# Uvod

Bookshelf je sustav koji omogućava članovima knjižnice pretraživanje, pregledavanje i rezervaciju knjiga dostupnih u svim knjižnicama nekog knjižničnog sustava. Intuitivno i jednostavno korisničko sučelje omogućava korisnicima svih uzrasta jednostavno i učinkovito navigiranje po aplikaciji i korištenje usluga knjižničnog sustava.

Aplikacija omogućava korisniku da prije odlaska u knjižnicu, kako bi posudio knjigu, provjeri dostupnost pojedine knjige i eventualno je rezervira. Na taj način članovi knjižnice ne moraju obilaziti više knjižnica u slučaju da tražena knjiga ne postoji u knjižnici ili je posuđena/rezervirana. Korisnik se pri početku rada prijavljuje u sustav informacijama koje su mu dane pri učlanjivanju kako bi potvrdio da je član knjižnice. Nakon uspješne prijave, korisnik može koristiti funkcionalnosti sustava.

Prijavom u sustav, korisnik može pretraživati knjige dostupne u knjižničnom sustavu. Pretragom korisnik vidi listu relevantnih knjiga i nude mu se opcije za pregled detalja o knjizi ili pregled dostupnosti odabrane knjige u svim knjižnicama sustava.

Pregledom detalja o knjizi, korisnik pregledava informacije o autoru, izdavaču i godini izdanja odabrane knjige kako bi razlučio između različitih izdanja istog djela i kako bi lakše odabrao izdanje koje traži.

Pregledom dostupnosti odabrane knjige u svim knjižnicama koje su dio sustava, korisnik može vidjeti sve knjižnice koje imaju navedeni naslov, te u kojim je sve knjižnicama tražena knjiga na raspolaganju.

Pregledom informacija o pojedinoj knjižnici, korisnik može pronaći nove knjižnice u njegovoj blizini za koje nije ni znao da postoje. Nude se informacije o nazivu, ulici, telefonskom broju i radnom vremenu odabrane knjižnice.

U slučaju da postoje knjige koje su vrlo tražene i često nisu dostupne u knjižnicama, sustav nudi funkcionalnost rezervacije odabrane knjige kako bi se izbjegao uzaludan odlazak člana u knjižnicu, u slučaju da su svi primjerci tražene knjige već posuđeni.

Rad korisnika u sustavu završava odjavom korisnika.

Sustav Bookshelf je građen po uzoru na postojeći sustav „Katalog Knjižnica grada Zagreba“ u svrhu promicanja modernizacije knjižničnih sustava za brojna mjesta koja još nemaju organiziran sustav koji bi olakšao i popravio korisničko iskustvo korištenja usluga knjižnice.

# Zahtjevi

|  |  |
| --- | --- |
| **Naziv** | **UC1 – Prijava korisnika u sustav** |
| **Opis** | Korisnik se prijavljuje kako bi mogao pregledavati knjige dostupne za rezervaciju |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik postoji u sustavu |
| **Okidač** | Korisnik pokušava pristupiti sustavu uz pomoć izbornika za pristup |
| **Glavni uspješni scenarij** | 1. Korisnik unosi korisničko ime i lozinku kako bi se prijavio. 2. Sustav provjerava postoji li korisnik sa navedenim podacima. 3. Uneseni podaci su autentični – korisniku se omogućuje pristup sustavu. |
| **Ekstenzije** | 1. Uneseni podaci nisu autentični.    1. Sustav obavještava korisnika o neispravnosti unesenih podataka.    2. Korisnik ponavlja postupak započevši od koraka 1 glavnog scenarija. |
| **Uvjeti koji trebaju biti zadovoljeni** | Korisnik je prijavljen. |

|  |  |
| --- | --- |
| **Naziv** | **UC2 – Pretraga knjiga** |
| **Opis** | Korisnik pretražuje kolekciju knjiga unosom naslova knjige. |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Korisnik odabire opciju pretrage klikom na gumb |
| **Glavni uspješni scenarij** | 1. Korisnik unosi naslov knjige u odgovarajuću formu za unos 2. Korisnik odabire opciju pretrage 3. Sustav pretražuje bazu podataka i vraća kolekciju knjiga koje zadovoljavaju dani uvjet 4. Korisnik odabire pregled detalja o knjizi ili pregled dostupnosti knjige u knjižnicama |
| **Ekstenzije** | 1. Knjiga ne postoji u bazi podataka    1. Sustav obavještava korisnika da ne postoji knjiga koja zadovoljava rezultate pretrage    2. Korisnik odustaje od pretrage ili mijenja pojam koji pretražuje |
| **Uvjeti koji trebaju biti zadovoljeni** |  |

|  |  |
| --- | --- |
| **Naziv** | **UC3 – Pregled detalja o knjizi** |
| **Opis** | Korisnik pregledava detalje o knjizi. |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Korisnik odabire opciju pregleda detalja knjige klikom na gumb iz prikaza liste knjiga dobivene pretragom |
| **Glavni uspješni scenarij** | 1. Korisnik odabire knjigu za koju želi vidjeti detalje iz liste dobivene pretragom knjiga 2. Sustav prikazuje detalje za danu knjigu |
| **Ekstenzije** |  |
| **Uvjeti koji trebaju biti zadovoljeni** |  |

|  |  |
| --- | --- |
| **Naziv** | **UC4 – Pregled dostupnosti knjige u knjižnicama** |
| **Opis** | Korisnik pregledava dostupnost odabrane knjige u svim dostupnim knjižnicama. |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Korisnik odabire opciju pregleda dostupnosti knjige klikom na gumb iz prikaza liste knjiga dobivene pretragom |
| **Glavni uspješni scenarij** | 1. Korisnik odabire knjigu za koju želi vidjeti dostupnost iz liste dobivene pretragom knjiga 2. Sustav prikazuje dostupnost u svim knjižnicama za danu knjigu |
| **Ekstenzije** |  |
| **Uvjeti koji trebaju biti zadovoljeni** |  |

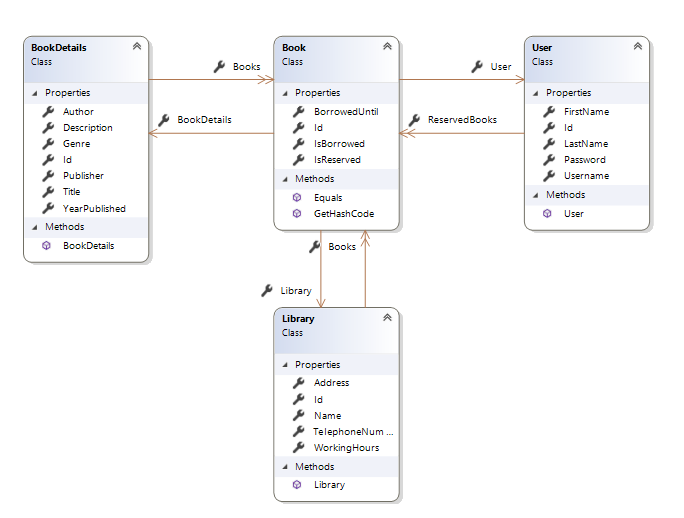
|  |  |
| --- | --- |
| **Naziv** | **UC5 – Rezervacija knjige** |
| **Opis** | Korisnik rezervira odabranu knjigu kako bi je mogao posuditi. |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Korisnik odabire opciju rezervacije knjige klikom na gumb iz prikaza dostupnosti odabrane knjige u knjižnicama |
| **Glavni uspješni scenarij** | 1. Iz prikaza pregleda dostupnosti knjige u knjižnicama, korisnik odabire u kojoj knjižnici želi rezervirati knjigu 2. Zahtijevana rezervacija je izvršena i knjiga je u odabranoj knjižnici rezervirana za korisnika |
| **Ekstenzije** | 1. Knjiga je već rezervirana od nekog korisnika – korisnik obaviješten |
| **Uvjeti koji trebaju biti zadovoljeni** | Knjiga je rezervirana korisniku u odabranoj knjižnici |

|  |  |
| --- | --- |
| **Naziv** | **UC6 – Pregled postojećih rezervacija** |
| **Opis** | Korisnik pregledava informacije o rezerviranim knjigama. |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Korisnik odabire opciju pregleda detalja o knjižnici klikom na odgovarajući element u formi |
| **Glavni uspješni scenarij** | 1. Korisnik odabire gumb za prikaz forme sa rezerviranim knjigama 2. Sustav prikazuje detaljne informacije o postojećim rezervacijama |
| **Ekstenzije** |  |
| **Uvjeti koji trebaju biti zadovoljeni** |  |

|  |  |
| --- | --- |
| **Naziv** | **UC7 – Brisanje postojeće rezervacije** |
| **Opis** | Korisnik briše prethodno napravljenu rezervaciju na odabranu knjigu u knjižnici. |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Odabir rezervirane knjige i klik na gumb „Obriši“ |
| **Glavni uspješni scenarij** | 1. Korisnik knjigu za koju želi obrisati rezervaciju 2. Klikom na gumb, sustav briše odabranu rezervaciju |
| **Ekstenzije** |  |
| **Uvjeti koji trebaju biti zadovoljeni** |  |

|  |  |
| --- | --- |
| **Naziv** | **UC8 – Odjava** |
| **Opis** | Odjava iz sustava |
| **Glavni akter** | Korisnik – član knjižnice |
| **Sporedni akteri** |  |
| **Preduvjet** | Korisnik je prijavljen u sustav |
| **Okidač** | Korisnik odabire opciju odjave iz sustava |
| **Glavni uspješni scenarij** | 1. Korisnik je odjavljen iz sustava |
| **Ekstenzije** |  |
| **Uvjeti koji trebaju biti zadovoljeni** | Korisnik je odjavljen. |

# Opis objektnog modela (entiteti)



Razred *User* predstavlja korisnika aplikacije, odnosno člana knjižnice. Koristi se prilikom prijave u sustav kako bi korisnik potvrdio da je član knjižnice. Razred također sadrži podatke o korisničkim informacijama: ime i prezime te korisničko ime i šifru koja mu je dana od knjižnice za prijavu u sustav. Korisnik ima informacije o svim knjigama koje je rezervirao kako bi vodio evidenciju i eventualno maknuo rezervaciju knjige.

Razred *Book* apstrahira objekt knjige. Svaka knjiga sadrži informacije o knjižnici u kojoj je dostupna za rezervaciju i posuđivanje te evidentira podatke o dostupnosti za rezervaciju i posuđivanje. Također, kroz *BookDetails* objekt višestruki primjeri iste knjiga (knjige sa istim naslovom, autorom, izdavačem i godinom izdanja) dijele jednake detalje jer su knjige kopije. Neke knjige imaju i korisnika od kojih su trenutno rezervirane.

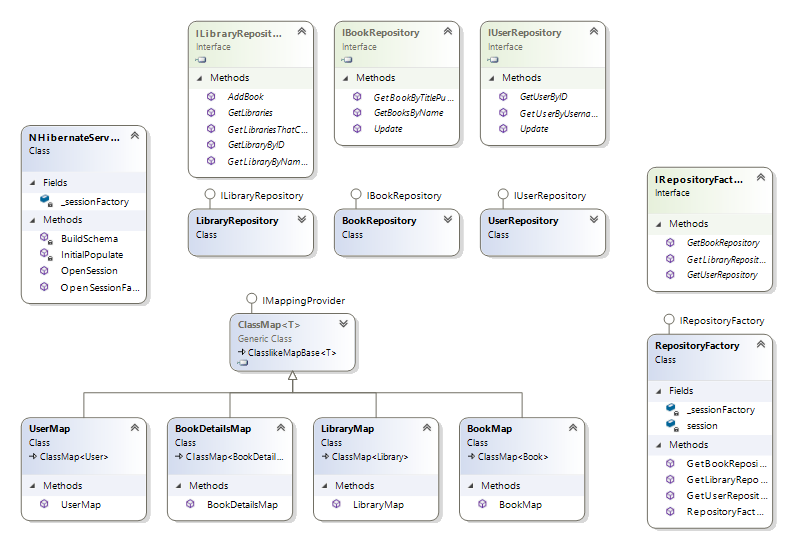
Razred *Library* predstavlja jednu knjižnicu u sustavu. Svaka knjižnica ima ime, adresu, telefonski broj i informacije o radnom vremenu. Također, knjižnica vodi evidenciju o knjigama koje su dostupne za rezervaciju i posuđivanje.

Razred *BookDetails* vodi evidenciju o detaljima neke knjige. To su: naslov djela, autor, izdavač, godina izdanja, žanr i opis.

# Opis modela za perzistenciju

Ove komponente služe za inicijalizaciju baze podataka i komunikaciju sa bazom podataka za vrijeme cijelog životnog ciklusa aplikacije. Sloj pristupa podacima (*Data access layer*) komunicira sa bazom podataka, dok upravljači preko repozitorija dohvaćaju te podatke i služe kao posrednici između modela domene i prikaza promjena stanja domene (*view-a*).

## 4.1. DAL (Data access layer)



Repozitoriji su organizirani kako bi komunicirali preko sučelja. *RepositoryFactory* je tvornica za stvaranje repozitorija. Svaki repozitorij nudi metode preko kojega upravljač kroz jasno definirana sučelja vrši operacije dohvata, spremanja i ažuriranja podataka. Za to, repozitoriji koriste *ISession* objekt kako bi prilikom izmjene nad podacima u bazi mogli stvarati i zaključiti transakcije.

Razred *NHibernateService* preko *OpenSessionFactory* metode stvara *SessionFactory* koji otvara konekciju na bazu i omogućava stvaranje *ISession* objekata koji su dostupni repozitorijima.

## 4.1.1. Mapiranja

U projektu sam koristio O/R mapiranje i *FluentNHibernate* *framework* kako bi objektni model mapirao u relacijski. U nastavku slijede sva mapiranja koje aplikacija sadrži. Sva mapiranja su izvedena iz *ClassMap* razreda.

## UserMap

Entitet *User* je mapiran preko *UserMap* razreda. Svako svojstvo (osim identifikatora koji je mapiran metodom *Id*) je mapirano putem metode *Map*, dok je svojstvo liste rezerviranih knjiga *ReservedBooks* mapirano sa *HasMany* jer jedan korisnik može imati više rezerviranih knjiga.

## BookMap

Entitet *Book* je mapiran preko *BookMap* razreda. Svako svojstvo (osim identifikatora koji je mapiran metodom *Id*) je mapirano putem metode *Map*, *BookDetails* je mapiran preko *References* jer svaka knjiga sadrži informacije o detaljima. *References* se također koristi za mapiranje knjižnice u kojoj se nalazi ta knjiga i korisnika (ako je trenutno rezervirana od nekog korisnika).

## LibraryMap

Entitet *Library* je mapiran preko *LibraryMap* razreda. Svako svojstvo (osim identifikatora koji je mapiran metodom *Id*) je mapirano putem metode *Map*, dok je svojstvo za listu knjiga koje knjižnica posjeduje mapirano koristeći metodu *HasMany* jer jedna knjižnica može imati više knjiga.

## BookDetailsMap

Entitet *BookDetails* je mapiran preko *BookDetailsMap* razreda. Svako svojstvo (osim identifikatora koji je mapiran metodom *Id*) je mapirano putem metode *Map*, dok je svojstvo za listu knjiga koje imaju isti *BookDetails* mapirano koristeći metodu *HasMany* jer postoji više knjiga koje imaju isti *BookDetails* u slučajevima kada postoji više primjerka jedne knjige za jednakim naslovom, autorom, izdavačem i godinom izdavanja.

# Arhitektura i opis aplikacije

Bookshelf je desktop aplikacija bazirana na MVC arhitekturi. *View-*ovi su izgrađeni kao Windows forme, dok se između svake forme i modela nalazi pripadajući kontroler. *View* reagira na korisnikovu interakciju sa sustavom te potom poziva odgovarajuću metodu kontrolera koji *view*-u govori na koji način se mora promijeniti.

## 5.1. Kontroleri

Svaki kontroler implementira sučelje koje definira ugovor kojeg se svaka implementacija kontrolera mora pridržavati. Aplikacija sadrži navedene kontrolere.

## 5.1.1. MainFormController

Upravlja prijavom i odjavom člana knjižnice u sustav. Sadrži funkcije *CheckAuthentication,* *Logout* i *ShowSearch.* *CheckAuthentication* služi za provjeru unesenih korisničkih podataka pri prijavi, *Logout* odjavljuje prijavljenog korisnika, a *ShowSearch* stvara formu za prikaz *Lobby*-a te odgovarajući kontroler koji njome upravlja.

## 5.1.2. LobbyController

Upravlja prikazivanjem rezultata pretrage, pripremom za prijelaz na prikaz detalja o knjizi, pripremom za prijelaz na prikaz ekrana sa rezerviranim knjigama i pripremom za prijelaz na prikaz ekrana gdje je moguće rezervirati knjigu. Metoda *ViewLobby* poziva se nakon uspješne prijave i korisniku prikazuje Lobby. Metoda *SearchBooks* formi vraća listu svih knjiga koje u naslovu sadrže riječ koju je korisnik unio u tražilicu te forma prikazuje te rezultate. *ShowDetails* kreira formu za prikaz detalja o odabranoj knjizi i pripadajući kontroler i prikazuje novu formu korisniku. *ShowMakeReservation* kreira formu za prikaz liste knjižnica u kojima je odabrana knjiga dostupna i pripadajući kontroler. *ShowViewReservations* stvara formu za prikaz svih rezervacija trenutno prijavljenog korisnika i stvara odgovarajući kontroler.

## 5.1.3. MakeReservationController

Ovaj kontroler ima metodu *MakeReservation* koja služi za rezervaciju odabrane knjige u formi.

## 5.1.4. BookDetailsController

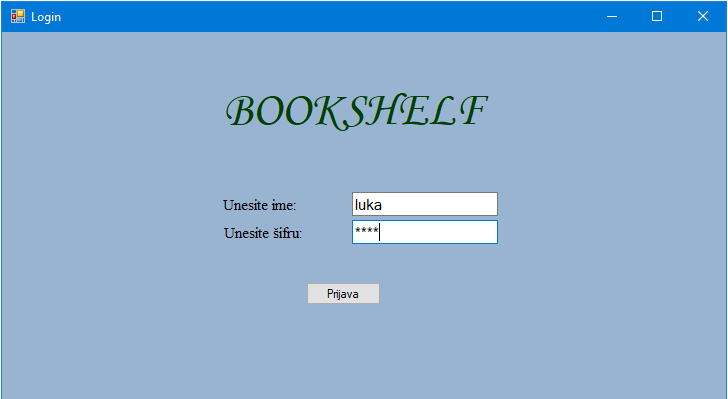
Upravlja prikazivanjem detalja odabrane knjige.

## 5.1.5. ShowReservationsController

Upravlja pregledom korisničkih rezervacija i brisanja odabranih rezervacija kroz metodu *DeleteReservation*.

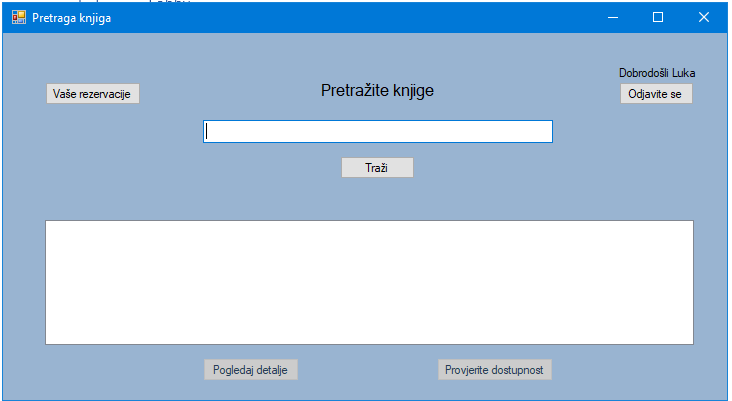
## Prijava

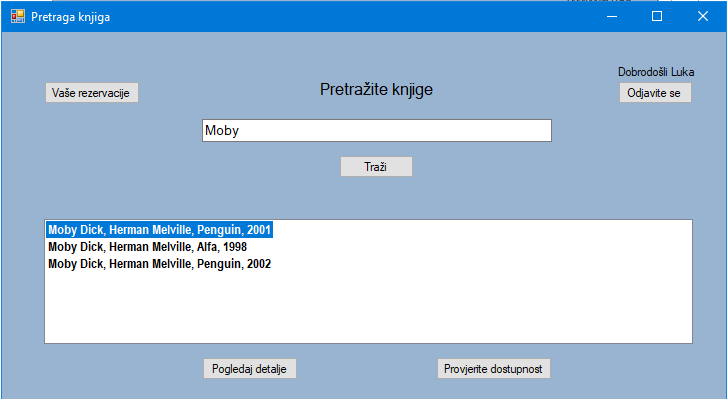
Prilikom pokretanja aplikacije korisniku se prikazuje početni ekran koji se sastoji od forme (frmMainWindow) za unos korisničkog imena i šifre. Kada su oba podatka ispunjena, klikom na gumb „Prijava“, iz forme se *MainFormController*-u prosljeđuje zahtjev za autorizacijom podataka. Ako je autorizacija uspješna, korisnika se prijavljuje u sustav i prikazuje mu se *Lobby* forma. U slučaje neuspjele prijave, korisnik je obavješten o nemogućnosti prijave u sustav (podaci za testnog korisnika; ime: „luka“, šifra: „luka“).



## Lobby

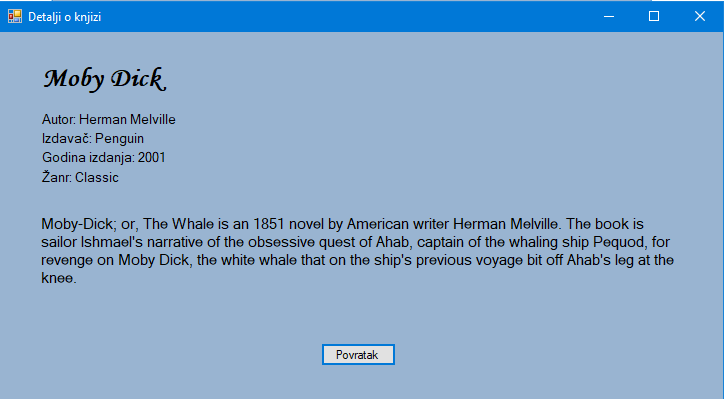
Nakon uspješne prijave, korisniku se prikazuje forma frmLobby. Prikaz se sastoji od forme za unos naslova knjige koju se želi pretražiti, komponente u kojoj se prikazuju rezultati pretrage te gumbovi za navigaciju po aplikaciji. Korisnik sada može pretraživati bazu podataka knjižnice kako bi pronašao postojeće knjige i provjerio njihovu dostupnost u knjižnicama. Klikom na gumb „Odjavite se“ korisnika se odjavljuje iz sustava. Klikom na gumb „Vaše rezervacije“ otvara se zaslon sa korisničkim rezervacijama. Gumbovi „Pogledaj detalje“ i „Provjerite dostupnost“ su onemogućeni sve dok nije odabran samo jedan element iz komponente u kojoj se prikazuju rezultati pretrage. Unosom teksta „Moby“ u tražilicu i klikom na gumb „Traži“, korisniku se u komponenti prikazuju relevantne knjige. Istovjetni naslovi se razlikuju prema izdavaču i godini izdanja.





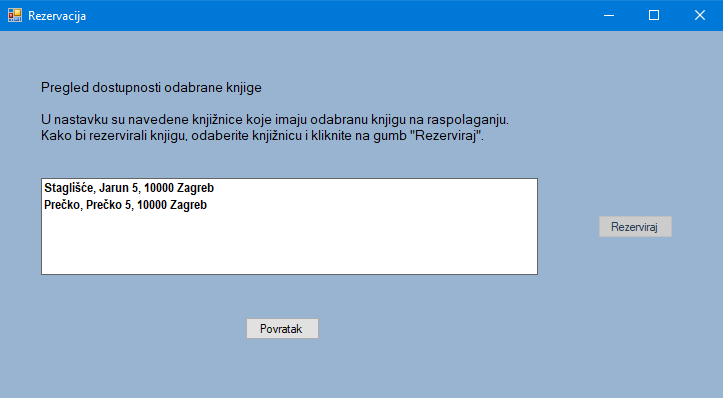
## Pregled detalja odabrane knjige

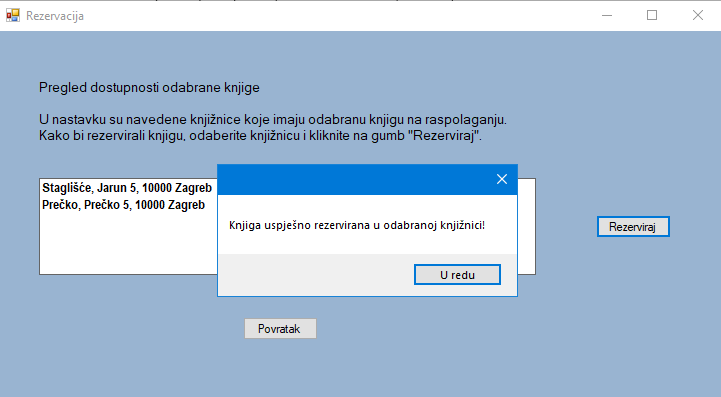
Kada se u frmLobby formi odabere rezultat i klikne na gumb „Pogledaj detalje“ prikazuje se forma (frmBookDetails) sa prikazom detalja za odabranu knjigu. Klikom na gumb „Povratak“ korisnika se vraća u Lobby.



## Provjera dostupnosti odabrane knjige

Klikom na „Provjerite dostupnost“ iz Lobby-a prikazuje se forma (frmMakeReservation) za pregled dostupnosti odabrane knjige u svim knjižnicama. Korisnik odabire knjižnicu u kojoj želi rezervirati odabranu knjigu i klikom na gumb „Rezerviraj“ provodi rezervaciju. Ako je knjiga već rezervirana, korisnik je obavješten o nemogućnosti provođenja rezervacije. Prilikom uspješne rezervacije, korisnika se vraća u Lobby. Klikom na gumb „Povratak“, korisnik se vraća u Lobby.





## Pregled postojećih korisničkih rezervacija

Ako se iz Lobby-a klikne na gumb „Pregled rezervacija“ otvara se forma (frmShowReservations) koja korisniku prikazuje njegove postojeće rezervacije svih knjiga u odabranim knjižnicama. Ako korisnik odabere neku rezervaciju, omogućen mu je gumb „Obriši“ koji briše odabranu korisničku rezervaciju. Brisanjem rezervacije, korisnika se obavještava o uspješnosti i vraća u Lobby.

