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](#_Toc4703569)

[**1.1** **Temat pracy dyplomowej** 2](#_Toc4703570)

[**1.2 Cel i zakres pracy dyplomowej** 3](#_Toc4703571)

[**1.2** **Wprowadzenie do problemu** 3](#_Toc4703572)

[2. **Analiza** 4](#_Toc4703573)

[**2.1 Założenia ogólne** 4](#_Toc4703574)

[2.1.1 Podstawowe pojęcia w projekcie 5](#_Toc4703575)

[2.1.2 Szczegóły opisu 5](#_Toc4703576)

[**2.5 Diagram klas** 25](#_Toc4703577)

[**3.Projektowanie** 26](#_Toc4703578)

[**3.1 Projekt bazy danych** 26](#_Toc4703579)

[3.1.1 Tabele bazy danych systemu 27](#_Toc4703580)

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
* **Notebook –** obiekt zawierający treningi oraz notatki osobiste
* 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.

Tabela 1 Wymagania funkcjonalne nr 1

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

Tabela 2 Wymagania funkcjonalne nr 2

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

Tabela 3 Wymagania funkcjonalne nr 3

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

Tabela 4 Wymagania funkcjonalne nr 4

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

Tabela 5 Wymagania funkcjonalne nr 5

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

Tabela 6 Wymagania funkcjonalne nr 6

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

Tabela 7 Wymagania funkcjonalne nr 7

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

Tabela 8 Wymagania funkcjonalne nr 8

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

Tabela 9 Wymagania funkcjonalne nr 9

|  |  |
| --- | --- |
| 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:

2.4.1 Diagramy przypadków użycia

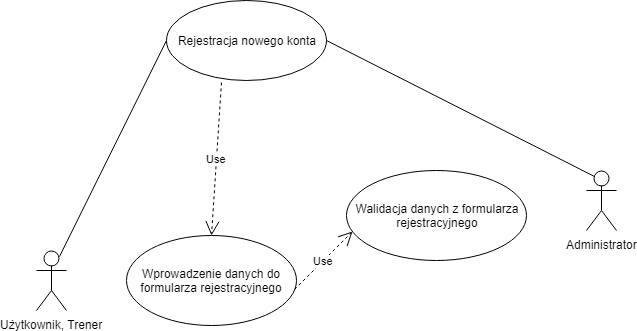


Diagram 1. Przypadki użycia; żródło: opracowanie własne

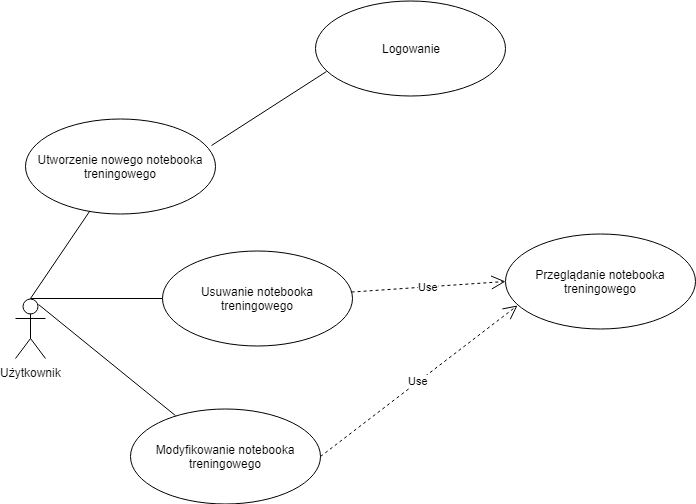


Diagram 2. Przypadki użycia; żródło: opracowanie własne

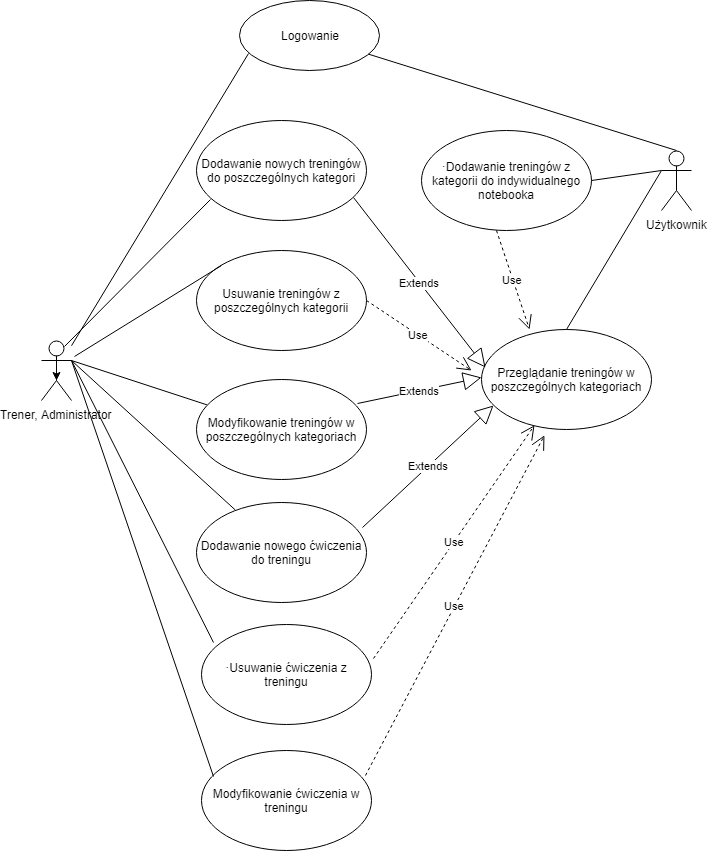


Diagram 3. Przypadki użycia; żródło: opracowanie własne

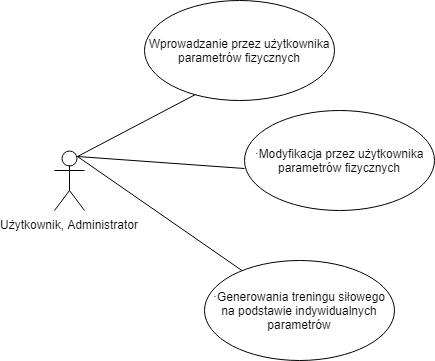


Diagram 4. Przypadki użycia; żródło: opracowanie własne

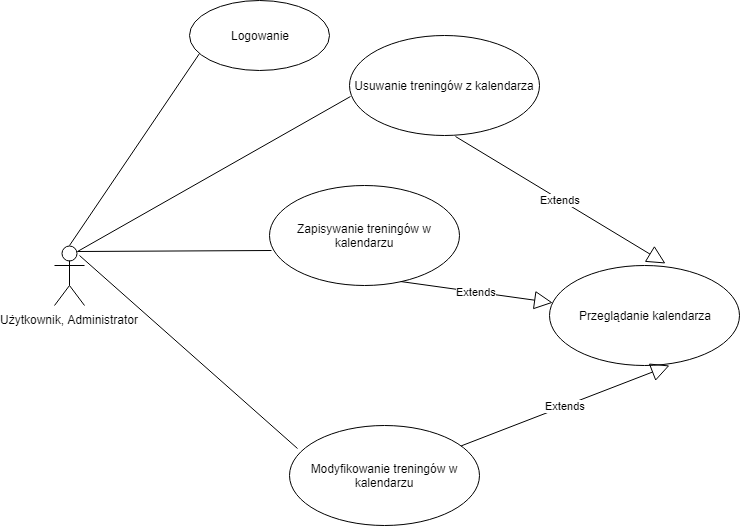


Diagram 5. Przypadki użycia; żródło: opracowanie własne

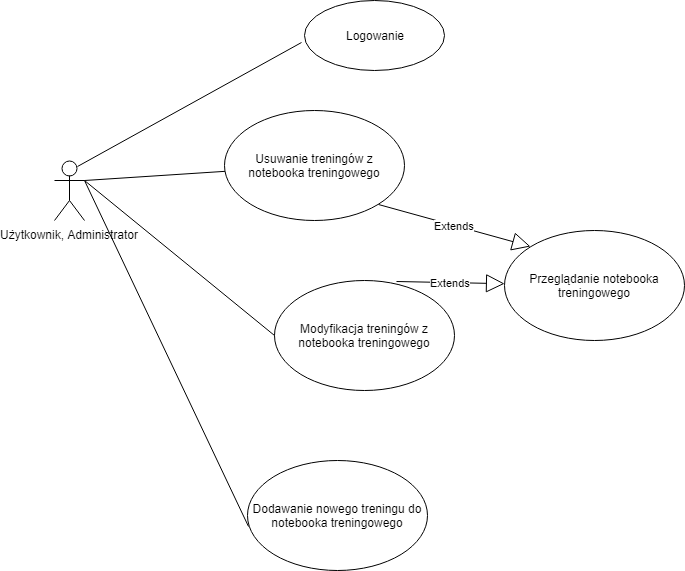


Diagram 6. Przypadki użycia; żródło: opracowanie własne

Tabela 10 Przypadki Użycia, PU1: Rejestracja nowego użytkownika

|  |
| --- |
| **Przypadek użycia:** PU2: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 |

|  |
| --- |
| **Przypadek użycia:** PU1:Rejestracja nowego konta |
| **Aktorzy:** Administrator, Trener, Użytkownik |
| **Opis:** Przypadek użycia dotyczy rejestracji nowego konta w aplikacji |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest formularz rejestracyjny zawierające pola:  * Imię * Nazwisko * Adres email * Login * Hasło   Dostępne przyciski:   * Zarejestruj * Anuluj  1. System czeka na aktywność użytkownika |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez wpisanie danych rejestracyjnych |
| **Scenariusz główny** |
| 1. Użytkownik wpisuje dane do formularza i wybiera opcje *Zarejestruj* 2. System sprawdza dane są poprawnie podane 3. System potwierdza poprawność danych rejestracyjnych przez wyświetlenie odpowiedniego komunikatu   4. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 1.A Użytkownik błędnie wypełnił formularz rejestracyjny  1.A.1 System podaje informację o tym jakie pole zostało źle uzupełnione  1.A.2 System wyświetla formularz rejestracyjny  1.B System kończy działanie |

Tabela 11

Przypadki Użycia, PU2:Logowanie

Tabela 12 Przypadki Użycia, PU3: Dodawanie nowego notebooka treningowego

|  |
| --- |
| **Przypadek użycia:** PU3 : Dodawanie nowego notebooka treningowego |
| **Aktorzy:** Użytkownik, Administrator |
| **Opis:** Przypadek użycia dotyczy dodawania notebooka treningowego |
| **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 utworzonych przez niego notebooków 2. Dostępnę są przyciski:  * dodaj * edytuj * usuń  1. Użytkownik wybiera opcję dodania nowego notebooka 2. System umożliwia wpisanie wymaganych danych 3. Użytkownik wprowadza wymagane dane 4. Użytkownik akceptuje wprowadzone dane przez wybranie opcji dodaj 5. System zapisuje dane 6. System wyświetla informację o zapisanych danych 7. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 3.A Użytkownik chce anulować tworzenie nowego notebooka  3.A.1 Użytkownik wybiera opcję anuluj  3.A.2 System wyświetla panel głowny zalogowanego użytkownika |

|  |
| --- |
| **Przypadek użycia:** PU4 : Dodawanie nowego treningu do notebooka |
| **Aktorzy:** Administrator, Użytkownik |
| **Opis:** Przypadek użycia dotyczy dodawanie nowego treningu do notebooka |
| **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 użytkownika |
| **Zdarzenie inicjujące** |
| Użytkownik rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. System wyświetla zakładkę na której można wyświetlić wybrany trening 2. Użytkownik zaznacza kliknięciem myszki wybrany trening 3. Użytkownik wybiera jedną z dostępnych opcji  * dodaj * anuluj  1. System wyświetla komunikat potwierdzający dodanie treningu |
| **Rozszerzenia** |
| 4.A Użytkownik chce anulować dodawanie treningu  4.A.1 Użytkownik nie potwierdza dadania treningu  4.A.2 System wyświetla panel głowny zalogowanego użytkownika |

Tabela 13 Przypadki Użycia, PU4: Dodawanie nowego treningu do notebooka

Tabela 14 Przypadki Użycia, PU5: Dodawanie nowego treningu do kategorii

|  |
| --- |
| **Przypadek użycia:** PU5 : Dodawanie nowego treningu do kategorii |
| **Aktorzy:**Trener, Administrator |
| **Opis:** Przypadek użycia dotyczy dodawania noweg treningu do kategorii |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs trenera   1. System czeka na aktywność trenera 2. Trener jest zalogowany do systemu |
| **Zdarzenie inicjujące** |
| Trener rozpoczyna akcję poprzez żądanie za pośrednictwem GUI |
| **Scenariusz główny** |
| 1. System wyświetla zakładkę na której można zobaczyć listę dodanych przez niego treningów 2. Dostępnę są przyciski:  * dodaj * edytuj * usuń  1. Trener wybiera opcję dodania nowego treningu 2. System umożliwia wpisanie wymaganych danych 3. Trener wprowadza wymagane dane 4. Trener akceptuje wprowadzone dane przez wybranie opcji dodaj 5. System zapisuje dane 6. System wyświetla informację o zapisanych danych 7. Przypadek użycia się kończy |
| **Rozszerzenia** |
| 3.A Trener chce anulować tworzenie nowego trening  3.A.1 Trener wybiera opcję anuluj  3.A.2 System wyświetla panel główny zalogowanego trenera |

Tabela 15 Przypadki Użycia, PU6: Modyfikacja parametrów użytkownika

|  |
| --- |
| **Przypadek użycia:** PU6: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  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 16 Przypadki Użycia, PU7: Usuwanie treningów

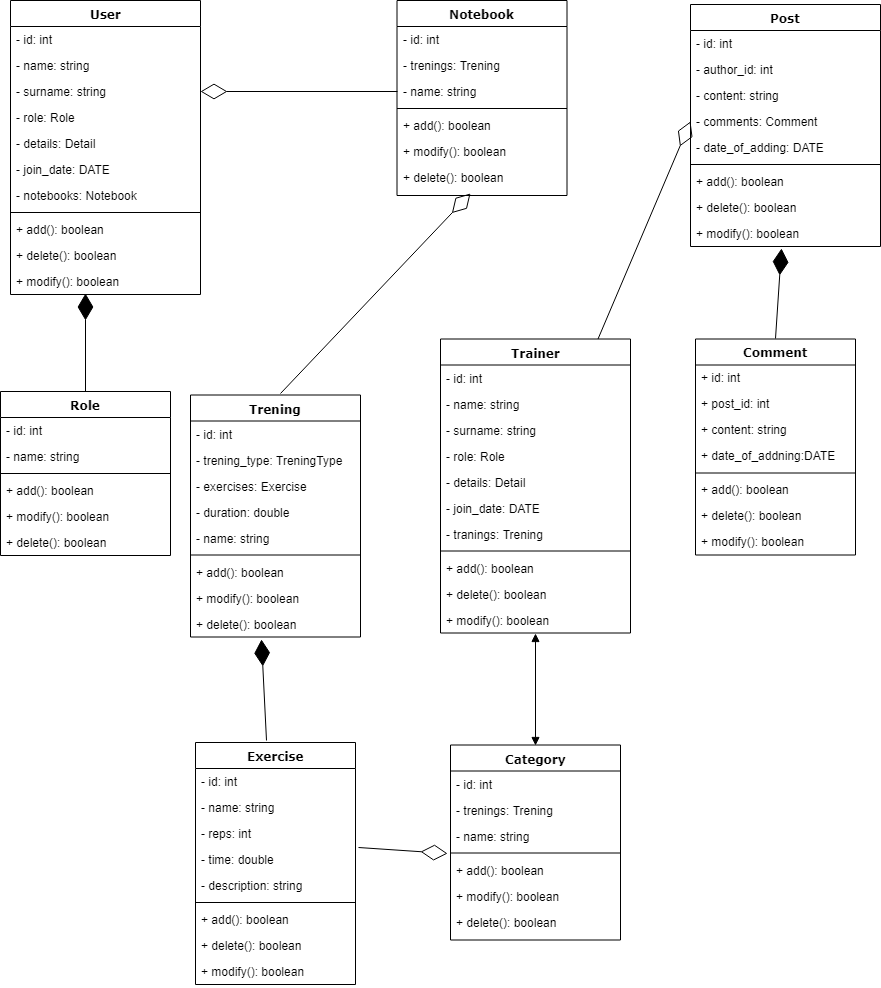
|  |
| --- |
| **Przypadek użycia:** PU07: Usuwanie treningów |
| **Aktorzy:** Trener, Administrator |
| **Opis:** Przypadek użycia dotyczy usuwania treningów |
| **Warunki początkowe** |
| 1. Na ekranie wyświetlony jest graficzny interfejs użytkownika 2. System czeka na aktywność użytkownika 3. Użytkownik 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 17 Przypadki Użycia, PU8: Modyfikacja parametrów użytkownika

|  |
| --- |
| **Przypadek użycia:** PU08: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 |

### **2.5 Diagram klas**

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



# **3.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**

# 

# 3.1.1 Tabele bazy danych systemu

Tabela 18 – Tabela konto użytkowników aplikacji-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | User | | | |
| **Opis** | Tabela przechowuje podstawowe dane użytkownika | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Imię | varchar | 50 | NIE | Imię użytkownika |
| Nazwisko | varchar | 50 | NIE | Nazwisko użytkownika |
| Email | varchar | 50 | NIE | Adres email użytkownika |
| Hasło | varchar | 32 | NIE | Hsło do aplikacji |
| Data dołączenia | date | - | NIE | Data utworzenia konta w aplikacji |
| Id roli użytkownika | int | - | NIE | Identyfikator danych użytkownika |
| Id danych użytkownika | int | - | TAK | Identyfikator do tabeli z szczegółowymi danymi użytkownika |

Tabela 19 – Tabela z danymi szczegółowymi użytkowników aplikacji-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | App\_User\_Details | | | |
| **Opis** | Tabela przechowuje szczegółowe dane żytkownika | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Waga | doulbe | - | TAK | Waga |
| Wzrost | double | 50 | TAK | Nazwisko |
| Specjalizacja | varchar | 100 | NIE | Identyfikator do tabeli z danymi uwieżytelniającymi |

Tabela 20 – Tabela role użytkowników aplikacji-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | App\_Roles | | | |
| **Opis** | Tabela przechowuje nazwy roli dla kont w aplikacji | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | nvarchar | 50 | NIE | Nazwa roli w aplikacji |

Tabela 21– Tabela notebooki utworzone przez użytkowników -źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Notebooks | | | |
| **Opis** | Tabela przechowuje informacje o zawartości notebooka treningowego | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Nazwa | nvarchar | 50 | NIE | Nazwa notebooka |
| Id\_Treningów | int | - | TAK | Nazwa użytkownika |
| Id\_użytkownika | int | - | NIE | Identyfikator użytkownika który utworzył notebook |

Tabela 22 – Tabela treningi notebooków-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Notebooks\_Trenings | | | |
| **Opis** | Tabela przechowuje identyfikatory treningów dla poczególnego notebooka | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Notebook\_id | int | - | NIE | Identyfikator notebooka |
| Trening\_id | int | - | NIE | Identyfikator treningu |

Tabela 23 – Tabela z treningami-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Trenings | | | |
| **Opis** | Tabela przechowuje dane szczegółowe użytkownika | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | varchar(50) | - | NIE | Nazwa treningu |
| Czas | double | - | TAK | Opcjonalnie czas na wykonanie treningu |
| Opis | Long text | - | NIE | Opis treningu |
| Id\_kategorii | Int | - | NIE | Identyfikator do nazwy kategori treningu |
| Id\_trening\_type | int | - | TAK | Identyfiaktor typu treningu ze słownika |

Tabela 24 – Tabela z nazwami typów treningów-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Trening\_Types | | | |
| **Opis** | Tabela przechowuje nazwy typów treningów | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id | int | - | NIE | Klucz główny |
| Name | Nvarchar(50) | 50 | NIE | Nazwa konkretnego typu treningu |
| Opis | Long Text | - | NIE | Opis typu treningu |

Tabela 25 – Tabela z nazwami typów treningów-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Trening\_Categories | | | |
| **Opis** | Tabela przechowuje informacja o do jakiej kategori należy konkretny trening | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id\_trening | int | - | NIE | Indentyfikator treningu |
| Id\_kategorii | int | - | NIE | Identyfikator kategorii |

Tabela 26– Tabela przechowuje ćwiczenia dla poszczególnego treningu-źródło: opracowanie własne

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa tabeli** | Trening\_Exercise | | | |
| **Opis** | Tabela przechowuje informacja o do jakiej kategori należy konkretny trening | | | |
| **Nazwa pola** | **Typ** | **Rozmiar** | **Czy puste** | **Opis pola** |
| Id\_ćwiczenia | Int | - | NIE | Identyfikator ćwiczenia |
| Id\_trening | int | - | NIE | Indentyfikator treningu |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 |