Spis treści

[1. Diagram przypadków użycia 2](#_Toc465886491)

[2. Scenariusze 3](#_Toc465886492)

[2.1. Okno 1 3](#_Toc465886493)

[2.2. Konto - opcje 3](#_Toc465886494)

[2.3. Zbiór biblioteki 4](#_Toc465886495)

[3. Instalacja środowiska 5](#_Toc465886496)

[3.1. Serwer bazy - lokalny 5](#_Toc465886497)

[3.2. Tworzenie bazy danych 5](#_Toc465886498)

[3.3. Konfiguracja środowiska Visual Studio 2015 5](#_Toc465886499)

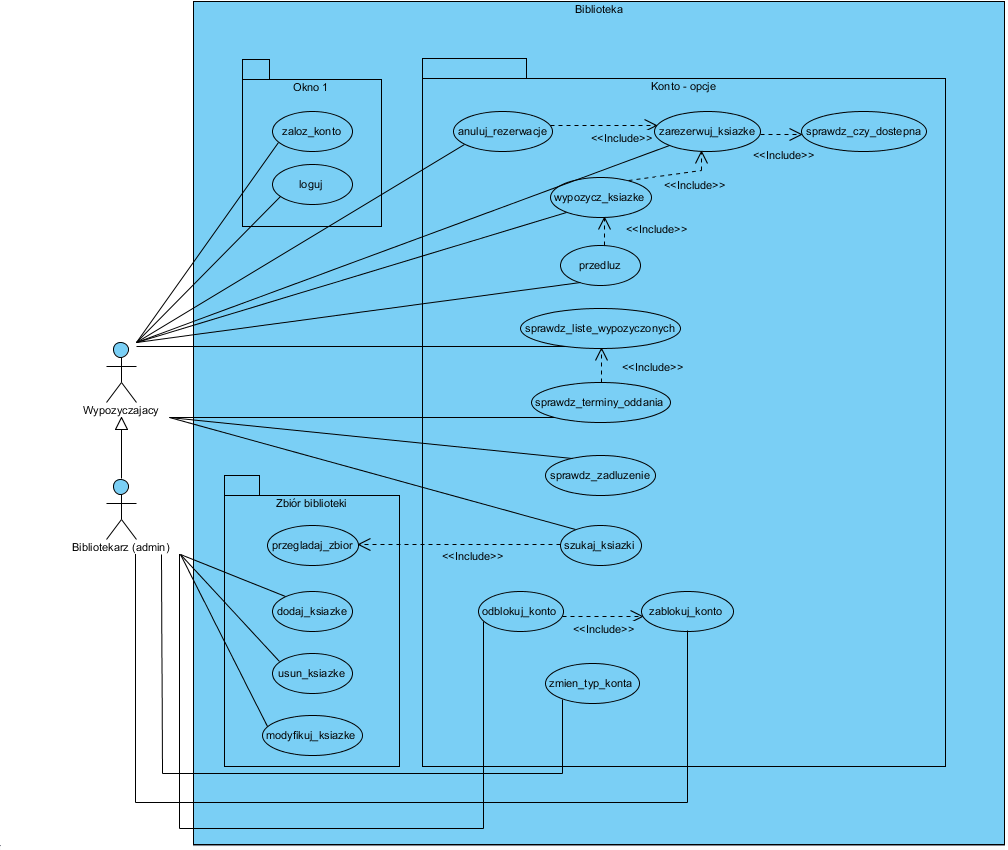
[4. Projekt interfejsu 6](#_Toc465886500)

[4.1. Okno logowania/rejestracji 6](#_Toc465886501)

[4.2 Okno właściwe programu 7](#_Toc465886502)

# Diagram przypadków użycia

**(DO ZMIANY)**



# Scenariusze

## Okno 1

**zaloz\_konto**

Scenariusz:

1. Należy podać imię oraz nazwisko pesel, login do konta, hasło oraz email.

2. Należy sprawdzić czy pesel oraz login jest unikalny.

3. Jeśli dane z punktu 2 są unikalne to załóż konto, w przeciwnym wypadku wyświetl błąd.

**loguj**

Scenariusz:

1. Należy wpisać login oraz hasło, podane podczas rejestracji.

2. W przypadku błędnego loginu lub hasła użytkownik zostaje o tym poinformowany poprzez odpowiedni komunikat.

## Konto – opcje

**(tam gdzie pusto trzeba dopisać)**

**anuluj\_rezerwacje**

**zarezerwuj\_ksiazke**

**sprawdz\_czy\_dostepna**

**wypożycz\_ksiazke**

**przedluz**

**zarezerwuj\_ksiazke**

**sprawdz\_liste\_wypozyczonych**

Scenariusz:

1. Po kliknięciu system podaje użytkownikowi listę wypożyczonych przez niego książek poprzez sprawdzenie tabeli „Rezerwacje” (dla zalogowanego użytkownika ).

**sprawdz\_terminy\_oddania**

Scenariusz:

1. Po wybraniu tej opcji system sprawdza terminy oddania szukając w odpowiedniego rekordu „termin\_oddania” (dla danej książki) w tabeli „Rezerwacje”.

**sprawdz\_zadluzenie**

Scenariusz:

1. Po kliknięciu w przycisk "Sprawdź zadłużenie" system wyświetla aktualny stan zadłużenia zalogowanego użytkownika poprzez sprawdzenie wartości „zadluzenie” (dla danej książki) w tabeli „Rezerwacje”.

**szukaj\_ksiazki**

**odblokuj\_konto**

**zablokuj\_konto**

**zmien\_typ\_konta**

## Zbiór biblioteki

**przeglądaj\_zbior**

**dodaj\_ksiazke**

**usun\_ksiazke**

**modyfikuj\_ksiazke**

# Instalacja środowiska

## Serwer bazy - lokalny

Umożliwia nam to: **MySQL Community Server** (<https://dev.mysql.com/downloads/mysql/>)

## Tworzenie bazy danych

Tworzenie bazy danych odbywa się w programie: **MySQL Workbench 6.3 CE** (<https://dev.mysql.com/downloads/workbench/>)

## Konfiguracja środowiska Visual Studio 2015

Aby mieć możliwość połączenia z bazą danych z poziomu aplikacji napisanej w C++ w środowisku VS 2015 musimy odpowiednio je zmodyfikować:

1. Należy zainstalować „sterownik” **Conncector/Net**(<http://dev.mysql.com/downloads/file/?id=463758>)
2. Następnie w VS dodać odwołanie: **MySql.Data**. (References -> PPM -> Add Reference..)

Podczas połączenia zarówno w bazie danych jak i w pliku konfiguracyjnym połączenia należy ustawić odpowiednie parametry. W naszym przypadku:

**datasource**=127.0.0.2; **port**=3306; **username**=root; **password**=root123; **database**=biblioteka

Za łączenie z bazą odpowiada poniższa linijka kodu:

MySqlConnection^ laczBaze = gcnew MySqlConnection(config);

Wysyłamy zapytanie i pobieramy wartości z dóch kolumn z tabeli

MySqlCommand^ zapytanie = gcnew MySqlCommand("SELECT \* FROM użytkownicy WHERE uzytkownik\_id = 1, laczBaze);

MySqlDataReader^ dane;

try {

laczBaze->Open(); //otwieramy połączenie z bazą

dane = zapytanie->ExecuteReader();

while (dane->Read()) {

jakisTextBox->Text = dane->GetString("Imie");

jakisTextBox->Text = dane->GetString("Nazwisko");

}

laczBaze->Close();//zamykamy połączenie z bazą

}

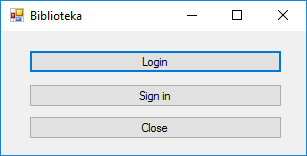
catch (Exception^ ex) {

MessageBox::Show(ex->Message);

}

# Projekt interfejsu

## Okno logowania/rejestracji



|  |  |
| --- | --- |
| Login: | Sign in: |

## Okno właściwe programu

Przykład: zakładka admina (widoczna tylko dla niego)

