

**Inżynieria oprogramowania - projekt**

*Program do odtwarzania muzyki*

Prowadzący: Autor:

mgr inż. Ewa Żesławska Patryk Szkafarak

nr albumu: W62048

Kierunek: 6IID/2019, grupa GPL01

Rzeszów 2022

Spis treści

[1. Założenia projektu 3](#_Toc109402631)

[1.1 Opis świata rzeczywistego 3](#_Toc109402632)

[2. Wymagania funkcjonalne i niefunkcjonalne 3](#_Toc109402633)

[2.1 Wymagania funkcjonalne 3](#_Toc109402634)

[2.2 Wymagania niefunkcjonalne 3](#_Toc109402635)

[3. Wykorzystany język, środowisko oraz biblioteki 3](#_Toc109402636)

[4. Diagramy i opisy 3](#_Toc109402637)

[4.1 Opis świata rzeczywistego 3](#_Toc109402638)

[4.2 Diagram UML 4](#_Toc109402639)

[4.3 Definicja aktorów 4](#_Toc109402640)

[4.4 Scenariusze przypadków użycia 5](#_Toc109402641)

[4.5 Diagram aktywności 6](#_Toc109402642)

[4.6 Diagram klas 8](#_Toc109402643)

[4.7 Diagram sekwencji 9](#_Toc109402644)

[4.8 Diagram stanów 10](#_Toc109402645)

[5. Prezentacja aplikacji 11](#_Toc109402646)

[6. Repozytorium 14](#_Toc109402647)

[7. Literatura 14](#_Toc109402648)

[8. Podsumowanie 14](#_Toc109402649)

# 1. Założenia projektu

Tematem projektu jest aplikacja desktopowa do odtwarzania muzyki w formacie MP3. Jest ona stworzona w języku **C**# przy pomocy **WindowsForms** (do stworzenia GUI). Korzysta ze stosownych API do pomocy w wyszukiwaniu. Użytkownik ma do wyboru kilka opcji dotyczących samego wyszukiwania i pobierania filmów.

## 1.1 Opis świata rzeczywistego

Ludzie słuchają muzyki codziennie. Przy pomocy tej aplikacji każdy użytkownik może słuchać dowolnie wybranej przez siebie muzyki w formacie mp3. Aplikacja pozwala również na przewijanie utworów, stopowanie, dodawanie i usuwanie z listy jak i zmianę głośności od strony interfejsu.

# 2. Wymagania funkcjonalne i niefunkcjonalne

## 2.1 Wymagania funkcjonalne

* Dodawanie utworów do listy,
* Usuwanie utworów z listy,
* Możliwość interakcji z paskiem trwania piosenki,
* Zmiana piosenki na następną i wcześniejszą,
* Możliwość pauzowania i zatrzymywania utworu.

## 2.2 Wymagania niefunkcjonalne

* Prosty i intuicyjny interfejs,
* Wyświetlanie miniatury utworu.

# 3. Wykorzystany język, środowisko oraz biblioteki

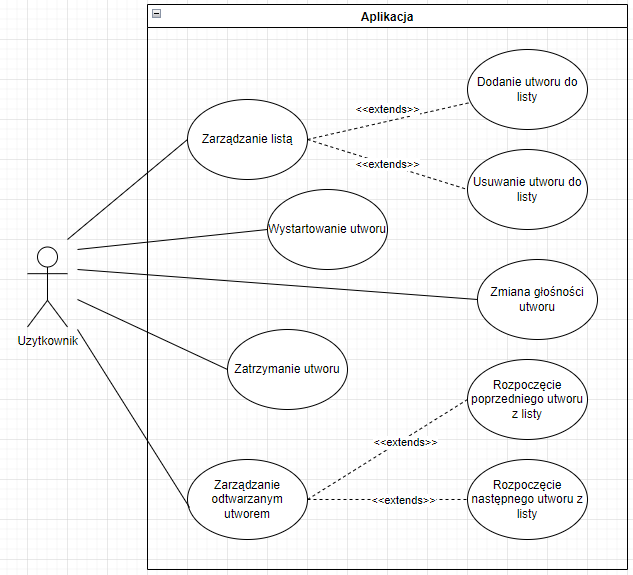
Aplikacja została stworzona przy pomocy języka **C#** oraz **Windows Forms** w środowisku programistycznym **Visual Studio Code 2019**. Wykorzystano biblioteki WMPLib(Windows Media Player) oraz TagLib(odczytywanie metadanych plików mp3).

# 4. Diagramy i opisy

## 4.1 Opis świata rzeczywistego

Użytkownik może dodawać utwory do listy jak i je usuwać. Dodawane utwory muszą być w formacie mp3. Do odtworzenia utworu użytkownik wybiera jeden z listy i naciska play żeby go odsłuchać. Posiada też możliwość zmiany piosenki za pomocą przycisków "Previous" oraz "Next". Z dostępnych opcji jest też w stanie zatrzymać utwór całkowicie za pomocą przycisku "Stop", a przyciskiem "Pause" utwór spauzować. Użytkownik w dowolnym momencie jest w stanie zmienić głośność utworu.

## 4.2 Diagram UML



*Rysunek.1 - diagram UML*

## 4.3 Definicja aktorów

|  |  |  |
| --- | --- | --- |
| **AKTOR** | **OPIS** | **PRZYPADEK UŻYCIA** |
| Użytkownik | Może on dodawać utwory jak i je usuwać. Może utwory odtwarzać oraz zatrzymywać. Posiada również możliwość zmiany głośności jak i opcję zmiany utworu na poprzedni lub następny z listy. | -PU Zarządzanie listą powiązany relacją <<extends>> z PU Dodanie utworu do listy oraz Usuwanie utworu z listy  -PU Wystartowanie utworu  -PU Zatrzymanie utworu  -PU Zmiana głośności  -PU Zarządzanie odtwarzanym utworem powiązany relacją <<extends>> z PU Rozpoczęcie poprzedniego utworu z listy oraz Rozpoczęcie następnego utworu z listy. |

*Tabela.1 - definicja aktorów*

## 4.4 Scenariusze przypadków użycia

**PU Wybór utworu do dodania do listy.**

**OPIS**

**CEL: Dodanie utworu do listy.**

WS: Wybranie przycisku "List" i wybranie odpowiedniego utworu.

WK: Utwór zostaje dodany do kolejki i można go odtworzyć.

**PRZEBIEG:**

Użytkownik wybiera przycisk "List" w aplikacji by następnie wybrać na swoim komputerze utwór, który chce dodać do listy. Po wybraniu utworu znajduje się już on na liście i można go odtwarzać.

**PU Zmiana utworu na następną z listy.**

**OPIS**

**CEL: Zmiana utworu na następny na liście.**

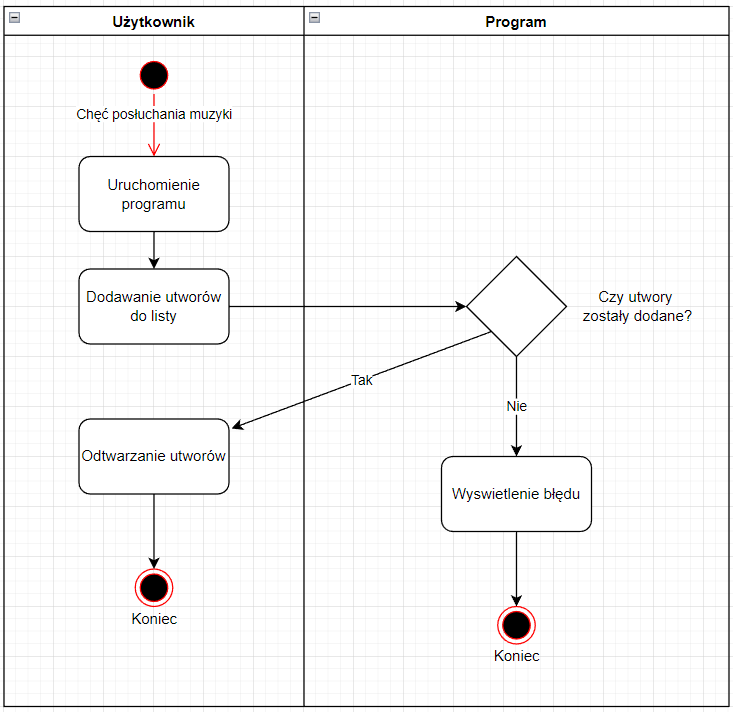
WS: Na liście muszą znajdywać się przynajmniej 2 utwory.

WK: Utwór można zmienić na następny.

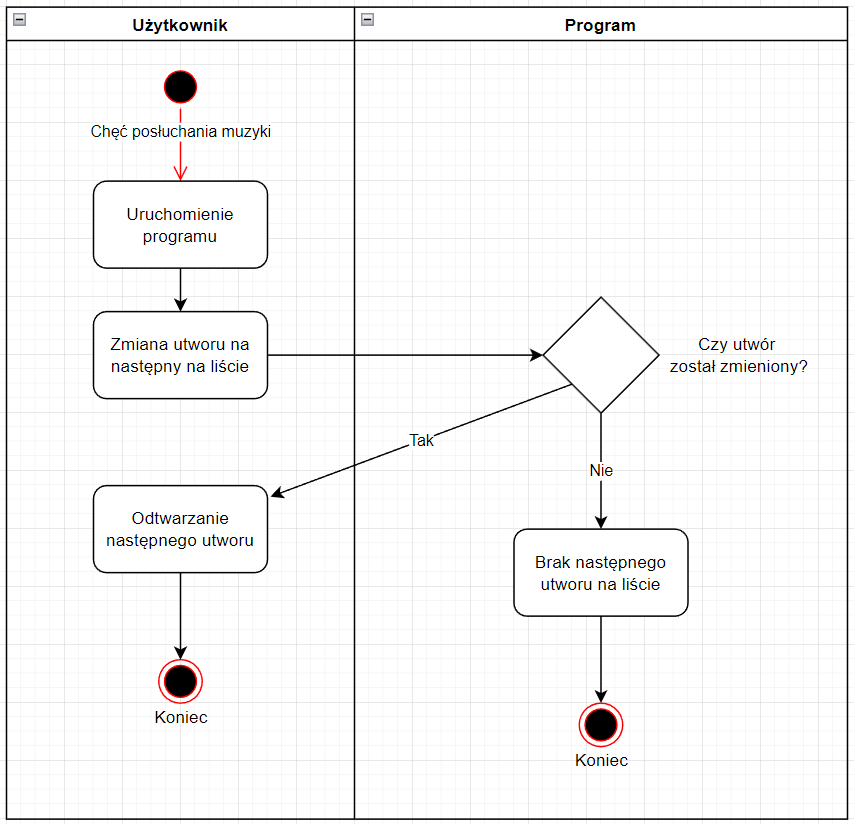
**PRZEBIEG:**

Gdy użytkownik posiada na liście przynajmniej 2 utwory jest w stanie za pomocą przycisku "Next" zmienić odtwarzany utwór na następny.

## 4.5 Diagram aktywności

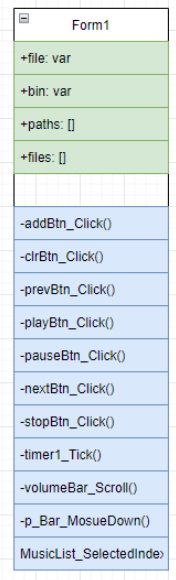


*Rysunek.2 – dodawanie utworu do listy*



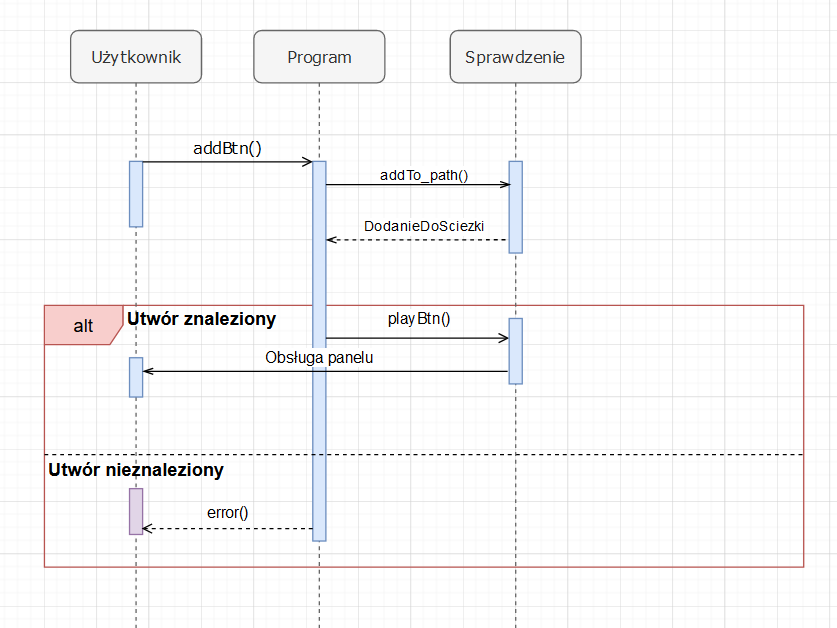
*Rysunek.3 – zmiana odtwarzanego utworu na następny na liście*

## 4.6 Diagram klas



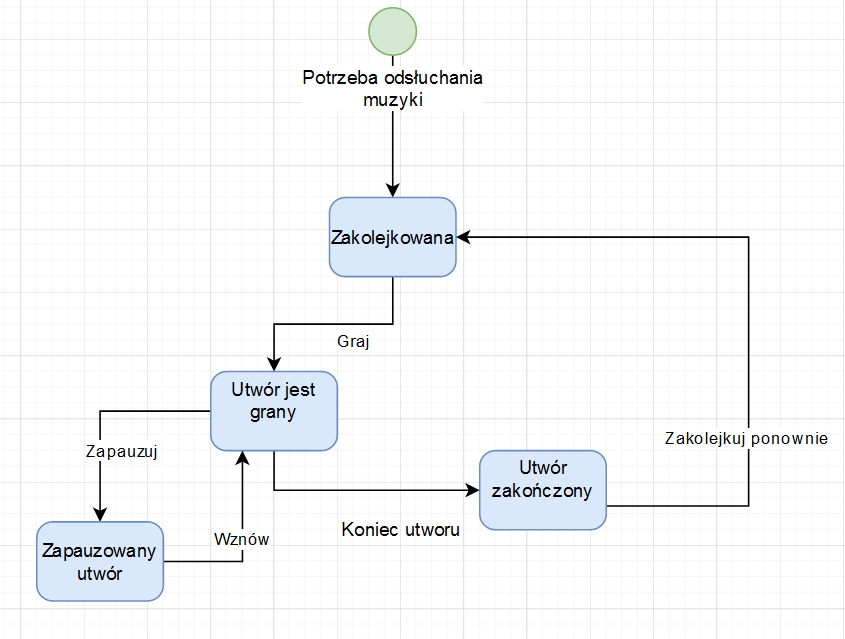
*Rysunek.4 - diagram klas*

## 4.7 Diagram sekwencji



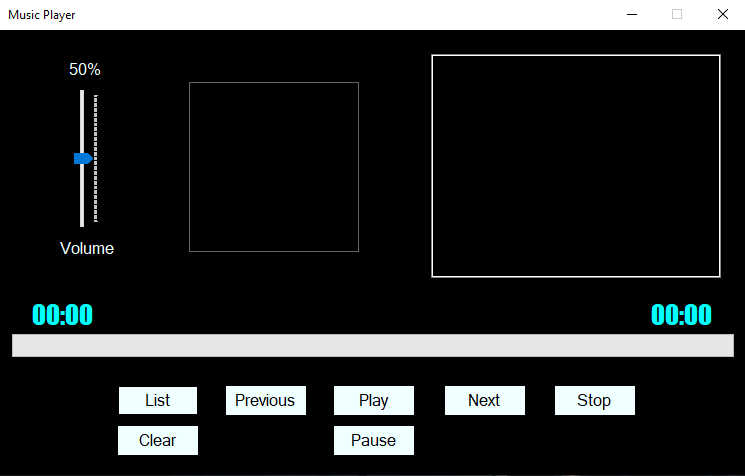
*Rysunek.5 - diagram sekwencji*

## 4.8 Diagram stanów



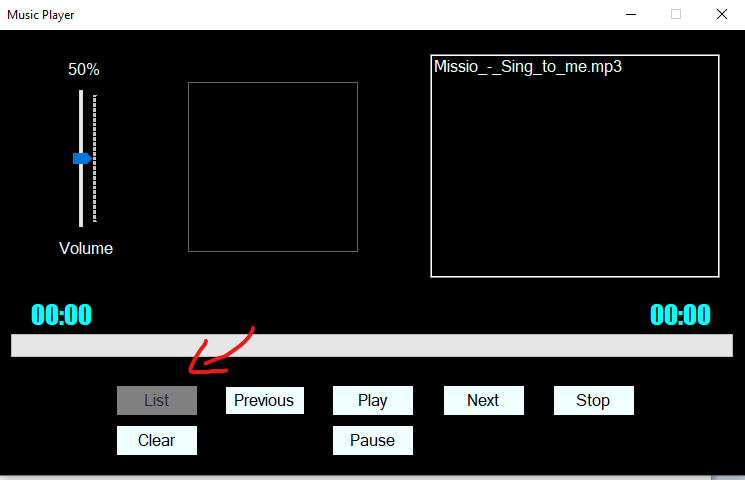
*Rysunek.6 - diagram stanów*

# 5. Prezentacja aplikacji



*Rysunek.7 - główny ekran*

Użytkownik po uruchomieniu programu widzi panel kontrolny. Może tutaj przeprowadzić kilka operacji. Jedną z nich jest opcja „List”, przy pomocy której można dodać utwór do kolejki granych utworów.



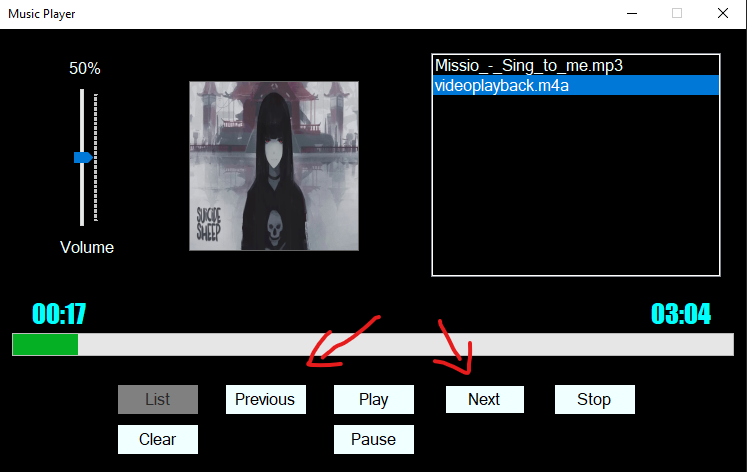
*Rysunek.8 - dodanie utworu do listy*

Przy odtwarzaniu muzyki użytkownik widzi pasek postępu, który charakteryzuje również długość utworu – po lewej widać aktualną minutę a po prawej całkowita długość utworu. Utwór można wznawiać i pauzować przy pomocy guzików „Play” oraz „Pause”.



*Rysunek.9 - odtwarzanie utworu*

Jeśli użytkownik ma na liście kolejek różne utwory to może się między nimi przełączać. Wykonuje to przy pomocy przycisków „Previous” oraz „Next”.



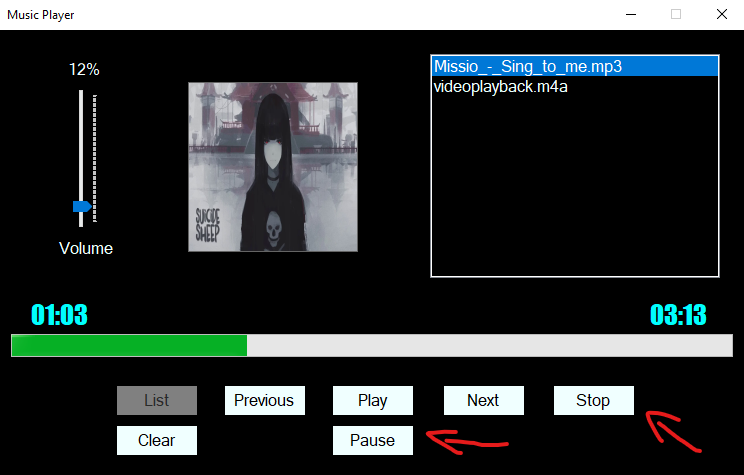
*Rysunek.10 - zmiana utworu*

Jedną z najważniejszych opcji jest możliwość podnoszenia i obniżania poziomu głośności granego utworu. Suwak znajduje się po lewej strony miniatury i obecną głośność wyraża się w procentach.



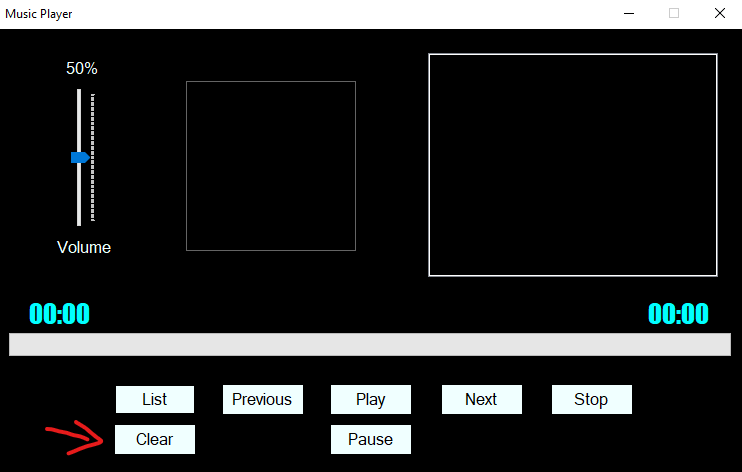
*Rysunek.11 - zmiana głośności*

Obecnie grany utwór można za pauzować lub go zupełnie zastopować.



*Rysunek.12 - pauzowanie i stopowanie utworów*

Ostatnią opcją jest możliwość wyczyszczenia listy kolejki. Po użyciu tej opcji wszystkie utwory zostaną usunięte i użytkownik będzie musiał ponownie dodać nowe utwory.



*Rysunek.13 - czyszczenie listy*

# 6. Repozytorium

https://github.com/patryksz1021/Inzynieria-oprogramowania-projekt

# 7. Literatura

* <https://www.w3schools.com/cs/index.php> (data dostępu 20.07.2022)
* <https://docs.microsoft.com/pl-pl/> (data dostępu 20.07.2022)

# 8. Podsumowanie

Dzięki temu projektowi utrwaliłem i podniosłem swoje umiejętności przy korzystaniu z języka C# oraz WindowsForms'a. Powyższy projekt wciąż ma możliwość rozwoju poprzez dodanie np. tekstów wyświetlanych piosenek, poprawienie wizualnej strony aplikacji oraz umożliwić odtwarzanie utworów w formacie MP4.