



# **Aplikacja dla kibiców piłkarskich**

## **Dokumentacja**

**Szymon Adach, Jacek Dziwulski**

**Wersja 1.3**

**16.01.2017**

## Spis treści

1.Specyfikacja.....	3
1.1 Opis.....	3
1.2 Wymagania funkcjonalne.....	3
1.3 Wymagania нефункционалне.....	7
1.4 Harmonogram projektu.....	8
1.5 Architektura rozwiązania.....	8
2.Dokumentacja końcowa (powykonawcza).....	9
2.1 Wymagania systemowe.....	9
2.2 Biblioteki wraz z określeniem licencji.....	9
2.3 Instrukcja instalacji.....	10
2.3.1 Instalacja na maszynie klienckiej.....	10
2.3.2 Instalacja na serwerze aplikacyjnym.....	10
2.4 Instrukcja uruchomienia.....	10
2.5 Instrukcja użytkowania.....	11
2.6 Instrukcja utrzymania.....	20
2.7 Raport odstępstw od specyfikacji wymagań.....	20
2.8 Dokumentacja usług Web Services.....	21

Data	Autor	Opis zmian	Wersja
17.12.2016	Jacek Dziwulski	Pierwsza wersja dokumentu.	1.0
17.12.2016	Szymon Adach	Opis i wymagania funkcjonalne.	1.1
07.01.2017	Szymon Adach	Dokumentacja końcowa.	1.2
16.01.2017	Jacek Dziwulski	Dokumentacja Web Services	1.3

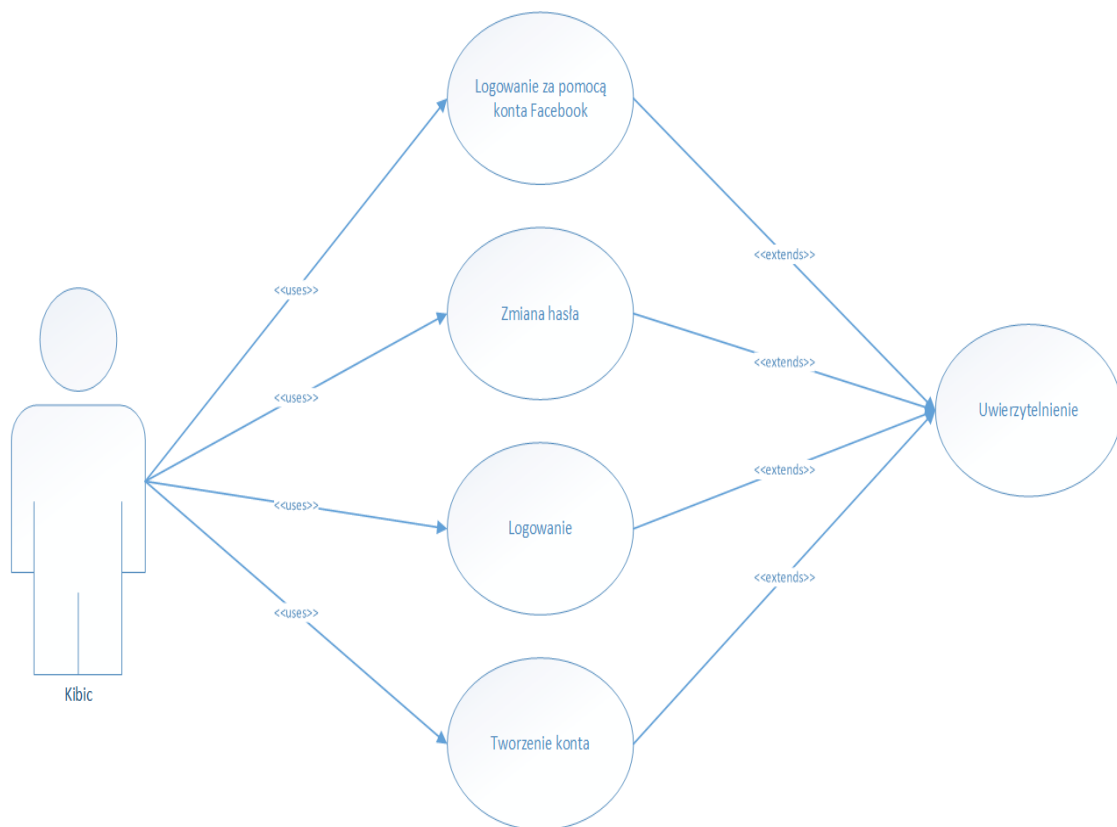
# 1. Specyfikacja

## 1.1 Opis

Niniejsza aplikacja przeznaczona jest dla kibiców piłki nożnej. Za pomocą strony internetowej mogą oni przeglądać dane ligi piłkarskiej: statystyki drużyn, zawodników, historie spotkań oraz tabelę ligi i ranking strzelców. Po zarejestrowaniu użytkownik otrzymuje dostęp do nowych opcji, powiadomień o meczach oraz oceniania ich, a także dodawania opinii o spotkaniach. Dodatkowo strona zapewniać będzie wskazówki dojazdu na mecz dla kibiców.

## 1.2 Wymagania funkcjonalne

Na poniższym rysunku przedstawiono w postaci diagramów UML zbiór przypadków użycia platformy dla aktora (kibica).





**Rysunek 1. i 2. Diagram przypadków użycia**

User stories:

Interfejs użytkownika:

1. Tworzenie konta  
Użytkownik tworzy konto w systemie poprzez podanie wybranego loginu oraz hasła (dwukrotnie).
2. Logowanie do aplikacji za pomocą konta Facebook.  
Użytkownik loguje się do aplikacji przy pomocy konta w serwisie Facebook.
3. Logowanie do aplikacji.  
Użytkownik loguje się do aplikacji przy pomocy konta uprzednio zarejestrowanego w systemie.
4. Zmiana hasła.  
Użytkownik po uwierzytelnieniu może zmienić swoje hasło systemowe, podając stare hasło oraz dwukrotnie nowe hasło.
5. Wyświetlenie tabeli ligowej dla sezonu  
Dowolny użytkownik może wyświetlić listę drużyn posortowanych według punktów (istnieje możliwość zmiany sortowania poprzez kliknięcie w wybraną kolumnę)
6. Wyświetlenie rankingu strzelców  
Dowolny użytkownik może wyświetlić określoną liczbę zawodników, posortowanych malejąco pod względem liczby strzelonych w sezonie bramek.
7. Wyświetlenie danych drużyny.  
Dowolny użytkownik może wyświetlić dane drużyny: listę zawodników, listę meczów oraz historyczne dane takie jak miejsca w poprzednich sezonach.
8. Wyświetlenie danych zawodnika.  
Dowolny użytkownik może wyświetlić dane zawodnika: jego drużyny, mecze, w których brał udział, a także strzelone bramki.
9. Wyświetlenie danych trenera.  
Dowolny użytkownik może wyświetlić dane trenera: prowadzone przez niego drużyny oraz ich sukcesy.
10. Wyświetlenie danych sędziego.  
Dowolny użytkownik może wyświetlić dane sędziego: mecze, które prowadził oraz statystyki kar (średnia liczba żółtych i czerwonych kartek na mecz).
11. Dodanie opinii o meczu.  
Zalogowany użytkownik może dodać opinię o meczu w postaci tekstowej oraz oceny liczbowej w skali od 1 do 5.
12. Zapisanie na powiadomienia o meczu.  
Zalogowany przez konto Facebook użytkownik może wybrać z jakim wyprzedzeniem powinien otrzymać powiadomienie o zbliżającym się spotkaniu.
13. Wyświetlenie wskazówek dojazdu na mecz.  
Zalogowany użytkownik może po wpisaniu adresu otrzymać wskazówki dojazdu na stadion w postaci tekstowej, a także trasy na mapie.

**Tabela 1 Opisy przypadków użycia dla aktora – kibica**

ID	Aktor	Nazwa	Opis	Odpowiedź systemu
USR	Użytkownik	Tworzenie konta.	Utworzenie konta użytkownika w systemie.	Informacja logiczna o powodzeniu operacji i ewentualny powód błędu.
		Logowanie do aplikacji za pomocą konta Facebook.	Logowanie do aplikacji przy pomocy istniejącego konta z serwisu Facebook.	Informacja logiczna o powodzeniu operacji i ewentualny powód błędu.
LOG	Zalogowany użytkownik	Logowanie do aplikacji.	Logowanie do aplikacji przy pomocy konta utworzonego w systemie.	Informacja logiczna o powodzeniu operacji i ewentualny powód błędu.
		Zmiana hasła.	Zmiana hasła dla konta utworzonego w systemie.	Informacja logiczna o powodzeniu operacji i ewentualny powód błędu.
		Wyświetlenie tabeli ligowej dla sezonu.	Wysłanie zapytania o listę drużyn dla danego sezonu.	Przesłanie danych drużyn z wybranego sezonu.
		Wyświetlenie rankingu strzelców.	Wysłanie zapytania o listę najlepszych strzelców sezonu.	Przesłanie danych określonej liczby zawodników, którzy strzelili najwięcej goli.
		Wyświetlenie danych drużyny.	Wysłanie zapytania o dane drużyny.	Przesłanie danych wybranej drużyny z aktualnego i zakończonych sezonów.
		Wyświetlenie danych zawodnika.	Wysłanie zapytania o dane zawodnika.	Przesłanie danych zawodnika z aktualnego i zakończonych sezonów.
		Wyświetlenie danych trenera.	Wysłanie zapytania o dane trenera.	Przesłanie danych trenera z aktualnego i zakończonych sezonów.
		Wyświetlenie danych sędziego	Wysłanie zapytania o dane sędziego.	Przesłanie danych sędziego z aktualnego i zakończonych sezonów.
		Dodanie opinii o meczu.	Dodanie komentarza tekstowego i oceny punktowej meczu.	Informacja logiczna o powodzeniu operacji i ewentualny powód błędu.
		Zapisanie na powiadomienia o meczu.	Wysłanie zapytania o powiadomienia Push o zbliżającym się meczu z odpowiednim wyprzedzeniem.	Informacja logiczna o powodzeniu operacji i ewentualny powód błędu. Wysłanie powiadomienia w przypadku spełnienia zadanych warunków czasowych.
		Wyświetlenie wskazówek dojazdu na mecz.	Wysłanie zapytania o szczegóły dojazdu z wybranego adresu na adres, gdzie rozgrywany jest mecz.	Tekstowe wskazówki dojazdu na mecz oraz mapa z trasą dojazdu,

Sumaryczna lista ról użytkowników występujących w historiach użytkownika:

<b>Użytkownik</b>	Dowolny użytkownik systemu.
<b>Zalogowany użytkownik</b>	Zalogowany w systemie użytkownik (przez konto systemowe bądź konto Facebook).

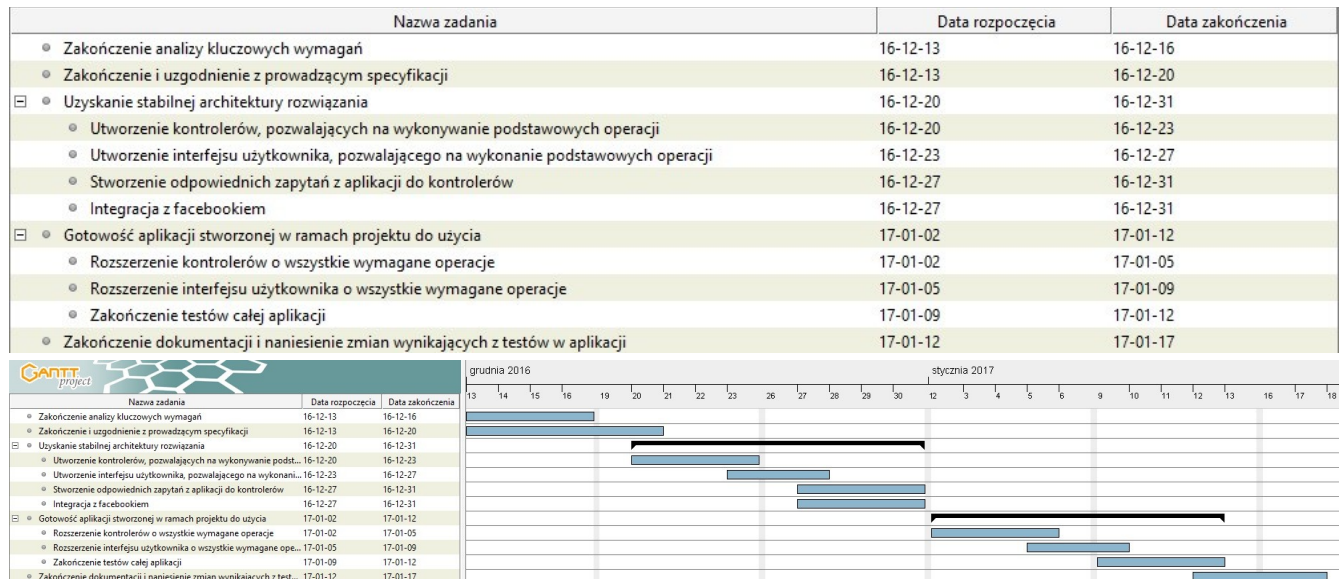
## 1.3 Wymagania niefunkcjonalne

**Tabela 2 Lista wymagań niefunkcjonalnych**

Obszar wymagań	Nr wymagania	Opis
Użyteczność	1	Wszystkie funkcjonalności aplikacji dostępne dla użytkownika muszą mieścić się na pojedynczym ekranie przy rozdzielczości 1920x1080 i czcionce nie mniejszej niż 12pt
	2	Aplikacja musi być stworzona z wykorzystaniem technologii ASP.NET.
	3	Aplikacja musi pobierać i zapisywać dane w bazie danych za pomocą ASP.NET Web Api.
	4	Interfejs użytkownika musi być w języku angielskim.
Niezawodność	5	Baza danych musi być utrzymywana w stanie gotowym do użytku, nawet w przypadku awarii aplikacji.
Wydajność	6	Aplikacja powinna odpowiadać na zapytania użytkownika w czasie nie dłuższym niż 5 sekund.
Utrzymanie	7	Aplikacja powinna zachować wsteczną zgodność z interfejsem dostępu do poszczególnych obiektów
	8	Aplikacja powinna działać na systemach Windows 7 i nowszych

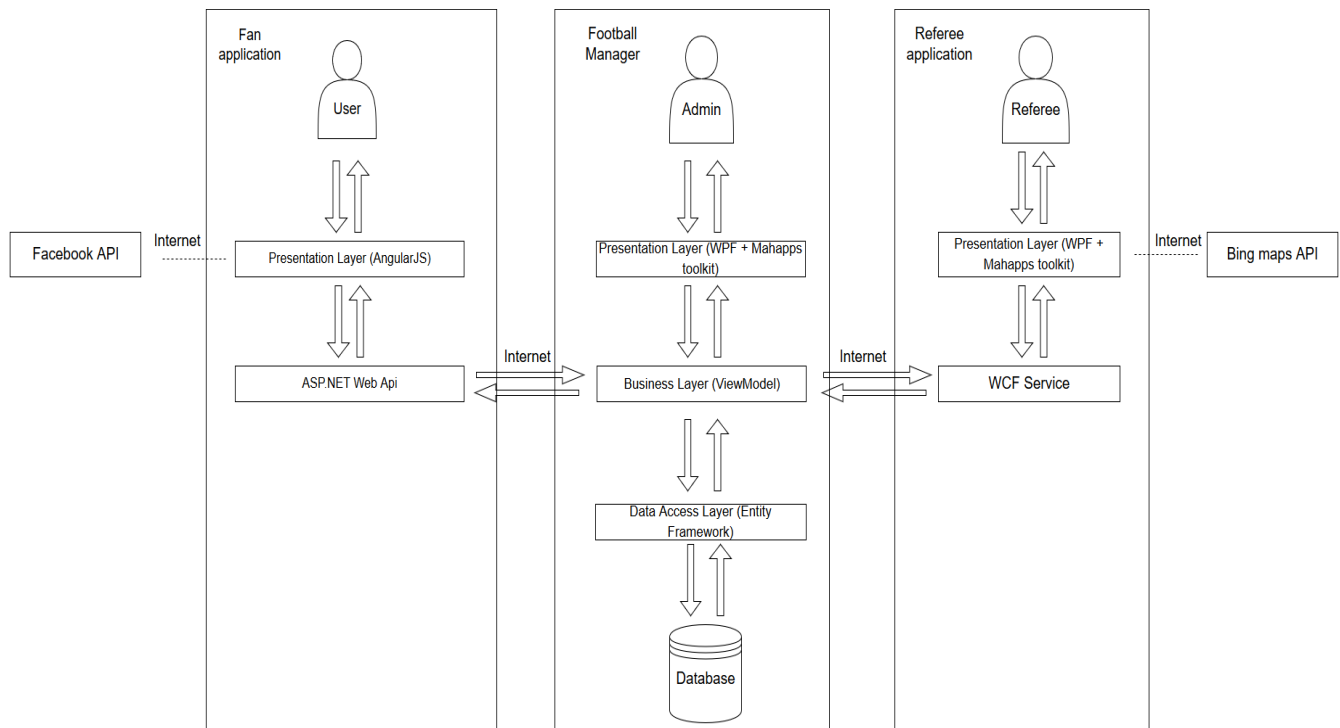
## 1.4 Harmonogram projektu

Planowana jest realizacja projektu zgodnie z następującym harmonogramem:



Rysunek 2. i 3. Harmonogram projektu

## 1.5 Architektura rozwiązania



Rysunek 5. Schemat architektury rozwiązania



## 2. Dokumentacja końcowa (powykonawcza)

### 2.1 Wymagania systemowe

Na maszynie klienckiej nie jest wymagana instalacja żadnego komponentu systemu. Maszyna kliencka powinna mieć dostęp do Internetu oraz zainstalowaną przeglądarkę, preferowane są Google Chrome (wersja 55 lub wyższa) i Mozilla Firefox (wersja 50 lub wyższa).

Na serwerze aplikacyjnym:

- Internet Information Services 7
- Dostęp do internetu
- Zainstalowany .NET framework w wersji 4.5.2

### 2.2 Biblioteki wraz z określeniem licencji

Nr.	Komponent i wersja	Opis	Licencja
1.	FluentAssertions 4.16.0	Zestaw metod rozszerzających pozwalający w naturalny sposób określać oczekiwany rezultat w testach jednostkowych.	<a href="#">Apache License</a>
2.	Moq 4.5.23	Biblioteka do testów typu mock.	<a href="#">The BSD 2-Clause License</a>
3.	Castle.Core 3.3.3	Biblioteka wymagana do działania komponentu Moq.	<a href="#">Apache License</a>
4.	CommonServiceLocator 1.3.0	Biblioteka wspomagająca implementację wzorca Inversion of Control.	<a href="#">Microsoft Public License</a>
5.	Antlr 3.5.0.2	Biblioteka zawierająca zestaw narzędzi do rozpoznawania języków.	<a href="#">BSD License</a>
6.	Autofac 4.2.1	Konterer dla wzorca Inversion of Control.	<a href="#">MIT License</a>
7.	Autofac.WebApi2 4.0.1	Integracja Autofac z ASP.NET WebAPI 2.2	<a href="#">MIT License</a>
8.	AutoMapper 5.2.0	Mapper typu obiekt-obiekt.	<a href="#">License</a>
9.	FluentValidation 6.2.1	Tworzenie reguł walidacji z silnym typowaniem.	<a href="#">License</a>
10.	Jquery 1.8.0	Biblioteka JavaScript pozwalająca	<a href="#">License</a>

		na prostą integrację z API Facebooka, łatwe zarządzanie eventami.	
11.	Newtonsoft.Json 9.0.1	Wydajna biblioteka obsługująca format JSON.	<a href="#">MIT License</a>
12.	AngularJS 1.6.1	Framework ułatwiający tworzenie webaplikacji przy użyciu wzorca MVC.	<a href="#">MIT License</a>
13.	Toastr 2.1.3	Biblioteka z nieblokującymi powiadomieniami.	<a href="#">MIT License</a>
14.	Bootstrap 3.3.5	Framework ułatwiający tworzenie strony.	<a href="#">MIT License</a>

## 2.3 Instrukcja instalacji

### 2.3.1 Instalacja na maszynie klienckiej

Instalacja nie jest wymagana. Klient musi dysponować jedynie nowoczesną przeglądarką internetową.

### 2.3.2 Instalacja na serwerze aplikacyjnym

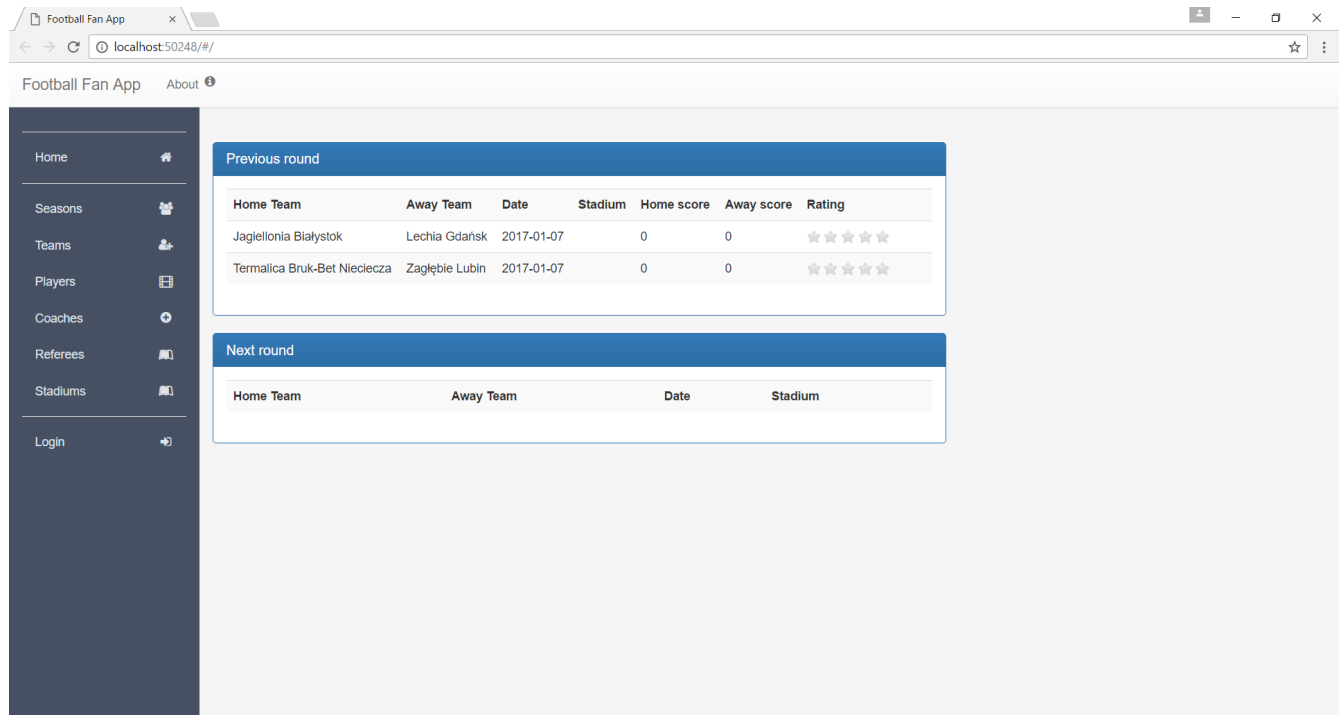
Odbiorca systemu podaje administratorowi technicznemu adres IP, nazwę użytkownika oraz hasło do serwera aplikacyjnego, na którym zainstalowane będą usługi WebApi. Administrator techniczny wykonuje akcję publikowania WebApi na maszynę-serwer. Należy poprawnie skonfigurować uprawnienia dla grupy IIS\_USRS. Instrukcje jak tego dokonać oraz jak zainstalować wymagane oprogramowanie znajdują się w dokumentacji technicznej producentów.

## 2.4 Instrukcja uruchomienia

W pasek adresu przeglądarki należy wpisać adres przekazany przez administratora systemu.

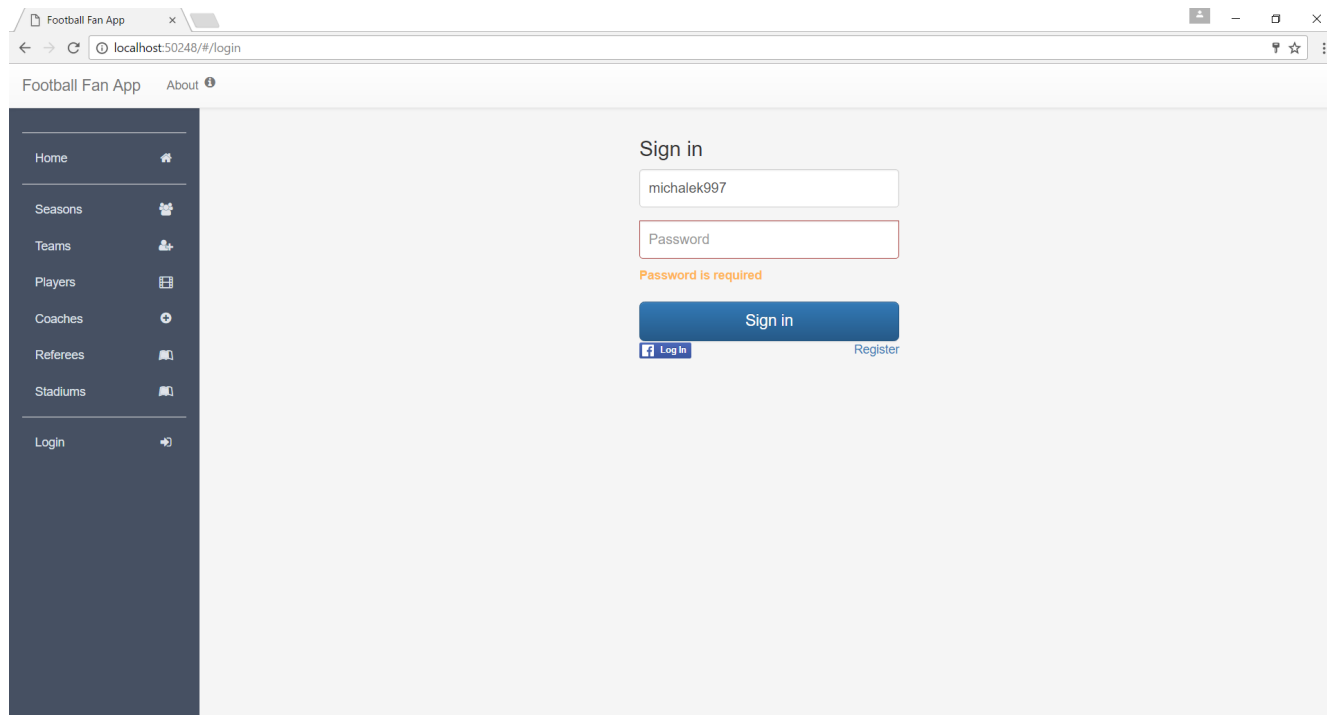
## 2.5 Instrukcja użytkowania

Po przejściu na stronę główną użytkownikowi prezentowany jest następujący widok:

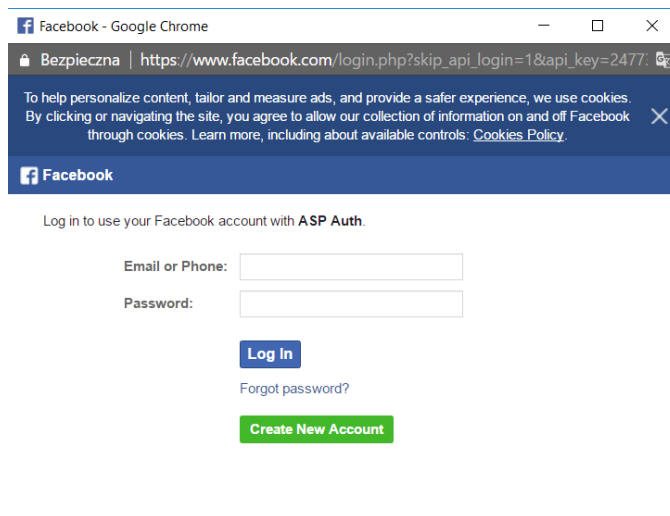


Belki górna oraz dolna prezentowane są na każdej podstronie aplikacji. Zadaniem lewej belki jest ułatwienie nawigowania w obrębie aplikacji, na górnej umieszczane są powiadomienia dla użytkownika.

Po wybraniu opcji Login prezentowany jest widok:

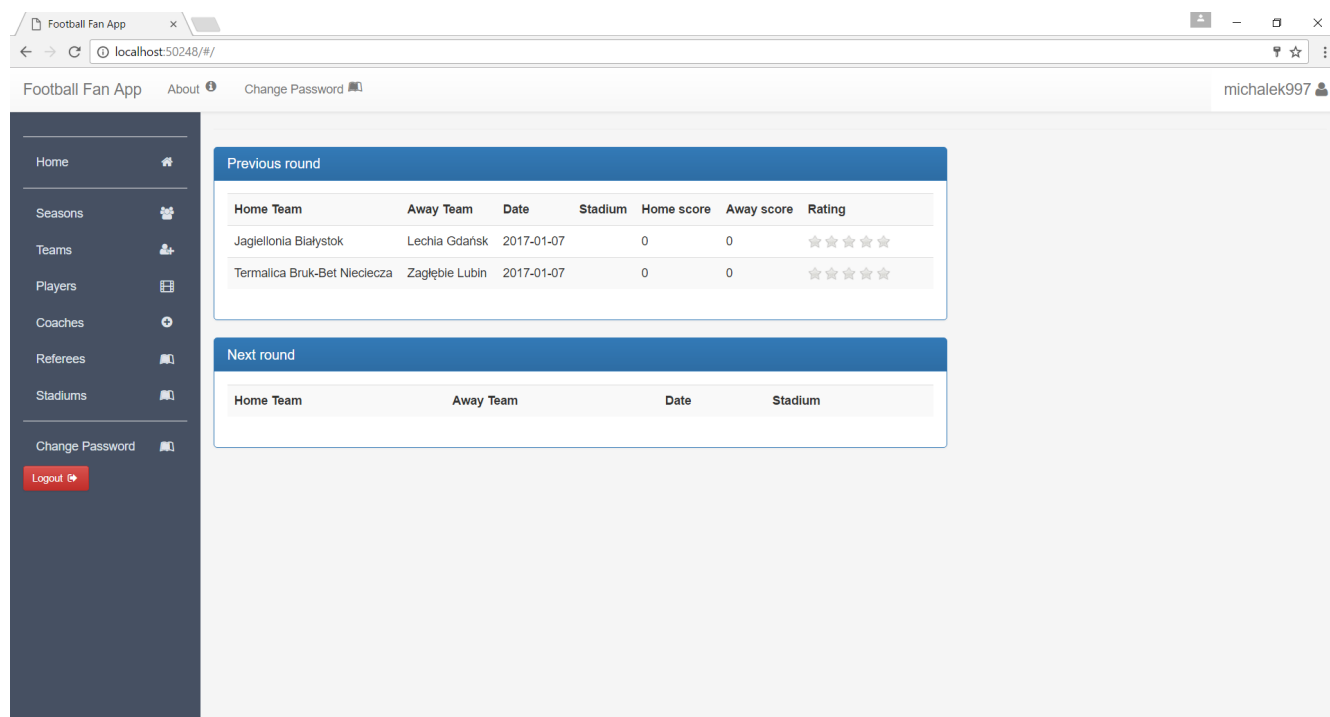


Należy tu wprowadzić poprawne dane użytkownika, które podano podczas rejestracji w systemie. Możliwa jest też opcja zalogowania poprzez konto w serwisie Facebook. Jeżeli nie ma aktywnej sesji w tym serwisie, to po przyciśnięciu przycisku Log In wyświetla się okno:

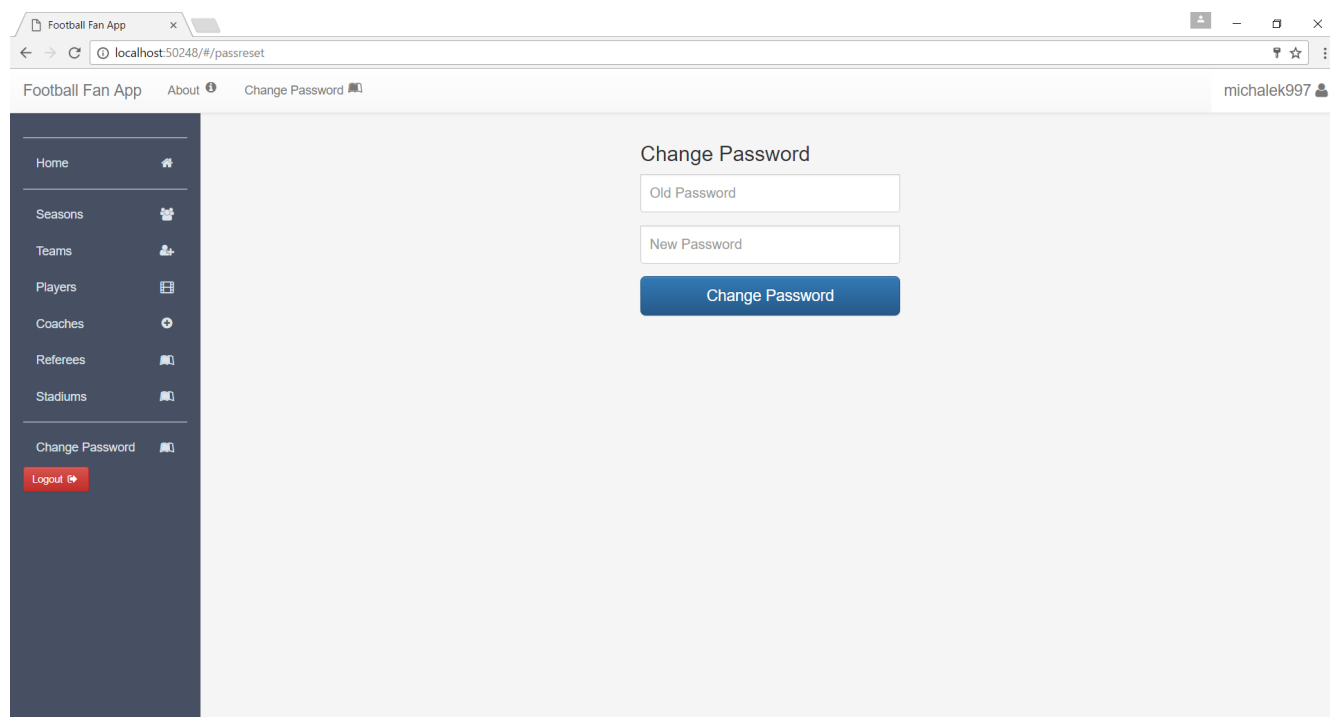


Do widoku rejestracji przechodzimy wciskając w widoku logowania przycisk register, a następnie podając nazwę użytkownika oraz niepuste hasło. Aplikacja powiadomi użytkownika, gdy wprowadzone dane są niepoprawne (nazwa użytkownika nie jest unikalna).

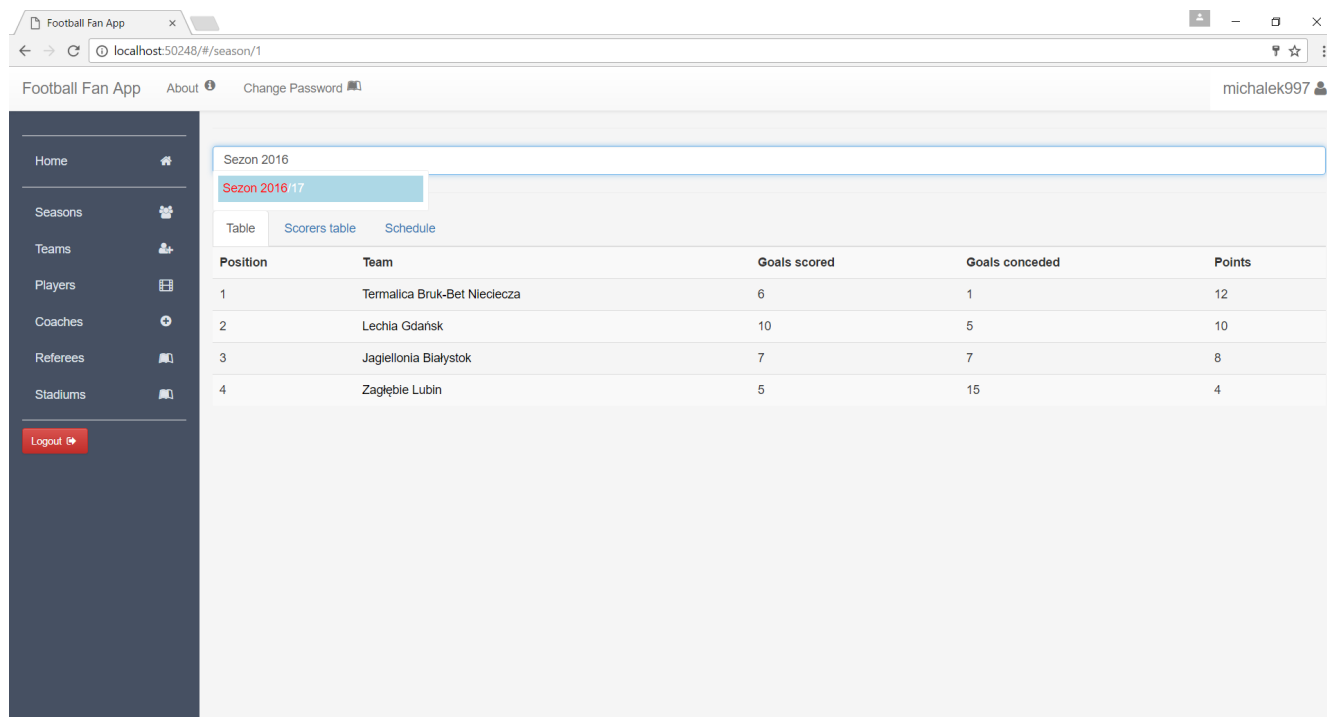
Po poprawnym zalogowaniu w górnej belce pojawia się nazwa użytkownika, a w menu po lewej stronie udostępniana jest opcja wylogowania.



Ponadto użytkownicy zarejestrowani przez konto w systemie aplikacji (tj. nielogujący się przez konto Facebook) mogą zmienić swoje hasło w odpowiednim widoku, do którego dostęp daje przycisk Change Password w górnym menu:



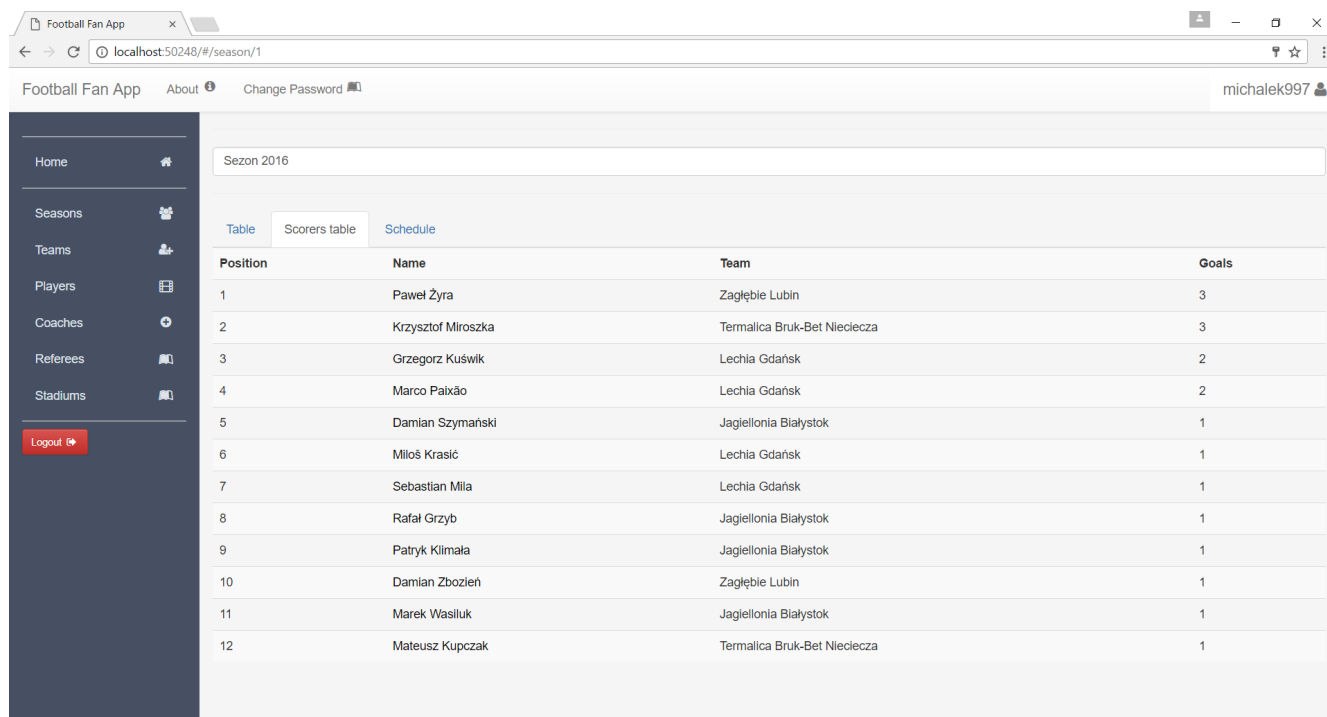
W widoku sezonu użytkownikowi udostępniana jest wyszukiwarka sezonów oraz trzy zakładki. Pierwsza z nich prezentuje tabelę drużyn posortowaną malejąco według liczby zdobytych punktów:



The screenshot shows the 'Table' tab for the 2016/17 season. The table lists teams and their performance metrics:

Position	Team	Goals scored	Goals conceded	Points
1	Termalica Bruk-Bet Nieciecza	6	1	12
2	Lechia Gdańsk	10	5	10
3	Jagiellonia Białystok	7	7	8
4	Zagłębie Lubin	5	15	4

Zakładka Scorers prezentuje listę najlepszych strzelców w wybranym sezonie:



The screenshot shows the 'Scorers table' for the 2016 season. The table lists players, their teams, and the number of goals scored:

Position	Name	Team	Goals
1	Paweł Zyra	Zagłębie Lubin	3
2	Krzysztof Miroszka	Termalica Bruk-Bet Nieciecza	3
3	Grzegorz Kuświk	Lechia Gdańsk	2
4	Marco Paixão	Lechia Gdańsk	2
5	Damian Szymański	Jagiellonia Białystok	1
6	Mikoł Krasieć	Lechia Gdańsk	1
7	Sebastian Mili	Lechia Gdańsk	1
8	Rafał Grzyb	Jagiellonia Białystok	1
9	Patryk Klimala	Jagiellonia Białystok	1
10	Damian Zbozień	Zagłębie Lubin	1
11	Marek Wasiliuk	Jagiellonia Białystok	1
12	Mateusz Kupczak	Termalica Bruk-Bet Nieciecza	1

Zakładka Schedule to lista meczów w kolejnych rundach, podobna do tej ze strony startowej.

Podobnie jak poprzednio, widok Teams udostępnia wyszukiwarkę drużyn w lidze oraz zakładki:

The screenshot shows the 'Football Fan App' interface. The left sidebar contains navigation links: Home, Seasons, Teams, Players, Coaches, Referees, Stadiums, and a Logout button. The main content area displays the details for 'KS LECHIA' (Lechia Gdańsk). At the top is a search bar labeled 'Search teams'. Below it are tabs for 'Details', 'Players', 'Matches', and 'Seasons'. The 'Details' tab is active, showing the team's flag and the following information:

- Name:** Lechia Gdańsk
- Founded:** 1945-01-01
- City:** Gdańsk
- Zipcode:** 80-560
- Street:** Pokoleń Lechii Gdańsk
- Number:** 1
- Coach:** Piotr Nowak
- Stadium:** Stadion Energa Gdańsk

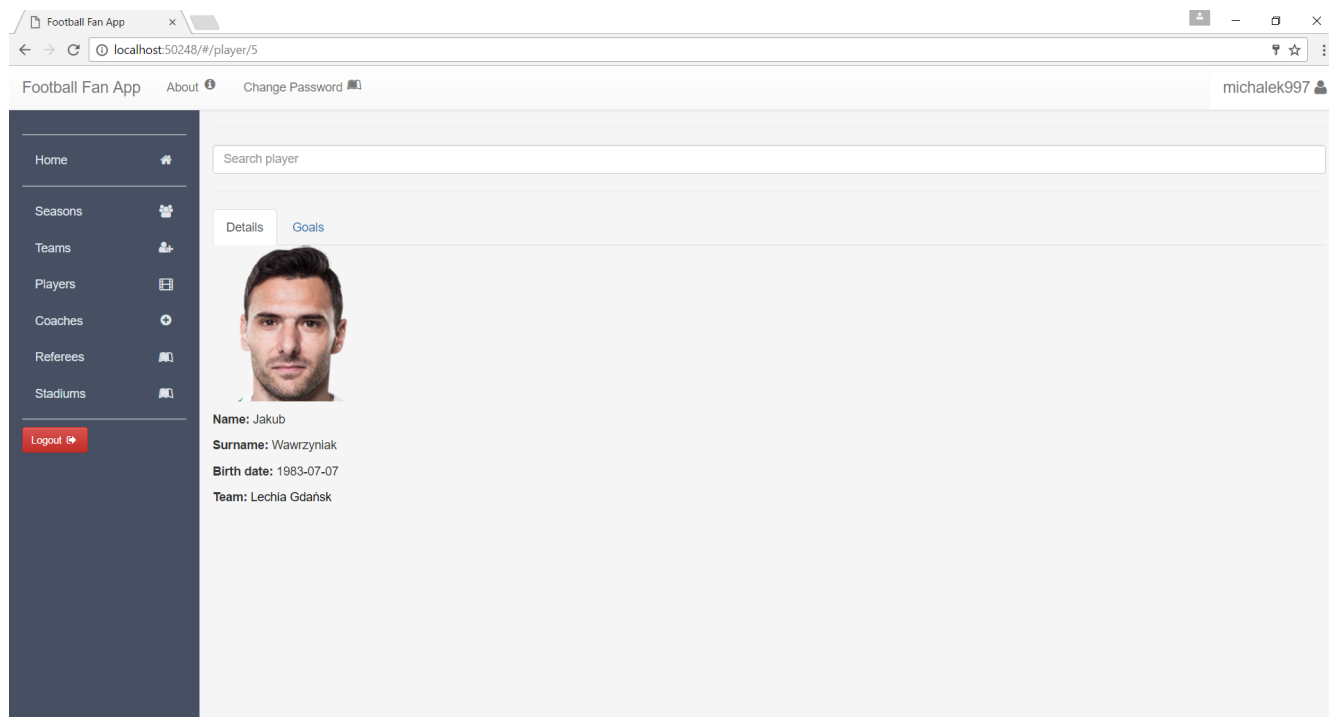
Zakładka Players prezentuje listę piłkarzy zarejestrowanych w rozgrywkach w drużynie:

The screenshot shows the 'Football Fan App' interface with the 'Players' tab selected. The main content area displays a table of players registered in the league. The table has three columns: 'First name', 'Surname', and 'Birth date'.

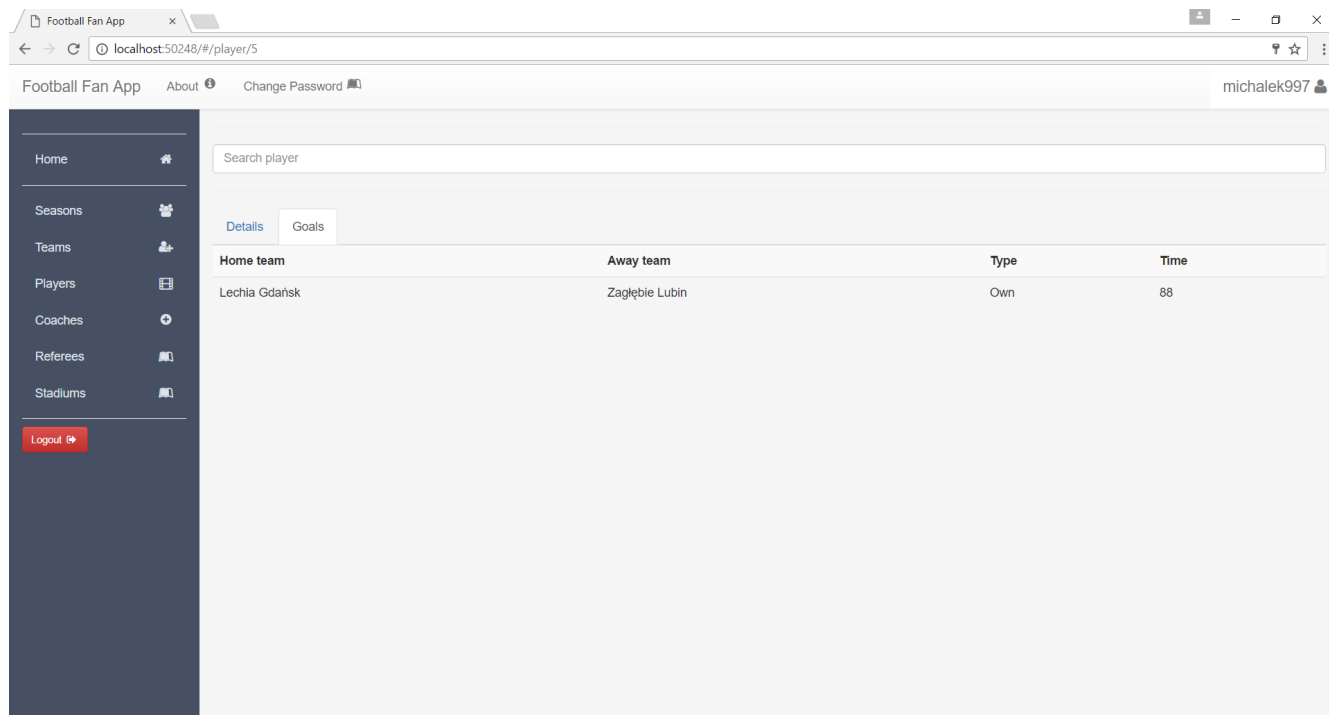
First name	Surname	Birth date
Mateusz	Bąk	1983-02-24
Adam	Chrzanowski	1999-03-31
Rafał	Janicki	1992-07-05
Paweł	Stolarski	1996-01-28
Jakub	Wawrzyniak	1983-07-07
Flávio	Paixão	1984-09-19
Martin	Kobyliński	1994-03-08
Miloš	Krašić	1984-11-01
Sebastian	Mila	1982-07-10
Grzegorz	Kuświk	1987-05-23
Marco	Paixão	1984-09-19

Zakładki Matches i Seasons prezentują listy meczów drużyny oraz sezonów, w których drużyna uczestniczyła w rozgrywkach.

Widok Players udostępnia wyszukiwarkę graczy oraz dwie zakładki:



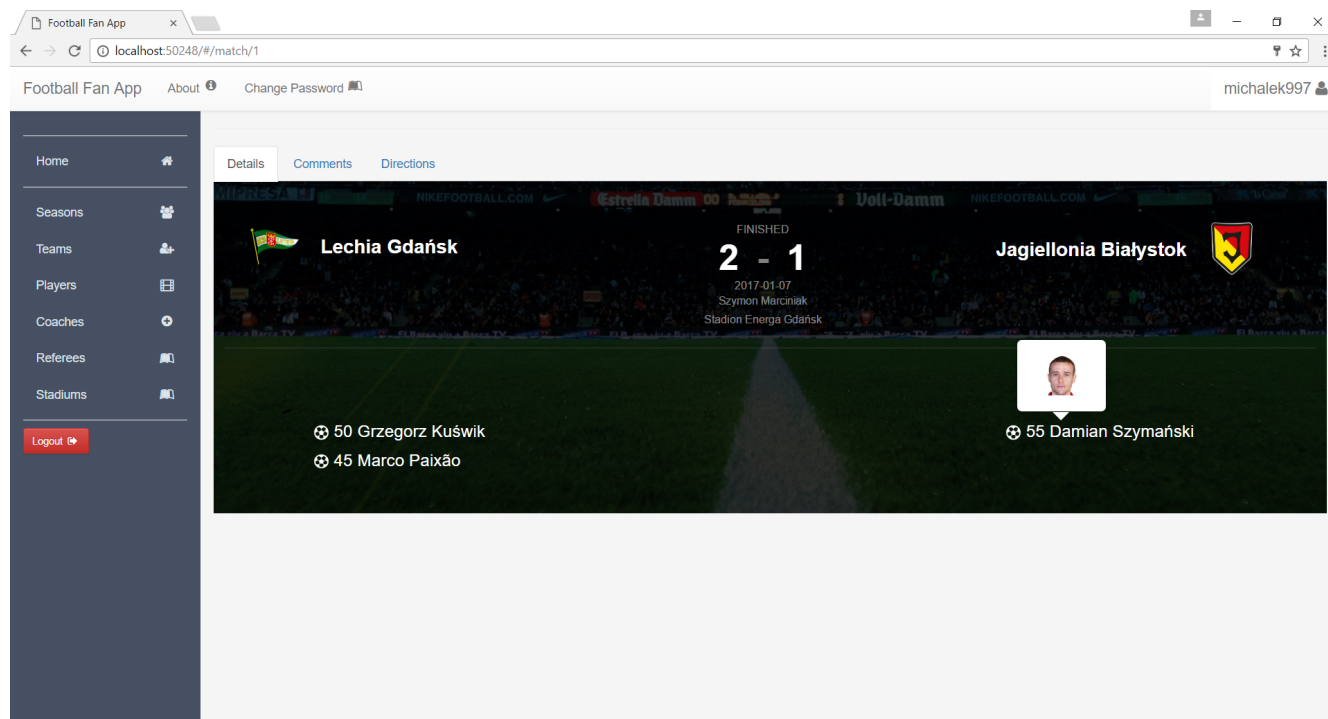
Zakładka Goals to lista wszystkich bramek zdobytych przez wybranego piłkarza, zawierająca informacje o bramce: typ, czas strzelenia oraz drużyny, które ze sobą grały.



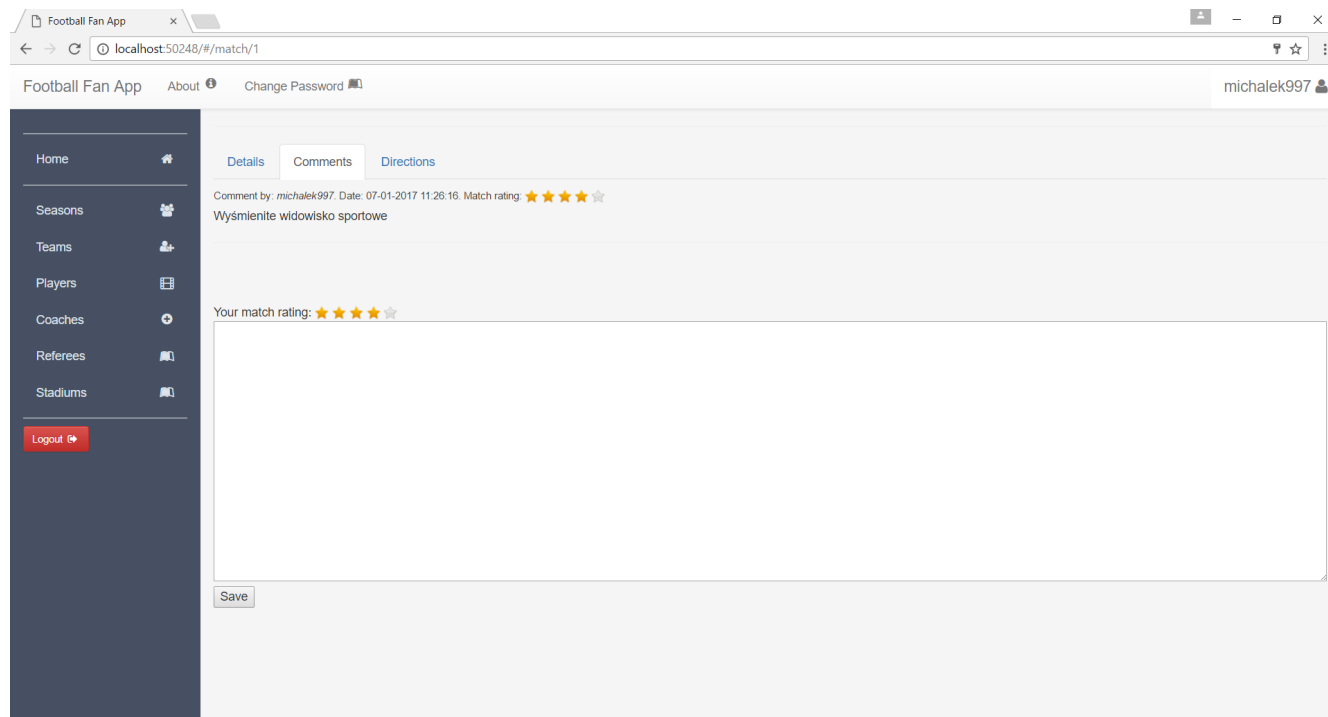


Widok sędziego udostępnia wyszukiwarkę sędziów oraz zakładki z danymi arbitra oraz meczami. Widok trenera udostępnia wyszukiwarkę trenerów oraz zakładki z danymi szkoleniowca oraz prowadzonymi przez niego drużynami. Podobnie widok stadionu, który oprócz zakładki z danymi zawiera listę drużyn, rozgrywających spotkania na danym obiekcie. Powyższe trzy widoki działają na tej samej zasadzie, co widoki opisane powyżej, tj. należy wybrać z wyszukiwarki odpowiedni obiekt, a następnie wyświetlone zostaną jego dane.

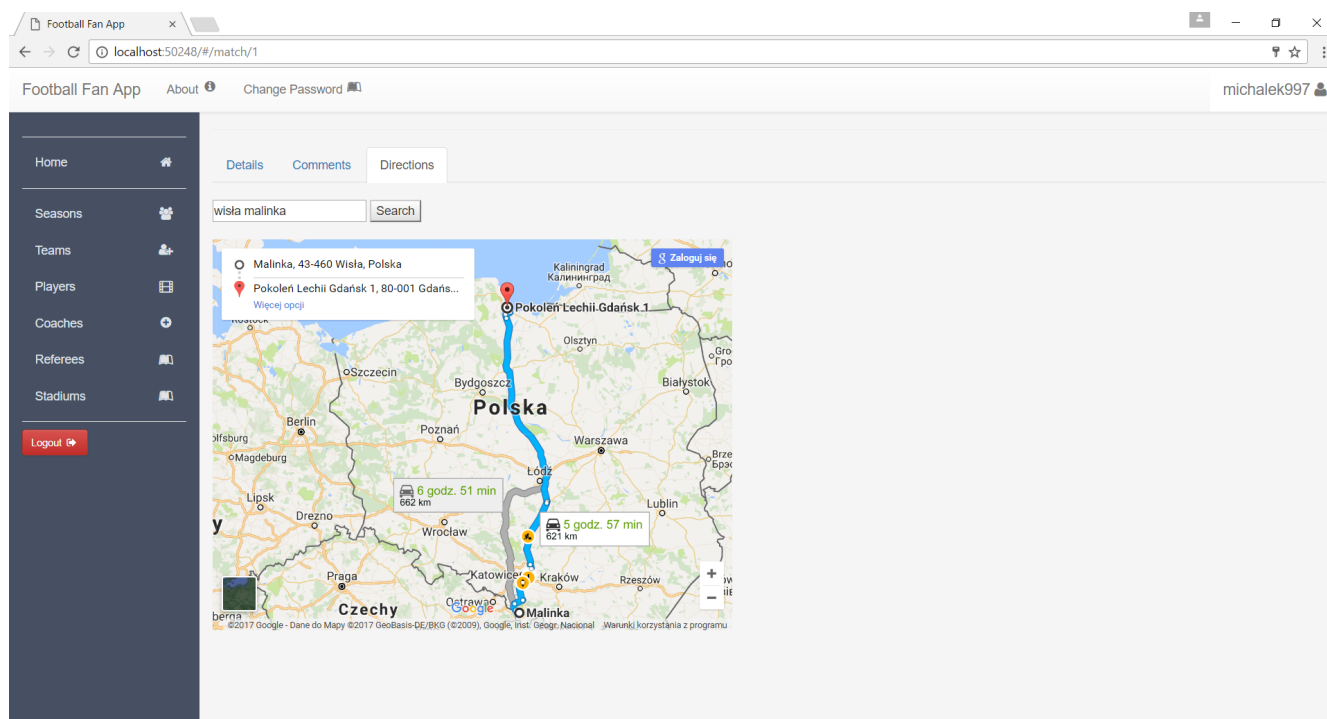
Do widoku meczu można przejść jedynie klikając na odpowiednie pola w powyższych widokach. Prezentowany widok zawiera trzy zakładki. Pierwsza z nich prezentuje wynik meczu, drużyny, datę oraz miejsce rozgrywania sędzię oraz listy strzelców. Na liście strzelców czerwoną ikoną piłki wyróżniono bramki samobójcze. Na strzelca można najechać kursorem aby zobaczyć jego zdjęcie.



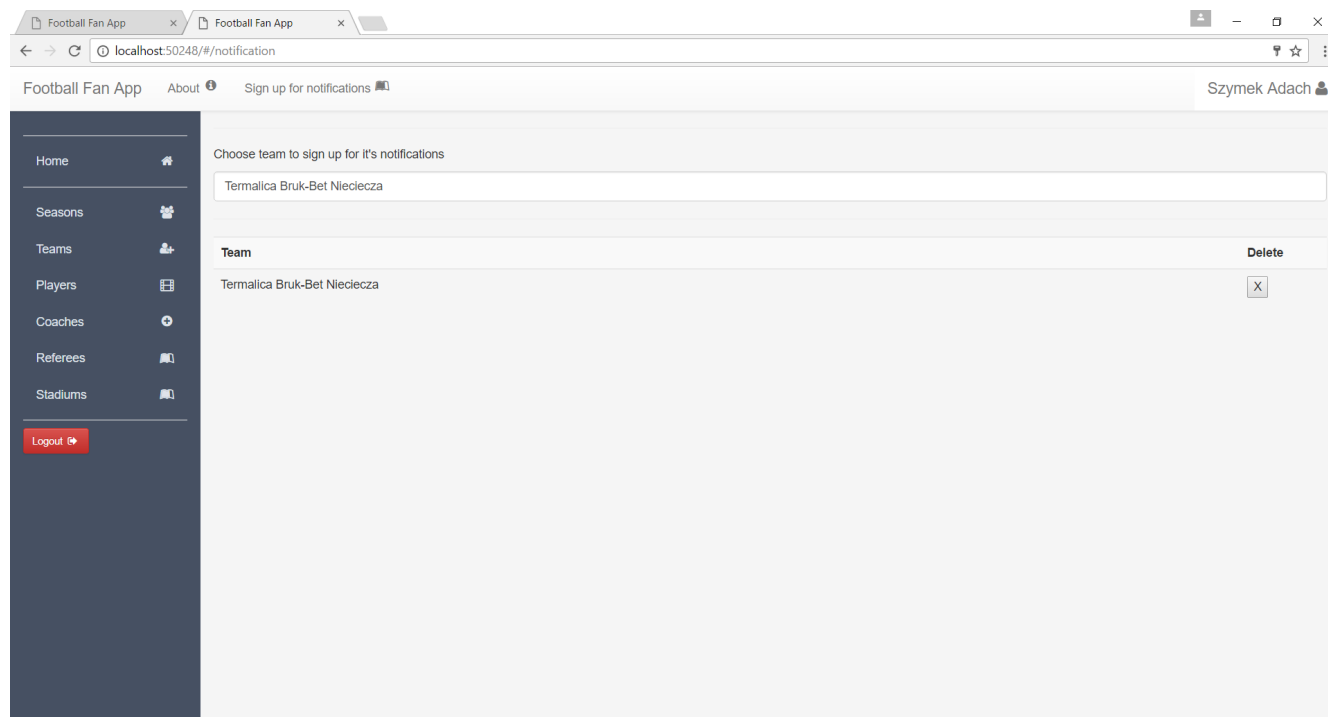
Widok Comments pozwala na ocenę meczu oraz dodanie komentarza tekstowego.



Widok Directions prezentuje za pomocą serwisu Google Maps trasę dojazdu z wybranego przez użytkownika miejsca do miejsca rozgrywania spotkania:



Jeżeli użytkownik zalogowany jest przez konto w serwisie Facebook, to możliwe jest zapisanie się (oraz wypisanie) na powiadomienia o zbliżającym się meczu za pomocą widoku, do którego prowadzi przycisk w górnym menu.



Ważnym faktem jest, że w każdym widoku poprzez kliknięcie na odpowiednie łącze, możemy przejść do wybranego zawodnika, drużyny, meczu, itd., co eliminuje konieczność ponownego ich wyszukiwania. Np. w widoku meczu możemy, poprzez kliknięcie na zawodnika, drużynę, sędziego bądź stadion przenieść się do widoku, który zawiera szczegółowe informacje o obiekcie.

## 2.6 Instrukcja utrzymania

Błędy w działaniu aplikacji spowodowane są problemami w komunikacji klient-serwer. Należy sprawdzić czy mamy łączność z Internetem, np. za pomocą konsoli cmd, wpisując w niej:

```
ping google.com
```

Jeżeli tylko próba połączenia z serwerem systemu powoduje timeout, to należy powiadomić Administratora systemu. W przeciwnym przypadku należy skontaktować się z dostawcą internetu.

## 2.7 Raport odstępstw od specyfikacji wymagań

Brak odstępstw.

## 2.8 Dokumentacja usług Web Services

### API sezonów

Poprzez publiczne API dostępna jest przefiltrowana lista dostępnych sezonów.

#### Przykład użycia – wywołanie

Metoda HTTP GET

#### Parametry wywołania:

filter – ciąg znaków, który ma być zawarty w zwracanych wynikach

#### Przykładowe wywołanie (zwraca obiekty zawierające ciąg “sezon” w nazwie)

`http://[adres serwera]/api/season/list/sezon`

#### Przykładowa odpowiedź

```
[{"Id":1,"Name":"Sezon 2016/17"}]
```

### API zespołów

Poprzez publiczne API dostępna jest przefiltrowana lista dostępnych zespołów.

#### Przykład użycia – wywołanie

Metoda HTTP GET

#### Parametry wywołania:

filter – ciąg znaków, który ma być zawarty w zwracanych wynikach

#### Przykładowe wywołanie (zwraca obiekty zawierające ciąg “termalica” w nazwie)

`http://[adres serwera]/api/team/list/termalica`

#### Przykładowa odpowiedź

```
[{"Id":4,"Name":"Termalica Bruk-Bet Nieciecza"}]
```

### API zawodników

Poprzez publiczne API dostępna jest przefiltrowana lista dostępnych zawodników.

#### Przykład użycia – wywołanie

Metoda HTTP GET

#### Parametry wywołania:

filter – ciąg znaków, który ma być zawarty w zwracanych wynikach

### **Przykładowe wywołanie (zwraca obiekty zawierające ciąg “mar” w nazwie)**

http://[adres serwera]/api/player/list/mar

Przykładowa odpowiedź

```
[{"Id":7,"Name":"Martin Kobylański"}, {"Id":11,"Name":"Marco Paixão"}, {"Id":16,"Name":"Marek Wasiluk"}]
```

### **API trenerów**

Poprzez publiczne API dostępna jest przefiltrowana lista dostępnych trenerów.

#### **Przykład użycia – wywołanie**

Metoda HTTP GET

#### **Parametry wywołania:**

filter – ciąg znaków, który ma być zawarty w zwracanych wynikach

### **Przykładowe wywołanie (zwraca obiekty zawierające ciąg “mich” w nazwie)**

http://[adres serwera]/api/coaches/list/mich

Przykładowa odpowiedź

```
[{"Id":4,"Name":"Czesław Michniewicz"}, {"Id":2,"Name":"Michał Probierz"}]
```

### **API sędziów**

Poprzez publiczne API dostępna jest przefiltrowana lista dostępnych sędziów.

#### **Przykład użycia – wywołanie**

Metoda HTTP GET

#### **Parametry wywołania:**

filter – ciąg znaków, który ma być zawarty w zwracanych wynikach

### **Przykładowe wywołanie (zwraca obiekty zawierające ciąg “pierluigi” w nazwie)**

http://[adres serwera]/api/referees/list/pierluigi

Przykładowa odpowiedź

```
[{"Id":2,"Name":"Pierluigi Colina"}]
```

## API stadionów

Poprzez publiczne API dostępna jest przefiltrowana lista dostępnych stadionów.

### Przykład użycia – wywołanie

Metoda HTTP GET

#### Parametry wywołania:

filter – ciąg znaków, który ma być zawarty w zwracanych wynikach

### Przykładowe wywołanie (zwraca obiekty zawierające ciąg “stadion” w nazwie)

`http://[adres serwera]/api/stadium/list/stadion`

Przykładowa odpowiedź

```
[{"Id":1,"Name":"Stadion Energa Gdańsk"}, {"Id":2,"Name":"Stadion Miejski"}, {"Id":4,"Name":"Stadion Nieciecza KS"}, {"Id":3,"Name":"Stadion Zagłębia"}]
```

## API spotkań

Poprzez publiczne API dostępna jest przefiltrowana lista spotkań w poprzedniej i następnej kolejce.

### Przykład użycia – wywołanie

Metoda HTTP GET

### Przykładowe wywołanie (zwraca spotkania z poprzedniej kolejki)

`http://[adres serwera]/api/matches/previous`

### Przykładowe wywołanie (zwraca spotkania z następnej kolejki)

`http://[adres serwera]/api/matches/next`

Przykładowa odpowiedź

```
{ "Referee": "Szymon Marciniak", "RefereId": 1, "Id": 1, "HomeTeamId": 1, "HomeTeamName": "Lechia Gdańsk", "HomeTeamLogoUrl": "lechia-gdansk.png", "AwayTeamId": 2, "AwayTeamName": "Jagiellonia Białystok", "AwayTeamLogoUrl": "jagiellonia-bialystok.png", "Stadium": { "Id": 1, "Name": "Stadion Energa Gdańsk", "Capacity": 43165, "PictureUrl": "gdansk.jpg", "Address": { "Id": 1, "City": "Gdańsk", "Zipcode": "80-560", "Street": "Pokoleń Lechii Gdańsk", "Number": "1" }, "Teams": [ { "Id": 1, "Name": "Lechia Gdańsk" } ], "Date": "2017-01-15T00:00:00", "HomeGoalsCount": 2, "AwayGoalsCount": 1, "HomeGoals": [ { "Id": 1, "Footballer": { "Id": 10, "Name": "Grzegorz Kuświk", "Team": "Lechia Gdańsk", "PictureUrl": "gdansk (10).png" }, "HomeTeam": "Lechia Gdańsk", "AwayTeam": "Jagiellonia Białystok", "Type": "Normal", "Time": 50 }, { "Id": 2, "Footballer": { "Id": 11, "Name": "Marco Paixão", "Team": "Lechia Gdańsk", "PictureUrl": "gdansk (11).png" }, "HomeTeam": "Lechia Gdańsk", "AwayTeam": "Jagiellonia Białystok", "Type": "Penalty", "Time": 45 }, "AwayGoals":
```

```
[{"Id":3,"Footballer":{"Id":20,"Name":"Damian Szymański","Team":"Jagiellonia Białystok","PictureUrl":"jaga (9).png"},"HomeTeam":"Lechia Gdańsk","AwayTeam":"Jagiellonia Białystok","Type":"Normal","Time":55},"Comments":[],"AverageRating":0},{Referee":"Pierluigi Colina","RefereeId":2,"Id":7,"HomeTeamId":3,"HomeTeamName":"Zagłębie Lubin","HomeTeamLogoUrl":"zagłębie-lubin.png","AwayTeamId":4,"AwayTeamName":"Termalica Bruk-Bet Nieciecza","AwayTeamLogoUrl":"termalica.png","Stadium":{"Id":3,"Name":"Stadion Zagłębia","Capacity":16100,"PictureUrl":"lubin.jpg","Address":{"Id":3,"City":"Lubin","Zipcode":"56-301","Street":"Marii Skłodowskiej-Curie","Number":"98"},"Teams":[{"Id":3,"Name":"Zagłębie Lubin"}]},"Date":"2017-01-15T00:00:00","HomeGoalsCount":0,"AwayGoalsCount":0,"HomeGoals":[],"AwayGoals":[],"Comments":[],"AverageRating":0}]
```