



AKADEMIA GÓRNICZO-HUTNICZA

Dokumentacja do projektu

**Terminalowa aplikacja do zarządzania
lokalną bazą danych na platformie
MySQL Workbench**

z przedmiotu

Języki Programowania Obiektowego

Elektronika i telekomunikacja, III rok

Kamil Drożdż

Grupa 4, piątek 11:30

prowadzący: mgr inż. Jakub Zimnol

09.01.2025

1. Wstęp

Celem projektu było zaprojektowanie i stworzenie aplikacji terminalowej przeznaczonej do zarządzania lokalną bazą danych księgarni. Aplikacja została stworzona w celu ułatwienia zarządzania kluczowymi aspektami funkcjonowania księgarni. Zastosowana baza danych została uruchomiona na platformie MySQL Workbench, co pozwoliło na zapewnienie niezawodności oraz szybkości przetwarzania danych. Przyjęto podejście pozwalające na intuicyjną obsługę aplikacji poprzez interfejs terminalowy, co umożliwia łatwe wykonywanie operacji takich jak dodawanie, edytowanie. Projekt miał na celu stworzenie rozwiązania przyjaznego dla użytkownika, które znacznie przyspieszy codzienne zarządzanie księgarnią i jednocześnie zminimalizuje ryzyko błędów wynikających z manualnego przetwarzania danych.

2. Instalacja i konfiguracja

1. Sklonuj repozytorium: git clone https://github.com/kamildrozdzz/JPO_Project.git
2. Przejdź do katalogu projektu: cd JPO_Project
3. Skonfiguruj bazę danych MySQL, importując plik libraryonline.sql.
4. Skonfiguruj pliki nagłówkowe i biblioteki MySQL w środowisku programistycznym.
5. Skompiluj projekt za pomocą Makefile: make
6. Uruchom aplikację: ./JPO_Project

3. Opis technologiczny

I. Baza danych

Wszystkie operacje są wykonywane na lokalnej bazie danych o następującej strukturze:

Name	Engine	Version	Row Format	Rows	Avg Row Length	Data Length	Max Data Length	Index Length	Data Free	Auto Incre...	Create Time	Update Time	Check Time	Collation	Checksum
ksiazki	InnoDB	10	Dynamic	30	546	16.0 KB	0.0 bytes	0.0 bytes	0.0 bytes	36	2024-12-19 14:16:52			utf8mb4_genera...	
uzytkownicy	InnoDB	10	Dynamic	19	862	16.0 KB	0.0 bytes	16.0 KB	0.0 bytes	121	2024-12-17 11:15:09			utf8mb4_genera...	
wypozyczenia	InnoDB	10	Dynamic	7	2340	16.0 KB	0.0 bytes	32.0 KB	0.0 bytes	12	2024-12-10 11:21:10			utf8mb4_genera...	

A tak wyglądają poszczególne tabele:

Ksiazki:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
id_produktu	int		NO			select,insert,update,references	auto_increment	
tytul	varchar(255)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
seria	varchar(255)		YES	utf8mb4	utf8mb4_genera...	select,insert,update,references		
autor	varchar(255)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
wydawnictwo	varchar(255)		YES	utf8mb4	utf8mb4_genera...	select,insert,update,references		
liczba_stron	int		YES			select,insert,update,references		
rok_wydania	int		YES			select,insert,update,references		

Uzytkownicy:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
id_uzytkownika	int		NO			select,insert,update,references	auto_increment	
imie	varchar(50)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
nazwisko	varchar(50)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
email	varchar(100)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
haslo	varchar(255)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
status	enum('aktywny','zaw...)	aktywny	YES	utf8mb4	utf8mb4_genera...	select,insert,update,references		

Wypozyczenia:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
id_uzytkownika	int		NO			select,insert,update,references	auto_increment	
imie	varchar(50)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
nazwisko	varchar(50)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
email	varchar(100)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
haslo	varchar(255)		NO	utf8mb4	utf8mb4_genera...	select,insert,update,references		
status	enum('aktywny','zaw...)	aktywny	YES	utf8mb4	utf8mb4_genera...	select,insert,update,references		

II. Struktura programu

Plik main.cpp

Kluczowy plik, zawierający logikę i zarządzanie najważniejszymi funkcjami programu. Uruchamia mechanizmy niezbędne do poprawnego działania programu, takich jak wywołanie funkcji, inicjalizacja zmiennych, tworzenie obiektów i pobieranie danych od użytkownika.

Plik database.cpp

W pliku tym znajdują się wszystkie mechanizmy, które są wykorzystywane do połączenia i interakcji z bazą danych. Odpowiedzialny jest za zarządzanie danymi, ich odczyt, zapis itd. Dzięki temu plikowi komunikacja została znacznie usprawniona.

Plik user.cpp

Plik ten implementuje do programu funkcjonalności związane z zarządzaniem użytkownika w projekcie. Obejmuje on definicje metod umożliwiających tworzenie i edycje danych użytkownika.

Plik book.cpp

Implementuje logikę związaną z obsługą danych o książkach w projekcie. Definiuje metody umożliwiające dodawanie oraz edycję książek w bazie danych.

Zawartość katalogu include

Zawiera pliki nagłówkowe, które definiują klasy i funkcje używane w aplikacji. Zapewnia to większą czytelność kodu.

Zawartość katalogu lib

Znajduje się tutaj biblioteka zewnętrzna, która wspomaga aplikację w połączeniu z bazą danych. Natomiast folder include przechowuje pliki nagłówkowe definiujące zmienne używane do połączenia w bazie danych.

Makefile

Definiuje reguły kompilacji i budowy aplikacji. Określa zależność między plikiem źródłowym, nagłówkowymi oraz docelowymi plikami wykonywalnymi. Zawiera polecenia do kompilowania kodu. Ułatwia automatyzację procesu kompilacji.

Plik libraryonline.sql

Skrypt SQL do tworzenia i inicjalizacji bazy danych w projekcie.

4. Postumowanie

Projekt ten to prosty system zarządzający biblioteką, stworzony w C++ z integracją bazy danych MySQL. Aplikacja zapewnia zarządzanie kluczowymi operacjami, takimi jak dodawanie, aktualizowanie i przeglądanie elementów bazy. Projekt demonstruje zrozumienie podstaw programowania w C++ oraz integrację z bazą danych. Może stanowić podstawę do dalszych prac na systemem zarządzania opartym na podobnych technologiach.