe care nu a imprumutat-o 1a. Daca cartea nu apartine bibliotecii, se termina CU. 1b1. Altfel daca cartea apartine bibliotecii, dar a fost imprumutata de alt abonat, sau nu apare in lista de imprumuturi a niciunui abonat, atunci se aplica masurile legale corespunzatoare in baza termenilor si conditiilor. 1b2. Se termina CU.

ID and name	UC04 AdaugaCarte		
Primary actor	AdministratorIT	Secondary actors	
Description	Administratorul adauga o noua carte in sistem spre a putea fi imprumutata.		
Trigger	Un donator ofera Bibliotecii una sau mai multe carti care trebuie inregistrate in sistem. Administratorul se afla in meniul "AfiseazaListaCarti" si apasa butonul "adauga carte noua".		
Preconditions	PRE-1. Administratorul este logat in sistem		
Postconditions	POST-1. Cartea adaugata este vizibila in lista cartilor Bibliotecii.		
Normal flow	1.0. Adauga un nou titlu in sistem 1. Administratorul apasa butonul "Adauga o carte" din fereastra de administrator. 2. Administratorul introduce datele de identificare ale cartii (titlu, autor, ISBN etc). 3. Biblioteca valideaza datele introduse. 4. Administratorul introduce numarul de exemplare pe care sa le adauge. 5. Biblioteca confirma adaugarea cartilor. 6. Biblioteca actualizeaza listele cartilor disponibile.		
Alternative flows			
Exceptions	1.0.E1. Datele sunt invalide 1. Biblioteca instiinteaza Administratorul despre eroarea produsa (ex. Numele autorului nu e specificat). 2. Biblioteca anuleaza CU.		

ID and name	UC05 Modifica carte		
Primary actor	AdministratorIT	Secondary actors	



Description	Administratorul modifica informatiile unei carti deja stocate in sistem.
Trigger	Administratorul a selectat un tip de carte, fie din lista cartilor, fie prin ISBN/cod carte.
Preconditions	PRE-1. Administratorul este logat in sistem
Postconditions	POST-1. Modificarile sunt vizibile in lista cartilor Bibliotecii.
Normal flow	1.0. Modifica o carte 1. Administratorul apasa butonul “Modifica” din dreptul unei carti selectate. 2. Biblioteca afiseaza informatiile despre carte. 3. Administratorul efectueaza modificarile dorite. 4. Biblioteca confirma modificarea cartii. 5. Biblioteca actualizeaza listele cartilor disponibile.
Alternative flows	
Exceptions	1.0.E1. Datele sunt invalide 1. Biblioteca instiinteaza Administratorul despre eroarea produsa (ex. Numele autorului nu e specificat). 2. Biblioteca anuleaza CU.

ID and name	UC06 Sterge carte		
Primary actor	AdministratorIT	Secondary actors	
Description	Administratorul sterge o carte stocata in sistem.		
Trigger	Administratorul a selectat un tip de carte, fie din lista cartilor, fie prin ISBN/cod carte.		
Preconditions	PRE-1. Administratorul este logat in sistem		
Postconditions	POST-1. Cartea stearsa nu mai apare in lista cartilor Bibliotecii.		
Normal flow	1.0. Sterge un tip de carte 1. Administratorul apasa butonul “Sterge” din dreptul unei carti selectate. 2. Biblioteca confirma stergerea cartii. 3. Biblioteca actualizeaza listele cartilor disponibile.		
Alternative flows	1.1. Sterge un exemplar de carte 1. Administratorul selecteaza o carte. 2. Biblioteca afiseaza informatiile despre carte. 3. Administratorul decrementeaza numarul de exemplare. 4. Biblioteca confirma stergerea exemplarelor. 5. Biblioteca actualizeaza listele cartilor disponibile. 1.1.2 Sterge singurul exemplar al unei carti 1. Administratorul selecteaza o carte. 2. Biblioteca afiseaza informatiile despre carte. 3. Administratorul decrementeaza numarul de exemplare (devine 0). 4. Biblioteca confirma stergerea intregii carti. 5. Biblioteca actualizeaza listele cartilor disponibile.		
Exceptions	1.1.E1. Exemplarele de sters depasesc stocul		



	1. Biblioteca instiinteaza Administratorul ca vrea sa stearga mai multe exemplare decat exista in stoc. 2. Biblioteca reseteaza inputul si restarteaza CU.
--	---

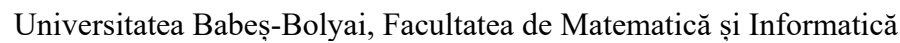
ID and name	UC07 LogheazaUtilizator		
Primary actor	Utilizator	Secondary actors	
Description	Utilizatorul se logheaza in sistem		
Trigger	Utilizatorul incepe o sesiune noua.		
Preconditions	PRE-1. Exista curent electric PRE-1. Aplicatia ruleaza		
Postconditions	POST-1. Utilizatorul este logat in sesiunea curenta.		
Normal flow	1.0. Logheaza Abonat 1. Utilizatorul da dublu click pe iconita aplicatiei. 2. Biblioteca afiseaza fereastra de login. 3. Utilizatorul introduce credentialele si apasa butonul de Login. 4. Biblioteca valideaza credentialele utilizatorului. 5. Biblioteca incepe CU01. Afiseaza lista carti intr-o fereastra de tip Abonat.		
Alternative flows	1.1. Logheaza Bibliotecar Pasii 1., 3. din 1.0. 3. Biblioteca incepe CU03. Returneaza carti 1.2. Logheaza Administrator Pasii 1., 3. din 1.0. 3. Biblioteca incepe CU01. Afiseaza lista carti intr-o fereastra de tip Administrator.		
Exceptions	1.1.E1. Date de identificare incorecte 1. Biblioteca anunta Utilizatorul ca a introdus credentiale gresite 2. Biblioteca anuleaza CU.		

ID and name	UC08 CreeazaContUtilizator		
Primary actor	AdministratorIT	Secondary actors	Utilizator
Description	Administratorul creeaza un nou cont de utilizator.		
Trigger	Un nou angajat (Bibliotecar/AdministratorIT) are nevoie de cont. Un Abonat doreste sa se inscrie in Biblioteca.		
Preconditions	PRE-1. Administratorul este logat in sistem		
Postconditions	POST-1. Contul noului utilizator este creat.		
Normal flow	1.0. Inregistreaza Utilizator 1. Administratorul introduce datele de identificare ale abonatului, precum si rolul acestuia in sistem. 2. Biblioteca valideaza datele introduse. 3. Administratoul inmaneaza utilizatorului datele de autentificare ale contului.		

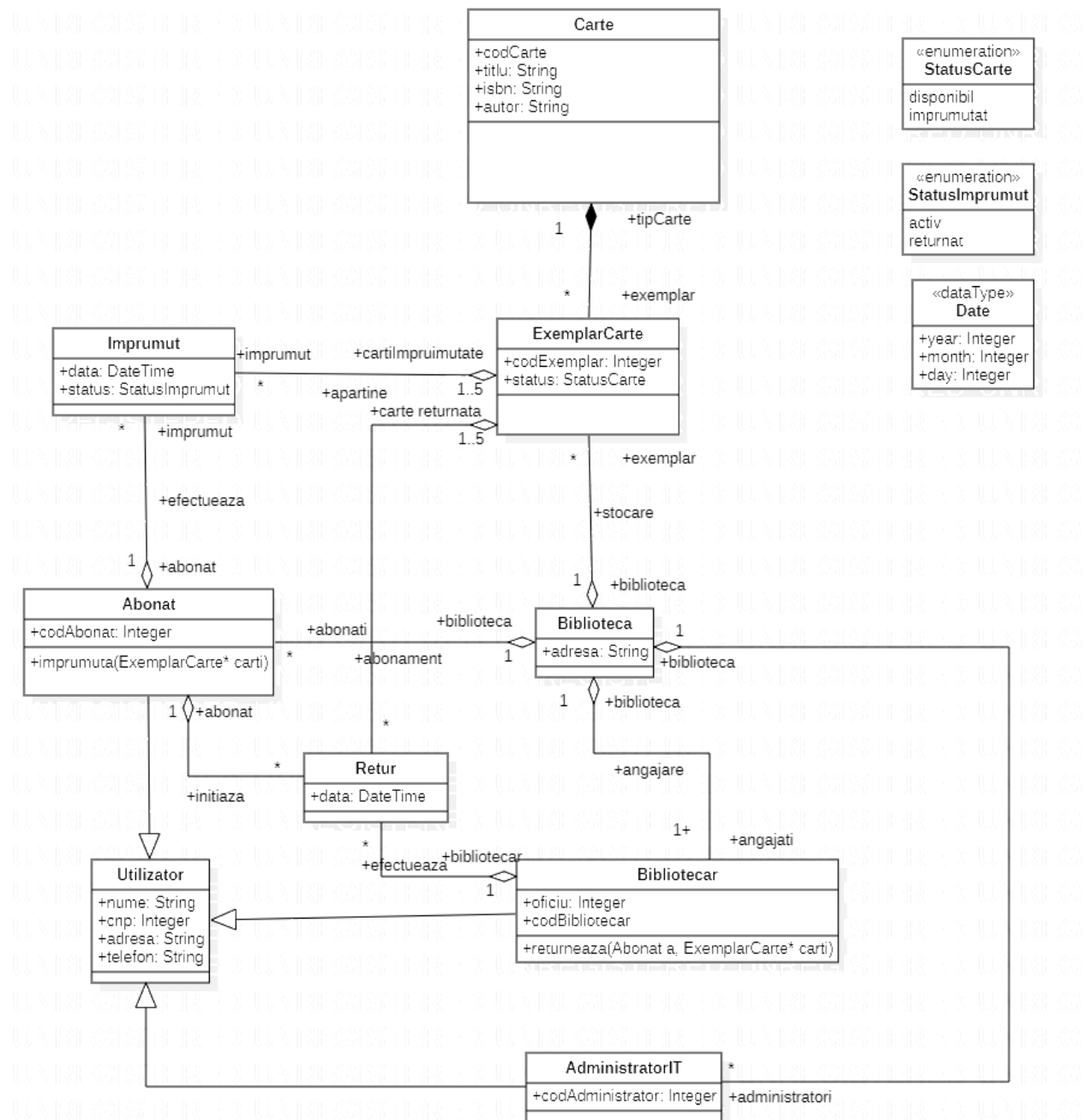


Alternative flows	
Exceptions	1.1.E1. Datele de inregistrare sunt invalide 1. Biblioteca anunta Administratoul despre eroare produsa (ex. Nume/CNP deja existent) 2. Biblioteca anuleaza CU.

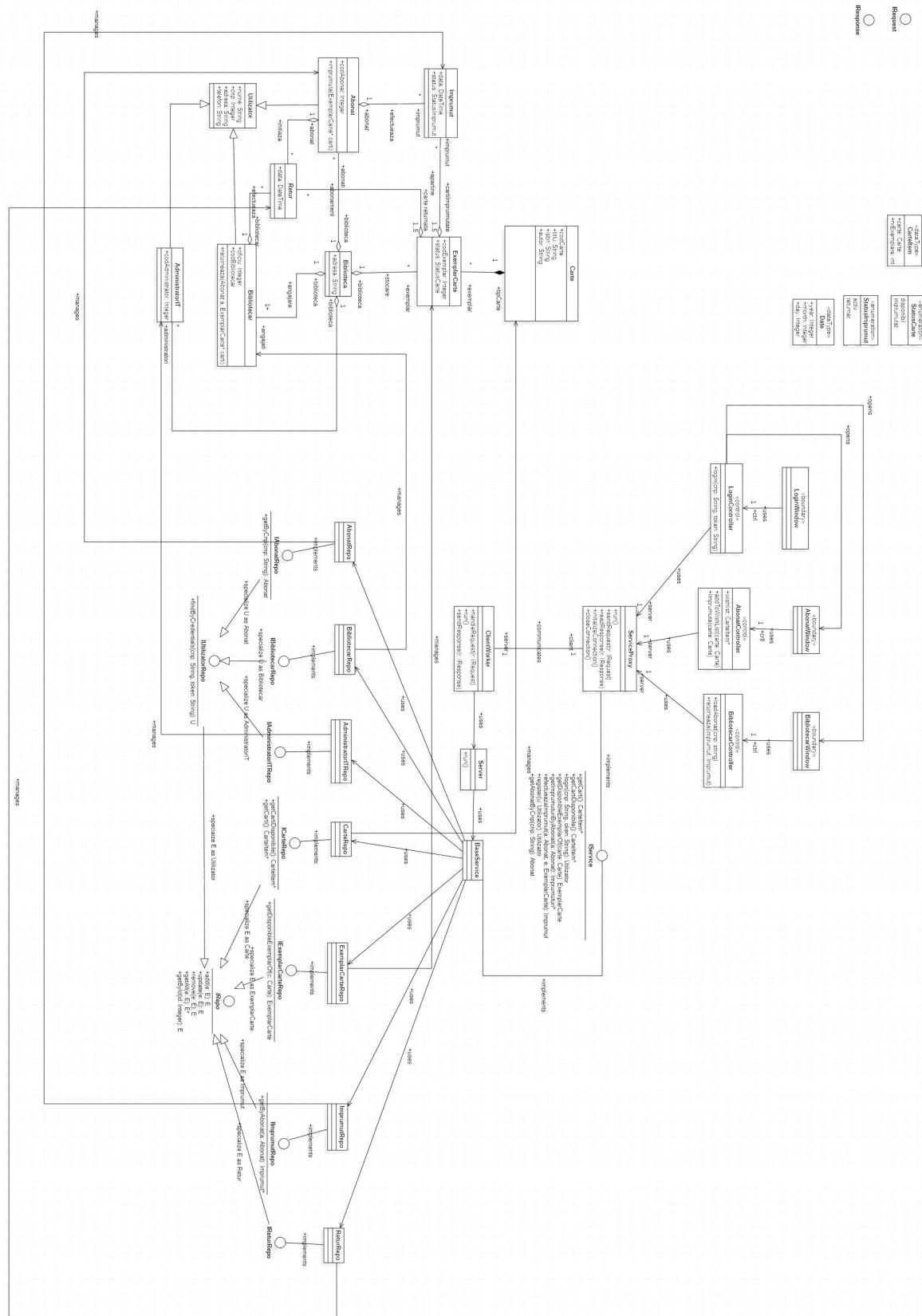
ID and name	UC08 ModificaContUtilizator		
Primary actor	AdministratorIT	Secondary actors	Utilizator
Description	In situatii speciale, administratorul poate modifica datele utilizatorilor din sistem – corectarea datelor personale, schimbarea parolei la solicitarea utilizatorului, stergerea contului		
Trigger	Un utilizator reclama ca a gasit greseli in datele contului personal. Un utilizator doreste sa isi schimbe parola. Un utilizator se retrage din sistem (un abonat isi inchide abonamentul, un angajat isi da demisia).		
Preconditions	PRE-1. Administratorul este logat in sistem		
Postconditions	POST-1. Modificarile contului sunt inregistrate in sistem.		
Normal flow	1.0. Modifica cont utilizator 1. Administratorul cauta datele de identificare ale contului de utilizator (cele care sunt presupuse corecte ex. CNP, sau nume). 2. Biblioteca valideaza datele introduse si afiseaza utilizatorul interesat. 3. Administratorul modifica datele contului semnalate ca incorecte. 4. Biblioteca actualizeaza datele contului de utilizator		
Alternative flows	1.1. Schimba parola 1. Administratorul introduce datele de identificare ale contului de utilizator. 2. Biblioteca valideaza datele introduse si afiseaza utilizatorul interesat. 3. Administratorul solicita generarea unei noi parola 4. Biblioteca afiseaza noua parola. 5. Administratoul inmaneaza utilizatorului datele de autentificare ale contului. 1.2. Sterge cont 1. Administratorul introduce datele de identificare ale contului de utilizator. 2. Biblioteca valideaza datele introduse si afiseaza utilizatorul interesat. 3. Administratorul solicita stergerea contului. 4. Biblioteca invalideaza contul selectat.		
Exceptions	1.1.E1. Datele de identificare sunt invalide 1. Biblioteca anunta Administratoul despre eroare produsa (ex. Nume/CNP inexistent) 2. Biblioteca anuleaza CU.		



3.1. Diagrama de clase a modelului

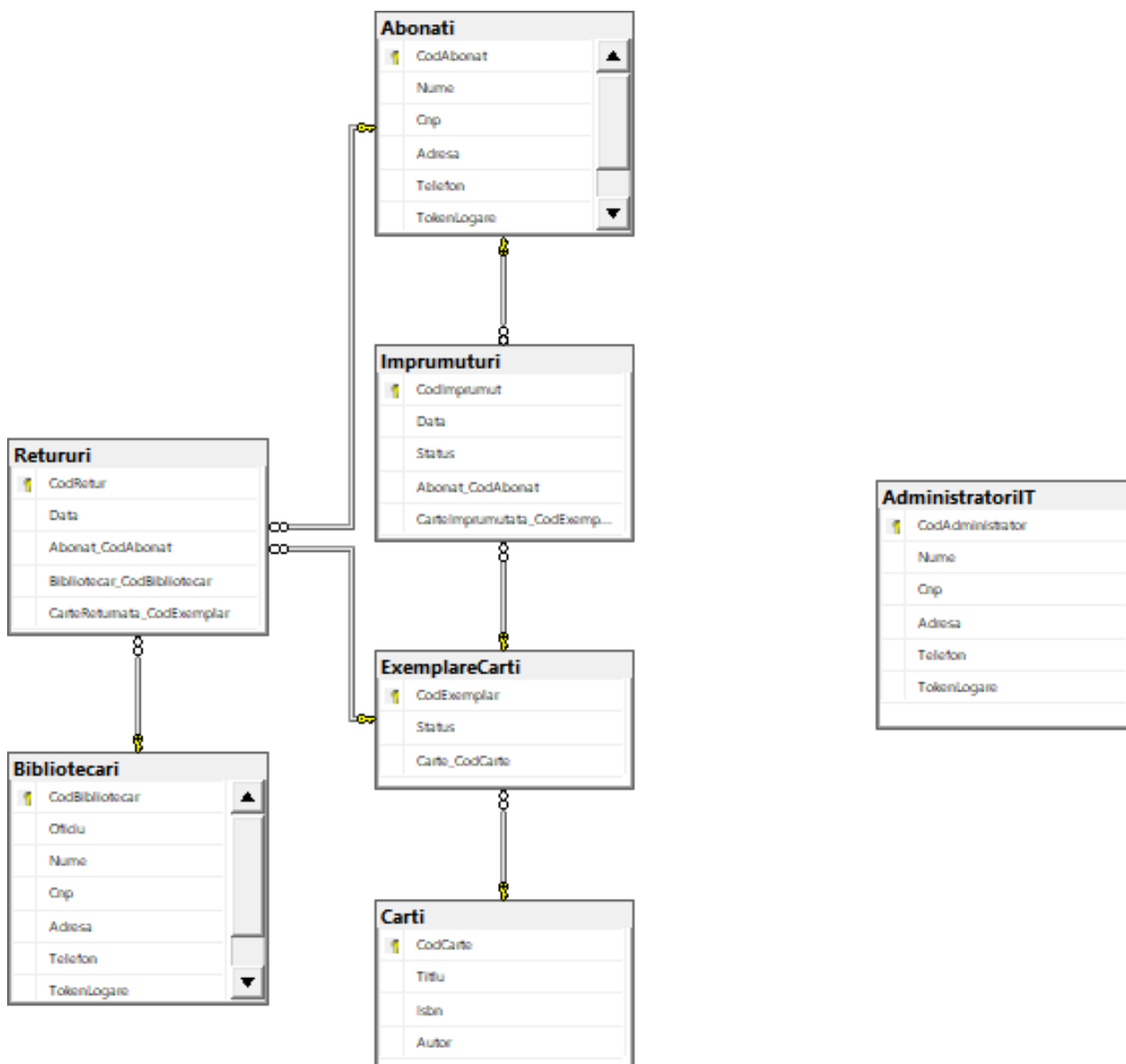


3.2. Diagrama de clase finisate





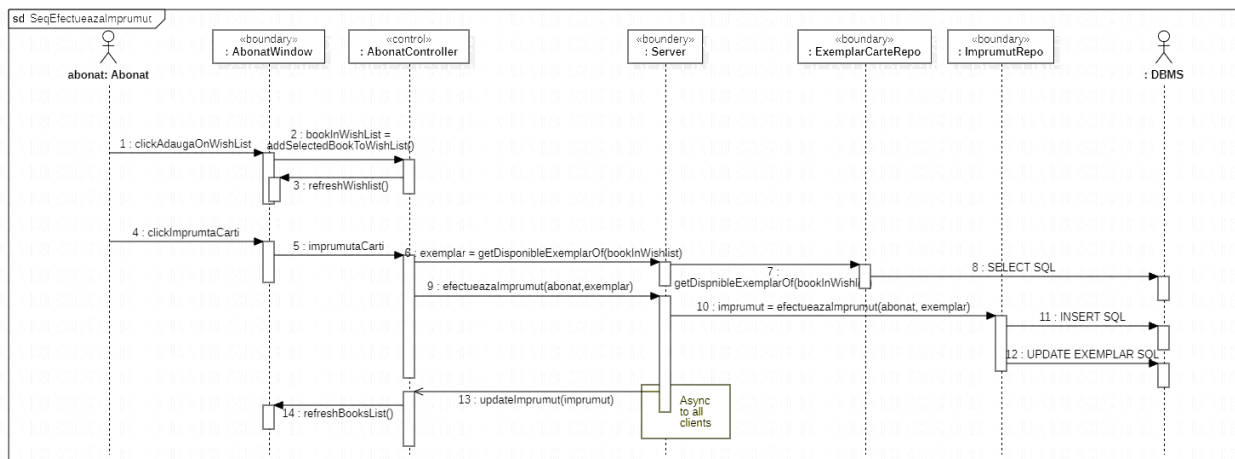
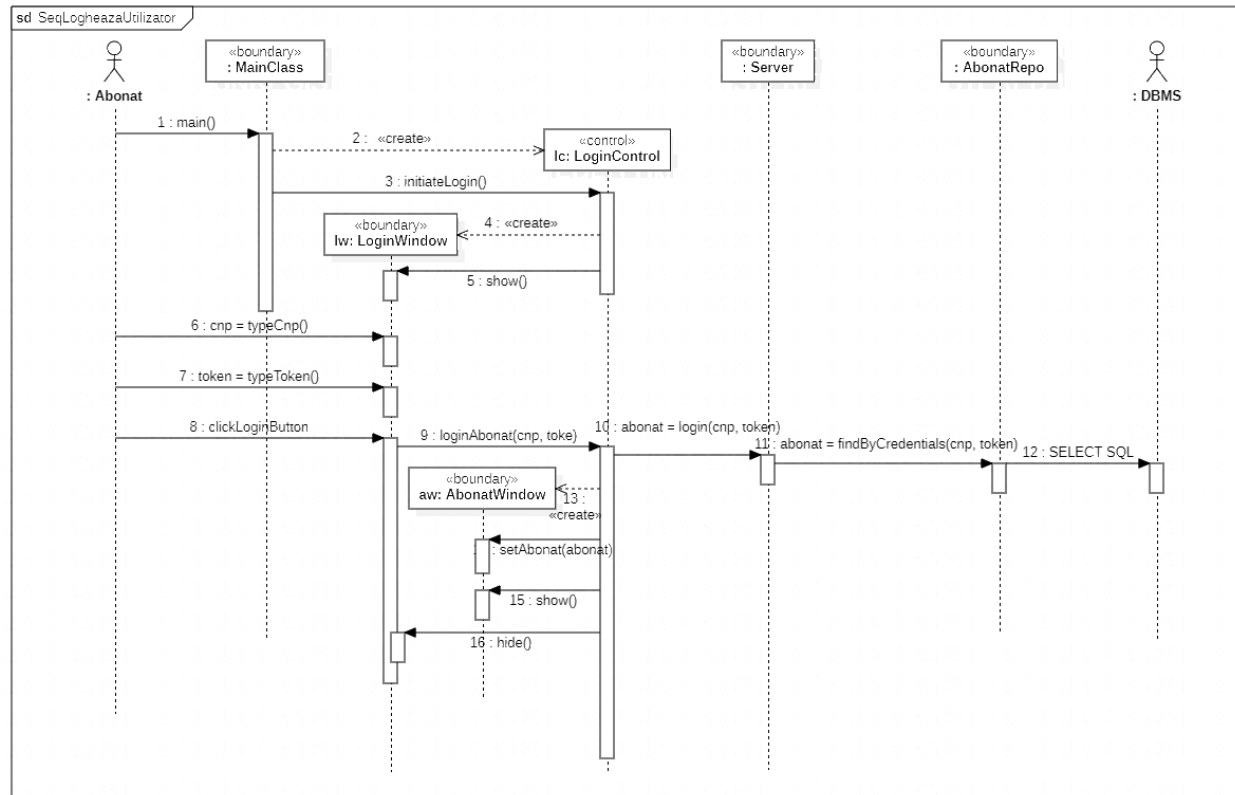
4. Diagrama bazei de date

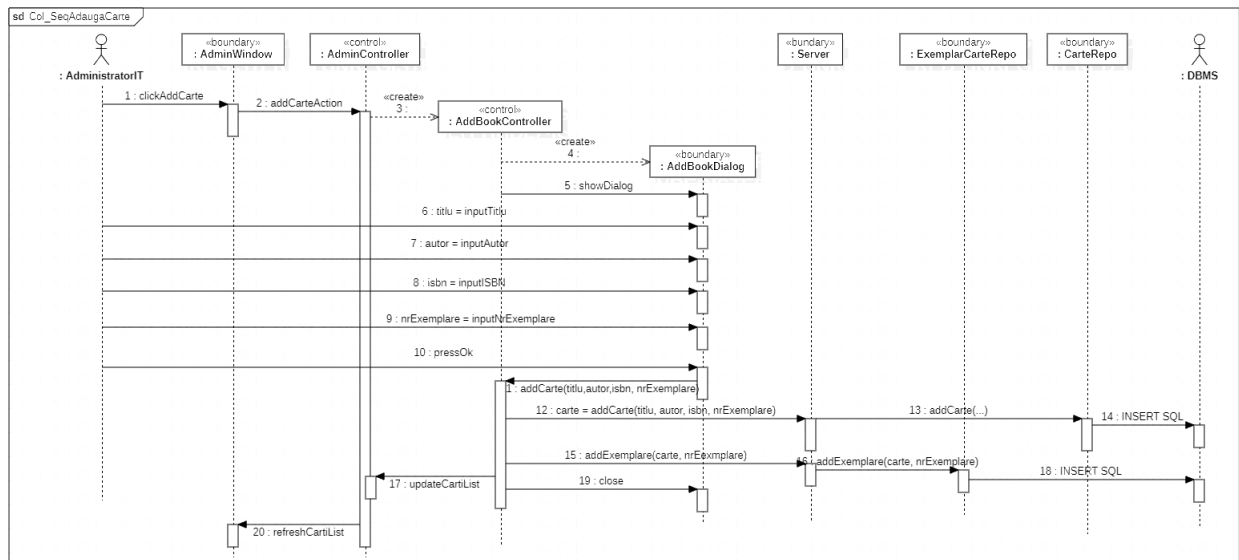
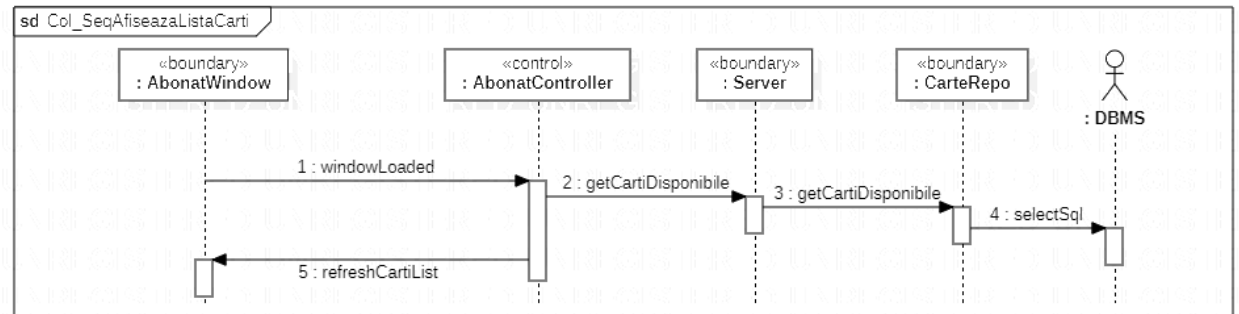
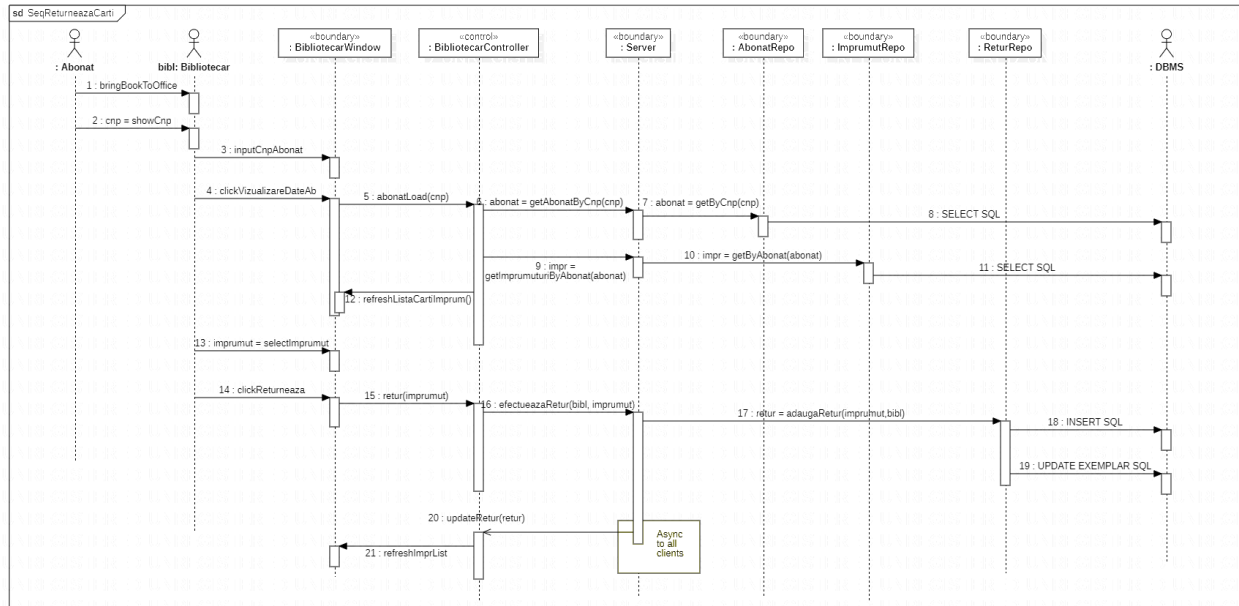


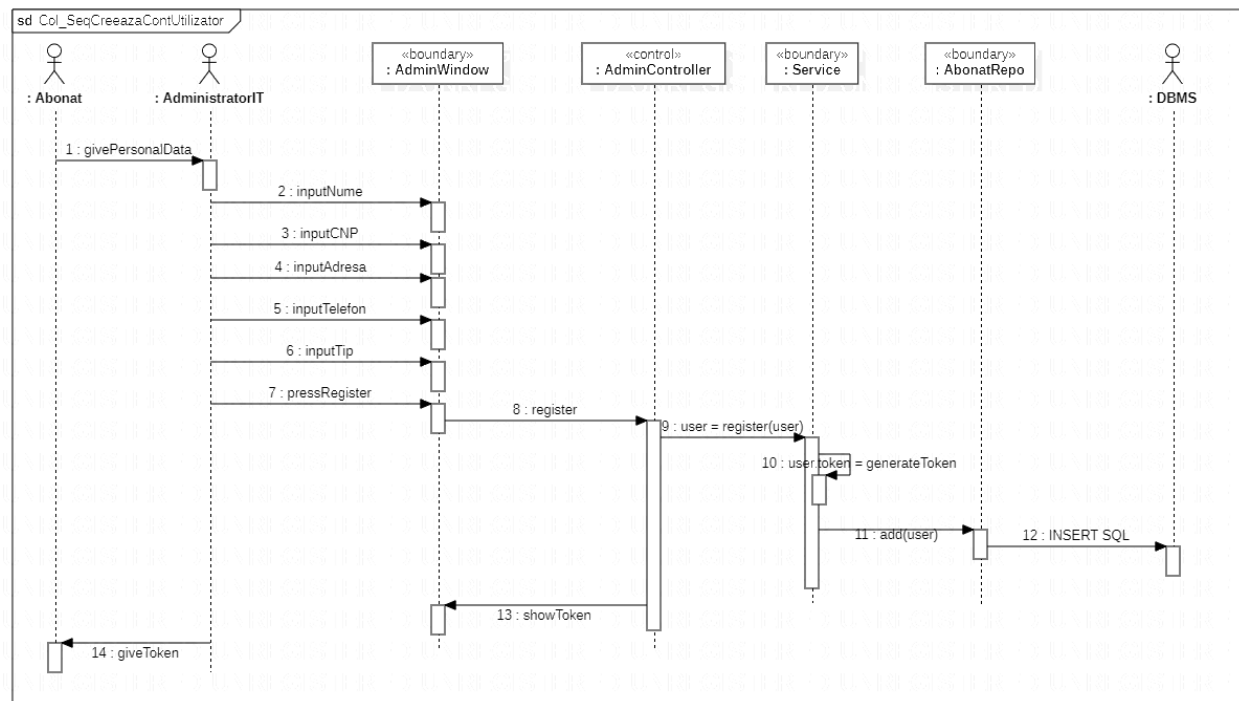
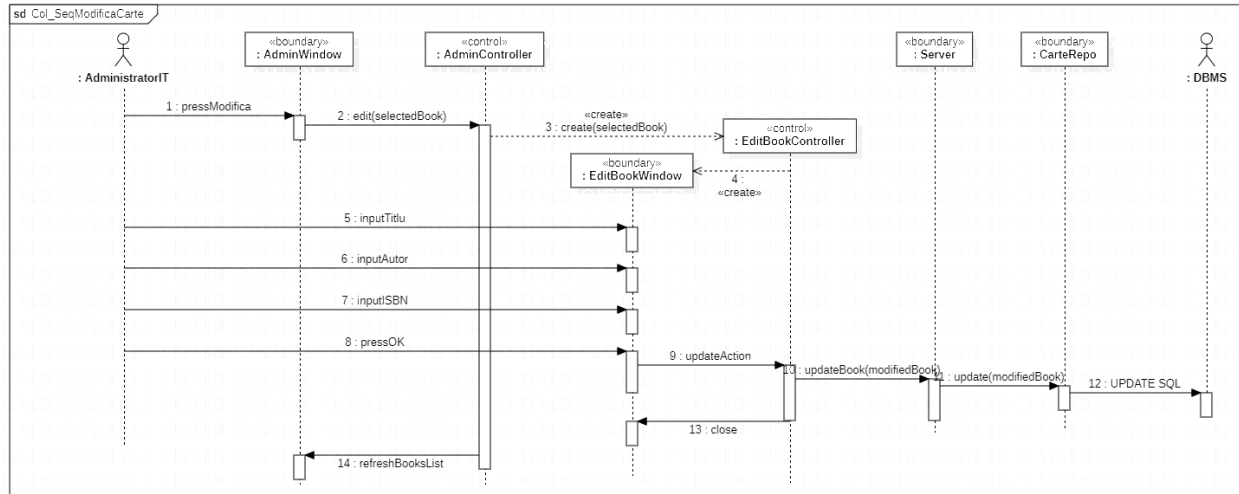


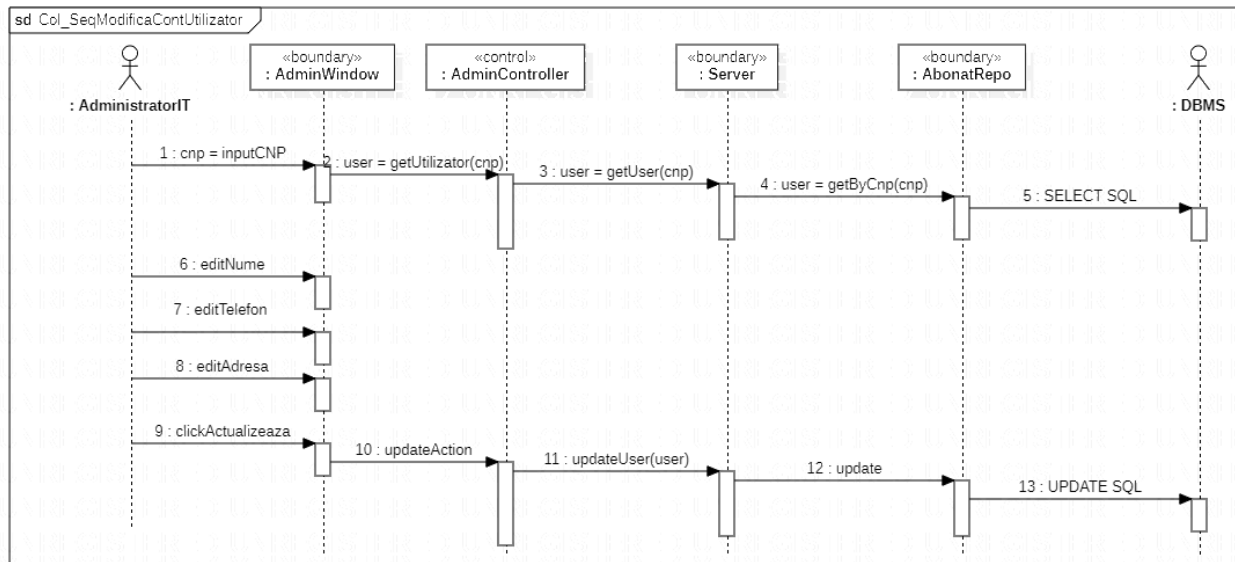
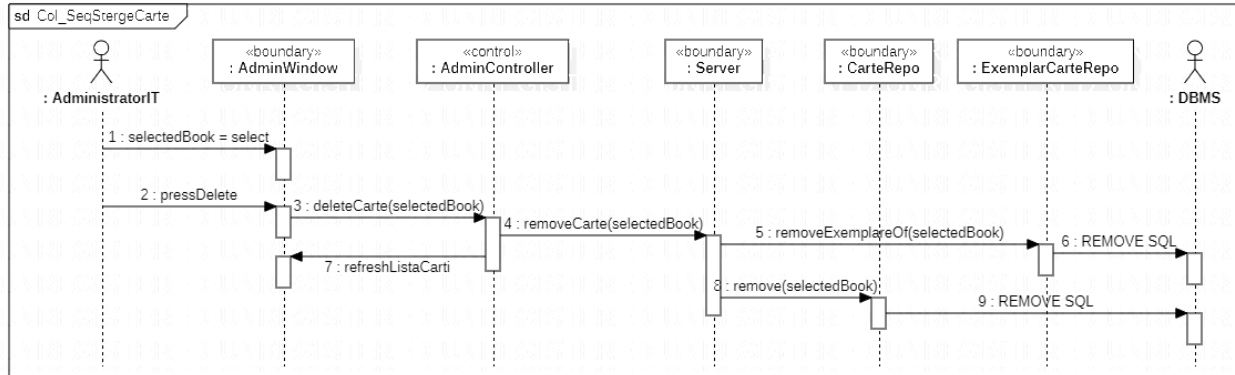
5. Modelul Dinamic

5.1. Diagramele de secvență

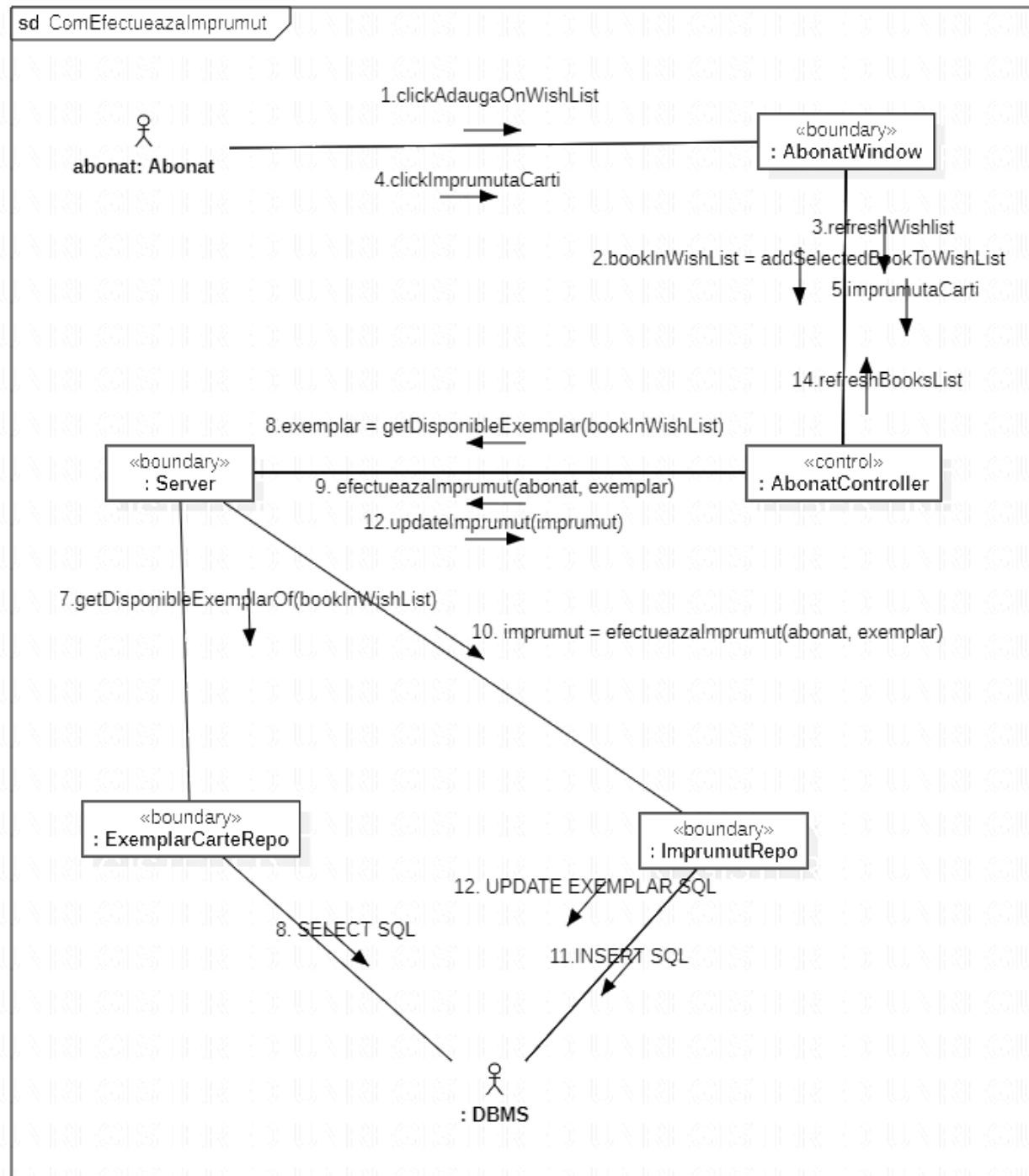






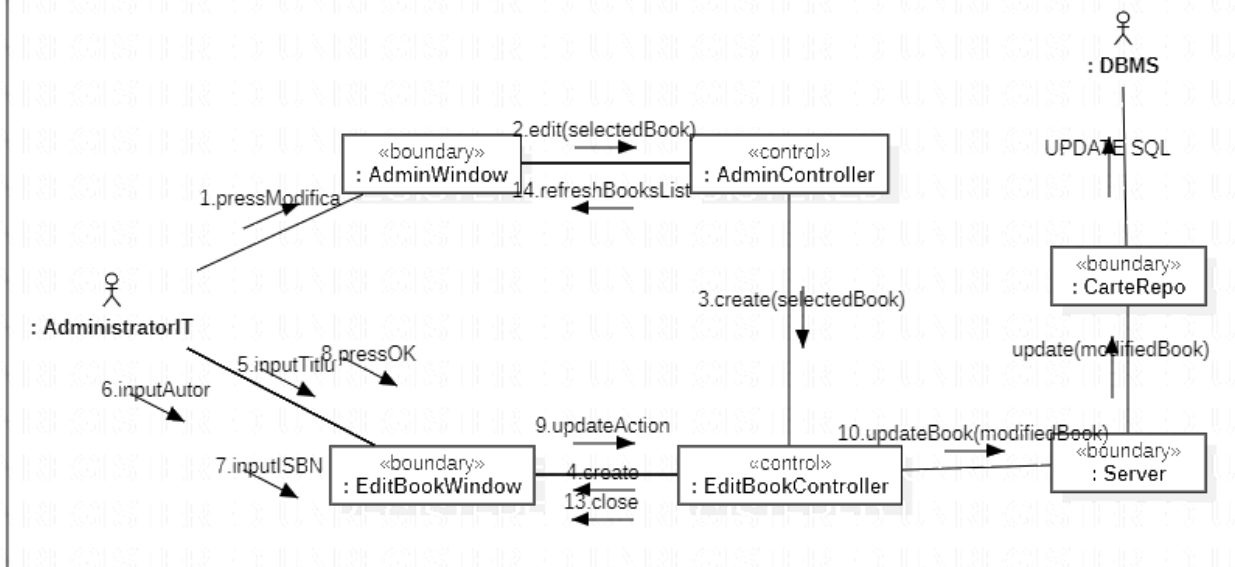


5.2. Diagrame de comunicare

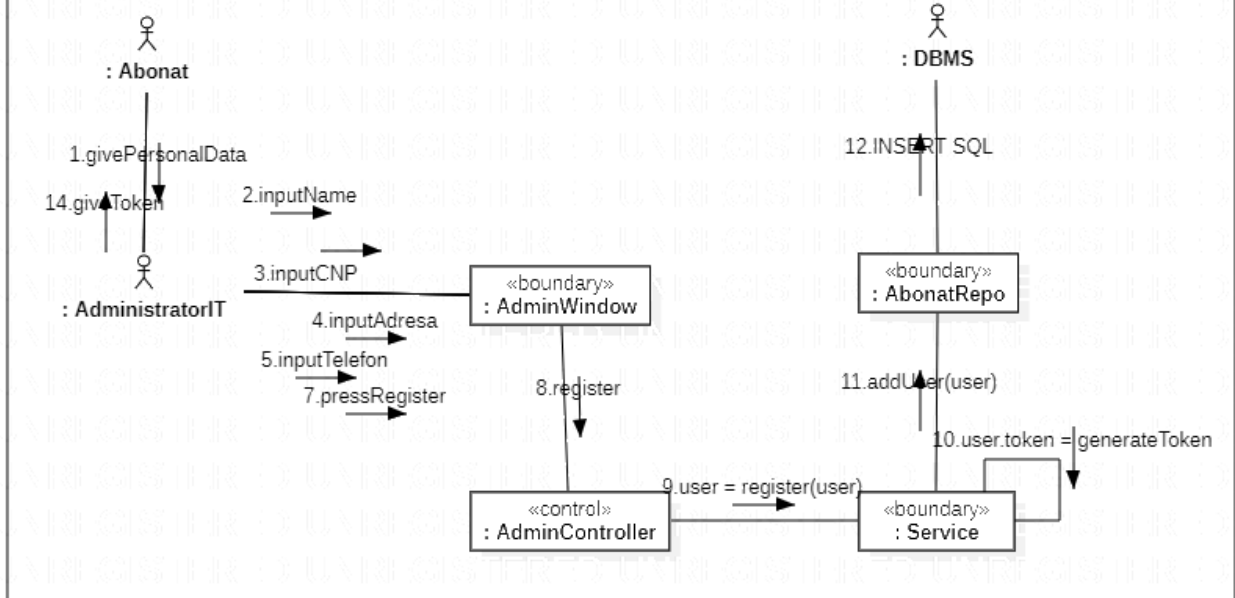




sd ComModificaCarte



sd ComCreeazaContUtilizator





6. Documentație tehnică

6.1. Bază de date

SQLite este un sistem de baze de date simplu și ușor de folosit. O bază de date SQLite are forma unui singur fișier care poate fi utilizat fără a fi nevoie să instalăm sisteme de gestiune specializate. Am ales SQLite datorită proprietăților sale lightweight, a portabilității, a ușurinței de a face backup și a capacității sale de integrare (embedded database), ceea ce îl face potrivit pentru aplicațiile de mici și medii dimensiuni, cum ar fi gestiunea unei biblioteci.

6.2. Mediul de dezvoltare

Aplicația a fost dezvoltată folosind **Visual Studio 2022**, datorită performanțelor sale ridicate în dezvoltarea software pentru Windows, a ușurinței de a descărca și folosi dependențele prin intermediul pachetelor Nuget, precum și a debugger-ului performant și a interfeței intuitive care sporește capacitatea de concentrare a programatorului.

6.3. Limbajul de programare

C# este un limbaj de programare de nivel înalt dezvoltat de Microsoft și este un instrument foarte eficient de a dezvolta aplicații Windows în special. Este un limbaj orientat obiect și dispune de calități deosebite, cum ar fi extensiile de clase, atributele sau sistemul de interogări LINQ foarte adaptabil. Spre deosebire de alte limbaje, codul C# tinde să fie mai scurt, ușor de înțeles și de întreținut și, pentru un limbaj rulat pe o mașină virtuală, tinde să aibă performanțe comparabile cu limbajele de nivel mediu, de exemplu C++. După analizele noastre, C# este un punct de echilibru între claritate și eficiență.

6.4. ORM

Obiect-relational mapping (ORM) este o tehnică de programare care asigură un liant între o bază de date relațională și memoria gestionată de un limbaj de programare. Cu ajutorul ei, se poate scrie cod care este independent de sursa care furnizează datele. ORM-ul utilizat în proiectul de față este **Entity Framework 6.0**, mai precis metoda de generare a bazei de date din definițiile claselor, datorită intuitivității sale ridicate și a libertății de expresie în cod a interogărilor.

6.5. Proiectarea diagramelor

Am folosit **StarUML** pentru proiectarea diagramelor deoarece permite gestionarea implicită a entităților componente ale proiectului, planificarea cazurilor de utilizare și a iterațiilor.

6.6. GUI

Am folosit **WPF** pentru interfața grafică deoarece implementarea sa bazată pe vectori, nu pe pixeli, permite realizarea unui design modern și adaptabil, iar posibilitatea de a defini ferestrele și stilurile în XAML oferă control larg și opțiuni variate, sporind totodată productivitatea.

6.7. Arhitectura aplicației

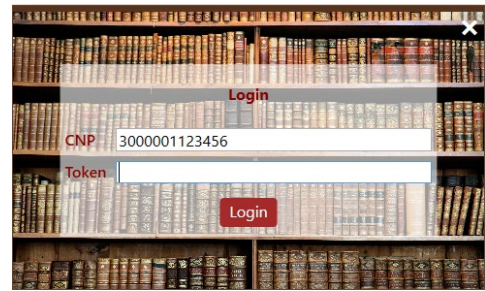
Codul este structurat folosind șablonul Domain-Driven Design, având următoarele straturi:

- **Domain:** definește entitățile și proprietățile lor, care se reflectă și în structura bazei de date
- **Repository:** conține metode de manipulare a ORM-ului
- **Service:** implementează funcționalitățile aplicației
- **Controller:** face accesibile metodele din Service utilizatorului, prin intermediul interfeței grafice

Aplicația este de tipul client-server și folosește o implementare bazată pe servicii proxy.

7. Help

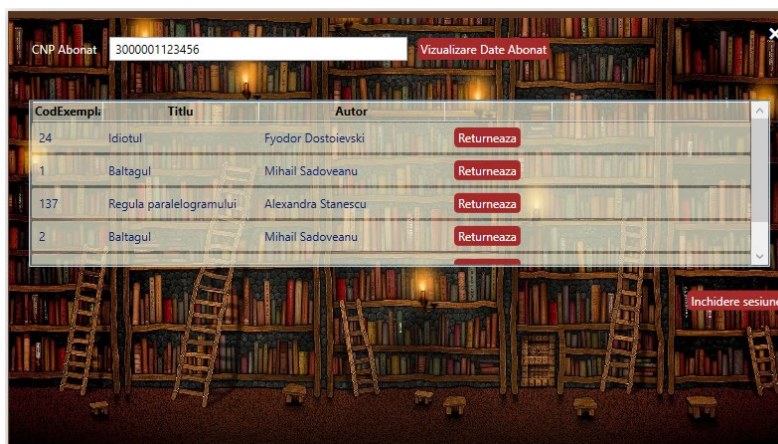
- **Pornirea aplicației.** Utilizatorul apasă iconița „ISS.Biblioteca.exe” din directorul aplicației
- **Login.** Se deschide fereastra de login, în care utilizatorul își introduce CNP-ul și token-ul personal primit de la un sediu al administrației bibliotecii. La apăsarea butonului „Login”, în funcție de statutul utilizatorului (Abonat, Bibliotecar, Administrator), se va deschide o fereastră corespunzătoare.



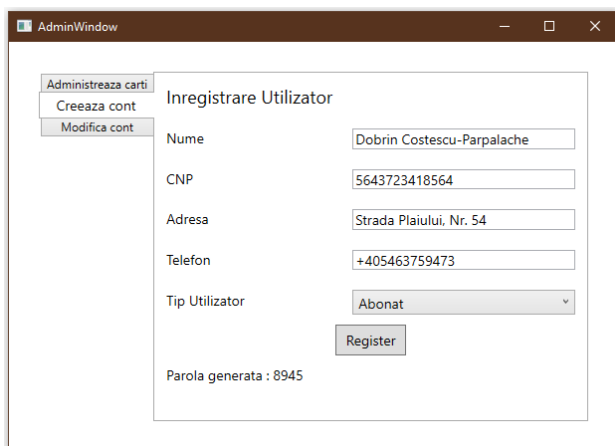
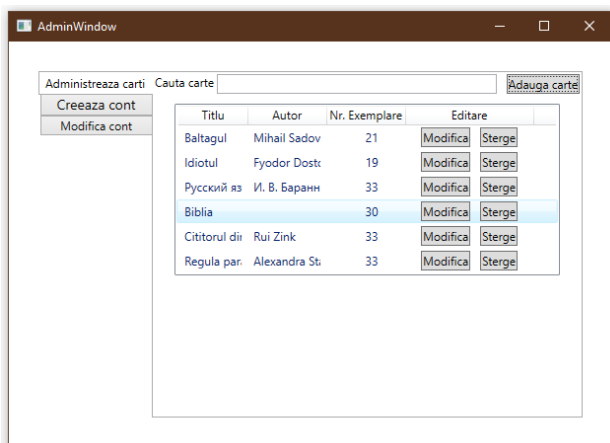
- **Abonat – împrumută cărți.** În fereastra abonatului, acesta poate alege o carte dorită, și apasă butonul „Adaugă în wishlist” din dreptul ei pentru a o plasa într-un wishlist temporar, din care poate modifica dacă dorește și numărul de exemplare. Un abonat poate avea maxim 5 exemplare împrumutate. Finalizarea împrumutului se realizează prin apăsarea butonului „Împrumută cărți”. Ulterior, abonatul este așteptat la un sediu al bibliotecii pentru a ridica exemplarele

împrumutate. Dacă abonatul se răzgândește, poate apăsa butonul „Anulează”.

- **Bibliotecar – efectuează retur.** În fereastra bibliotecarului, acesta introduce CNP-ul abonatului și apasă butonul „Vizualizare Date Abonat” pentru a vedea împrumuturile sale. În lista împrumuturilor, acesta poate apăsa butonul „Returnează” din dreptul unei cărți pentru a efectua returnul. Pentru ieșirea din aplicație, se folosește iconița „X” sau butonul „Închidere sesiune”.

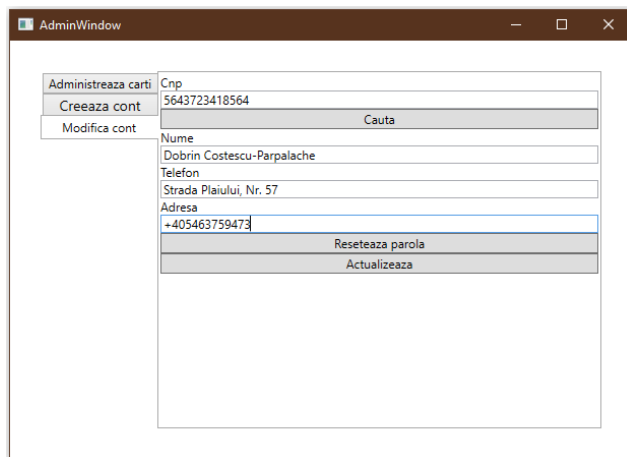


- **Administrator – Editează cărți.** Din fereastra administratorului, tabul „Administrează cărți”, acesta poate alege să adauge sau să modifice o carte, prin apăsarea butonului „Adaugă o carte”, sau prin apăsarea butonului „Modifică” din dreptul unei cărți. Se va deschide o fereastră de dialog în care administratorul introduce (sau actualizează) titlul cărții, numele autorului, ISBN-ul și numărul de exemplare, apoi se apasă OK. Cartea adăugată/modificată va apărea în lista cărților din fereastra principală. Ștergerea unei cărți se realizează prin acționarea butonului „Șterge” din dreptul acesteia.



- **Administrator – Creează cont.** Din fereastra administratorului, tabul „Creează cont”, acesta poate crea un cont nou unui utilizator al bibliotecii. Se introduc datele personale ale utilizatorului în câmpurile destinate și se alege rolul noului utilizator (Abonat, Bibliotecar, Administrator). După apăsarea butonului Register, se va genera un token de autentificare care este înmănat utilizatorului, și care va fi folosit la autentificare în clienții bibliotecii.

- **Administrator – Modifică cont.** Din fereastra administratorului, tabul „Modifica cont”, acesta poate căuta un utilizator după CNP și se vor afișa datele utilizatorului existente în baza de date. Administratorul poate corecta aceste date, care se vor actualiza în baza de date după apăsarea butonului „Actualizează”, sau poate cere resetarea parolei (în cazul în care utilizatorul a uitat-o), prin apăsarea butonului „Resetează parola”.





8. Bibliografie

1. <https://www.entityframeworktutorial.net/code-first/what-is-code-first.aspx>
2. <https://en.wikipedia.org/wiki/SQLite>
3. <https://stackoverflow.com/questions/52718652/ef-core-sqlite-sqlite-error-19-unique-constraint-failed>
4. <https://sqlitestudio.pl/>
5. <https://wpf-tutorial.com/listview-control/listview-data-binding-item-template/>
6. <https://www.c-sharpcorner.com/uploadfile/mahesh/listview-in-wpf/>
7. <https://github.com/msallin/SQLiteCodeFirst>
8. <https://stackoverflow.com/questions/8142498/howto-sqlite-with-entityframework-and-code-first>
9. Medii de proiectare și programare 2022-2023, Curs 7, Conf. COJOCAR Grigoreta
10. <https://simaioan.wordpress.com/2023/03/13/seminarii-iss-2021/>