



**Politechnika  
Śląska**

Dokumentacja projektu

2021/2022

**Programowanie obiektowe i graficzne**

*Football centre - projekt zaliczeniowy*

Kierunek: Informatyka

Członkowie zespołu:

Miłosz Flasz

Jakub Nocoń

Artur Łukaszek

Gliwice, 2021/2022

# Spis treści

<b>1</b>	<b>Część I</b>	<b>2</b>
1.1	Temat projektu . . . . .	2
1.2	Baza danych . . . . .	2
1.3	Założenia . . . . .	2
<b>2</b>	<b>Część II</b>	<b>3</b>
2.1	Działanie programu - instrukcja obsługi . . . . .	3
2.2	Implementacja . . . . .	4
2.3	Diagram UML . . . . .	6
2.4	Testy . . . . .	8
<b>3</b>	<b>Część III</b>	<b>11</b>
3.1	Podsumowanie . . . . .	11
3.2	Potencjał rozwoju . . . . .	11

# 1 Część I

## 1.1 Temat projektu

Tematem naszego projektu jest aplikacja okienkowa napisana w języku C# z wykorzystaniem architektury MVVM (Model-View-Viewmodel) połączona z utworzoną przez nas bazą danych skupioną wokół lig i klubów piłkarskich z całego świata. Głównym zadaniem aplikacji jest danie możliwości przeglądania danych zawartych w bazie w łatwy, przejrzysty i usystematyzowany sposób, dzięki przystępnemu interfejsowi. Daje ona też możliwość rozwiązania quizu opartego na wyżej wymienionej bazie.

## 1.2 Baza danych

*Piłkarska baza danych* jest tworem mający na celu odwzorowanie realiów piłkarskich z całego świata. Baza podzielona jest na 5 encji:

- Federacje -> nazwa federacji (w formie skrótowca), kontynent, prezes
- Ligi -> nazwa ligi, kraj, poziom rozgrywkowy, pozycja w rankingu kontynentalnym
- Kluby -> nazwa klubu, miasto, rok założenia, przydomek, wartość kadry, ilość trofeów, liga, stadion
- Stadiony -> nazwa stadionu, adres, data otwarcia, pojemność
- Pracownicy -> id pracownika, imię, nazwisko, funkcja, klub

## 1.3 Założenia

Do głównych założeń naszego projektu, należy:

- Utworzenie przyjaznej dla oka oprawy graficznej,
- zaprojektowanie spójnego i intuicyjnego mechanizmu obsługi programu,
- przedstawienie danych w sposób czytelny i przystępny,
- zaprojektowanie aplikacji z myślą o dalszym rozwoju,
- efektywne wykorzystanie wzorca MVVM.

## 2 Część II

### 2.1 Działanie programu - instrukcja obsługi

Aplikacja składa się z następujących części:

1. **MENU** - zawiera krótki opis aplikacji;
2. **KLUBY** - wybieramy klub, dla którego program ma wyświetlić informacje, takie jak: miasto pochodzenia, rok utworzenia klubu, przydomek, wartość drużyny, liczba trofeów, listę pracowników lub informacje na temat stadionu: nazwa, adres, pojemność, data budowy. Upřednio przed wyborem klubu, wybieramy federację, z której później wybieramy interesującą nas ligę;
3. **GRY** - zaimplementowana rozgrywkę polega na wyborze 1 z 2 opcji (quiz typu A lub B). Spośród wylosowanych par klubów lub stadionów wybieramy, który z nich ma: w przypadku klubów - więcej trofeów albo droższą kadrę, a w przypadku stadionów - więcej miejsc;
4. **RANKINGI** - dla klubów, lig oraz stadionów może wywołać listę, posortowaną według różnych kryteriów
  - dla klubów: ilość trofeów, wartość drużyny, rok założenia(od najstarszego),
  - dla lig: pozycja w federacyjnych rankingach,
  - dla stadionów: pojemność oraz rok budowy(od najstarszego).

Listy można przefiltrować - dla klubów można ograniczyć się do tych w samej lidze, a dla lig ograniczyć się do tych, znajdujących się w tej samej federacji;

po których można się poruszać używając paska przycisków po lewej stronie okna.

W każdej sekcji zastosowano rozwijane listy dla wyboru odpowiadających użytkownikowi wariantów. Na pasku opcji znajduje się również przycisk **WYJDŹ** kończący działanie aplikacji.

## 2.2 Implementacja

Program został zaimplementowany w kilku folderach, można wyróżnić:

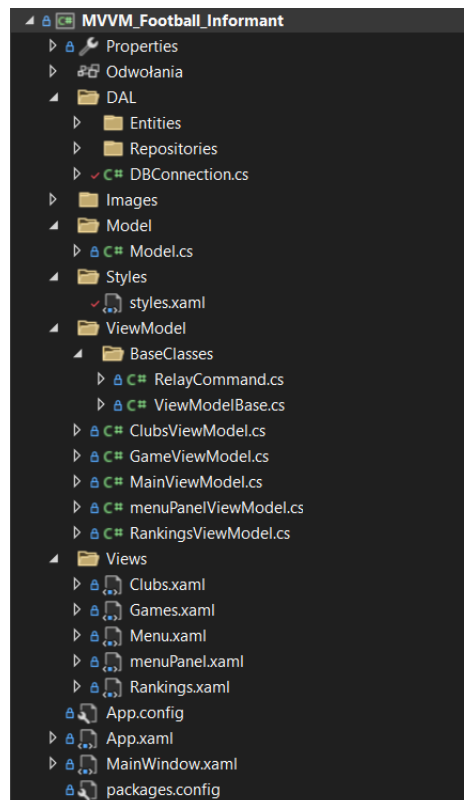
DAL - na który składają się foldery Entities(konstrukcje jednostek) i Repositories(odczytywanie jednostek z bazy danych),

Model - gdzie przetrzymywany jest odzwierciedlenie bazy danych,

Styles - tutaj przetrzymywane są zasoby - style do przycisków, comboboxów i textblocków,

ViewModel - dla każdego View stworzono ViewModel, który wiąże dane, można wyróżnić menuPanelViewModel.cs, który służy do sterowania widocznością poszczególnych widoków z poziomu View menuPanel, gdzie stworzono instancje wszystkich potrzebnych widoków,

Views - folder ze stworzonymi widokami, składającymi się na całość aplikacji - Clubs, Games, Menu, Rankings.



```

private const string SELECT_TO_RANKING_1 = "SELECT * FROM ";
private const string SELECT_TO_RANKING_2 = " ORDER BY ";
private const string SELECT_TO_RANKING_3 = " WHERE ";
#endregion

#region Methods
public static List<dynamic> DownloadToRanking(string targetType, string orderType, string whereType)
{
    var list = new List<dynamic>();

    using (var connection = DBConnection.Instance.Connection)
    {
        var where = "";
        var SELECT_COMMAND = "";

        if (whereType != null)
        {
            if (targetType.Equals("leagues"))
            {
                where = "federationName = ";
            }
            else if (targetType.Equals("clubs"))
            {
                where = "leagueName = ";
            }

            SELECT_COMMAND = SELECT_TO_RANKING_1 + targetType + SELECT_TO_RANKING_3 + where + "" +
                whereType + "" + " AND " + orderType + " IS NOT NULL " + SELECT_TO_RANKING_2 + orderType;
        }
        else
        {
            SELECT_COMMAND = SELECT_TO_RANKING_1 + targetType + " WHERE " + orderType + " IS NOT NULL " + SELECT_TO_RANKING_2 + orderType;
        }

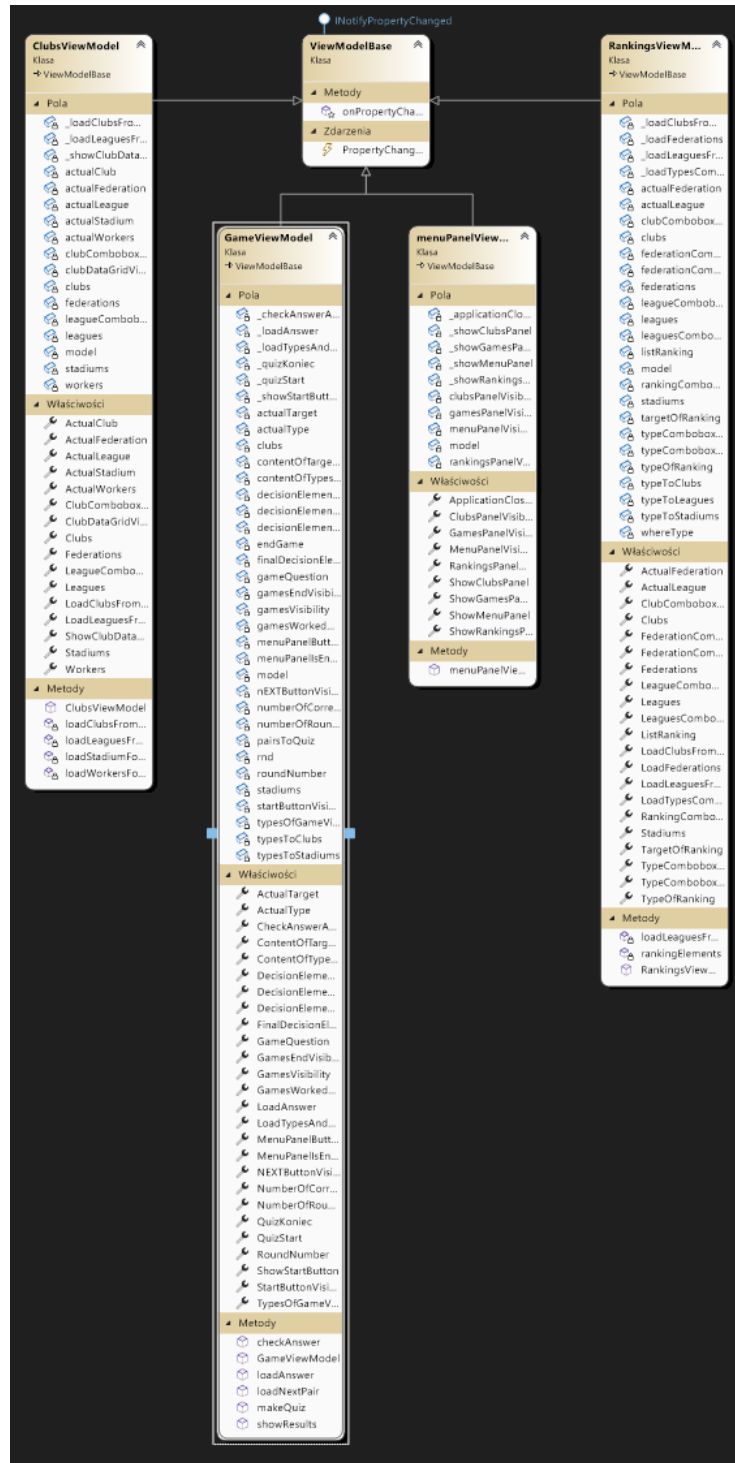
        if (targetType.Equals("stadiums"))
        {
            SELECT_COMMAND = SELECT_TO_RANKING_1 + targetType + SELECT_TO_RANKING_2 + orderType;
        }

        if (orderType.Equals("teamValue") || orderType.Equals("trophiesNumber") || orderType.Equals("capacity"))
        {
            SELECT_COMMAND += " DESC"; // ponieważ zależy na największej wartości, największej ilości trofeów i miejsc na stadionie :)
        }
    }
}

```

Jako ciekawą część kodu aplikacji, można przywołać powyższy kod odpowiedzialny za tworzenie zapytań do bazy danych, dzięki któremu baza zwraca odpowiednio posortowane listy, wykorzystywane w zakładce „RANKINGI”. Kod tworzy odpowiednie zapytanie w zależności, od tego jakie elementy naszej bazy będą wykorzystywane (kluby, ligi, stadiony) - 'targetType', według czego będzie przeprowadzane sortowanie - 'orderType' oraz czy elementy naszej bazy będą przefiltrowane w poszukiwaniu konkretnych elementów - 'whereType'.

## 2.3 Diagram UML







## 2.4 Testy

Poniższy zrzut ekranu prezentuje zakładkę - "KLUBY", gdzie został przedstawiony przykładowy wybór - federacji, ligi oraz klubu. Program sprawnie i prawidłowo wyświetlił informacje o drużynie - Borussia Dortmund oraz o stadionie, na którym na co dzień występują żółto-czarni.

Football Centre


**MENU**

Federacja: UEFA [Europa] ▼ Prezes: Aleksander Ceferin

**KLUBY**

Liga: 1. Bundesliga [Niemcy] ▼ Klub: Borussia Dortmund ▼

**GRY**

 Dortmund 1909  
Die Schwarzgelben  
Wartość: 560mln € Trofea: 19

**RANKINGI**

Nazwa stadionu i adres:  
Signal Iduna Park  
Strobelallee 50

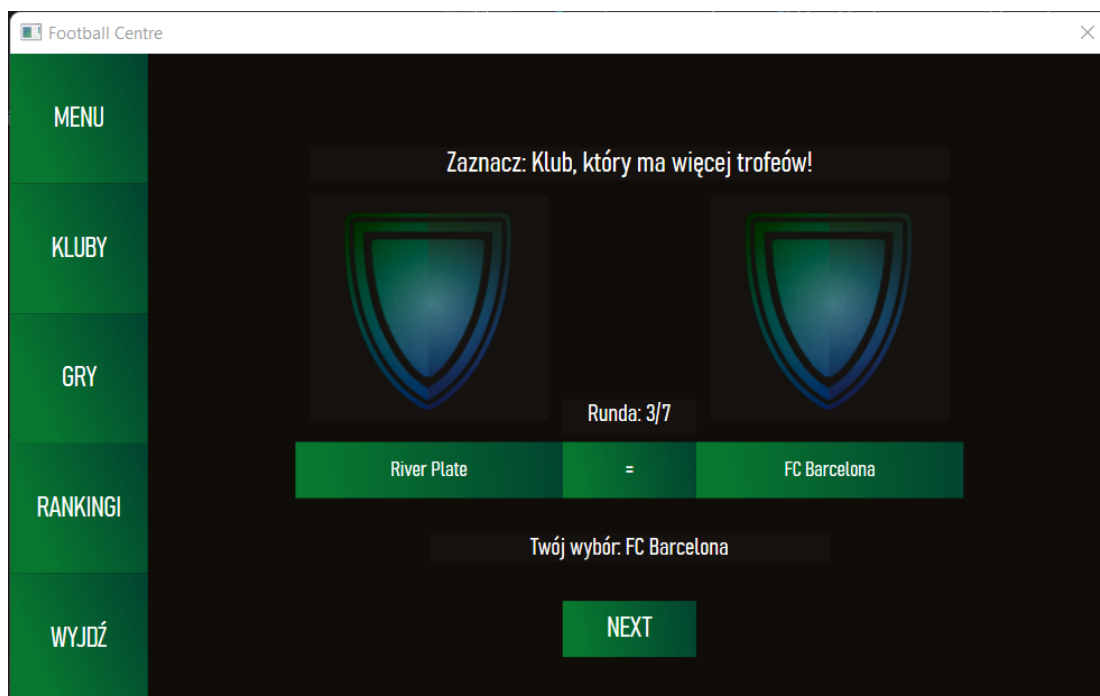
**WYJÓZ**

Liczba miejsc: 81365  
Rok otwarcia: 1974

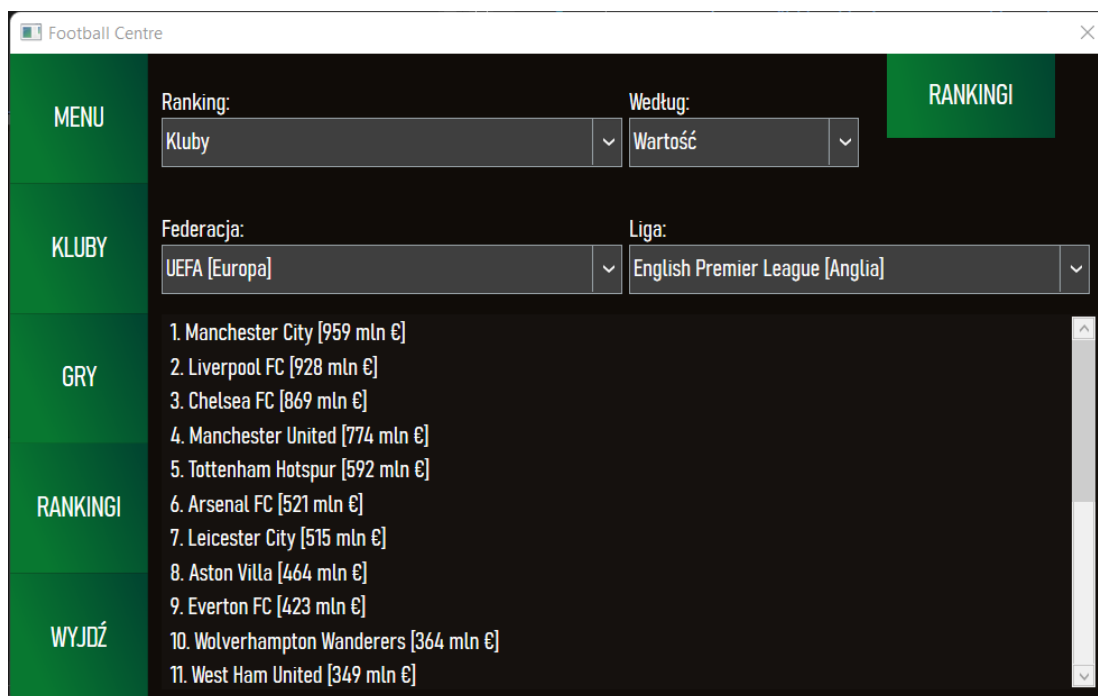
**Pracownicy:**

- Marco Reus - kapitan
- Lars Ricken - koordynator młodzieżówki
- Sebastian Kehl - dyrektor sportowy
- Edin Terzić - trener
- Hans-Joachim Watzke - prezes
- Matthias Sammer - doradca zarządu

Poniższy zrzut ekranu przedstawia zaimplementowaną rozgrywkę, gdzie użytkownik wybrał opcję gry w odgadywanie - który z wylosowanych klubów w parze ma więcej trofeów.



Następnie, obrazek ukazuje tryb Rankingów, który bez zarzutów wyświetlił ranking najdroższych klubów(od najbardziej wartościowego) w lidze angielskiej Premier League.



## 3 Część III

### 3.1 Podsumowanie

Udało nam się utworzyć aplikację realizującą ideę struktury MVVM i w pełni wykorzystującą zawartość utworzonej przez nas bazy danych. W ramach czasowych naszego projektu nie było takiego elementu, który zająłby znaczącą część czasu realizacji całości i odbiegał od reszty. Szczególną uwagę musieliśmy zwrócić na tworzenie projektu zgodnie z ideą MVVM.

### 3.2 Potencjał rozwoju

Spekulując, co można by było zrobić w celu rozwinięcia aplikacji to np. dodanie grafik dających większą estetykę aplikacji jak herby drużyn, loga federacji i lig, czy zdjęcia stadionów lub pracowników. Ciekawym pomysłem odnośnie aplikacji byłoby również rozwinięcie poszczególnych sekcji oraz stworzenie funkcjonalności, która aktualizowałaby informacje w bazie danych, aby nie były nieaktualne.

Do KLUBÓW można by dodać opcję przekierowania do odpowiednich stron internetowych dotyczących poszczególnych federacji, lig, klubów, pracowników i stadionów takich jak artykuły na Wikipedii, profile na Transfermarkt albo FlashScore, oficjalne strony internetowe wybranych podmiotów.

Największe pole do popisu leży w GRACH. Oprócz dodania innych form quizu pojawił się pomysł stworzenia symulatora rozzgrywek, którego działanie opierać by się mogło na zaangażowaniu sztucznej inteligencji. Przeprowadzała by ona symulacje meczów na podstawie danych z bazy i tych pobranych z internetu (aktualna forma drużyn i kadra).