W A R S Z A W S K AW Y Ż S Z A S Z K O Ł A I N F O R M A T Y K I

**P R A C A D Y P L O M O W A  
S T U D I A P I E R W S Z E G O S T O P N I A**

**Grzegorz Malarski**

Numer albumu 7063

„Portal społecznościowy na temat zdrowego trybu życia.”

Promotor:

mgr inż. Jerzy Stankiewicz

*Praca spełnia wymagania stawiane pracom dyplomowym na studiach pierwszego stopnia.*

WARSZAWA 2018

Spis treści

[**1 . Wstęp** 2](#_Toc524863275)

[**1.1** **Temat pracy dyplomowej** 2](#_Toc524863276)

[**1.2 Cel i zakres pracy dyplomowej** 3](#_Toc524863277)

[**1.2 Wprowadzenie do problemu** 3](#_Toc524863278)

[2. **Analiza** 3](#_Toc524863279)

[**2.1 Założenia ogólne** 3](#_Toc524863280)

[**2.1.1 Podstawowe pojęcia w projekcie** 4](#_Toc524863281)

[**2.1.2 Szczegóły opisu** 4](#_Toc524863282)

[**2.1.3 Rejestracja do serwisu** 5](#_Toc524863283)

[**2.1.4 Rodzaje treningów dostępnych w serwisie** 5](#_Toc524863284)

[**2.1.5 Algorytm przeliczający maksymalne wyniki** 5](#_Toc524863285)

[**2.2 Aktorzy** 6](#_Toc524863286)

[**2.3 Modelowanie** 6](#_Toc524863287)

[**2.3.1 Diagram przypadków użycia** 6](#_Toc524863288)

[**2.3.2 Diagram Sekwencji** 17](#_Toc524863289)

[**2.5.4 Diagram klas** 18](#_Toc524863290)

[**3.** **Projektowanie** 19](#_Toc524863291)

[**3.1 Projekt bazy danych** 19](#_Toc524863292)

[20](#_Toc524863293)

Wprowadzenie

# 1 . Wstęp

## **Temat pracy dyplomowej**

Tematem pracy dyplomowej jest: „Portal społecznościowy na temat zdrowego trybu życia.”

## **1.2 Cel i zakres pracy dyplomowej**

Celem projektu jest stworzenie aplikacji internetowej dla osób interesujących sięaktywnym trybem życia. Głównym zadaniem serwisu będzie przechowywanie oraz planowanie treningów indywidualnych.

Zakres pracy obejmować będzie:

* Dokonanie analizy funkcjonalnej rozwiązania
* Zaprojektowanie i implementacja bazy danych
* Zaprojektowanie i implementacja mechanizmów wewnętrznych bazy danych
* Zaprojektowanie interfejsu systemu
* Utworzenie aplikacji
* Wprowadzenie testowych danych do aplikacji
* Testowanie aplikacji

## **Wprowadzenie do problemu**

Wiele osób trenujących sport ma problem z przechowywaniem oraz z wyszukiwaniem nowych treningów.Nie każdy może sobie pozwolić na opłacenie trenera personalnego. Celem aplikacji jest pomoc w gromadzeniu oraz znależieniu nowych planów treningowych dostosowanych pod potrzeby indywidualne.

System który będzie spełniał wymagania powinien mieć możliwość:

* zapisywania oraz zarządzania swoimi planami treningowymi
* doboru treningu siłowego na podstawie indywidualnych parametrów
* pomoc w wyborze treningów osobą początkującym
* dopasowanie treningów dla indywidualnego użytkownika
* możliwość zaplanowania konkretnych treningów w kalendarzu
* wyświetlanie artykułów oraz filmów na temat zdrowego trybu życia

Aplikacja powinna pomóc efektywniejszym zarządaniu swoimi planami treningowymi, jak i również pomagać mniej doświadczonym osobą w wyborze odpowiednich ćwiczeń.

# 2. **Analiza**

W tym rozdziale przedstawiono założenia ogólne do aplikacji, wymagania funkcjonalne i pozafunkcjonalne aplikacji. Dodatkowo został przeprowadzony wiwiad środowiskowy z osobami trenującymi jak i równierz trenerami co do funkcjonalności jakie powinna posiadać aplikacja. Zmodelowano rownież projekt przy wykorzystaniu aplikacji drawio.

## **2.1 Założenia ogólne**

Aplikacja ma za zadanie pomóc w gromadzeniu, planowaniu oraz dostosowniu treningów fizycznych.

Po zalogowaniu do aplikacji na konto użytkownika, dostępna będzie możliwość utworzenia notatnika treningowego. Ideą notatnika jest dostarczenie możliwości przechowywania własnych treningów, dodawania swoich notatek oraz rozplanowania poszczególnych treningów w kalendarzu.

W aplikacji będą dostępne gotowe treningi dodane przez trenerów, treningi będą podzielone na sekcje w zależności od poziomy zawansowania. Do każdego treningu będzie notatka opisująca jak ćwiczenie powinno być poprawnie wykonane. Użytkownicy mogą stworzyć też swój własny trening na podstawie gotowych ćwiczeń lub dodać własne ćwiczenie, bedzie on jednak widoczny tylko w ich prywatnym notatniku.

Kolejną dostępna funkcjonalnością jest możliwość doposowania treningów pod indywidaulne osiagnięcia w określonych ćwiczeniach siłowych, po zgromadzeniu odpowiednich danych algorytm przeliczy proporcjonalny ciężar oraz ilość powtórzeń i zaplanuje trening.

Dostępna będzie też sekcja w której zalogowani użytkownicy będą mogli przeglądać artykuły oraz filmiki na temat treningów oraz zdrowego trybu zycia.

Po zalogowaniu do aplikacji na konto trenera, dostępna będzie funkcjonalność dodawania treningów widocznych dla zwykłego użytkownika, każdy trening oraz ćwiczenie z których będzie się składał powinno posiadać odpowiedni opis.

Trener dodatkowo jest odpowiedzialny za obsługę sekcji z artykułami, posiada możliwość dodawania nowych postów oraz modyfikowania wcześniejszych które dodał.

### 2.1.1 Podstawowe pojęcia w projekcie

* Użytkownik – osoba rejestrująca się w aplikacji
* Trener – osoba rejestrująca się w aplikacji w roli trenera
* Administrator – konto posiadające dostęp do wszystkich funkcjonalności serwisu
* Trening – obiekt zawierający zestaw ćwiczeń
* Ćwiczenie – obiekt opisujący ćwiczenie
* Notebook – obiekt zawierający ćwiczenia lub notatki tekstowe
* Kalendarz – obiekt zawierający treningi przypisane do konkretnej daty
* Artykuł – obiekt zawierający artykuł

## 

### 2.1.2 Szczegóły opisu

Użytkownik – potrzeba przechowywania informacji : login, hasło, imię, nazwisko, data urodzenia,lista zapisanych treningów, parametry indywidualne, progres.

Trener – potrzeba przechowywania informacji : login, hasło, imię, nazwisko, specjalizacja, treningi.

Administrator – potrzeba przechowywania informacji : login, hasło.

Trening zwyczajny - potrzeba przechowywania informacji : nazwa, rodzaj, treść.

Trening dopasowany - potrzeba przechowywania informacji : nazwa, rodzaj, ćwiczenie dopasowane indywidualnie

Artykuł - potrzeba przechowywania informacji :tekst zawierający materiał dotyczący zdrowego trybu życia.

2.2 Wymagania dotyczące systemu

Wymagania dotyczące systemu możemy ogólnie podzielić na dwa typy:funkcjonalne i pozafunkcjonalne. Opisują one jak system ma działać, jakie ma mieć funkcjeużytkowe oraz jakie są oczekiwania jakościowe od systemu.

2.2.1 Wymaganie funkcjonalne

* Rejestracja użytkownika
* Wprowadzanie danych personalnych do formularza rejestracyjnego
* Walidacja danych z formularza rejestracyjnego
* Dodawanie kont użytkowników oraz trenerów
* Logowanie
* Utworzenie nowego notebooka treningowego
* Usuwanie notebooka treningowego
* Modyfikacja notebooka treningowego
* Przeglądanie notebooka treningowego
* Dodawanie nowego treningu do notebooka treningowego
* Usuwanie treningu z notebooka treningowego
* Modyfikacja treningu w notebooku treningowym
* Zapisywanie treningów w kalendarzu
* Usuwanie treningów w kalendarzu
* Modyfikowanie treningów w kalendarzu
* Przeglądanie kalendarza
* Dodawanie nowych kategori z treningami
* Modyfikowanie kategori z treningami
* Usuwanie kategori z treningami
* Przeglądanie kategori z treningami
* Dodawanie nowych treningów do poszszczególnych kategori
* Usuwanie treningów w poszszczególnych kategoriach
* Modyfikowanie treningów w poszszczególnych kategoriach
* Przeglądanie treningów w poszszczególnych kategoriach
* Dodawanie treningów z kategorii do indywidualnego notebooka
* Dodawanie nowego ćwiczenia do treningu
* Usuwanie ćwiczenia z treningu
* Modyfikowanie ćwiczenia w treningu
* Dodawanie artykułów do bloga
* Usuwanie artykułów z bloga
* Modyfikowanie artykułów z bloga
* Przeglądanie artykułów na blogu
* Wprowadzanie przez uzytkownika parametrów fizycznych
* Modyfikacja przez uzytkownika parametrów fizycznych
* Generowania treningu siłowego na podstawie indywidualnych parametrów

2.2.2 Wymagania Pozafunkcjonalne

Wymagania pozafunkcjonalne czyli inaczej jakościowe podobnie jakfunkcjonalne zostały określone na podstawie analizy wstępnej.

|  |  |
| --- | --- |
| Identyfikator | WPF1 |
| Opis | Aplikacja będzie działać w oparciu o bazę danych MySQL 8.0 lub wyższą |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF2 |
| Opis | Aplikacja będzie tworzona w środowisku Spring Framework 5 |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF3 |
| Opis | Aplikacja będzie działać na przeglądarkach internetowych nie starszych wersji niż Mozilla 61, Chrome 68. |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF4 |
| Opis | Aplikacja będzie przechowywać hasła do systemu w postaci zaszyfrowanej. |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF5 |
| Opis | Aplikacja musi działać stabilnie, nawet przy dużej liczbie użytkowników |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF6 |
| Opis | Aplikacja musi być responsywna, działać poprawnie na urządzaniach o różnej rodzielczości ekranu |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF7 |
| Opis | Wszyscy użytkownicy mogą zgłaszać błędy na dedykowany adres email |
| Priorytet | Średni |

|  |  |
| --- | --- |
| Identyfikator | WPF8 |
| Opis | Czas reakcji aplikacji na zapytania ze strony użytkowników powinien być przyjazny dla użytkownika |
| Priorytet | Wysoki |

|  |  |
| --- | --- |
| Identyfikator | WPF9 |
| Opis | Poruszanie się po aplikacji powinno być intuicyjne dla użytkownika |
| Priorytet | Wysoki |

2.3 Aktorzy

Po analizie wstępnej zdefiniowano że system będzie obsługiwany przez 3 typy aktorów biorących udział w działaniu programu:

* Administrator
* Trener
* Użytkownik

2.4 Modelowanie

W podrozdziale przedstawiono projekt graficzny systemu. Do jego wykonaniawykorzystano i przedstawiono poniżej wybrane (ze względu na ograniczenia objętościowePracy Dyplomowej) diagramy języka UML:

### **2.1.3 Rejestracja do serwisu**

Do portalu będzie mógł zarejestrować się zarówno trener jaki i osoba trenująca. Te dwa konta różnią się dostępem do poszczególnych funkcji serwisu.

Trener ma możliwość dodawania nowych treningów oraz artykułów. Osoba która chce zarejestrować się jako trener zobowiązana jest przesłać drogą mailowa potwierdzenie uprawnień do wykonywania tego zawodu.

Zwykły użytkownik ma dostęp do treningów, może komentować artykuły. Posiada też dostęp do statystyk jego progresu,

### **2.1.4 Rodzaje treningów dostępnych w serwisie**

Trening dopasowany

Każdy dzień treningowy może być uzależniony od wyboru trenującego, będzie on mógł podążać za planem rozpisanym na pewien okres lub też danego dnia zmienić rodzaj treningu, wybrać inna partię mięśniowa lub sam typ np. wykonać trening wydolnościowy lub typu Crossfit zamiast siłowego.

Po ukończeniu takiego treningu wymagane jest aby potwierdził czy zrealizował w pełni swój plan.

Trening zwyczajny

Będą to gotowe zestawy ćwiczeń dla danej kategorii lub partii mięśniowej, nie będzie tu ćwiczeń spersonalizowanych.

Osoba zarejestrowana na stronie chcąc wybrać sobie trening będzie mogła przefiltrować listę na podstawie konkretnego ćwiczenia.

### **2.1.5 Algorytm przeliczający maksymalne wyniki**

Algorytm ten będzie miał za zadanie wyliczyć na podstawie kilku zmiennych, z jakim obciążeniem ma być wykonane dane ćwiczenie oraz ile powtórzeń ma być zrobione.

Wpływ na wynik wyliczeń będą miały parametry indywidualne oraz to czy trenujący wykonał swój plan danego dnia w stu procentach.

## **2.2 Aktorzy**

Po analizie wstępnej zdefiniowano że portal będzie obsługiwany przez 3 typów aktorów biorących udział w działaniu programu:

• Administrator

• Użytkownik

• Trener

## 2.3 Modelowanie

W podrozdziale przedstawiono projekt graficzny systemu. Do jego wykonania wykorzystano i przedstawiono poniżej wybrane (ze względu na ograniczenia objętościowe Pracy Dyplomowej) diagramy języka UML:

* 4 diagramy przypadków użycia ze szczegółowym opisem każdego przypadku w postaci 22 tabel,
* 2 diagramy sekwencji,
* 2 diagramy czynności,
* 1 diagram hierarchii funkcji
* 1 diagram klas

Do tworzenia diagramów został użyty program Microsoft Visio.

### 2.3.1 Diagram przypadków użycia

Wprowadzanie słowników i definicji



Rysunek 1 Przypadki użycia; źródło własne



Rysunek 2 Przypadki użycia; źródło własne

Rysunek 3 Przypadki użycia; źródło własne

Tabela 1 Przypadki Użycia, PU1: Logowanie

|  |
| --- |
| **Przypadek użycia:** PU1:Logowanie |
| **Aktorzy:** Administrator, Trener, Użytkownik |
| **Opis:** Przypadek użycia dotyczy logowania do systemu |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlone jest okno logowania zawierające pola:  * Login * Hasło   Dostępne przyciski:   * Zaloguj * Anuluj  1. System czeka na aktywność użytkownika |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez wpisanie danych logowania |
| **Scenariusz główny** |
| 1. Użytkownik wpisuje login i hasło do systemu i wybiera opcje *Zaloguj* 2. System sprawdza czy login i hasło do siebie pasują 3. System potwierdza poprawność danych logowania przez wyświetlenie ekranu głównego interfejsu użytkownika   4. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 1.A Użytkownik podał niewłaściwe login lub hasło  1.A.1 System podaje informację o błędnych danych logowania i pozostałą ilość prób wpisania  1.A.2 System wyświetla okno logowania  1.B Użytkownik podał niewłaściwe dane logowania powyżej dozwolonej ilości  1.B.1 System podaje informację o wykorzystanej ilości prób i kończy działanie  1.B.2 System kończy działanie |

Tabela 2 Przypadki użycia- PU2 : Dodawanie parametrów użytkownika

|  |
| --- |
| **Przypadek użycia:** PU2 : Dodawanie parametrów użytkownika |
| **Aktorzy:** Użytkownik |
| **Opis:** Przypadek użycia dotyczy dodawania parametrów nowego użytkownika |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik jest zalogowany do systemu |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. System wyświetla zakładkę na której można zobaczyć listę wymaganych parametrów 2. Na formularzu przyciski:  * dodaj * edytuj * usuń  1. Użytkownik wybiera opcję wprowadzenia nowych parametrów 2. System umożliwia wpisanie nowych parametrów 3. Użytkownik wprowadza wymagane dane 4. Użytkownik akceptuje wprowadzone dane przez wybranie opcji zapisu 5. System zapisuje dane 6. System wyświetla informację o zapisanych danych 7. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 3.A Użytkownik chce anulować wprowadzanie nowych parametrów  3.A.1 Użytkownik wybiera opcję anuluj  3.A.2 System wyświetla główne menu programu |

**Tabela 3 Przypadki użycia – PU03:** Usuwanie parametrów użytkownik

|  |
| --- |
| **Przypadek użycia:** PU3 : Usuwanie parametrów użytkownika |
| **Aktorzy:** Administrator, Użytkownik |
| **Opis:** Przypadek użycia dotyczy usuwania parametrów użytkownika |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik jest zalogowany do systemu z uprawnieniami administratora   Lub zwykłego użytkownika |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. System wyświetla zakładkę na której można zobaczyć listę parametrów 2. Na formularzu przyciski:  * nowy * edytuj * usuń  1. Użytkownik wybiera opcję usuwania 2. System wyświetla potwierdzenie usunięcia 3. Użytkownik akceptuje usunięcie   6. System usuwa dane  7. System wyświetla informację o usunięciu danych  8. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 4.A Użytkownik chce anulować usuwanie parametru  4.A.1 Użytkownik nie potwierdza usunięcia parametru  4.A.2 System wyświetla główne menu programu |

Tabela 4 Przypadki użycia- PU4:Modyfikacja parametrów użytkownika

|  |
| --- |
| **Przypadek użycia:** PU4:Modyfikacja parametrów użytkownika |
| **Aktorzy:** Administrator, Użytkownik |
| **Opis:** Przypadek użycia dotyczy modyfikacji danych wybranego użytkownika |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik jest zalogowany do systemu z uprawnieniami użytkownika lub administratora |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. System wyświetla zakładkę na której można zobaczyć listę pracowników 2. Na formularzu przyciski:  * nowy * edytuj * usuń   Formularz zawiera również pola z danymi pracownika:   * imię * nazwisko * data urodzenia * waga * wzrost * cel treningowy  1. Użytkownik wybiera opcję wprowadzenia edycji 2. System wyświetla formularz z parametrami użytkownika które były już wprowadzone i umożliwia ich edycję. 3. Użytkownik wprowadza nowe dane i wybiera opcję zapisu. 4. System zapisuje dane 5. System wyświetla informację o zapisaniu danych 6. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 4.A Użytkownik chce anulować modyfikację parametrów użytkownika  4.A.1 Użytkownik wybiera opcję anuluj  4.A.2 System wyświetla główne menu programu |

Tabela 5 Przypadki użycia – PU05:Dodawanie nowych treningów

|  |
| --- |
| **Przypadek użycia:** PU05:Dodawanie typów obowiązków |
| **Aktorzy:** Trener |
| **Opis:** Przypadek użycia dotyczy dodawania nowych treningów |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik jest zalogowany do systemu z uprawnieniami trenera |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. Użytkownik wybiera opcję wprowadzenia nowego treningu 2. Na ekranie pojawia się okno z polami:  * nazwa, * ćwiczenie i powtórzenie, * opcjonalnie czas na wykonanie ćwiczenia, * opis  1. Oprócz tego pojawiają się też przycisk Zapisz oraz Anuluj 2. Użytkownik wprowadza wymagane dane 3. Użytkownik akceptuje wprowadzone dane przez wciśnięcie przycisku Zapisz 4. System zapisuje dane 5. System wyświetla informację o zapisanych danych 6. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 4.A Użytkownik chce anulować wprowadzanie nowego trening  4.A.1 Administrator wybiera przycisk anuluj  4.A.2 System wyświetla główne menu programu |

Tabela 6 Przypadki użycia – PU06:Usuwanie treningów

|  |
| --- |
| **Przypadek użycia:** PU06: Usuwanie treningów |
| **Aktorzy:** Trener, Administrator |
| **Opis:** Przypadek użycia dotyczy usuwania treningów |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik jest zalogowany do systemu z uprawnieniami trenera lub administratora |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. Użytkownik zaznacza na liście wybrany trening 2. System wyświetla przycisk usuń 3. Użytkownik wciska przycisk usuń 4. System wyświetla potwierdzenie usunięcia 5. Użytkownik akceptuje usunięcie treningu 6. System usuwa dane 7. System wyświetla informację o usunięciu danych 8. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 4.A Użytkownik chce anulować usuwanie treningu  4.A.1 Użytkownik wybiera opcję anuluj  4.A.2 System wyświetla główne menu programu |

Tabela 7 Przypadek użycia- PU07:Modyfikacja treningów

|  |
| --- |
| **Przypadek użycia:** PU07:Modyfikacja treningów |
| **Aktorzy:** Trener, Administrator |
| **Opis:** Przypadek użycia dotyczy modyfikacji treningów |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik jest zalogowany do systemu z uprawnieniami trenera lub administratora |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. Użytkownik wybiera trening 2. System wyświetla dane treningu oraz przycisk edytuj 3. Użytkownik wybiera opcję edytuj 4. System umożliwia edycję danych i wyświetla przycisk zapisz 5. Użytkownik wprowadza zmiany w wybranych polach i naciska przycisk zapisz 6. System zapisuje dane 7. System wyświetla informację o zapisanych danych 8. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 4.A Użytkownik chce anulować modyfikację trening  4.A.1 Użytkownik wybiera opcję anuluj  4.A.2 System wyświetla główne menu programu |

### **2.3.2 Diagram Sekwencji**



### **2.5.4 Diagram klas**

Na podstawie założeń do systemu poniżej przedstawiono strukturę systemu w postaci diagramu klas zawierającą strukturę klas systemu oraz związki między nimi.



# **Projektowanie**

Rozdział zawiera projekt bazy danych ze szczegółowym opisem poszczególnych tabel. Ponadto rozdział zawiera wykaz i opis zastosowanych mechanizmów systemu zarządzania bazą danych takich jak procedury składowane czy procedury wyzwalane. Dodatkowo umieszczony został model fizyczny bazy danych, który w sposób graficzny pokazuje powiązania referencyjne między tabelami.

## **3.1** **Projekt bazy danych**

# 

## **3.2 Tabele bazy danych aplikacji**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | UserAccount | | | |
| **Opis** | Tabela przechowuje podstawowe dane użytkownika | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Imię | nvarchar | 50 | NIE | Imię użytkownika |
| Nazwisko | nvarchar | 50 | NIE | Nazwisko użytkownika |
| Email | nvarchar | 50 | NIE | Adres email użytkownika |
| IdParameters | int | - | TAK | Identyfiaktor do tabeli z parametrami |
| IdUserCredentials | int | - | NIE | Identyfikator do tabeli z danymi uwieżytelniającymi |
| IdUserTrening | int | - | TAK | Identyfikator do tabeli z trenigami użytkownika |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | AdminAccount | | | |
| **Opis** | Tabela przechowuje dane uwierzytelniające konta administratora | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 50 | NIE | Imię |
| Surname | nvarchar | 50 | NIE | Nazwisko |
| IdUserCredentials | int | - | NIE | Identyfikator do tabeli z danymi uwieżytelniającymi |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | TrainerAccount | | | |
| **Opis** | Tabela przechowuje dane uwierzytelniające konta trenera | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 50 | NIE | Imię |
| Surname | nvarchar | 50 | NIE | Nazwisko |
| Specialization | nvarchar | 50 | NIE | Specjalizacja trenera |
| Roles\_id | Int | - | NIE | Identyfikator roli użytkownika |
| IdUserCredentials | int | - | NIE | Identyfikator do tabeli z danymi uwieżytelniającymi |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | UserCredentials | | | |
| **Opis** | Tabela przechowuje dane uwierzytelniające użytkownika | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Login | nvarchar | 50 | NIE | Identyfikator użytkownika. Klucz obcy |
| Hasło | nvarchar | 50 | NIE | Nazwa użytkownika |
| IdRole | int | - | NIE | Identyfikator roli użytkownika |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | UserParameters | | | |
| **Opis** | Tabela przechowuje dane szczegółowe użytkownika | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| DateOfBirth | date | - | NIE | Data urodzenia |
| Weight | double | - | NIE | Waga |
| Height | int | - | NIE | Wzrost |
| IdTreningType | int | - | TAK | Identyfiaktor typu treningu który może wybrać użytkownik |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Roles | | | |
| **Opis** | Tabela przechowuje informacje o roli poszczególnego użytkownika w aplikacji | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 45 | NIE | Nazwa roli w aplikacji |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | UserTrening | | | |
| **Opis** | Tabela przechowuje trening który użytkownik doda do swojej listy | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| IdTreningExercises | int | - | NIE | Identyfikator konkretnego treningu |
| PercentCompleted | int | - | NIE | Liczba zawierająca procent ukończenia treningu |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | TreningTypes | | | |
| **Opis** | Tabela przechowuje nazwy kategori treningów | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | Nvarchar | 50 | NIE | Nazwa konkretnego typu treningu |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | NormalTrening | | | |
| **Opis** | Tabela przechowuje informacje o zwyczajnym programie treningowym | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 50 | NIE | Nazwa treningu |
| IdTreningTypes | int | - | NIE | Identyfikator kategori do jakiej należy trening |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | PersonalizedTrening | | | |
| **Opis** | Tabela przechowuje informacje o spersonalizowanym programie treningowym | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 50 | NIE | Nazwa treningu |
| IdTreningTypes | int | - | NIE | Identyfikator kategori do jakiej należy trening |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | TreningExercises | | | |
| **Opis** | Tabela przechowuje ćwiczenia które składają się na jeden konkretny trening | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| ExerciseId | int | - | NIE | Identyfikator konkretnego treningu |
| IdNormalTrening | Int | - | TAK | Identyfikator treningu personalnego do ktorego należy |
| IdPersonalizedTrening | int | - | TAK | Identyfikator treningu personalnego do ktorego należy |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Exercise | | | |
| **Opis** | Tabela przechowuje informacje o konkretnym ćwiczeniu | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 50 | NIE | Nazwa ćwiczenia |
| Reps | Int | - | TAK | Ilość powtórzeń ćwiczenia w seri |
| Time | int | - | TAK | Czas przez jaki można wykonywać ćwiczenie |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | UserExerciseMax | | | |
| **Opis** | Tabela przechowuje informacje o maksymalnych wynikach użytkownika w konkretnym ćwiczeniu | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| UserAccountId | int | - | NIE | Identyfikator użytkownika |
| Exercise\_id | Int | - | NIE | Identyfikator ćwiczenia |
| UserMax | double | - | NIE | Maksymalne obciążenie z jakim użytkownik wykonywał ćwiczenie |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Artykuł | | | |
| **Opis** | Tabela przechowuje informacje o konkretnym ćwiczeniu | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Content | longtext | - | NIE | Zawartość artykułu |
| AddedDate | date | - | NIE | Data dodania |
| IdAuthor | int | - | NIE | Identyfikator autora |