Politechnika Śląska

Dokumentacja projektu

2021/2022

Płytoteka

Projekt zaliczeniowy z przedmiotu
Programowanie obiektowe i graficzne

Kierunek: Informatyka

Członkowie zespołu:

Michał Pawełek
Jan Walica

Gliwice, 27 czerwca 2022

Spis treści

1 Opis projektu

Celem projektu jest stworzenie aplikacji będącej interfejsem bazy danych, który umożliwi katalogowanie i przeglądanie informacji o wydawnictwach muzycznych oraz podmiotach z nimi związanych – zespołach i artystach.

Do podstawowych zadań programu należą:

1. Komunikacja z bazą danych
2. Przeglądanie informacji o Albumach, Artystach, Utworach i Zespołach
3. Dodawanie kolejnych wpisów w każdej z kategorii
4. Edycja dotychczasowych wpisów
5. Usuwanie wybranych wpisów
6. Obsługa zależności pomiędzy obiektami w bazie

2 Przebieg realizacji

Projekt zrealizowany został w modelu MVVM. Dodatkowo oddzielona została warstwa dostępu do danych, w której zrealizowano połączenie z bazą danych. Interfejs powstał z wykorzystaniem technologii WPF.

2.1 Data Access Layer

W tej warstwie zaimplementowano klasy odpowiadające tabelom w bazie danych (Encje). Każda z nich implementuje interfejs *ICRUDStrings* zapewniający metody zwracające ciągi dodawane do zapytań do bazy. Zbudowane są w podobny sposób: posiadają publiczne własności odpowiadające kolumnom tabeli w bazie, a następnie przeciążone konstruktory: jeden tworzący obiekt na podstawie danych otrzymanych z bazy, drugi na podstawie danych przekazanych od użytkownika i trzeci – kopiujący. Metody: przeciążone *ToString()*, *Equals()* i *GetHashCode()* oraz dwie implementujące wspomniany interfejs: *ToInsert()* oraz *ToUpdate()*.

W tej warstwie zamieszczono także repozytoria – statyczne klasy narzędziowe. Każda z nich ponownie odpowiada tabelom bazy. Oprócz stałych ciągów znakowych przechowujących zapytania w języku SQL posiadają metody:

- *PobierzWszystko()* – pobierającą z bazy wszystkie rekordy z danej tabeli i zwracającą je jako listę,

- *Dodaj* – dodającą do bazy pojedynczy wpis na podstawie przekazanego obiektu,
- *Edytuj()* – wykonującą zapytanie modyfikujące wpis o podanych id w bazie,
- *Usun()* – usuwającą wpis o podanym id.

2.2 Model

Warstwa zawiera jedną klasę – *Model*, w której skład wchodzą list obiektów klas z DAL, a także szereg metod pobierających dane z wykorzystaniem klas warstwy DAL. Przykładowe z nich, to:

- *OdswiezAlbumy()* – pobiera aktualną listę obiektów i aktualizuje ją w modelu,
- *ZnajdzZespolPoId()* – wyszukuje w liście obiektów tego, o podanym numerze id,
- *PobierzSkladoweAlbumu()* – spośród wszystkich składowych zwraca listę tych, które są w związku z przekazanym albumem,
- *CzyAlbumJestJuzWRepozytorium()* – sprawdza, czy przekazany album występuje w bazie,
- *DodajAlbumDoBazy()* – jeśli przekazany album jest unikatowy, dodaje go do bazy danych i aktualizuje listę obiektów,
- *EdytujAlbumWBazie()* – modyfikuje wpis w bazie na podstawie przekazanego obiektu i id rekordu,
- *UsunAlbumZBazy()* – usuwa wpis o podanym id i aktualizuje listę obiektów.

2.3 ViewModel

Główny plik, *MainViewModel*, zawiera wyłącznie obiekty klas pomocniczych tej warstwy oraz konstruktor, który je tworzy. Każda z klas pomocniczych odpowiada jednej zakładce widoku. Posiada ona listy obiektów oraz szereg pomocniczych własności bezpośrednio połączonych z widokiem, takich jak informacje o zaznaczonych elementach list, danych wprowadzanych przez użytkownika, czy możliwych zachowaniach interfejsu. Klasy te dziedziczą

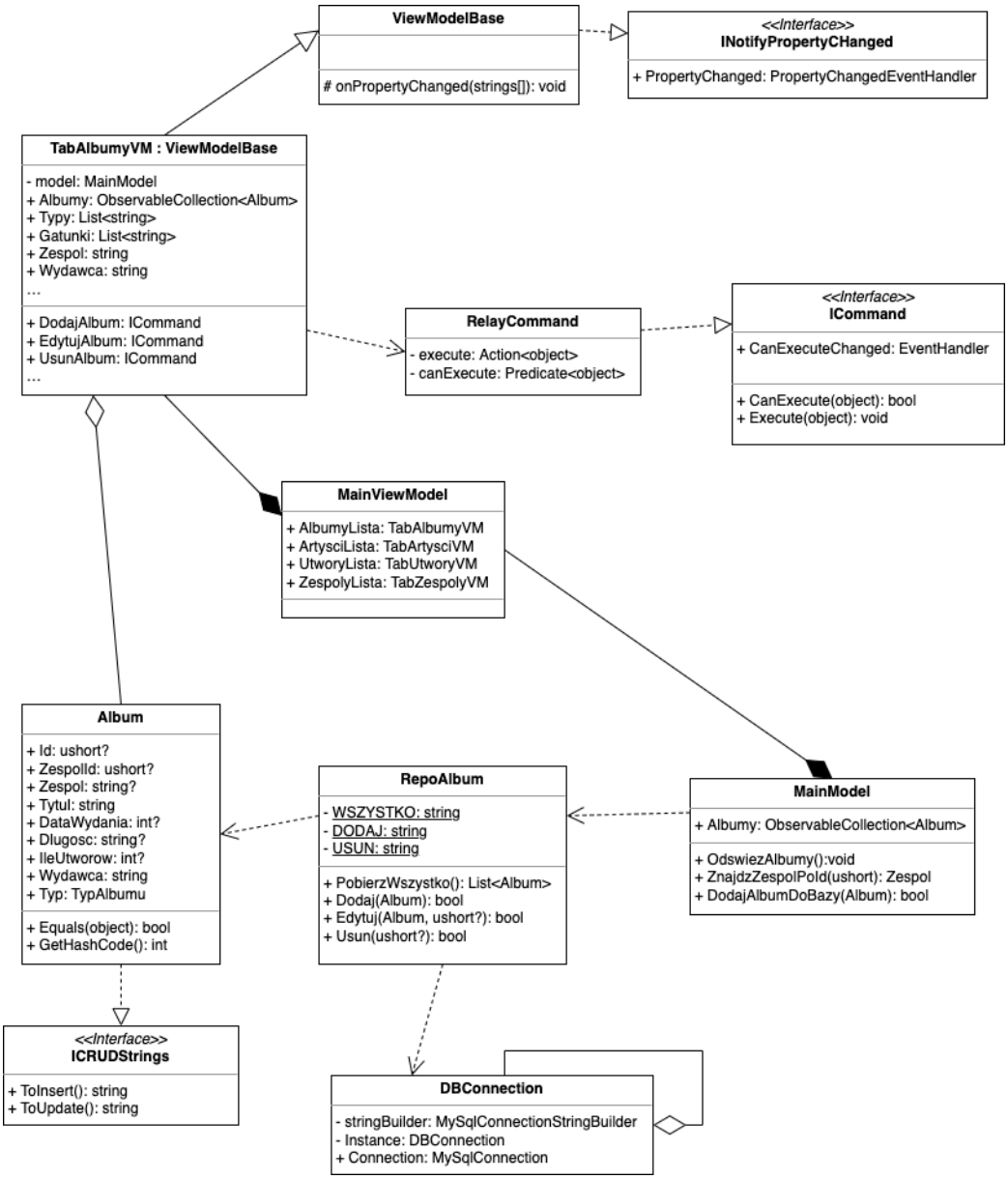
po klasie *ViewModelBase*, która zaś implementuje interfejs *INotifyPropertyChanged* i zapewnia obsługę zdarzeń potrzebną przy wiązaniu warstwy z widokiem.

2.4 Widok

Warstwa widoku została w całości zrealizowana w jednym oknie WPF, które zawiera wszystkie kontrolki powiązane z odpowiednimi klasami warstwy ViewModelu.

3 UML

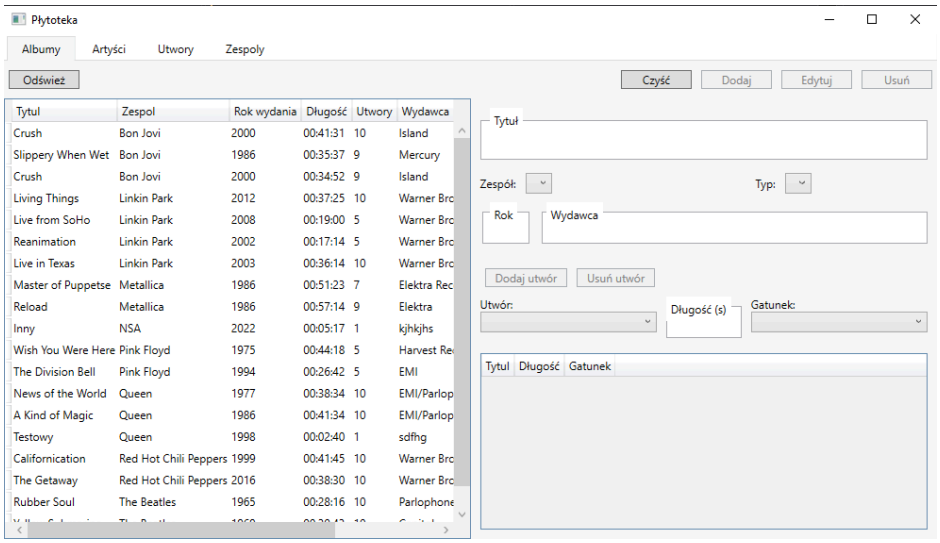
Poniższy rysunek prezentuje wycinek projektu odpowiadający za funkcjonowanie jednej zakładki interfejsu, co odpowiada jednej tabeli w bazie danych i powiązanym z nią związkiem.



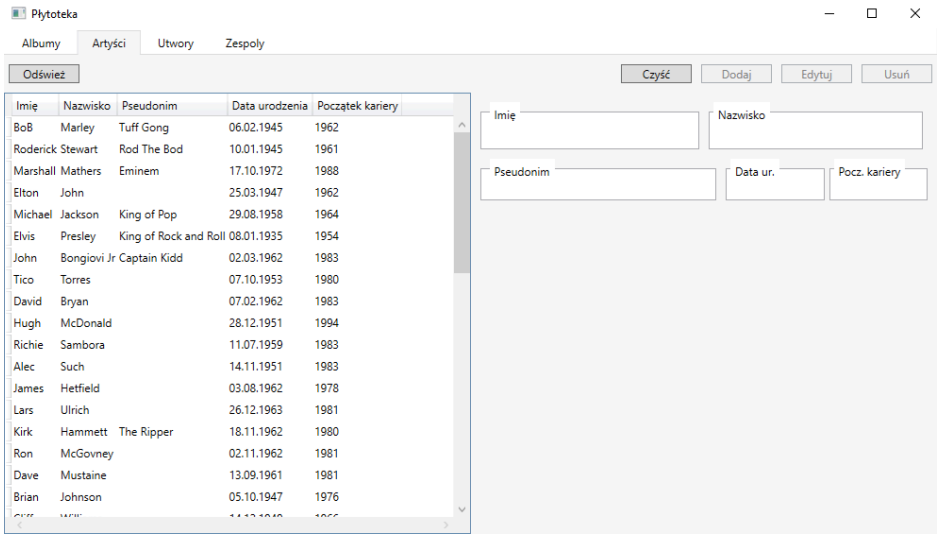
Rysunek 1: Diagram UML

4 Instrukcja użytkownika

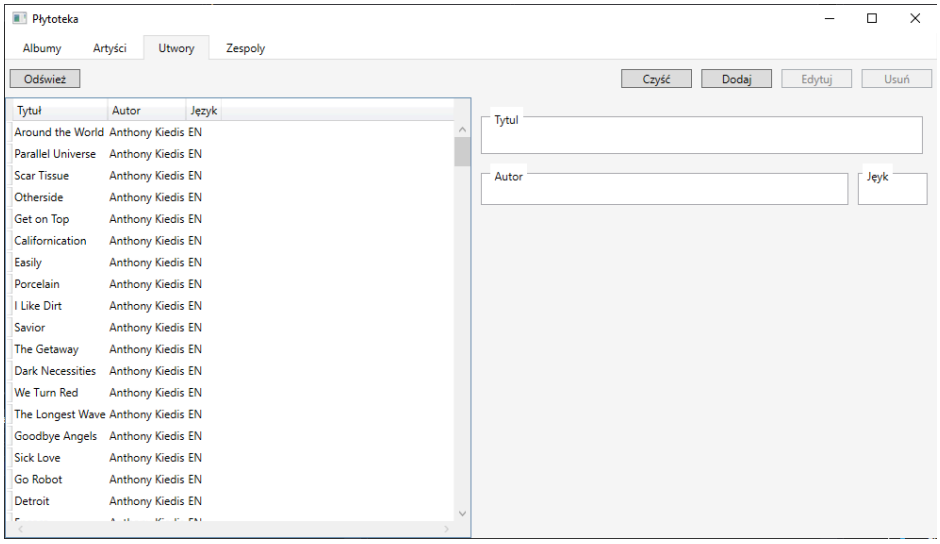
Aplikacja składa się z czterech zakładek przełączanych w górnej części ekranu – *Albumy*, *Artyści*, *Utwory*, *Zespoły*. Każda z nich odpowiada za prezentację danych zgodnie z jej nazwą.



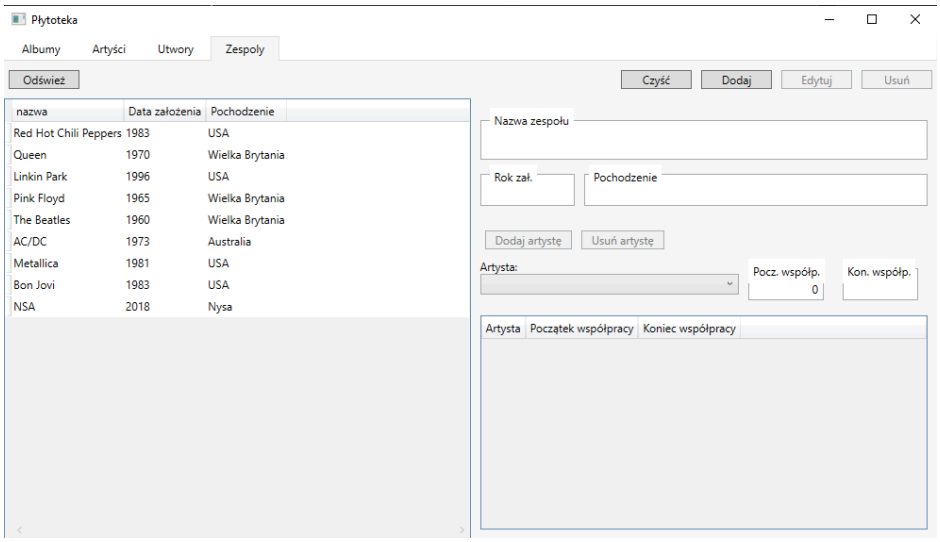
Rysunek 2: Zakładka *Albumy*



Rysunek 3: Zakładka *Artyści*



Rysunek 4: Zakładka *Utwory*



Rysunek 5: Zakładka *Zespoły*

Każda zakładka zostałą podzielona na dwa obszary. Po lewej znajduje się lista rekordów pobranych z bazy, a po prawej wyświetlany jest formularz umożliwiający dodanie, lub modyfikację danych. W celu przejrzenia szczegółów danego rekordu i jego ewentualnej modyfikacji należy dwukrotne kliknąć w wybrany wiersz listy, po czym dany zostaną wczytane do formularza. Po zmodyfikowaniu danych należy kliknąć przycisk *Edytuj*, co dodaje nowe rekordy do bazy danych.

Aby dodać nowy rekord należy wyczyścić formularz odpowiednim przyciskiem, a następnie uzupełnić go nowymi danymi i wybrać przycisk *Dodaj*

W przypadku dwóch zakładek dolna połowa formularza umożliwia niezależne dodawanie danych powiązanych z daną kategorią (utworów znajdujących się na albumie, czy muzyków będących członkami zespołu). Posiada ona niezależne przyciski dodawania i usuwania.

Każdą listę wyświetlaną przez aplikację można sortować względem dowolnej kolumny rosnaco lub malejąco.

5 Podsumowanie

Projekt został zrealizowane według założeń. Aplikacja dostarcza użytkownikowi interfejs, który pozwala na łatwą obsługę bazy danych. Zrozumienie i opanowanie podstawowych funkcji jest łatwe i nie zajmuje dużo czasu.

Podczas realizacji projektu napotkano pewne trudności związane z prawidłowym funkcjonowaniem warstwy widoku, jednak po przestudiowaniu wybranych zagadnień dokumentacji technologii WPF udało się zrealizować interfejs użytkownika.

Także niedogodności związane z funkcjonowaniem bazy danych sprawiły, że rozwiązywanie problemów zajmowało więcej czasu przez konieczność weryfikowania źródła problemu w dwóch odrębnych miejscach.

6 Wnioski

Podczas realizacji projektu rozpoznano potencjalne możliwości rozwoju aplikacji w przyszłości.

Jednym z udogodnień usprawniającym pracę z programem może być pole wyszukiwania/filtrowania wyników, które pozwoli na sprawniejsze poruszanie się po rekordach szczególnie, gdy baza rozrośnie się.

Także dalsza implementacja zależności między tabelami bazy jest możliwym obszarem dalszych prac ze względy na potencjał jak dałyby kolejne panele widoku prezentujące dane w przystępny sposób.