PROE 2020Z  
Zadanie Programistyczne nr 1

Rutkowski Hubert   
Numer albumu: 307462

Oceniający: mgr inż. Marcin Bączyk

„System obsługi danych w bibliotece – część dotycząca bazy danych biblioteki”

**Specyfikacja wstępna**

1. Temat

„System obsługi danych w bibliotece”

2. Interfejs aplikacji

- uruchamianie z poziomu powłoki

- menu użytkownika z wyborem opcji

3. Operacje wykonywane przez program

b) Baza danych biblioteki:

- informacja na temat stanu katalogu (wypożyczona, dostępna itp.)

- zmiana stanu katalogu (dodawanie, usuwanie katalogu),

- zapisywanie i wczytywanie katalogu z pliku,

4. Oczekiwany wynik działania programu

- stworzenie funkcjonalnej bazy danych biblioteki z możliwością łatwego dostępu i modyfikacji przez użytkowników i administratorów;

5. Obiekty i ich funkcjonalności tworzone w trakcie działania programu

Book

Przechowuje dane na temat pozycji w katalogu

Catalog

Obiekt-lista, przechowujący dane ogólne katalogu

Menu

Obiekt służący do porozumiewania się z użytkownikiem

UserInput

Obiekt do przetworzenia informacji wejściowych użytkownika

File

Obiekt do obsługi plików (zapisywania i wczytywania baz danych)

6. Zależności i orientacyjny typ poszczególnych obiektów

UserInput

Obiekt agregowany z klasą Menu

Catalog

Obiekt przechowujący obiekty typu Book i metody operacji na elementach listy

Menu

Klasa przechowująca metody operowania plikami i dostęp do listy

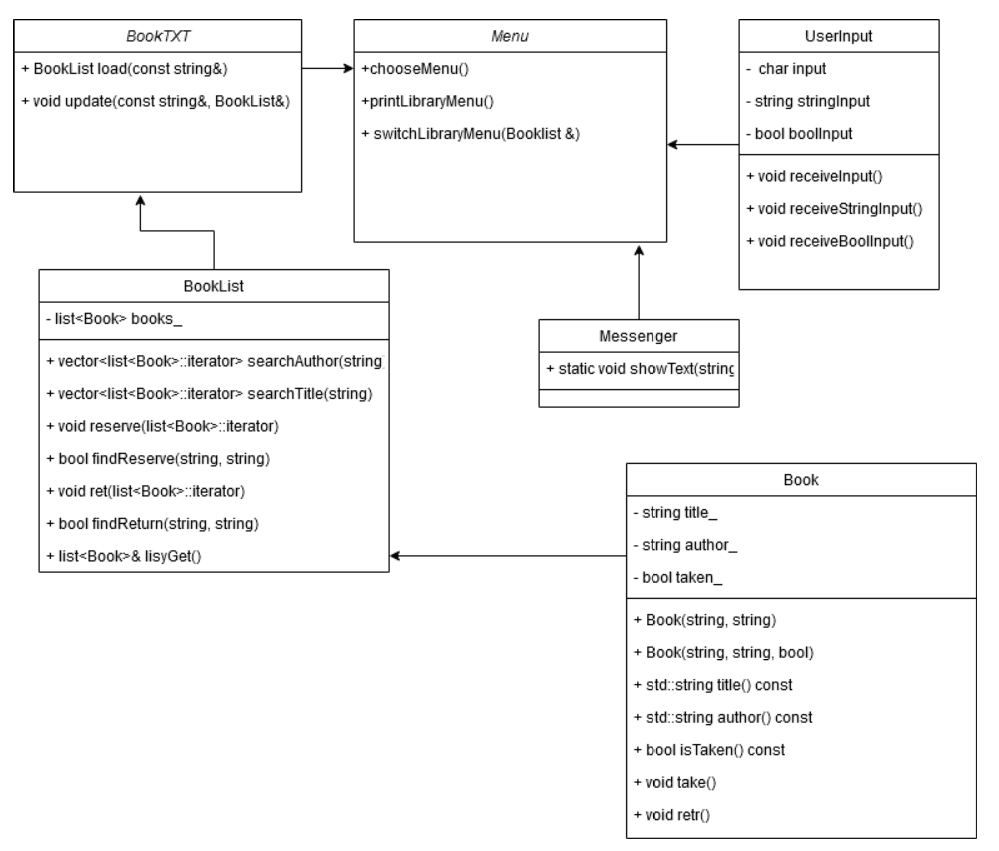
7. Przypadki użycia

Przykład wywołania programu: nazwa\_programu

Wyświetlenie menu -> pobranie wyboru od użytkownika -> pobranie danych wejściowych od użytkownika -> wykonanie pożądanej operacji -> wyświetlenie menu

Odstępstwa od specyfikacji wstępnej w kontekście klas:  
Messenger – Obsługiwanie wyświetlania tekstu przez program.  
BookList – Przechowuje książki (stary Catalog).  
BookTXT – Wykonuje prace związane z obsługą pliku tekstowym (stary File)

**Diagram klas**



**Opis najciekawszych fragmentów programu**

Najciekawszą funkcjonalnością programu jest możliwość wypożyczania książek i ich zwracania. Dodatkowo dane o wypożyczeniu zostają zapisane na stałe w pamięci programu (pliku tekstowym) dzięki czemu pomimo zamknięcia programu obecny stan książek jest wciąż pamiętany i przy ponownym uruchomieniu automatycznie wgrywany.

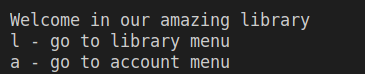
**Krótki opis najważniejszych klas i metod na poziomie funkcjonalnym**

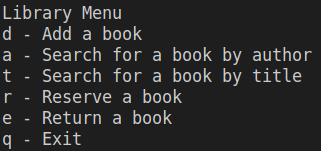
Według mnie najważniejszymi klasami są BookTXT oraz BookList. Ta pierwsza pozwala nam wgrać i aktualizować na bieżąco stanu biblioteki przy użyciu pliku tekstowego. Ta druga jest listą przechowującą obiekty klasy Book. Możemy wykonywać na niej wiele operacji potrzebnych w zarządzaniu biblioteką (dodawanie książek, wypożyczanie i zwracanie ich). Istotną klasą jest również Menu, które stanowi szatę graficzną umożlwiającą pracę z programem(jednak ta klasa jest wspólna i prowizoryczna, zostanie rozwinięta w drugiej części projektu). W niej operują jeszcze dwie klasy. UserInput, który pozwala na komunikację między programem, a użytkownikiem. Messenger, druga z tych klas, sprawia, że kod jest zdecydowanie łatwiejszy do zrozumienia, a komunikowanie się z użytkownikiem staje się przyjemnością.

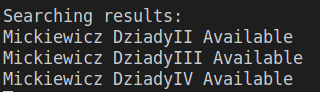
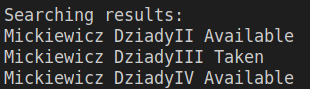
**Krótka instrukcja posługiwania się programem**

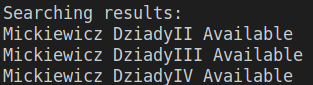
Kilka słów wstępu , by móc omówić działanie programu.  
Program został napisany w systemie operacyjnym Linux, zatem preferowanym jest, by właśnie w nim posługiwać się programem.  
Zaleca się wykonać kompilację wchodząc w folder z programem (obecnie src ale można go sobie zmienić wedle gustu).  
  
Kompilacji dokonujemy poprzez komendę:  
g++ -std=c++17 main.cpp -o wybrana\_dowolna\_nazwa  
  
Aby uruchomić program wystarczy wpisać komendę:  
./wybrana\_dowolna\_nazwa   
  
  
Po uruchomieniu programu należy użyć klawisza ‘l’ aby wywołać menu operacji na bazie książek biblioteki. Następnie oczom użytkownika objawia się moc funkcjonalności. Jest to dodawanie książki, szukanie książki po autorze lub tytule, rezerwowanie książki oraz jej zwracanie. Wystarczy, że użytkownik wciśnie wybrany klawisz, któremu przyporządkowana jest dana funkcja by jej użyć. Program jest dosyć intuicyjny lecz i tak podczas całego procesu po kolei „mówi” użytkownikowi co powinien teraz zrobić/wpisać. Dzięki temu nawet kompletnie nieobeznany użytkownik, na pewno poradzi sobie z jego obsługą.

**Przykładowe działanie programu**

Wybór „podmenu”:  


Menu biblioteki:  


Szukanie książek po autorze:  
  
  
  
Rezerwowanie książki:  
 (pytanie o autora)  
 (pytanie o tytuł)  
 (informacja o sukcesie)  
 (efekt końcowy – książka wypożyczona)

Oddawanie książki:  
 (pytanie o autora)  
 (pytanie o tytuł)  
 (informacja o sukcesie)  
 (efekt końcowy – książka została oddana i jest znów dostępna)

Wciśnięcie złego klawisza w menu:  
 (informacja dla użytkownika)