

Projekt zaliczeniowy



Programowanie obiektowe Rok akademicki 2024/2025

Autorzy:

Zofia Ekielska
Małgorzata Gdowska
Wiktoria Wróbel

Biblioteka

Projekt dotyczy systemu zarządzania biblioteką, który umożliwia rejestrację użytkowników, wypożyczanie książek oraz zarządzanie kontami bibliotecznymi. Obsługuje konta standardowe i premium, oferując m.in. przedłużanie wypożyczeń, regulowanie należności oraz rekomendacje książek. System zarządza książkami, użytkownikami, filiami, oferując funkcje wyszukiwania, filtrowania, klonowania obiektów i serializacji danych.

Podział ról

Zofia Ekielska:

Klasy: Ksiazka, KontoBiblioteczne, KontoBibliotecznePremium, Filia

Interferjsy: IWypożyczenie, IFilia

Małgorzata Gdowska:

Klasa Propozycja, Klasa KontoBibliotecznePodstawowe, odczyt danych do/z pliku, wykorzystanie interfejsów ICloneable, IComparable, IEquatable, diagram CRC, testy jednostkowe, wykorzystanie delegatów

Wiktoria Wróbel:

Zaprojektowanie i implementacja graficznego interfejsu użytkownika

Klasa Książka

Klasa jest odpowiedzialna za tworzenie książek. Zadeklarowałam w niej dwa pola enum, tak by tematyka książki i wydawnictwo były wybierane z określonych pozycji. Każda książka pobiera też parametry tytuł oraz autor. Dodatkowo każda książka ma swój unikalny kod: "(int)typKsiazki}/{(int)wydawnictwo}/{NumerKsiazki}" - do każdej tematyki oraz wydawnictwa przypisany jest konkretny numer, natomiast trzecim elementem jest numer, który dla każdej kolejnej książki będzie się zwiększał o jeden, tak by mieć pewność, że kody nie będą się powtarzać. Dodalam także właściwości polom tytuł i autor, by móc sprawdzić, czy dane zostały napisane z dużej litery - wtedy program deklaruje specjalny wyjątek. Każda książka posiada także pole bool, mówiące o tym czy jest aktualnie wypożyczona, co będzie ważne w kolejnych klasach. Wartość pola przy tworzeniu jest ustawiona na false - książka nie jest wypożyczona.

Klasa Konto Biblioteczne

Klasa stanowi bazę dla tworzenia dalszych klas, które ściślej będą określały parametry typów kont. W związku z tym klasa jest abstrakcyjna i pobiera od użytkownika niezbędne dane. Dla każdego użytkownika tworzy także unikalny numer konta. Sprawdzam także poprawność niektórych danych:

- czy imię oraz nazwisko zostały wprowadzone z dużej litery
- czy pesel składa się z 11 znaków
- czy nazwa użytkownika jest wystarczająco długa
- czy użytkownik jest pełnoletni

W związku z tym stworzyłam także odpowiednie wyjątki.

Klasa Konto Biblioteczne Premium

Klasa dziedziczy po klasie konto biblioteczne. Została stworzona by wyodrębnić inny typ konta, który pozwoli na większą ilość wypożyczeń oraz obniży stawkę kary za opóźnienia do 0,1 zł za każdy dzień przekroczenia terminu.

Klasa Filia

Stanowi zbiór książek, jakie ma do zaoferowania filia oraz listę swoich klientów, implementuje interfejs IFilia, która umożliwia wiele niezbędnych operacji.

Interfejs IFilia

a) *Funkcje dotyczące książek:*

- tworzenie nowych książek
- usuwanie konkretnej książki z oferty po jej niepowtarzalnym numerze
- wyszukiwanie wszystkich książek konkretnego autora
- wyszukiwanie wszystkich książek o konkretnej tematyce

- liczenie ile łącznie książek oferuje filia
- liczenie ilości książek o wskazanej tematyce
- liczenie ilości książek konkretnego autora

b) Funkcje dotyczące Kont:

- dodawanie konta do systemu filii
- usuwanie konta po unikalnym numerze użytkownika

Klasa Wypożyczenie

Konstruktor pobiera Konto oraz Książkę. Zgłasza błąd, gdy konto wyczerpało swoją maksymalną ilość wypożyczeń lub książka jest niedostępna. Każde wypożyczenie ma swój unikalny numer.

Przy poprawnym wypożyczeniu książki automatycznie liczba wypożyczonych książek danego użytkownika wzrasta o jeden, a status książki jest ustawiany na wypożyczony.

a) Pierwszy konstruktor:

Data zwrotu zostanie automatycznie przypisana na miesiąc od daty wypożyczenia.

b) Drugi konstruktor:

Możliwe jest także wprowadzenie dawnych wypożyczeń (data wypożyczenia nie jest dniem dzisiejszym) . Wtedy również data zwrotu ustawiana jest na miesiąc po dacie wypożyczenia.

Pole przechowujące informacje o tym , czy książka jest po terminie zostaje odpowiednio ustawione (wartość true w przypadku przetrzymania książki)

Interfejs IWypożyczenie

Pozwala na:

- obliczenie ile dni minęło od daty zwrotu książki (przydatne do obliczania wysokości kary)
- przedłużenie terminu wypożyczenia o miesiąc
- uregulowanie należności

-zarejestrowanie zwrotu książki, pod warunkiem, że należności wynoszą zero (kar nie ma lub zostały uregulowane)

-obliczenie ile wynosi kara za przetrzymanie książki

Klasa Propozycja

Klasa **Propozycja** odpowiada za generowanie propozycji książek dla użytkownika, bazując na ulubionym gatunku oraz informacjach o tym, w których filiach (obiektach klasy **Filia**) użytkownik ma swoje konto. Metoda **Proponuj()**:

- Sprawdza, czy konto biblioteczne użytkownika (konto) jest przypisane do jakiegokolwiek filii z listy **listaFili**.
- Dla każdej filii, w której użytkownik ma konto, wyszukuje książki danego gatunku (**ulubionyGatunek**).
- Losuje do 3 pozycji i wyświetla je jako „polecane książki”.

Modyfikatory dostępu:

Klasa posiada modyfikator dostępu **internal** - jest dostępna tylko w obrębie bieżącego assembly, ponieważ logika propozycji ma być wykorzystywana tylko w obrębie projektu bibliotecznego.

Pola **ulubionyGatunek**, **listaFili** oraz **konto** są prywatne. Tylko metody klasy mogą je zmieniać i odczytywać. Zapobiega to ingerencji z zewnątrz w stan klasy.

Klasa KontoBibliotecznePodstawowe

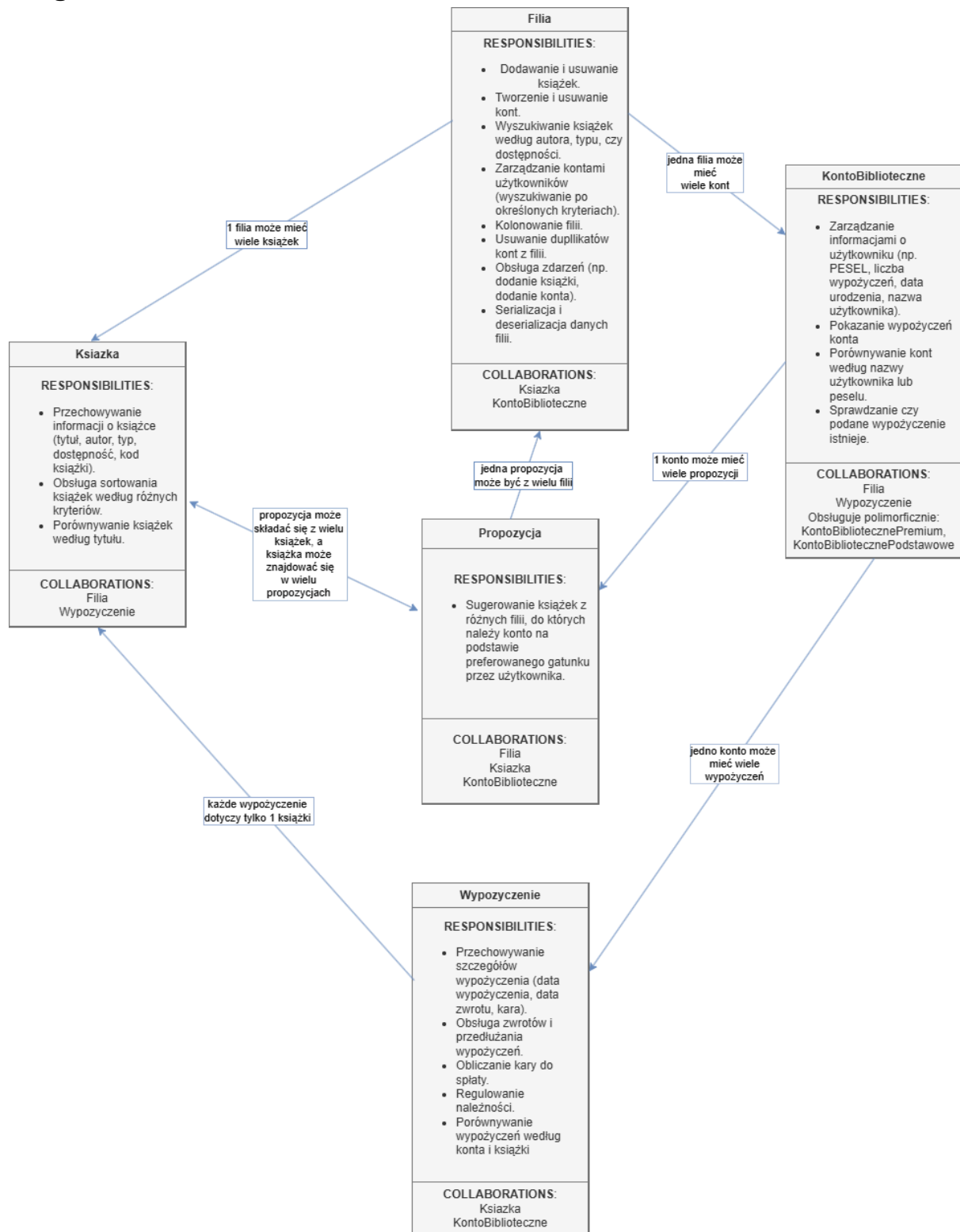
Klasa **KontoBibliotecznePodstawowe** rozszerza funkcjonalność bazowego konta bibliotecznego (**KontoBiblioteczne**). Dzięki temu może definiować specyficzne dla konta podstawowego właściwości: maksymalną liczbę wypożyczeń lub stawkę kary za przetrzymane książki.

Modyfikatory dostępu:

Klasa jest **public**, dzięki czemu można użyć ją w testach jednostkowych.

public override decimal StawkaKary - właściwość jest publiczna, ponieważ funkcje zarządzające karami mogą potrzebować odczytać aktualną stawkę kary.

Diagram klas



INNE FUNKCJONALNOŚCI W PROJEKCIE

Sortowanie książek: Klasa **Ksiazka** implementuje interfejs **Comparable** i pozwala na sortowanie według tytułu, autora lub obu. Metody sortujące według wybranego kryterium zostały zaimplementowane w Klasie **Filia**:

```
public void SortujKsiazkiPoTytule()
{
    Ksiazka.RodzajSortowania = Sortowanie.Tytul;
    ListaKsiazek.Sort();
}
Odwołania: 0
public void SortujKsiazkiPoAutorze()
{
    Ksiazka.RodzajSortowania = Sortowanie.Autor;
    ListaKsiazek.Sort();
}
1 odwołanie
public void SrotujKsiazkiPoAutorzeITytule()
{
    Ksiazka.RodzajSortowania = Sortowanie.AutorTytul;
    ListaKsiazek.Sort();
}
```

Filtrowanie kont: W Klasie **Filia** za pomocą delegata **KryteriumFiltracjiKont** oraz przy użyciu metody **FiltrujKonta** można wyszukiwać:

- konta po peselu,
- konta według nazwy użytkownika,
- wszystkie konta z maksymalnym limitem wypożyczeń,
- wszystkie konta bez wypożyczeń.
- wszystkie konta premium,
- wszystkie konta podstawowe

Wyszukiwanie książek: W klasie **Filia** za pomocą delegata **KryteriumWyszukiwaniaKsiazek** oraz przy użyciu metody **WyszukajKsiazki** można wyszukiwać:

- książki, które zawierają dane słowo w tytule,
- książki na podstawie wydawnictwa,
- wszystkie wypożyczone książki,
- wszystkie książki, które są dostępne,
- książki, których numery znajdują się w podanym zakresie

przykład:

```

List<Ksiazka> ksiazkiNumery= f2.WyszukajKsiazkiZZakresuNumerow(100, 105);
foreach (Ksiazka ksiazka in ksiazkiNumery)
{
    Console.WriteLine($"numer książki: {ksiazka.NumerKsiazki}");
    Console.WriteLine(ksiazka);
}

```

(Filia f2 zawiera książki z numerami od 103 do 106, a my chcemy wyszukać te z numerami od 100 do 105)

wynik użycia funkcji:

```

numer książki: 103
Tytuł: "Zielona Mila" | Autor: Artur Sikorski (1/50/103)
ScienceFiction, Muza
dostępna
numer książki: 104
Tytuł: "Lalka" | Autor: Bolesław Prus (7/20/104)
Romans, Czarne
dostępna
numer książki: 105
Tytuł: "Zdążyć przed Panem Bogiem" | Autor: Hanna Krall (6/1/105)
Reportaż, Znak
dostępna

```

(książki nie mają numerów po kolei, ponieważ domyślnie są filtrowane według autora, a następnie tytułu)

Klonowanie obiektów: Klasy **Filia**, **Ksiazka**, i **KontoBiblioteczne** implementują interfejs **ICloneable**, co pozwala tworzyć kopie obiektów.

Serializacja danych: Klasa **Filia** obsługuje zapis i odczyt danych za pomocą **DataContract** do XML.

Porównywanie: Klasy **Wypozyczenie**, **KontoBiblioteczne** i **Ksiazka** implementują interfejs **IEquatable**, umożliwiając porównywanie obiektów. Wykorzystywane np. w przypadku funkcji **UsunDuplikatyKont()**, która jest odpowiedzialna za usunięcie konta jeżeli pesel lub nazwa użytkownika się powtórzy.

Obsługa zdarzeń (za pomocą delegatów): Dodawanie kont wywołuje zdarzenie **DodanoKonto**, które informuje o tym, że książka została dodana. Analogiczna sytuacja w przypadku zdarzenia **DodanoKsiazke**.

Opis funkcjonalności

Głównym celem interfejsu jest zapewnienie łatwej interakcji użytkownika z wybraną filią, pozwalając na pełne wykorzystanie programu od strony konsumenta. Aplikacja umożliwia:

- wybór filii
- przegląd dostępnych zasobów filii bez konieczności logowania się
- rejestrację nowego użytkownika
- zalogowanie się na istniejące konto
- wypożyczenie książki
- przedłużenie wypożyczenia
- uregulowanie należności za książkę, która nie została oddana w terminie
- zwrot książki

Poszczególne okna aplikacji

Okno główne

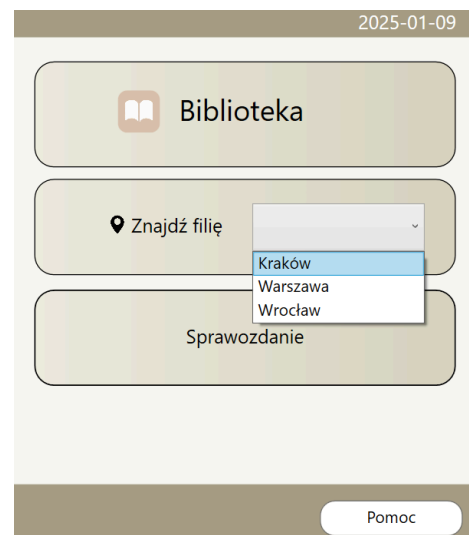
Okno, które pokazuje się jako pierwsze. Umożliwia:

- zobaczenie aktualnej daty
- wybór filii
- otwarcie sprawozdania
- uzyskanie pomocy (FAQ)

ComboBox Wybór Filii

Wybranie jednej z trzech opcji wywołuje działanie funkcji `WyborFilii()`.

Za pomocą instrukcji *switch* następuje wywołanie odpowiednich okien (np. dla wyboru *Kraków* zostaje otworzone okno *Kraków Filia*).



Przycisk Sprawozdanie

Przenosi użytkownika na stronę internetową, na której umieszczone jest sprawozdanie (wykorzystanie klasy `System.Diagnostics`).

Przycisk Pomoc

Po jego kliknięciu następuje otwarcie okna z FAQ. Pytania i odpowiedzi mogą rozjaśnić użytkownikowi najważniejsze założenia programu.

Pomoc

Okno z najpotrzebniejszymi pytaniami i odpowiedziami odnośnie programu. Opisuje jak wygląda proces rejestracji, czy podstawowe operacje, jakie może wykonać zalogowany użytkownik.

Przycisk Zamknij

Przycisk zamykający okno programu.

Proces logowania użytkownika

Po wybraniu filii, użytkownik jest przekierowywany na stronę logowania. W tym momencie zostaje wczytany plik xml z danej filii.

Okno umożliwia:

- zobaczenie aktualnej daty
- zalogowanie się na istniejące konto
- rejestracja nowego użytkownika
- sprawdzenie aktualnych zasobów filii
- powrót do ekranu głównego i wyboru filii

Przycisk Aktualne zasoby filii

Po kliknięciu użytkownik zostaje przeniesiony na stronę przeznaczoną do przeglądania aktualnych zasobów filii.

Należy zaznaczyć, że aby przeglądać dostępne pozycje, użytkownik nie musi być zalogowany. W oknie została wykorzystana funkcja *WyszukajKsiazkiDostepne()*, która automatycznie wyświetla wszystkie dostępne pozycje.

Istnieje możliwość filtrowania książek według gatunków, autorów lub gatunków i autorów. Wybór gatunku i autora jest możliwy tylko z pozycji dostępnych na dany moment (oznacza to, że jeżeli w bibliotece nie ma w danym momencie książki z gatunku *Romans*, to ten gatunek nie pojawi się w ComboBoxie).

Filtruj po gatunku: Thriller

Filtruj po autorze: Agatha Christie

I nie było już nikogo
Agatha Christie

Noc i ciemność
Agatha Christie

Czym jest Biblioteka?
Biblioteka to system służący do zarządzania kontem bibliotecznym użytkownika. Aplikacja obsługuje 3 filie - Kraków, Warszawę i Wrocław. W każdej z nich użytkownik może dokonać rejestracji, aby w pełni korzystać z usług biblioteki.

Jak wygląda proces rejestracji?
W celu utworzenia konta, użytkownik musi podać swoje imię, nazwisko, pesel i datę urodzenia, a także preferowaną nazwę użytkownika. Istnieje możliwość wyboru konta w trakcie rejestracji - konto premium wiąże się z większymi udogodnieniami w porównaniu do konta podstawowego.

Dostępne funkcje
Zalogowany użytkownik może wypożyczać książkę, przedłużyć jej termin, czy zwrócić książkę. W przypadku przekroczenia terminu zwrotu, naliczane są opłaty. Użytkownik musi uregulować opłatę przed oddaniem książki.

Przeglądanie zasobów
Użytkownik nie musi być zalogowany, aby przeglądać aktualne zasoby w danej filii. Użytkownik może przefiltrować książki według gatunków lub autorów, którzy obecnie znajdują się w filii.

Zamknij

Filia Kraków 2025-01-09

Nazwa użytkownika

Zaloguj się

Nie masz konta? Zarejestruj się

Aktualne zasoby filii Powrót

Filtruj po gatunku: [dropdown]

Filtruj po autorze: [dropdown]

Cywilizacja Słowian
Kamil Janicki

Miedzy wierszami
Izabella Nowaczyk

Pokochaj albo zabij
Weronika Ancerowicz

Inna strona świata
Wojciech Jagielski

Sekta
Marcin Walczak

I nie było już nikogo
Agatha Christie

Zagubiony ślad
Aleksandra Potter

Noc i ciemność
Agatha Christie

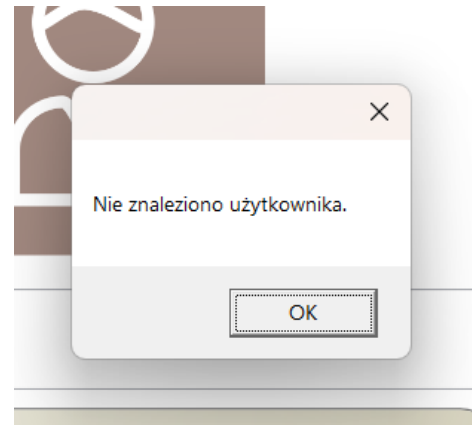
Zamknij

ComboBox Gatunki i ComboBox Autorzy działają na podobnej zasadzie. Na samym początku zostaną pokazane wszystkie możliwe opcje wyboru. Jeżeli użytkownik wybierze konkretny gatunek / autora, to odpowiedź zostaje przekazana jako parametr do funkcji *PokazDostepneKsiazki(string gatunek = " ", string autor = " ")*, która wywołuje wcześniej wspomnianą funkcję *WyszukajKsiazkiDostepne()* wraz z odpowiednim filtrowaniem.

TextBox Nazwa użytkownika

Po kliknięciu w pole użytkownik może wpisać swoją nazwę użytkownika. W celu zatwierdzenia należy nacisnąć przycisk *Zaloguj się* lub nacisnąć enter. Jeżeli w zasobach biblioteki istnieje konto o podanej nazwie użytkownika, to logowanie przebiegnie pomyślnie. W przeciwnym wypadku wyskoczy komunikat o braku konta. Do wyszukiwania konta po loginie została wykorzystana funkcja

WyszukajKontoWedlugNazwyUzytkownika(string login).



Przycisk Zarejestruj się

Przenosi użytkownika do nowego okna, w którym następuje proces rejestracji. Aby utworzyć nowe konto wymagane jest podanie:

- imienia
- nazwiska
- peselu
- nazwy użytkownika
- daty urodzenia
- opcjonalnie - wybór konta premium

A screenshot of a registration form. At the top is a header bar. Below it is a large square icon containing a stylized person silhouette. To the right of the icon is a checkbox labeled "Konto premium". Below these are five input fields: "Imię", "Nazwisko", "Pesel", "Nazwa użytkownika", and "Data Urodzenia". The "Data Urodzenia" field has a date picker icon and shows the date "15". At the bottom of the form is a wide button labeled "Utwórz konto".

* Imię i nazwisko muszą być pisane wielką literą.

W przeciwnym wypadku zostanie wyrzucony wyjątek *DuzaLiteraException()*.

* Pesel musi składać się z 11 cyfr. W przeciwnym wypadku zostanie wyrzucony wyjątek *WrongPeselException()*.

* Nazwa użytkownika nie może być krótsza niż 3 znaki. W przeciwnym wypadku zostanie wyrzucony wyjątek *ZaKrotkiFormatException()*.

* Wiek osoby nie może być mniejszy niż 18. W przeciwnym wypadku zostanie wyrzucony wyjątek *ZaMlodyException()*;



Po wypełnieniu wszystkich pól i kliknięciu w przycisk *Utwórz konto*, nowe konto w filii zostaje utworzone. W zależności od wartości zmiennej *rodzajKonta* zostaje utworzony nowy obiekt klasy *KontoBibliotecznePremium* lub *KontoBibliotecznePodstawowe*. Aby automatycznie dodać użytkownika do filii, wykorzystywana jest funkcja *UtworzKonto(KontoBiblioteczne konto)*.

Po pomyślnej rejestracji użytkownik automatycznie zostaje przekierowany (jako już zalogowany użytkownik) na okno *KontoUzytkownika*.

Konto Użytkownika

Okno zalogowanego użytkownika dostarcza użytkownikowi następujących informacji:

- login
- imię i nazwisko
- aktualne wypożyczenia
- rodzaj konta

Użytkownik może wykonać następujące czynności:

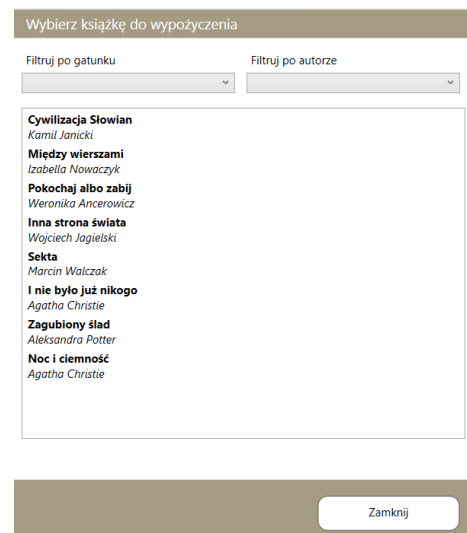
- wypożyczyć książkę
- przedłużyć termin wypożyczenia
- uregulować należność za książkę po terminie
- zwrócić książkę
- wylogować się
-

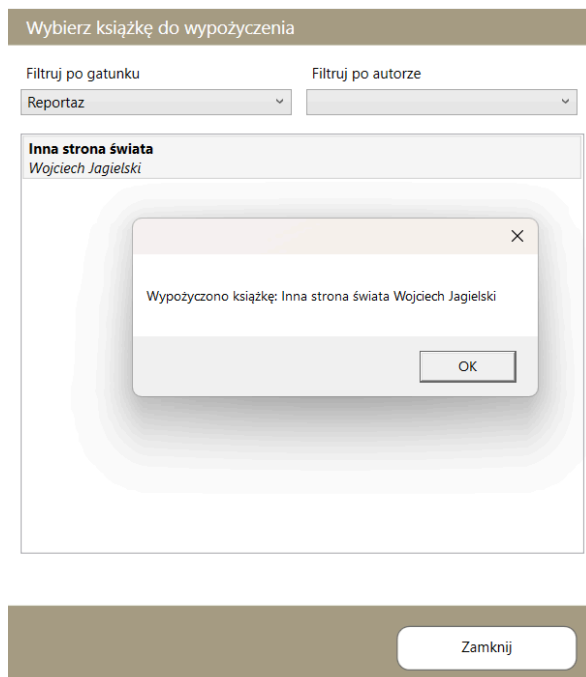
Wypożyczenie książki

W celu wypożyczenia książki użytkownik musi wybrać jedną z dostępnych pozycji. Sposób wyświetlania książek i ich filtrowania jest dokładnie taki sam, jak w przypadku okna *Aktualne zasoby filii*, które zostało omówione wyżej.

Jeżeli limit wypożyczeń zostanie osiągnięty, to zostanie wywołany komunikat (*MessageBox*) o takiej sytuacji.

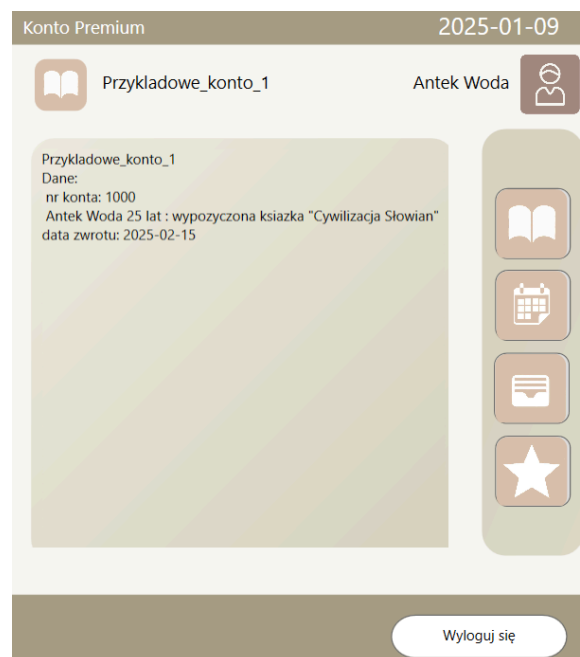
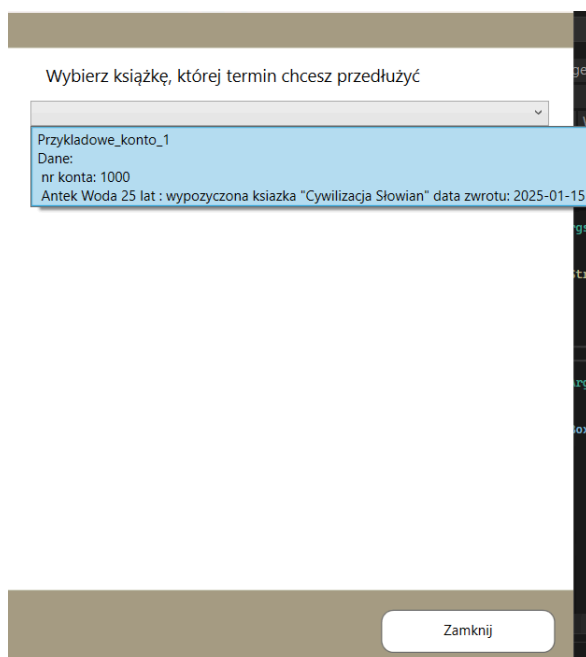
Kliknięcie na książkę oznacza jej wypożyczenie oraz automatyczny powrót do okna *Konto Użytkownika*. Nowe wypożyczenie zostaje wyświetlone na koncie użytkownika. Wypożyczenie książki następuje przez utworzenie nowego obiektu klasy *Wypozyczenie(KontoBiblioteczne konto, Ksiazka ksiazka)*, gdzie kontem jest aktualne konto biblioteczne, a książką wybrana pozycja z *ListBoxa*.





Przedłużenie wypożyczenia

Użytkownik może przedłużyć termin oddania książki. W tym celu należy kliknąć na ikonkę kalendarza i wybrać jedną z pozycji. Użytkownik może wybrać jedynie to wypożyczenie z listy, dla którego wynik funkcji *WysokoscKary()* wynosi 0.



W powyższym przypadku termin zwrotu został wydłużony o miesiąc. Jest to domyślna ilość dni.

Uregulowanie należności

Jeżeli książka nie została zwrócona w terminie, to zaczynają się naliczać należności. Ich uregulowanie jest niezbędne do oddania książki. Podobnie jak w poprzednim przypadku, ComboBox wyświetla tylko te pozycje, przy których należności są większe od 0 (wykorzystanie funkcji *WysokoscKary()* > 0).

Po uregulowaniu należności użytkownik automatycznie zostaje przekierowany do konta użytkownika. Użytkownik zostaje także poinformowany, że ma 1 dzień na oddanie książki. W przeciwnym razie kara zacznie się znowu naliczać.

Wybierz książkę, aby uregulować należność

zosiae
Dane:
nr konta: 1000
Zosia Ekielska 20 lat : wypożyczona książka "Ostrze"
Książka po terminie 30 dni do zapłaty: 9,00 zł

Zwrot książki

Pozycję można zwrócić tylko wtedy, gdy wszystkie należności (jeżeli takie występują) są uregulowane. Działanie okna zwrotu książki jest bardzo podobne do działania dwóch wyżej wymienionych okien.

Wyloguj się

Po kliknięciu na przycisk *Wyloguj się* aplikacja zostaje zamknięta, a jakiekolwiek zmiany zostają zapisane w pliku xml o tej samej nazwie i w tym samym miejscu, z którego to pliku dane były pobierane. Wprowadza to do aplikacji większą mobilność.