

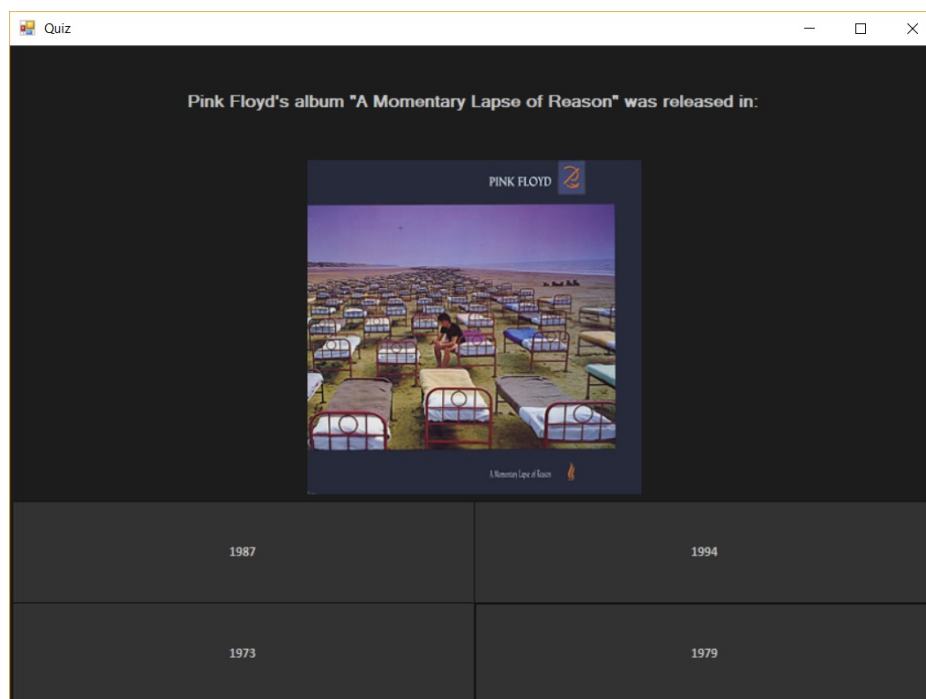
# Programowanie Obiektowe - dokumentacja projektu

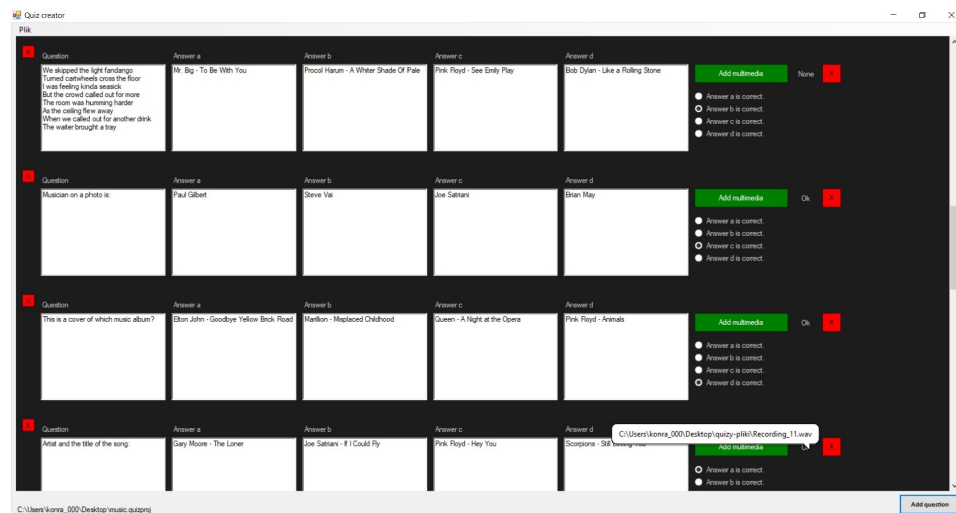
Konrad Werbliński

26 czerwca 2017

## 1 Ogólny opis

Tematem mojego projektu była obiektowa implementacja programu, umożliwiającego rozwiązywanie i tworzenie quizów. Program został napisany w języku C#, przy użyciu biblioteki Windows Forms.





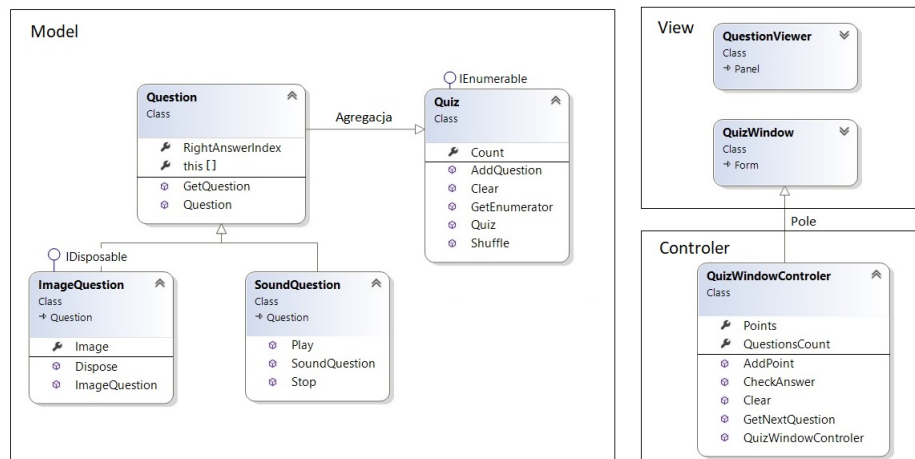
## 2 Implementacja - okno gry

Przy implementacji okna gry wykorzystałem następujące klasy:

1. **QuizWindow** - Klasa reprezentująca okno gry. Dziedziczy po klasie **Form**, z biblioteki **Windows Forms**.
2. **QuizWindowController** - Klasa odpowiedzialna za logikę rozgrywki: wczytuje quiz, rozdysponowuje pytania, sprawdza odpowiedzi oraz zlicza punkty. Zawiera właściwości **Points**, **QuestionsCount** oraz metody **AddPoint()**, **GetNextQuestion()**, **CheckAnswer()** i **Clear()** (wywołuje **Clear()** w obiekcie typu **Quiz**).
3. **Question** - Klasa reprezentująca pytanie w quizie. Zawiera treść pytania (metoda **GetQuestion()**), możliwe odpowiedzi (przechowywane w cztero-elementowej tablicy, dostęp do nich zaimplementowany za pomocą indeksa klasy **Question**) oraz informacje o indeksie poprawnej odpowiedzi (właściwość **RightAnswerIndex**).
4. **ImageQuestion** - Klasa reprezentująca pytanie z obrazem. Dziedziczy po klasie **Question** oraz implementuje interfejs **IDisposable**. Zawiera właściwość **Image** przechowującą obraz oraz metodę **Dispose()** służącą do zwalniania zasobów po zakończeniu korzystania z obiektu.
5. **SoundQuestion** - Klasa reprezentująca pytanie z dźwiękiem. Dziedziczy po klasie **Question**. Pozwala odtwarzać plik dźwiękowy. Do tego celu wykorzystuje klasę **SoundPlayer** z przestrzeni nazw **System.Media**. Implementuje metody **Play()** oraz **Stop()**.
6. **Quiz** - Klasa reprezentująca quiz. Jest agregacją obiektów klasy **Question**. Implementuje interfejs **IEnumerable<Question>**. Implementuje metody **AddQuestion()** (dodawanie pytań, wykorzystywane przy wczytywaniu quizu), **Clear()** (do zwalniania zasobów po zakończeniu pracy z obiektem), **GetEnumerator**, **Shuffle()** (mieszanie kolejności pytań).

7. **QuestionViewer** - Komponent służący do wyświetlania pytania. Dziedziczy po klasie **Panel** z biblioteki Windows Forms. Służy do wyświetlania treści pytań zawartej w obiektach typu **Question**. Sposób wyświetlania pytania jest zależny od typu obiektu pytania, które jest wyświetlane.

Okno gry zostało zaimplementowane z wykorzystaniem wzorca projektowego Model-View-Controller.



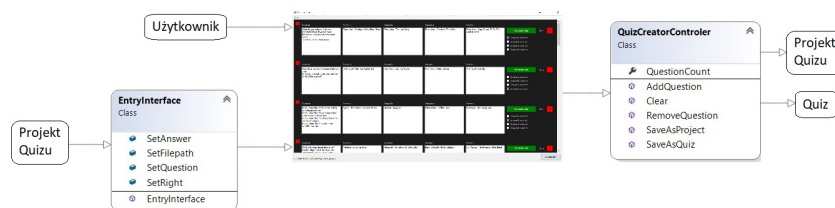
### 3 Implementacja - kreator quizów

Przy implementacji kreatora quizów wykorzystałem następujące klasy:

1. **QuizCreator** - Klasa reprezentująca okno kreatora. Dziedziczy po klasie **Form**, z biblioteki Windows Forms. Obsługuje interakcje z użytkownikiem oraz otwieranie i ładowanie plików projektu quizu. Zawiera zagnieżdżoną klasę **EntryInterface**, która reprezentuje operacje wpisywania danych do elementów interfejsu, co jest wykorzystywane przy ładowaniu projektu z pliku. Interfejs edycji pojedynczego pytania jest zamknięty w panelu.



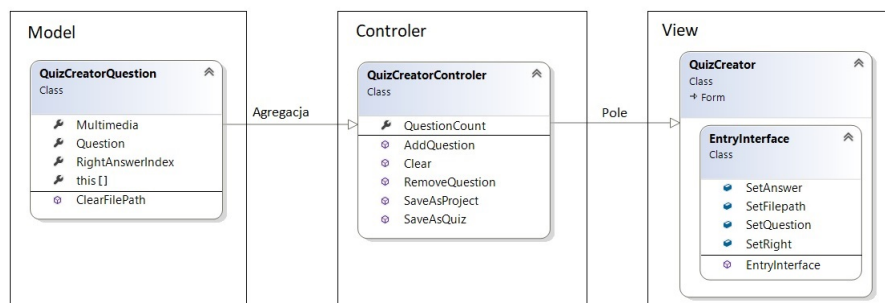
Program może zmieniać dane w zawartych w nim kontrolkach, za pomocą obiektu klasy **EntryInterface**, utworzonego dla konkretnego pytania. Obiekty klasy **EntryInterface** zawierają cztery pola reprezentujące operacje edycji danych w komponentach: **SetQuestion**, **SetAnswer**, **SetRight** oraz **SetFilepath**.



Schemat przepływu danych przez kreator.

2. **QuizCreatorController** - Klasa odpowiedzialna za logikę kreatora quizów. Pozwala na dodawanie oraz usuwanie pytań oraz obsługuje zapis jako projekt lub gotowy quiz. Implementuje właściwość **QuestionCount** oraz metody **AddQuestion()**, **RemoveQuestion()**, **Clear()** (usuwa wszystkie pytania), **SaveAsQuiz()** i **SaveAsProject()**.
3. **QuizCreatorQuestion** - Klasa reprezentująca pytanie w kreatorze. Zawiera treść pytania (właściwość **Question**), możliwe odpowiedzi (są one przechowywane w czteroelementowej tablicy, dostęp do nich jest zaimplementowany za pomocą indeksa klasy **QuizCreatorQuestion**), informacje o indeksie poprawnej odpowiedzi (właściwość **RightAnswerIndex**), ścieżkę do pliku multimedialnego (właściwość **FilePath**) oraz informację czy dane pytanie będzie wykorzystywać multimedia (właściwość **Multimedia**).

Kreator quizów został zaimplementowany z wykorzystaniem wzorca projektowego Model-View-Controller.



## 4 Implementacja - ustawienia

Okno ustawień pełni na tyle prostą funkcjonalność, że zostało zaimplementowane za pomocą jednej klasy **Settings**, która dziedziczy po klasie **Form**. Pobiera ona ustawienia od użytkownika i zapisuje je w pliku XML. Do stosowania wygenerowanych ustawień wyglądu programu, służy statyczna klasa **StyleApplier**, która posiada metody:

1. **Reload()** - ładuje ponownie ustawienia z pliku konfiguracyjnego.
2. **ApplyName()** - ustawia nazwę podanego w argumencie okna.
3. **ApplyColors()** - ustawia kolorystykę podanego w argumencie komponentu Windows Forms.

## 5 Implementacja - Menu główne

Okno menu głównego zostało zaimplementowane, za pomocą jednej prostej klasy (dziedziczącej po klasie `Form`), która jest odpowiedzialna za interakcje z użytkownikiem oraz uruchamianie innych części programu.

## 6 Implementacja - format plików wykorzystywanych przez program

Program zapisuje i otwiera dwa rodzaje plików:

1. `.quizproj` - format zapisu projektów quizów, pliki `quizproj` są zwykłymi plikami XML. Do ich odczytu i zapisu program wykorzystuje technologię LINQ to XML.
2. `.quiz` - format zapisu gotowych quizów, jest to archiwum zip, w którym znajdują się pliki multimediów oraz plik kontrolny(XML), zawierający treści pytań i ścieżki do odpowiednich multimediów w archiwum.

## 7 Hipotetyczne zastosowania stworzonego kodu (reusing)

Klasy `Question` oraz `ImageQuestion` mogłyby znaleźć zastosowanie w programie do typografii sprawdzianów oraz testów.

Klasa `SoundQuestion` mogłaby zostać użyta przy tworzeniu komputerowej adaptacji teleturnieju "Jaka to melodia?".

Statyczna klasa `StyleApplier`, może być stosowana do ustawiania kolorystyki dowolnych obiektów klasy `Control` z biblioteki Windows Forms.