

BasketBet

Łukasz Michalski

pzx110616

BasketBet to aplikacja umożliwiająca użytkownikom obstawianie wyników meczów NBA. Użytkownicy mogą tworzyć konta, logować się i śledzić historię swoich zakładów. System aktualizuje statystyki zespołów, zapewniając dostęp do najnowszych informacji potrzebnych do podejmowania decyzji. Tylko zalogowani użytkownicy mogą brać udział w obstawianiu, a wszystkie zakłady są przypisywane do ich kont, co pozwala na późniejszą analizę wyników.

Specyfikacja wykorzystanych technologii:

.NET 8 – nowoczesna platforma programistyczna używana do budowy aplikacji webowych.

ASP.NET MVC – framework zastosowany do stworzenia warstwy wizualnej aplikacji oraz logiki kontrolerów.

ASP.NET Web API – komponent obsługujący komunikację z zewnętrznymi źródłami danych oraz aktualizację bazy.

Entity Framework Core z MS SQL Server – narzędzie ORM wykorzystywane do zarządzania relacyjną bazą danych i uproszczenia operacji SQL.

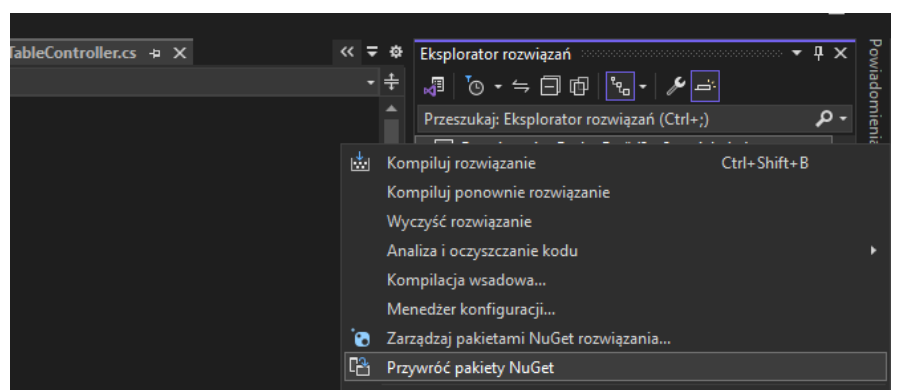
HTMLAgilityPack – biblioteka służąca do pobierania i przetwarzania danych ze źródeł HTML, wykorzystywana do zaciągania statystyk meczów NBA.

Odpalanie projektu:

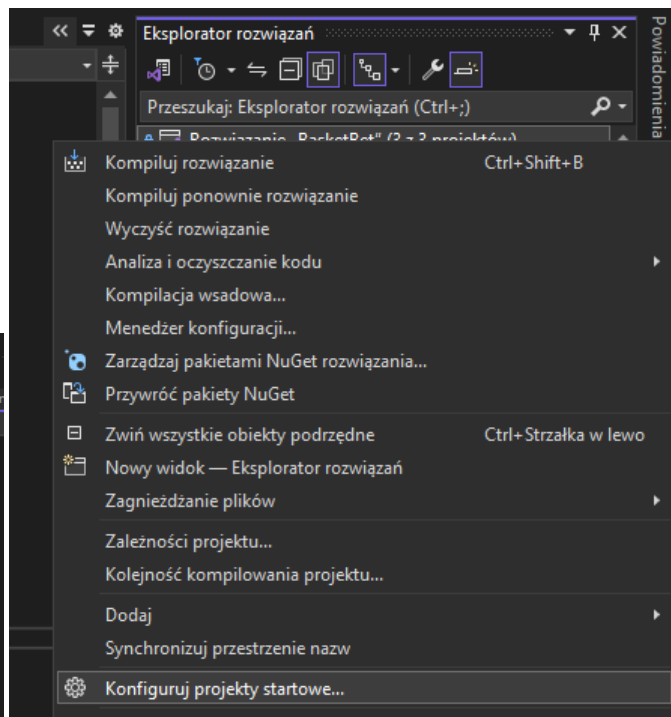
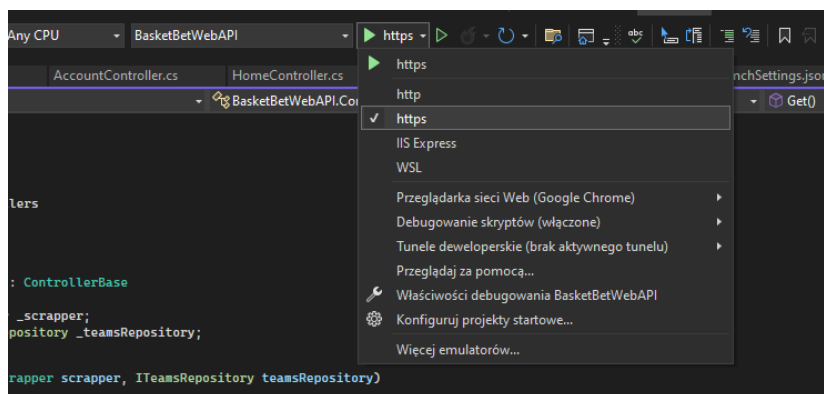
1. W pliku *BasketBet.EntityFramework/databasesettings.json* należy podać *connection string* do własnej bazy danych MS SQL Server

W pliku *BasketBetWebAPI/nlog.config* `<target filename="">` należy podać ścieżki do plików tekstowych, w których zapisywać będą się informacje o ewentualnych błędach. *allfile* dla wszelkich ostrzeżeń oraz błędów, *exceptions* tylko dla błędów

2. Należy przywrócić pakiety NuGet oraz zbudować projekt jeżeli Visual Studio nie zrobiło tego automatycznie

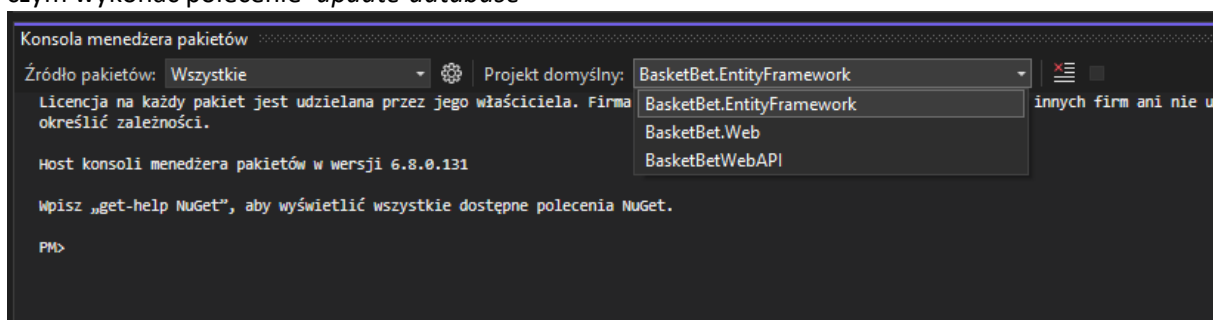


3. Następnie należy ustawić projekt BasketBet.WebAPI jako startowy i upewnić się, że wybranym profilem uruchamiania jest 'https'

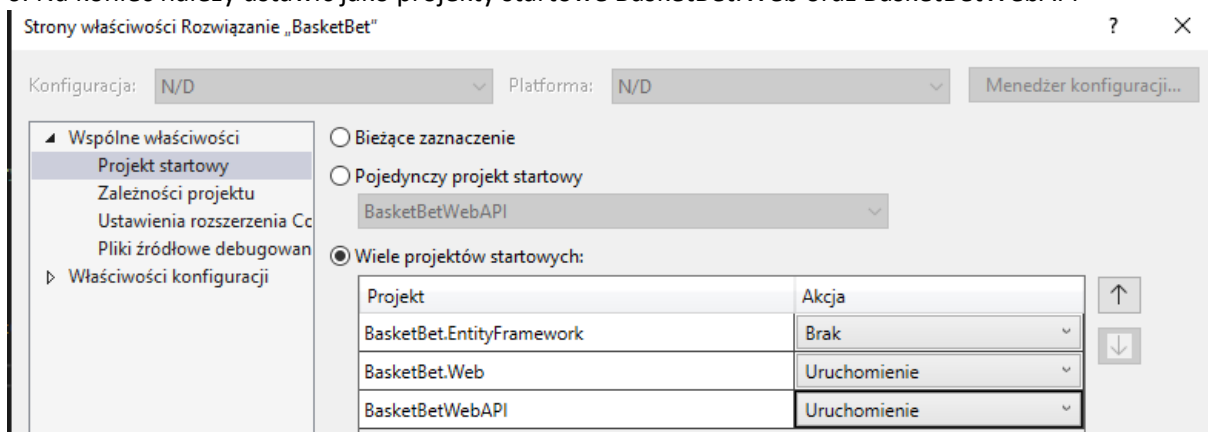


4. Następnie należy ustawić projekt BasketBet.Web jako projekt startowy

5. W konsoli pakietów nuget należy ustawić projekt domyślny na BasketBet.EntityFramework po czym wykonać polecenie 'update-database'



6. Na koniec należy ustawić jako projekty startowe BasketBet.Web oraz BasketBetWebAPI



7. Przy pierwszym odpaleniu aplikacji może dłużej potrwać zanim pobiorą się wszystkie dane. Wykonuje się to asynchronicznie przy przechodzeniu przez zakładki oraz przy logowaniu/rejestracji

Struktura projektu:

BasketBetWebAPI – WebApi służące do aktualizowania rzeczywistych danych z ligi NBA za pomocą pobierania danych z publicznej strony internetowej

- **ErrorHandlingMiddleware** – odpowiedzialne za logowanie wszelkich błędów.
- **Kontrolery**: ScheduleController oraz TableController – zarządzają logiką pobierania harmonogramów meczów oraz tabeli wyników.
- **Serwis**: Scraper – obsługuje pobieranie danych NBA z publicznych stron internetowych.
- **Repozytoria**: GamesRepository oraz TeamsRepository – odpowiadają za komunikację z bazą danych i zarządzanie danymi o meczach oraz zespołach.

BasketBet.Web – Aplikacja internetowa – symulator obstawiania meczów NBA

- **Kontrolery**: AccountController oraz HomeController, odpowiedzialne za logowanie, rejestrację oraz wyświetlanie danych o dostępnych meczach.
- **Repozytoria**: BetRepository, GamesRepository, TeamsRepository oraz UserRepository, obsługujące komunikację z bazą danych.
- **Widoki**: Powiązane z kontrolerami, renderujące formularze logowania, rejestracji oraz dostępne mecze do obstawiania.
- **Foldery statyczne**: Skrypty JavaScript w `wwwroot/js/site.js` oraz style CSS w `wwwroot/css/site.css`, odpowiedzialne za wygląd i interaktywność aplikacji.

BasketBet.EntityFramework – biblioteka klas odpowiedzialna za obsługę bazy danych

- **DataContext** – Klasa kontekstowa bazy danych, która dziedziczy po DbContext i zawiera definicje tabel w bazie danych (DbSet).
- **Modele**:
 - AppUser** – Dziedziczy po IdentityUser, reprezentuje użytkownika aplikacji.
 - Bet, BetItem, Game, Team** – Modele reprezentujące tabele w bazie danych.
- **databasesettings.json** – Plik konfiguracyjny zawierający łańcuch połączenia do bazy danych SQL Server.
- **ServiceCollectionExtensions** – Klasa rozszerzająca IServiceCollection, odpowiedzialna za rejestrację usług, w tym połączenie z bazą danych oraz konfigurację tożsamości użytkownika.

Modele

Modele encji BasketBet.EntityFramework/Entities

AppUser

Reprezentuje użytkownika aplikacji, który jest powiązany z zakładami i może zdobywać punkty.

- **Points**: Liczba punktów użytkownika.
- **LastPointsClaimTime**: Data i godzina ostatniego zgłoszenia punktów przez użytkownika.
- **Bets**: Kolekcja zakładów powiązanych z użytkownikiem.

Bet

Reprezentuje zakład użytkownika na konkretne mecze.

- **Id**: Unikalny identyfikator zakładu.
- **TotalOdds**: Łączne kursy zakładu.
- **Bid**: Kwota postawiona na zakład.
- **PotentialWinning**: Potencjalna wygrana z zakładu.
- **BetOutcome**: Wynik zakładu (True – wygrana, False – przegrana, Null – nie rozstrzygnięty).
- **Games**: Kolekcja meczów powiązanych z zakładem.
- **AppUserId**: Identyfikator użytkownika, który postawił zakład.
- **AppUser**: Obiekt użytkownika powiązanego z zakładem.

BetItem

Reprezentuje pojedynczy element zakładu, powiązany z jednym meczem i drużyną.

- **Id**: Unikalny identyfikator elementu zakładu.
- **SelectedTeamId**: Identyfikator drużyny na której zwycięstwo obstawiono w zakładzie.
- **ItemOdds**: Kurs na wygranie tego elementu zakładu.
- **BetId**: Identyfikator zakładu, do którego należy element.
- **Bet**: Obiekt zakładu, do którego należy element.
- **GameId**: Identyfikator meczu powiązanego z elementem zakładu.
- **Game**: Obiekt meczu powiązanego z elementem zakładu.
- **BetItemOutcome**: Wynik elementu zakładu (True – wygrana, False – przegrana, Null – nie rozstrzygnięty).

Game

Reprezentuje mecz pomiędzy dwiema drużynami.

- **Id**: Unikalny identyfikator meczu.
- **Date**: Data meczu.
- **HomeTeamId**: Identyfikator drużyny gospodarzy.
- **HomeTeam**: Obiekt drużyny gospodarzy.
- **HomeTeamScore**: Wynik drużyny gospodarzy.
- **OddsHomeTeam**: Kurs na drużynę gospodarzy.
- **AwayTeamId**: Identyfikator drużyny gości.
- **AwayTeam**: Obiekt drużyny gości.
- **AwayTeamScore**: Wynik drużyny gości.
- **OddsAwayTeam**: Kurs na drużynę gości.
- **Bets**: Kolekcja zakładów powiązanych z meczem.

Team

Reprezentuje drużynę w lidze NBA.

- **Id:** Unikalny identyfikator drużyny.
- **Name:** Nazwa drużyny.
- **Wins:** Liczba zwycięstw drużyny.
- **Looses:** Liczba porażek drużyny.
- **WinningPercentage:** Procent zwycięstw drużyny.
- **HomeRecord:** Rekord drużyny w meczach domowych.
- **AwayRecord:** Rekord drużyny w meczach wyjazdowych.
- **PointsPerGame:** Średnia liczba punktów zdobywana przez drużynę na mecz.
- **OpponentPointsPerGame:** Średnia liczba punktów zdobywana przez drużynę przeciwnika na mecz.
- **CurrentStreak:** Aktualna seria wygranych/porażek drużyny.
- **Last10Record:** Rekord drużyny w ostatnich 10 meczach.
- **Conference:** Konferencja, w której gra drużyna.
- **HomeGames:** Kolekcja meczów drużyny w roli gospodarza.
- **AwayGames:** Kolekcja meczów drużyny w roli gościa.

ViewModele w BasketBet.Web

BetVM - Reprezentuje widok zakładu użytkownika.

- **BetsList:** Lista pojedynczych zakładów (powiązanych z BetItem).
- **TotalCourse:** Łączny kurs zakładu.
- **Points:** Liczba punktów dostępnych dla użytkownika.
- **TotalWinning:** Potencjalna wygrana z zakładu.
- **BetOutcome:** Wynik zakładu (True – wygrana, False – przegrana, Null – nie rozstrzygnięty).

GameVM - Reprezentuje widok meczu.

- **Id:** Identyfikator meczu.
- **Date:** Data meczu.
- **HomeTeamVMId:** Identyfikator drużyny gospodarzy.
- **HomeTeamVM:** Obiekt drużyny gospodarzy (odpowiednik HomeTeam w Game).
- **HomeTeamScore:** Wynik drużyny gospodarzy.
- **AwayTeamVMId:** Identyfikator drużyny gości.
- **AwayTeamVM:** Obiekt drużyny gości (odpowiednik AwayTeam w Game).
- **AwayTeamScore:** Wynik drużyny gości.

- **OddsHomeTeam:** Kurs na drużynę gospodarzy.
- **OddsAwayTeam:** Kurs na drużynę gości.

HomeVM - Reprezentuje główny widok aplikacji (strona główna).

- **Matches:** Lista meczów (GameVM) do wyświetlenia na stronie.
- **LastPointsClaimTime:** Czas ostatniego zgłoszenia punktów przez użytkownika.

LeaderBoardsVM - Reprezentuje widok tablicy liderów.

- **CurrentUserPosition:** Pozycja aktualnego użytkownika na tablicy liderów.
- **CurrentUser:** Dane aktualnego użytkownika (UserVM).
- **Leaders:** Lista liderów na tablicy (lista UserVM).

LoginVM - Reprezentuje widok logowania użytkownika.

- **Email:** Adres e-mail użytkownika, wymagany do logowania. **[Required]**
- **Password:** Hasło użytkownika, wymagane do logowania. **[Required] [DataType.Password]**
- **RememberMe:** Określa, czy użytkownik chce zostać zapamiętany.

RegisterVM - Reprezentuje widok rejestracji nowego użytkownika.

- **Username:** Nazwa użytkownika, wymagana do rejestracji. **[Required]**
- **Email:** Adres e-mail użytkownika, wymagany do rejestracji. **[Required] [DataType.EmailAddress]**
- **Password:** Hasło użytkownika, wymagane do rejestracji. **[Required] [DataType.Password]**
- **ConfirmPassword:** Potwierdzenie hasła, wymagane do rejestracji, musi być takie same jak hasło. **[Required] [Compare("Password", ErrorMessage = "Passwords do not match")] [DataType.Password]**

SingleGameBetVM - Reprezentuje widok pojedynczego elementu zakładu na mecz.

- **Id:** Identyfikator elementu zakładu.
- **GameVMId:** Identyfikator meczu (GameVM) powiązanego z elementem.
- **GameVM:** Obiekt meczu (GameVM), powiązany z elementem zakładu.
- **TeamTypedOnId:** Identyfikator drużyny, na którą postawiono.
- **TeamTypedOn:** Obiekt drużyny (TeamVM), na którą postawiono.
- **Course:** Kurs przypisany do drużyny wybranej w zakładzie.
- **BetItemOutcome:** Wynik elementu zakładu (True – wygrana, False – przegrana, Null – nie rozstrzygnięty).

StandingsVM - Reprezentuje widok klasyfikacji drużyn.

- **WesternConference**: Lista drużyn z zachodniej konferencji (TeamVM).
- **EasternConference**: Lista drużyn ze wschodniej konferencji (TeamVM).

TeamVM - Reprezentuje widok drużyny.

- **Id**: Identyfikator drużyny.
- **Name**: Nazwa drużyny.
- **Wins**: Liczba zwycięstw drużyny.
- **Looses**: Liczba porażek drużyny.
- **WinningPercentage**: Procent zwycięstw drużyny.
- **HomeRecord**: Rekord drużyny w meczach domowych.
- **AwayRecord**: Rekord drużyny w meczach wyjazdowych.
- **PointsPerGame**: Średnia liczba punktów zdobywana przez drużynę.
- **OpponentPointsPerGame**: Średnia liczba punktów zdobywana przez przeciwników drużyny.
- **CurrentStreak**: Aktualna seria drużyny.
- **Last10Record**: Rekord drużyny w ostatnich 10 meczach.
- **Conference**: Konferencja, w której drużyna gra.
- **HomeGames**: Kolekcja meczów drużyny w roli gospodarza (GameVM).
- **AwayGames**: Kolekcja meczów drużyny w roli gościa (GameVM).

UserVM - Reprezentuje widok użytkownika.

- **Name**: Imię użytkownika.
- **Points**: Liczba punktów użytkownika.
- **BetSuccessRate**: Procentowy wskaźnik sukcesów zakładów użytkownika.

ViewModele BasketBetWebAPI

TeamDto - Reprezentuje dane drużyny w formacie DTO (Data Transfer Object) używane do przekazywania danych pomiędzy warstwami aplikacji.

- **Id**: Identyfikator drużyny.
- **Name**: Nazwa drużyny.
- **Wins**: Liczba zwycięstw drużyny.
- **Looses**: Liczba porażek drużyny.
- **WinningPercentage**: Procent zwycięstw drużyny.
- **HomeRecord**: Rekord drużyny w meczach domowych.
- **AwayRecord**: Rekord drużyny w meczach wyjazdowych.
- **PointsPerGame**: Średnia liczba punktów zdobywana przez drużynę.
- **OpponentPointsPerGame**: Średnia liczba punktów zdobywana przez drużynę przez przeciwników.
- **CurrentStreak**: Aktualna seria drużyny.

- Last10Record: Rekord drużyny w ostatnich 10 meczach.
- Conference: Konferencja, do której należy drużyna.
- HomeGames: Kolekcja meczów drużyny w roli gospodarza (GameDto).
- AwayGames: Kolekcja meczów drużyny w roli gościa (GameDto).

GameDto - Reprezentuje dane meczu w formacie DTO.

- Id: Identyfikator meczu.
- Date: Data meczu.
- HomeTeamDtoid: Identyfikator drużyny gospodarzy.
- HomeTeamDto: Obiekt drużyny gospodarzy (TeamDto).
- HomeTeamScore: Wynik drużyny gospodarzy.
- AwayTeamDtoid: Identyfikator drużyny gości.
- AwayTeamDto: Obiekt drużyny gości (TeamDto).
- AwayTeamScore: Wynik drużyny gości.
- OddsHomeTeam: Kurs na drużynę gospodarzy.
- OddsAwayTeam: Kurs na drużynę gości.

Opis kontrolerów i akcji

BasketBetWebAPI

1. ScheduleController

- **Route:** [controller]
- **Opis:** Kontroler odpowiedzialny za aktualizowanie informacji o meczach i wynikach w oparciu o dane pobrane za pomocą Scrapper z zewnętrznego źródła (np. strona internetowa).

UpdateGamesFromDate

- **HTTP Method:** PUT
- **Route:** /{date}
- **Parametry:**
 - date - Parametr w ścieżce (typ: DateOnly), reprezentuje datę, od której mają zostać pobrane mecze.
- **Opis działania:**

- Metoda uruchamia proces pobierania meczów na podstawie podanej daty.
- Używa metody UpdateGamesFromDate w klasie Scraper.
- **Zwracane dane:**
 - **NoContent (204)** - Oznacza, że operacja została wykonana pomyślnie, ale nie zwraca danych.

Refresh

- **HTTP Method:** PUT
- **Route:** ``
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Metoda wywołuje proces odświeżenia danych o meczach za pomocą metody UpdateGames w klasie Scraper.
- **Zwracane dane:**
 - **NoContent (204)** - Operacja zakończona sukcesem bez zwrócenia danych.

UpdateScores

- **HTTP Method:** PUT
- **Route:** scores/{date}
- **Parametry:**
 - date - Parametr w ścieżce (typ: DateOnly), reprezentuje datę, dla której mają zostać zaktualizowane wyniki.
- **Opis działania:**
 - Metoda uruchamia proces aktualizacji wyników meczów na podstawie daty.
 - Używa metody UpdateGamesResults w klasie Scraper i zwraca odpowiedni status w zależności od wyniku.
- **Zwracane dane:**
 - **Ok (200)** - Jeżeli wyniki zostały zaktualizowane pomyślnie.
 - **NoContent (204)** - Jeśli proces aktualizacji wyników nie przyniósł rezultatów.

2. TableController

- **Route:** [controller]
- **Opis:** Kontroler odpowiedzialny za zarządzanie tabelą drużyn, aktualizowanie jej i pobieranie danych drużyn z repozytorium.

Get

- **HTTP Method:** GET
- **Route:** ``
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Metoda pobiera wszystkie drużyny z repozytorium ITeamsRepository.
 - Zwraca dane w postaci listy obiektów TeamDto.
- **Zwracane dane:**
 - **Ok (200)** - Lista drużyn (IEnumerable<TeamDto>), która jest zwracana w odpowiedzi.

Refresh

- **HTTP Method:** PUT
- **Route:** ``
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Metoda wywołuje proces odświeżenia tabeli drużyn za pomocą metody UpdateTable w klasie Scrapper.
- **Zwracane dane:**
 - **NoContent (204)** - Operacja zakończona sukcesem bez zwrócenia danych.

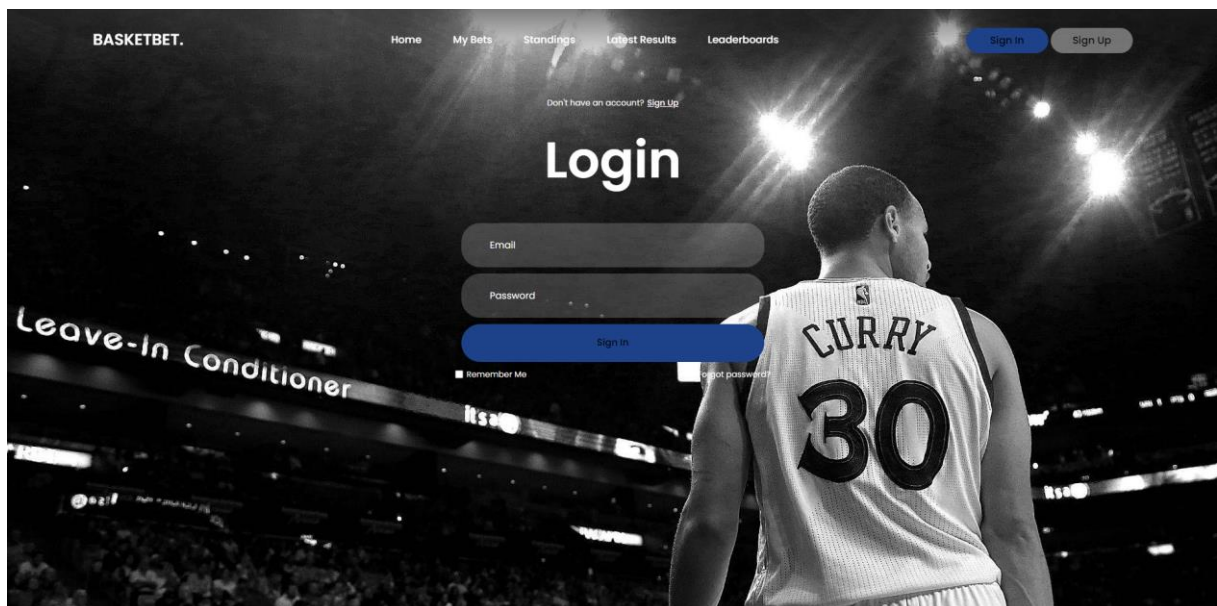
BasketBet.Web

1. AccountController

- **Opis:** Kontroler odpowiedzialny za obsługę logowania, rejestracji użytkowników oraz ich wylogowywania. Dodatkowo kontroler integruje się z API zewnętrznym, wysyłając zapytania do różnych punktów API po udanym logowaniu lub rejestracji użytkownika.

Login (GET)

- **HTTP Method:** GET
- **Route:** Account/Login
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Metoda renderuje formularz logowania dla użytkownika.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok Login, w którym użytkownik może wprowadzić swoje dane logowania.



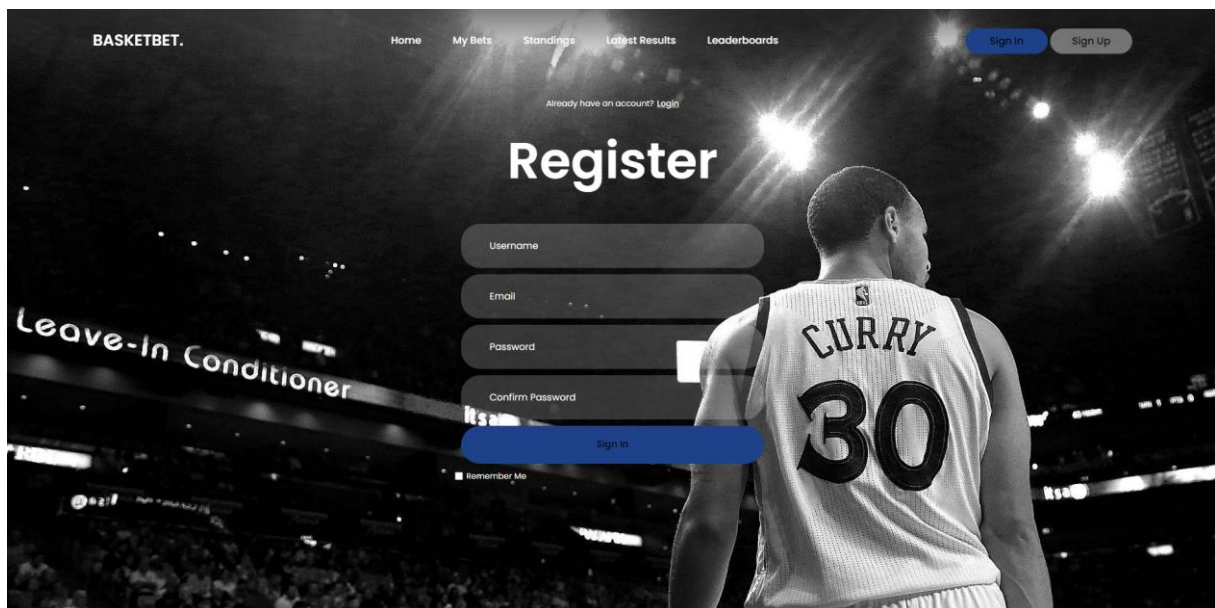
Login (POST)

- **HTTP Method:** POST
- **Route:** Account/Login
- **Parametry:**
 - loginVM - Obiekt typu LoginVM, zawierający dane logowania użytkownika.
- **Opis działania:**

- Metoda weryfikuje dane logowania (email i hasło) i jeśli są poprawne, użytkownik zostaje zalogowany. Po udanym logowaniu wysyła zapytania do zewnętrznego API, aby zaktualizować dane o meczach, a następnie przekierowuje użytkownika do strony głównej.
- **Zwracane dane:**
 - **Redirect (301)** - Jeśli logowanie jest udane, użytkownik zostaje przekierowany do strony głównej (Home/Index).
 - **View (Widok)** - Jeśli logowanie się nie uda, użytkownik otrzymuje widok Login z błędami.

Register (GET)

- **HTTP Method:** GET
- **Route:** Account/Register
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Metoda renderuje formularz rejestracji użytkownika.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok Register, w którym użytkownik może wprowadzić dane do rejestracji.



Register (POST)

- **HTTP Method:** POST
- **Route:** Account/Register
- **Parametry:**
 - registerVM - Obiekt typu RegisterVM, zawierający dane rejestracyjne użytkownika.
- **Opis działania:**

- Metoda rejestruje nowego użytkownika, a po udanej rejestracji loguje użytkownika i wysyła zapytania do API zaktualizujące dane o meczach. Następnie użytkownik zostaje przekierowany do strony głównej.
- **Zwracane dane:**
 - **Redirect (301)** - Jeśli rejestracja jest udana, użytkownik zostaje przekierowany do strony głównej (Home/Index).
 - **View (Widok)** - Jeśli rejestracja nie powiedzie się (np. z powodu istniejącego konta), użytkownik otrzymuje widok Register z błędami.

LogOut

- **HTTP Method:** GET
- **Route:** Account/LogOut
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Metoda wylogowuje użytkownika i przekierowuje go do strony logowania.
- **Zwracane dane:**
 - **Redirect (301)** - Po wylogowaniu użytkownik zostaje przekierowany do strony logowania (Account/Login).

2. HomeController

- **Opis:** Kontroler odpowiedzialny za wyświetlanie głównych widoków aplikacji związanych z zakładami, meczami, wynikami oraz rankingami. Zawiera metody do tworzenia zakładów, przeglądania wyników, zarządzania punktami użytkownika oraz interakcje z zewnętrznym API..

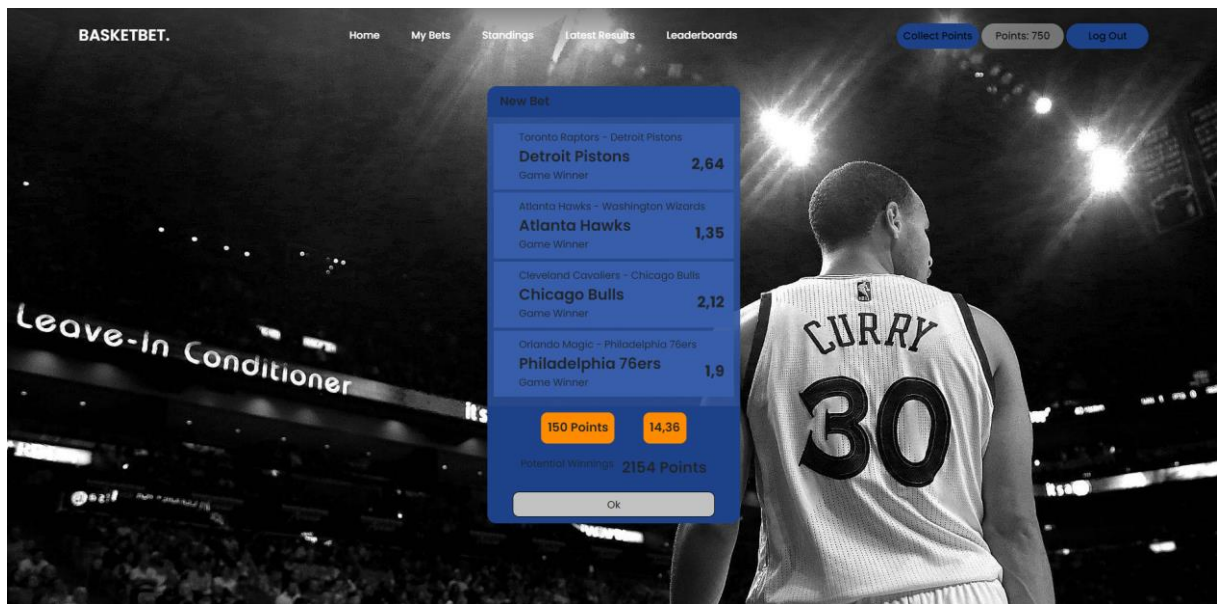
CreateBet

- **HTTP Method:** POST
- **Route:** Home/CreateBet
- **Parametry:**
 - betVM - Obiekt typu BetVM, zawierający dane zakładu (np. drużyna, kwota).
- **Opis działania:**
 - Tworzy nowy zakład na podstawie danych przekazanych w betVM i przypisuje go do zalogowanego użytkownika.
- **Zwracane dane:**
 - **JsonResponse:** { success: true, result: resultId } - Zwraca identyfikator nowo utworzonego zakładu, jeśli zakład został utworzony pomyślnie, lub informację o błędzie, jeśli operacja się nie powiedzie.

NewBet

- **HTTP Method:** GET
- **Route:** Home/NewBet

- **Parametry:**
 - BetId - Identyfikator zakładu, który ma zostać edytowany.
- **Opis działania:**
 - Wyświetla stronę z formularzem edycji zakładu na podstawie podanego identyfikatora.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok NewBet.cshtml, który wyświetla zakład stworzony przez użytkownika.



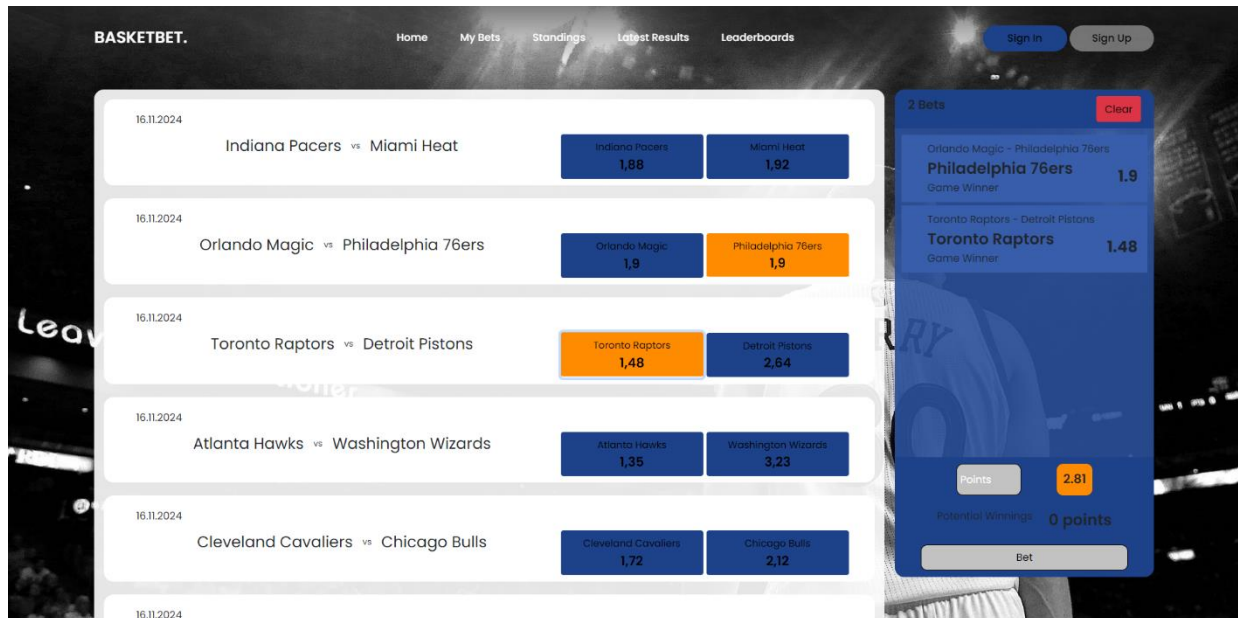
ClaimPoints

- **HTTP Method:** POST
- **Route:** Home/ClaimPoints
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Co godzinę pozwala zalogowanemu użytkownikowi na odebranie punktów.
- **Zwracane dane:**
 - **Ok (200)** - Zwraca status 200, jeśli operacja została wykonana pomyślnie.
 - **Redirect (301)** - Jeśli użytkownik nie jest zalogowany, zostaje przekierowany do strony logowania.

Index

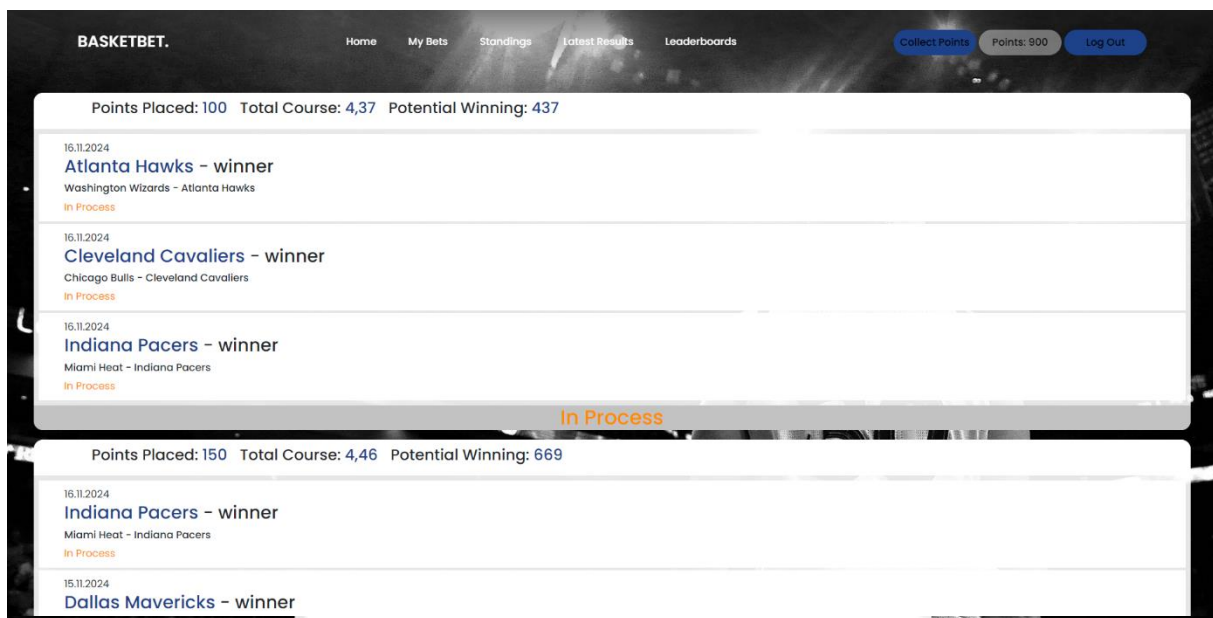
- **HTTP Method:** GET
- **Route:** Home/Index
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**

- Metoda odpowiedzialna za renderowanie strony głównej z ostatnimi meczami i informacjami o punktach użytkownika. Dodatkowo wykonuje zapytanie do API w celu zaktualizowania terminarza meczów.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok główny, wyświetlając najbliższe mecze oraz okienko z naszym zakładem, takie jak czas ostatniego odbioru punktów.



MyBets

- **HTTP Method:** GET
- **Route:** Home/MyBets
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Wyświetla listę zakładów użytkownika, a także aktualizuje wyniki zakładów po zakończeniu meczów (w tle wysyła zapytanie do API w celu zaktualizowania wyników).
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok, na którym użytkownik może zobaczyć swoje zakłady i ich wyniki.



Standings

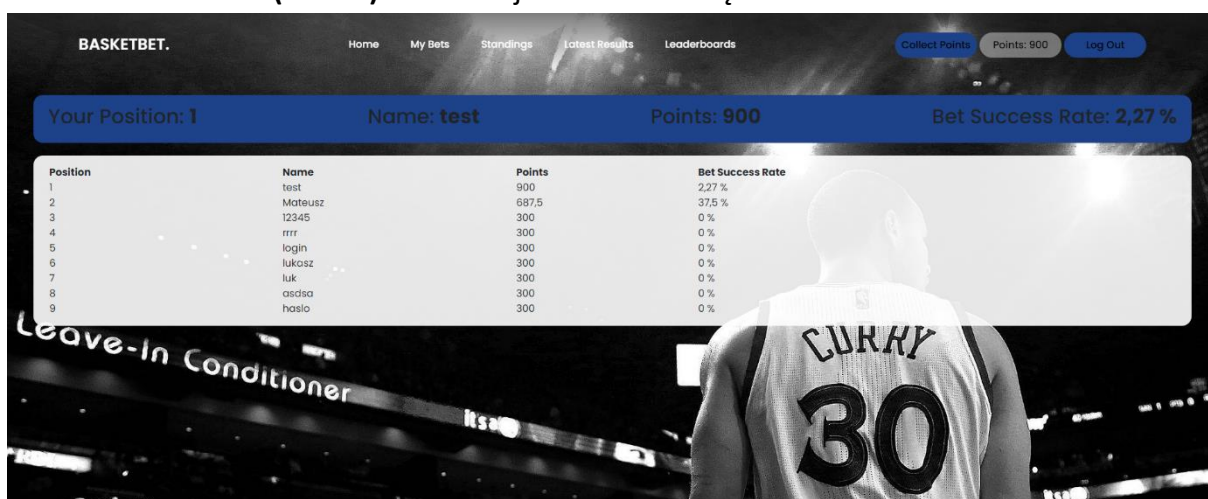
- **HTTP Method:** GET
- **Route:** Home/Standings
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Wyświetla tabelę ligową (ranking drużyn). Przed renderowaniem wysyła zapytanie do API w celu zaktualizowania tabeli.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok z aktualną tabelą ligową.

LatestResults

- **HTTP Method:** GET
- **Route:** Home/LatestResults
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Wyświetla wyniki ostatnich meczów. Przed renderowaniem wysyła zapytanie do API w celu zaktualizowania wyników za poprzedni dzień.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok z najnowszymi wynikami meczów.

Leaderboards

- **HTTP Method:** GET
- **Route:** Home/Leaderboards
- **Parametry:**
 - Brak parametrów.
- **Opis działania:**
 - Wyświetla tablicę liderów (ranking użytkowników) na podstawie punktów i wyników. Jeśli użytkownik jest zalogowany pokazuje też jego statystyki i pozycję w rankingu.
- **Zwracane dane:**
 - **View (Widok)** - Renderuje widok z tabelą liderów.



Position	Name	Points	Bet Success Rate
1	test	900	2,27 %
2	Mateusz	687,5	37,5 %
3	12345	300	0 %
4	rrrr	300	0 %
5	login	300	0 %
6	lukasz	300	0 %
7	luk	300	0 %
8	asdsa	300	0 %
9	haslo	300	0 %

System użytkowników

Nie ma podziału na role wśród użytkowników.

Użytkownicy niezalogowani mogą:

- Przeglądać zbliżające się mecze jednak nie mogą obstawić kuponu
- Przejrzeć tabele NBA
- Zobaczyć wyniki ostatnich meczów
- Zobaczyć ranking użytkowników

Użytkownicy zalogowani mogą:

- wszystko co niezalogowani
- obstawiać mecze na punkty
- co godzinę odbierać 100 punktów
- Przeglądać historię swoich zakładów
- W zakładce rankingu użytkowników widzą również swoje statystyki oraz pozycje w rankingu

W przypadku kiedy niezalogowany użytkownik spróbuje obstawić mecz lub wejść w zakładkę z historią jego zakładów przeniesie go do formularza logowania

Charakterystyka najciekawszych funkcjonalności

Akcje WebAPI używają scrappera, który używając paczki HTMLAgility Pack pobiera kod html strony i za pomocą różnych dostępnych w paczce narzędzi szuka elementów o podanych klasach następnie odczytując ich wartości. Potem dane są zapisywane w bazie danych i w taki sposób aplikacja korzystająca z tej samej bazy danych ma dostęp do rzeczywistych danych.

W głównym widoku aplikacji webowej można wybierać drużyny, na które chce się postawić. W momencie wybrania drużyny zmienia się kolor przycisku i dodaje się ona do pola z naszym zakładem po prawej stronie. Dzięki skryptowi javascript nie możemy obstawić tego samego meczu 2 razy lub np. obstawić w meczu na obie drużyny. Kiedy klikniemy z zaznaczony przycisk to mecz usunie się z listy. To znacznie wpływa na pozytywne odczucia użytkownika z korzystania z aplikacji.

