POLITECHNIKA ŁÓDZKA

Wydział Elektrotechniki, Elektroniki,

Informatyki i Automatyki

|  |
| --- |
| Praca dyplomowa inżynierska **Opracowanie aplikacji wspierającej proces decyzyjny przy zakładach bukmacherskich z wykorzystaniem sztucznych sieci neuronowych** Artur Widziński |

# Nr albumu: 195085

Opiekun pracy:

**Dr inż. Wojciech Zabierowski**

## Łódź, 2018

**Streszczenie**

Blablabla….

**Abstract**

Blablabla…

**Spis treści**

1. **Wstęp**

1. **Aplikacja webowa**

W tym rozdziale opiszę aplikację webową, jej funkcjonalności dla klienta i administratora oraz sposób działania.

* 1. **Opis serwisu**

Strona bukmacherska jest aplikacją skierowaną dla ludzi interesujących się rozgrywkami sportowymi. Wszystkich użytkowników mojej aplikacji można podzielić na kilka ról:

* Administrator
* Zalogowany użytkownik
* Niezalogowany użytkownik

Administrator jest główną osobą, która czuwa nad poprawnością działania serwisu oraz wprowadza dane związane z nadchodzącymi meczami. Zalogowany użytkownik tworzy kupony, za pomocą których zakłada przewidywany rezultat meczu. Większość funkcjonalności jest dostępnych dla wszystkich zalogowanych osób, jednak jako użytkownik niezalogowany mamy również kilka możliwości.

Oto najważniejsze funkcjonalności przeznaczone dla administratora:

* Stworzenie meczu
* Dodanie składu drużyny do meczu
* Rozegranie meczu

Serwis bukmacherski jest stroną, która musi posiadać dużą ilość danych na temat rozgrywanych meczów. Jest to ważne z punktu widzenia klienta, który decydując się, czy postawić zakład na daną drużynę, chętnie sprawdzi np. jak klub się spisywał w przeciągu kilku ostatnich gier, jakich piłkarzy posiada czy też na którym miejscu w lidze się znajduje. Dlatego moja aplikacja oferuje dużą ilość różnych możliwości skierowanych głównie dla klienta. Oto niektóre z nich:

* Rejestracja/logowanie
* Sprawdzenie tabeli ligi
* Sprawdzenie danych drużyny

Z materialnego punktu spojrzenia udostępniłem w aplikacji funkcjonalności odpowiadające za zarządzanie kuponami. Klient może sprawdzić jak wygląda jego sytuacja finansowa, a administrator może sprawdzić stan konta każdego zalogowanego użytkownika. Takie usługi są dostępne w tym celu:

* Doładowanie konta
* Stworzenie kuponu
* Sprawdzenie stanu konta
  1. **Schemat bazy danych**

Aplikacja korzysta z bazy danych MySQL. Zastanawiając się, którą bazę wybrać kierowałem się ilością wpisów w Internecie, które pomagały czy to w rozwijaniu aplikacji, czy to w rozwiązywaniu problemów[[1]](#footnote-2). W niej znajdują się tabele, w których aplikacja przechowuje dane, a także dodaje, modyfikuje i usuwa kolejne rekordy. Występują w niej również tabele, które przechowują dane, ale aplikacja z nich nie korzysta. Związane jest to z tym, że aplikacja będzie jeszcze rozwijana.

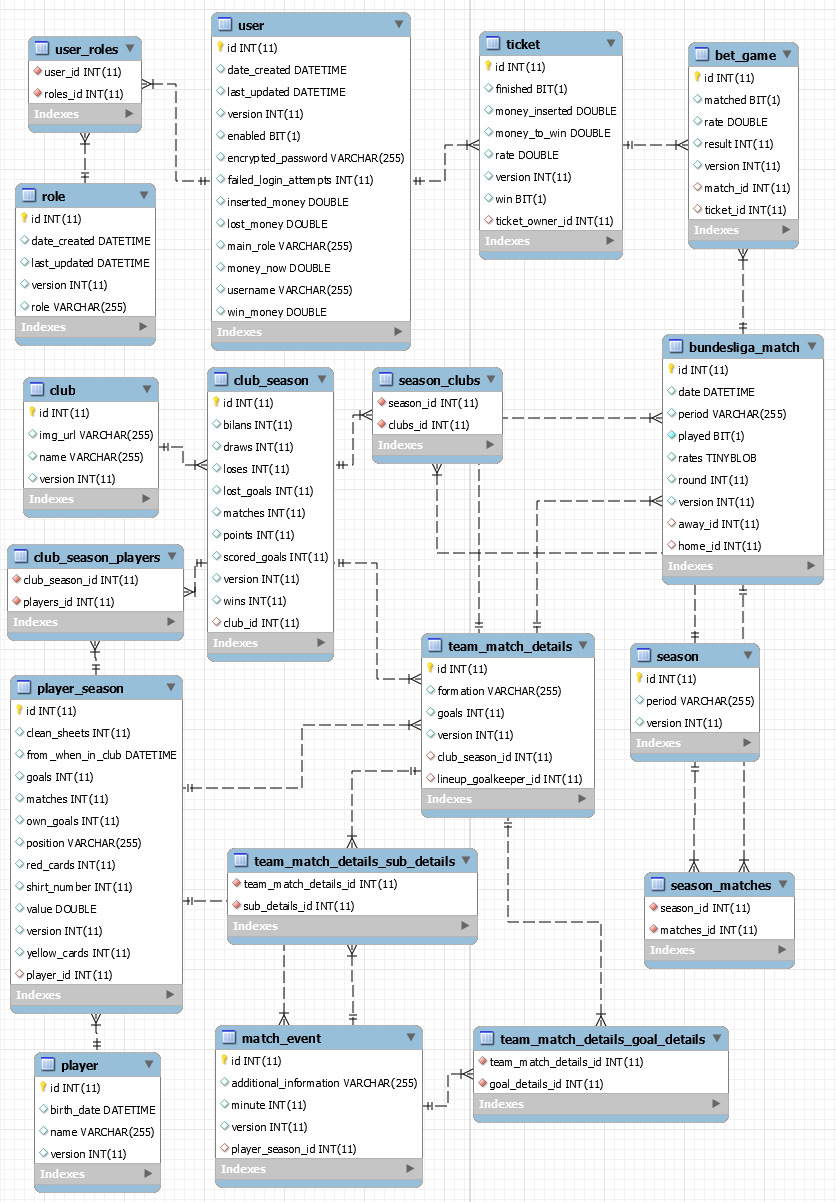
Podstawową tabelą w tej aplikacji jest tabela **USER**. Przechowuje ona dane użytkownika, takie jak nazwa użytkownika, zaszyfrowane hasło, ilość pieniędzy na koncie czy rolę użytkownika. Domyślnie po rejestracji każdy użytkownik posiada rolę *„user”*.

Każdy użytkownik ma przypisaną listę kuponów, które kiedyś stworzył. Trzymane są one w tabeli **TICKET**. Każdy kupon przechowuje informacje między innymi o tym ile użytkownik postawił pieniędzy, ile może wygrać oraz czy dany kupon został już rozstrzygnięty. Dodatkowo do każdego z nich jest przypisanych kolekcja tzw. singli, inaczej mówiąc zakładów pojedynczych, singlowych. Jest to zdarzenie, które dotyczy pojedynczego meczu. W zdarzeniu zapisane są informacje o tym jaki rezultat użytkownik przewidział, i czy się zgadzał z rozegraną już grą. Te dane znajdują się w tabeli **BET\_GAME**.

Najbardziej zewnętrzną tabelą trzymającą informacje o rozgrywkach jest tabela **SEASON**. Każdy rekord przechowywał lata trwania, mecze wtedy rozegrane oraz kluby, które w nim uczestniczyły. Rekordy z tabeli **CLUB\_SEASON** przechowują dane o tym jak drużyny spisywały się w danym sezonie (ilość wygranych, przegranych, remisów, strzelonych i straconych bramek). Dodatkowo do każdej encji z tej tabeli przypisana jest lista zawodników, którzy wtedy należeli do danego klubu. Każdy z nich jest zapisany w tabeli **PLAYER\_SEASON** i dodatkowo zawiera zarówno informacje personalne piłkarza (imię i nazwisko, data urodzenia) jak i dane piłkarskie (numer na koszulce, pozycja, wartość rynkowa, ilość meczów, strzelonych bramek oraz czystych kont bramkarzy).

Tabela **BUNDESLIGA\_MATCH** przechowuje encje wszystkich meczy, które się już odbyły oraz tych, które zostały już zaplanowane. Można tutaj znaleźć datę danej rozgrywki, wyliczone kursy na dany mecz oraz dane składów obu drużyn, które są trzymane w tabeli **TEAM\_MATCH\_DETAILS**. Każdy rekord zawiera informacje dotyczące piłkarzy składu wyjściowego oraz ławki rezerwowych, ilość strzelonych goli, a także zdarzenia w meczu dotyczące drużyny. Te ostatnie znajdziemy w tabeli **MATCH\_EVENT**. Jest tam zapisana minuta wydarzenia, piłkarz, którego ono dotyczy oraz dodatkowa krótka informacja, która dokładnie opisuje co to za rodzaj zdarzenia (np. gol, zmiana, samobój).

Bardziej szczegółowy schemat tej bazy danych przedstawia rysunek na następnej stronie.



*Rysunek 1. Schemat bazy danych*

* 1. **Jakiś rozdział**

1. **Spis rysunków**
2. Schemat bazy danych
3. **Spis tabel**
4. **Bibliografia**
5. Konfiguracja aplikacji napisanej w Springu pod bazę danych MySQL <https://spring.io/guides/gs/accessing-data-mysql/>, październik 2017

1. Konfiguracja aplikacji napisanej w Springu pod bazę danych MySQL: <https://spring.io/guides/gs/accessing-data-mysql/>, październik 2017 [↑](#footnote-ref-2)