

# AKADEMIA NAUK STOSOWANYCH W NOWYM SĄCZU

Instytut Techniczny  
Informatyka Stosowana

## DOKUMENTACJA PROJEKTOWA INŻYNIERIA OPROGRAMOWANIA

**GameZone**

Autorzy:  
Adamczyk Mateusz  
Bazielich Tomasz  
Cetera Maciej

Prowadzący:  
mgr inż. Daniel Drozd

Nowy Sącz 2022

## **Spis treści**

<b>1. Opis projektu</b>	<b>3</b>
<b>2. Podobne aplikacje</b>	<b>4</b>
<b>3. Definicja wymagań funkcjonalnych i нефункционалне</b>	<b>5</b>
<b>4. Identyfikacja problemów oraz proponowane rozwiązania</b>	<b>6</b>
<b>5. Dobór technologii.</b>	<b>7</b>
<b>6. Diagram przypadków użycia</b>	<b>8</b>
<b>7. Scenariusze przypadków użycia</b>	<b>8</b>
<b>8. Diagram ERD</b>	<b>16</b>
<b>9. Diagram Klas</b>	<b>17</b>
<b>10. Diagram Sekwencji</b>	<b>18</b>
<b>11. Diagram Aktywności</b>	<b>19</b>
<b>12. Wzorce projektowe</b>	<b>20</b>
<b>Literatura</b>	<b>23</b>
<b>Spis rysunków</b>	<b>23</b>
<b>Spis tabel</b>	<b>24</b>

## 1. Opis projektu

Celem projektu (GameZone) jest stworzenie aplikacji internetowej - sklepu z grami komputerowymi z funkcjami serwisu społecznościowego.

Platforma będzie przeznaczona przede wszystkim dla graczy. Każdy gracz będzie mógł kupować oraz kolekcjonować swoje ulubione tytuły ze świata gier. Aplikacja umożliwi użytkownikom na rozwijanie swojego profilu, a inni gracze będą mogli te postępy śledzić.

## 2. Podobne aplikacje

- Steam
- Epic games store
- GOG galaxy
- Origin
- Ubisoft Store
- Microsoft Store
- Battle.net
- G2A

Nasza strona połączy najistotniejsze funkcje wyżej wymienionych. Nasza aplikacja będzie posiadała przejrzysty i intuicyjny interfejs przyjazny dla użytkownika. Profil każdego użytkownika będzie posiadał swój poziom, który może wzrosnąć dzięki punktom, które można zdobyć przy zakupach gier. Coraz wyższe poziomy pozwolą użytkownikowi na odblokowanie kolejnych nagród do swojego profilu.

### **3. Definicja wymagań funkcjonalnych i нефункциональных**

Wymagania funkcjonalne:

- rejestracja
- logowanie
- kupowanie gry
- wyszukiwanie gier w sklepie
- dodanie środków do konta użytkownika
- poziom profilu
- lista obserwowanych użytkowników
- lista życzeń informująca użytkownika o promocjach
- koszyk
- oceny użytkowników
- dodanie, usunięcie, modyfikowanie gry do sklepu (Admin)
- moderowanie kontami użytkowników (nałożenie blokady) (Admin)

Wymagania нефункциональные:

- przejrzysty i intuicyjny interfejs
- szybkość działania aplikacji

## **4. Identyfikacja problemów oraz proponowane rozwiązania**

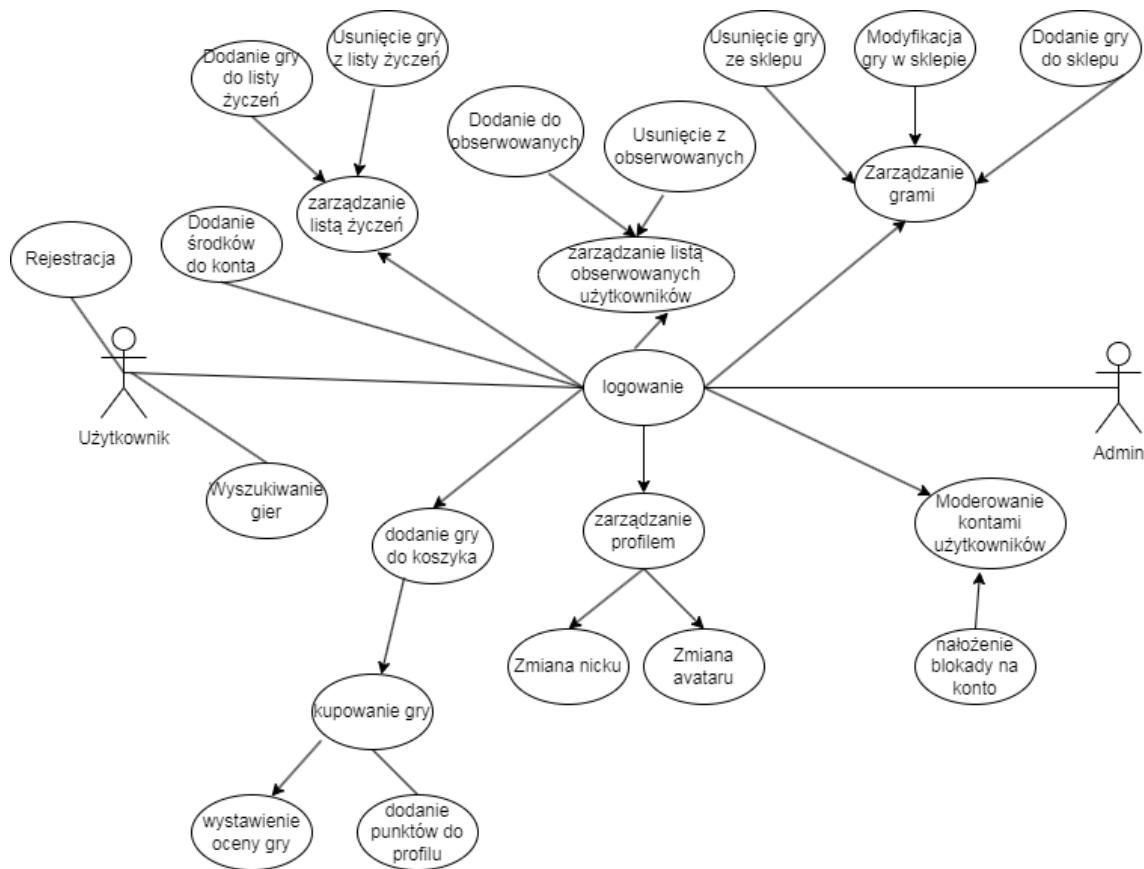
W naszym projekcie przewidujemy problemy podczas implementacji niektórych z naszych założeń. Najbardziej skomplikowanymi do wdrożenia funkcjonalnościami wydają się być lista obserwowanych profili oraz poziomy profilu.

Lista obserwowanych umożliwi na śledzenie kont innych użytkowników, ich poziomu, który będzie zależał od ilości posiadanych punktów.

## 5. Dobór technologii.

- Język PHP - framework Laravel
- Wygląd - Biblioteka Bootstrap z HTML oraz CSS
- Kompilator - Visual Studio Code

## 6. Diagram przypadków użycia



Rys. 6.1. Diagram przypadków użycia

## 7. Scenariusze przypadków użycia



1. Scenariusz
Nazwa: Scenariusz 1 - Rejestracja
Aktor: Użytkownik
Warunki początkowe: -
Opis: Użytkownik wybiera opcje "Rejestracja"
Ścieżki główne: 1. Użytkownik klika opcje "Rejestracja" 2. Wypełnia formularz rejestracyjny danymi 3. Klika "Zatwierdź" 4. System sprawdza poprawność danych
Ścieżka alternatywna: 4a. Wyświetlenie komunikatu o niepoprawnych danych
Warunki końcowe: Użytkownik został pomyślnie zarejestrowany

2. Scenariusz
Nazwa: Scenariusz 2 - Logowanie
Aktor: Użytkownik, Admin
Warunki początkowe: Użytkownik musi być zarejestrowany
Opis: Użytkownik wybiera opcje "Logowanie"
Ścieżki główne: 1. Użytkownik wybiera opcje "Logowanie" 2. Wypełnia formularz logowania danymi 3. Klika "Zatwierdź" 4. System sprawdza poprawność danych
Ścieżka alternatywna: 4a. Wyświetlenie komunikatu o niepoprawnych danych
Warunki końcowe: Użytkownik został pomyślnie zalogowany

3. Scenariusz
Nazwa: Scenariusz 3 - Wyszukiwanie gier
Aktor: Użytkownik
Warunki początkowe: -
Opis: Użytkownik wybiera opcje "Wyszukaj"
Ścieżki główne:
1. Użytkownik wpisuje nazwe gry
2. System sprawdza czy wyszukiwana fraza znajduje sie w bazie danych
Ścieżka alternatywna:
3. Wyświetlenie komunikatu o nie znalezieniu wyszukiwanej frazy
Warunki końcowe: System wyświetla wyniki wyszukiwana

4. Scenariusz
Nazwa: Scenariusz 4 - Dodanie środków do konta
Aktor: Użytkownik
Warunki początkowe: Użytkownik musi być zalogowany
Opis: Użytkownik wybiera opcje "Dodaj środki"
Ścieżki główne:
1. Użytkownik klika opcje "Dodaj środki"
2. Określa kwote którą chce doładować
3. Wybiera sposób płatności
4. Użytkownik wybiera "Zatwierdź transakcje"
5. System sprawdza wykonanie transakcji
Ścieżka alternatywna:
5a. System odrzuca wykonanie transakcji
Warunki końcowe: Środki zostały przelane na konto użytkownika

5. Scenariusz
Nazwa: Scenariusz 3 - Dodanie gry do koszyka
Aktor: Użytkownik
Warunki początkowe: Użytkownik musi być zalogowany oraz wyszukał grę
Opis: Użytkownik wybiera opcję "Dodaj grę do koszyka"
Ścieżki główne:
1. System sprawdza, czy dana gra jest już w koszyku
Ścieżka alternatywna:
2. Jeśli dana gra jest już w koszyku gra nie zostanie dod do niego dodana ponownie
Warunki końcowe: Gra została dodana do koszyka

6. Scenariusz
Nazwa: Scenariusz 6 - Dodanie gry do listy życzeń
Aktor: Użytkownik
Warunki początkowe: Użytkownik musi być zalogowany oraz wyszukał grę
Opis: Użytkownik dodaje grę do listy życzeń
Ścieżki główne:
1. System sprawdza, czy dana gra jest już w liście życzeń
Ścieżka alternatywna:
2. Jeśli dana gra jest już w liście życzeń gra nie zostanie dodana ponownie
Warunki końcowe: Gra została dodana do listy życzeń

7. Scenariusz
Nazwa: Scenariusz 7 - Kupowanie gry
Aktor: Użytkownik
Warunki początkowe: Użytkownik musi być zalogowany, wybrana gra znajduje się w koszyku, użytkownik posiada środki na koncie
Opis: Użytkownik wybiera opcje "Dokonaj zakupu"
Ścieżki główne:
1. Użytkownik dokonuje zakupu gry z koszyka.
2. System sprawdza dostępne środki na koncie.
3. System pobiera odpowiednią sumę z konta.
4. System przypisuje zakupioną grę do konta użytkownika.
5. System przyznaje odpowiednią ilość punktów za zakup gry.
Ścieżka alternatywna:
1a. Użytkownik nie posiada gry w koszyku.
2a. Użytkownik nie posiada odpowiednich środków na koncie.
Warunki końcowe:
Gra została pomyślnie zakupiona
Punkty za zakup gry zostały dodane do konta

8. Scenariusz
Nazwa: Scenariusz 8 - Usunięcie gry z listy życzeń
Aktor: Użytkownik
Warunki początkowe: Użytkownik jest zalogowany oraz gra jest na liście życzeń
Opis: Użytkownik usuwa grę z listy życzeń
Ścieżki główne:
1. Użytkownik przy wybranej grze z listy klika "usuń grę z listy życzeń"
2. System usuwa grę z listy życzeń
Ścieżka alternatywna: -
Warunki końcowe: Gra została usunięta z listy życzeń

9. Scenariusz
Nazwa: Scenariusz 9 - Wystawienie oceny gry
Aktor: Użytkownik
Warunki początkowe: Użytkownik posiada ocenianą grę
Opis: Użytkownik wybiera opcję "Oceń grę"
Ścieżki główne:
1. Użytkownik wybiera stopień oceny od 1 do 5
2. System przypisuje ocenę użytkownika grze
Ścieżka alternatywna: -
Warunki końcowe: Gra została oceniona przez użytkownika

10. Scenariusz
Nazwa: Scenariusz 10 - Dodanie użytkownika do listy obserwowanych
Aktor: Użytkownik
Warunki początkowe: Użytkownik jest zalogowany oraz użytkownik, który ma być obserwowany jest wyszukany
Opis: Użytkownik dodaje innego użytkownika do listy obserwowanych
Ścieżki główne:
1. System sprawdza, czy dany użytkownik jest już obserwowany
Ścieżka alternatywna:
2. Jeśli użytkownik jest już na liście obserwowanych nie zostanie on do niej dodany ponownie
Warunki końcowe: Użytkownik został dodany do listy obserwowanych

11. Scenariusz
Nazwa: Scenariusz 11 - Zmiana nicku
Aktor: Użytkownik
Warunki początkowe: Użytkownik jest zalogowany
Opis: Użytkownik wybiera opcję "Zmień nazwę profilu"
Ścieżki główne: <ol style="list-style-type: none"> <li>1. Użytkownik wybiera opcję "Zarządzaj profilem"</li> <li>2. W panelu zarządzania użytkownik wypisuje nową nazwę profilu</li> <li>3. System zmienia nazwę na podaną</li> </ol>
Ścieżka alternatywna: - <ol style="list-style-type: none"> <li>2a. Użytkownik podał tę samą nazwę lub nieobsługiwane znaki</li> </ol>
Warunki końcowe: Nazwa profilu została pomyślnie zmieniona na nową

12. Scenariusz
Nazwa: Scenariusz 12 - Usunięcie użytkownika z listy obserwowanych
Aktor: Użytkownik
Warunki początkowe: Użytkownik jest zalogowany
Opis: Użytkownik usuwa obserwowanego z listy obserwowanych
Ścieżki główne: <ol style="list-style-type: none"> <li>1. Użytkownik przy danym obserwowanym wybiera opcję "usuń z listy obserwowanych"</li> <li>2. System usuwa danego obserwowanego z listy obserwowanych</li> </ol>
Ścieżka alternatywna: -
Warunki końcowe: Dany użytkownik został usunięty z listy obserwowanych

13. Scenariusz
Nazwa: Scenariusz 13 - Zmiana avataru
Aktor: Użytkownik
Warunki początkowe: Użytkownik jest zalogowany
Opis: Użytkownik wybiera opcję "Zmień ikonę profilu"
Ścieżki główne: <ol style="list-style-type: none"> <li>1. Użytkownik wybiera opcję "Zarządzaj profilem"</li> <li>2. W panelu zarządzania użytkownik wybiera nową ikonę profilu</li> <li>3. System zmienia ikonę na wybraną</li> </ol>
Ścieżka alternatywna: -
Warunki końcowe: Ikona profilu została pomyślnie zmieniona.

14. Scenariusz
Nazwa: Scenariusz 14 - Dodanie gry do sklepu
Aktor: Admin
Warunki początkowe: Admin jest zalogowany
Opis: Admin dodaje grę do sklepu
Ścieżki główne: <ol style="list-style-type: none"> <li>1. Admin wybiera opcję "Zarządzaj grami"</li> <li>2. Admin wybiera opcję "Dodaj grę do sklepu"</li> <li>3. Admin wypełnia formularz danymi</li> <li>4. Admin wybiera "Zatwierdź"</li> <li>5. System sprawdza poprawność danych</li> </ol>
Ścieżka alternatywna: <ol style="list-style-type: none"> <li>6. Jeśli dane są niepoprawne lub gra jest już w sklepie, gra nie zostanie dodana i wyświetli się stosowny komunikat</li> </ol>
Warunki końcowe: Gra została dodana do sklepu

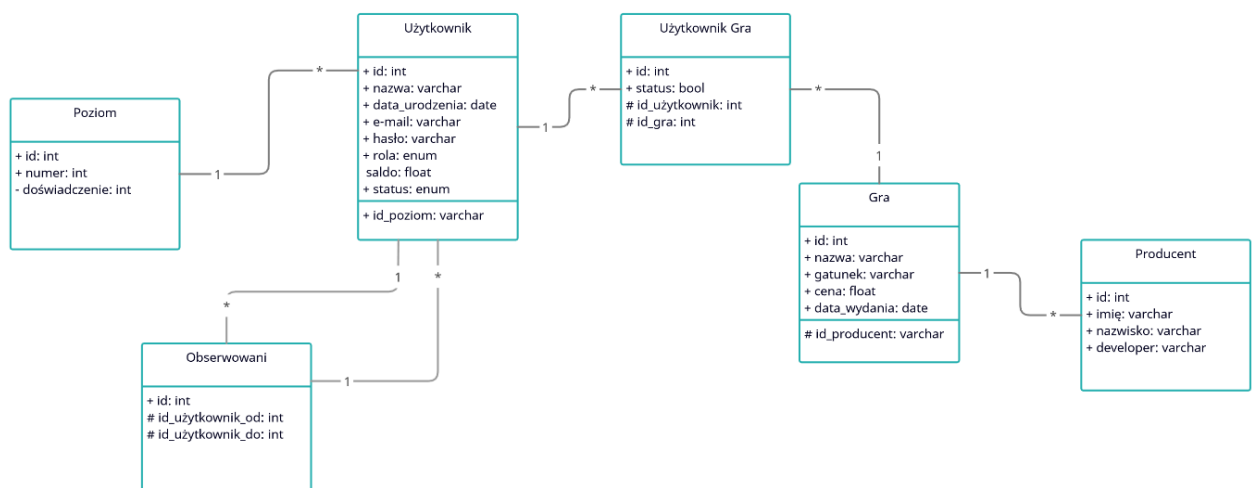
15. Scenariusz
Nazwa: Scenariusz 15 - Nałożenie blokady na konto
Aktor: Admin
Warunki początkowe: Admin jest zalogowany
Opis: Admin nakłada nakłada blokadę na danego użytkownika
Ścieżki główne: <ol style="list-style-type: none"> <li>1. Admin wyszukuje danego użytkownika</li> <li>2. Admin wybiera opcję "Zablokuj użytkownika"</li> <li>3. Admin określa długość trwania blokady i zatwierdza wybór</li> <li>4. System nakłada blokadę na konto użytkownika</li> </ol>
Ścieżka alternatywna: <ol style="list-style-type: none"> <li>2a. Użytkownik zablokowany permanentnie nie może być ponownie zablokowany</li> </ol>
Warunki końcowe: Na użytkownika zostaje nałożona blokada

16. Scenariusz
Nazwa: Scenariusz 16 - Zdjęcie blokady z konta
Aktor: Admin
Warunki początkowe: Admin jest zalogowany, dany użytkownik jest zablokowany
Opis: Admin usuwa blokadę z danego użytkownika
Ścieżki główne: 1. Admin wyszukuje danego użytkownika 2. Admin wybiera opcję "Odblokuj użytkownika" 3. System zdejmuję blokadę z konta użytkownika
Ścieżka alternatywna: -
Warunki końcowe: Z Użytkownika zostaje zdjęta blokada

17. Scenariusz
Nazwa: Scenariusz 17 - Usunięcie gry ze sklepu
Aktor: Admin
Warunki początkowe: Admin jest zalogowany oraz gra jest w sklepie
Opis: Admin usuwa grę ze sklepu
Ścieżki główne: 1. Admin wybiera opcję "Zarządzaj grami" 2. Admin wybiera opcję "Usuń grę do sklepu" 3. System usuwa grę ze sklepu
Ścieżka alternatywna: -
Warunki końcowe: Gra została usunięta ze sklepu

18. Scenariusz
Nazwa: Scenariusz 18 - Modyfikacja gry ze sklepu
Aktor: Admin
Warunki początkowe: Admin jest zalogowany oraz gra jest w sklepie
Opis: Admin modyfikuje grę ze sklepu
<p>Ścieżki główne:</p> <ol style="list-style-type: none"> <li>1. Admin wybiera opcję "Zarządzaj grami"</li> <li>2. Admin wybiera opcję "Modyfikuj grę"</li> <li>3. Admin wypełnia formularz danymi</li> <li>4. Admin wybiera "Zatwierdź"</li> <li>5. System sprawdza poprawność danych</li> </ol> <p>Ścieżka alternatywna:</p> <ol style="list-style-type: none"> <li>6. Jeśli dane są niepoprawne lub gra jest już w sklepie, gra nie zostanie dodana i wyświetli się stosowny komunikat</li> </ol> <p>Warunki końcowe: Gra została zmodyfikowana w sklepie</p>

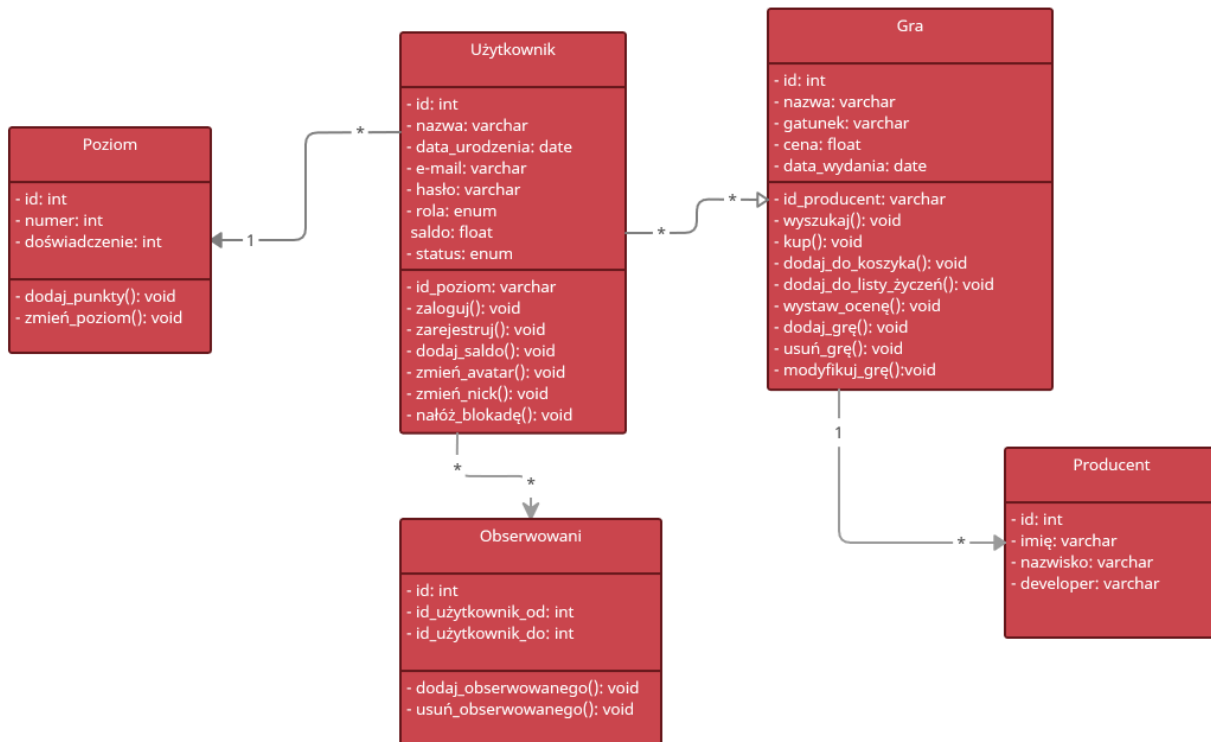
## 8. Diagram ERD



Rys. 8.1. Diagram ERD

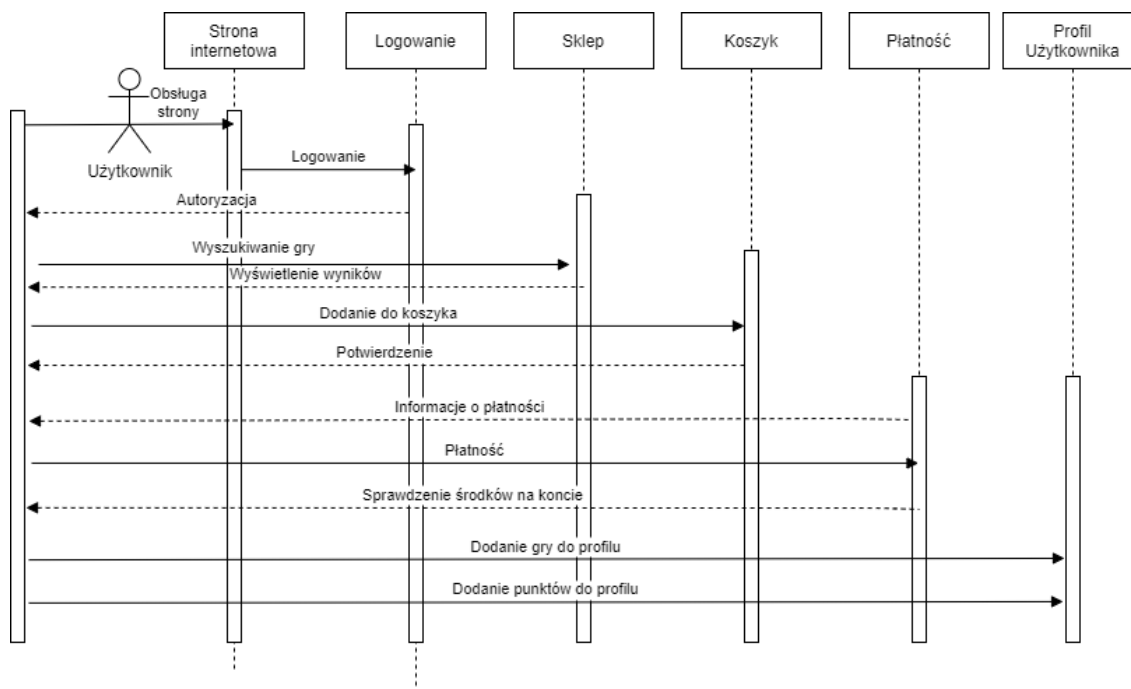


## 9. Diagram Klas



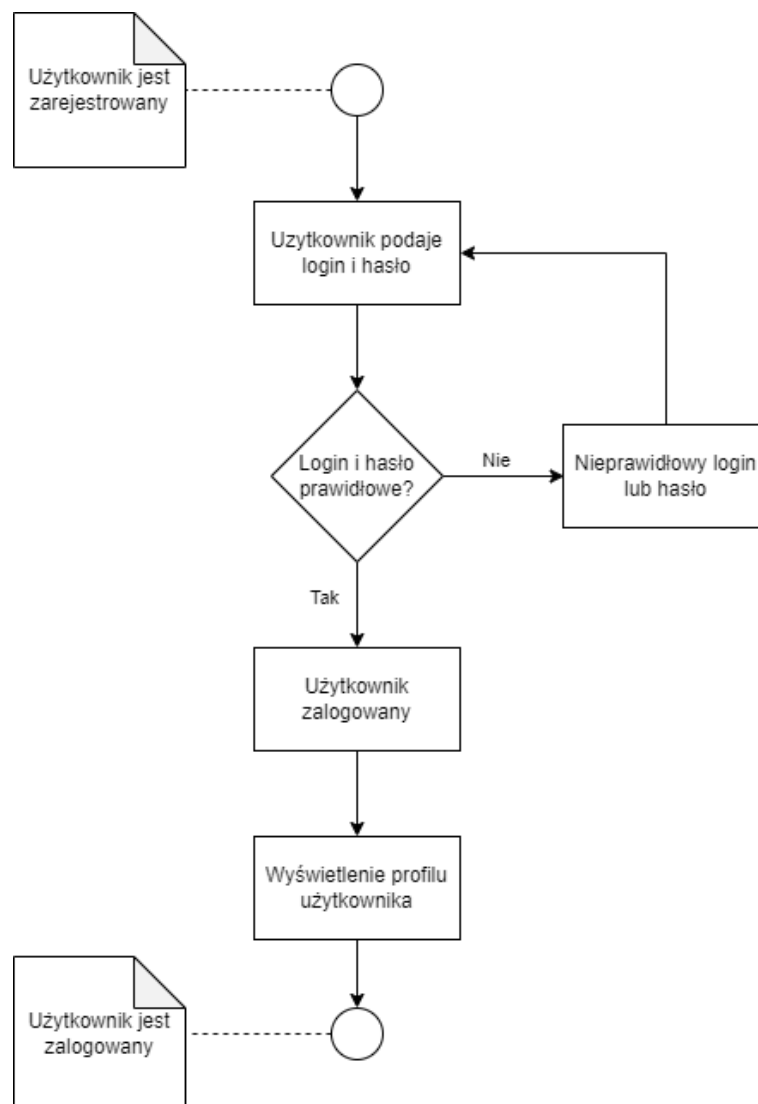
Rys. 9.1. Diagram klas

## 10. Diagram Sekwencji



Rys. 10.1. Diagram sekwencji

## 11. Diagram Aktywności



Rys. 11.1. Diagram aktywności

## 12. Wzorce projektowe

### 1. Metoda szablonowa

Jej zadaniem jest zdefiniowanie metody będącej szkieletem algorytmu. Jest to wzorzec projektowy który pozwala na nadpisanie pewnych etapów algorytmu bez konieczności zmiany jego struktury. Metoda szablonowa jest stosowana w przypadkach gdy chcemy pozwolić klientom na rozszerzanie niektórych etapów algorytmu bez dokonywania zmian jego całości. Wzorzec jest również przydatny w sytuacjach gdy kilka klas posiada prawie identyczne algorytmy

### 2. Budowniczy

Budowniczy jest to kreacyjny wzorzec projektowy, który umożliwia na budowanie zaawansowanych obiektów wg etapów. Tworzone Obiekty mogą być różnego typu oraz nie mają dostępu do innych obiektów przed ich ukończeniem. Główną zaletą jest możliwość ponownego użycia stworzonej struktury do tworzenia innych przedmiotów, wiąże się to jednak ze znacznym stopniem skomplikowania kodu.

### 3. Adapter

Adapter umożliwia współgranie ze sobą elementów obiektów, aby mogły ze sobą współpracować, lecz pojedynczo są one niekompatybilne. Pozwala on konwertować interfejsy w taki sposób że obiekty się rozumiały. Adapter nie tylko potrafi konwertować formaty ale pozwala na współpracę obiektów. Możliwe jest nawet stworzenie adaptera dwukierunkowego, który konwertuje dane w obu kierunkach. Analogią to prawdziwego życia jest przejściówka między wtyczką europejską a brytyjską.

### 4. Singleton

Singleton to wzorzec kreacyjny, ale często określa się go jako antywzorzec ponieważ poprzez ograniczenie możliwości tworzenia obiektów tylko do pojedynczej instancji, czyli do tworzenia wielu obiektów w jednym wykonaniu kodu, łamie się zasady obiektowego projektowania. Zaletą takiego rozwiązania

jest pewność, że dany obiekt zostanie stworzony po raz pierwszy wtedy, kiedy rzeczywiście jest on potrzebny. Wadą są wspomniane złamane zasady projektowania obiektowego.

## 5. Odwiedzający

Odwiedzający jest behawioralnym wzorcem projektowym. Umożliwia on na dokonanie rozdzielania algorytmów od obiektów, które je wykorzystują w swoim działaniu. Pozwala to na odpowiednie dopasowanie działań do typu obiektu poprzez “odwiedzającego”, który zbiera informacje o danym obiekcie. Minusem takiego rozwiązania jest brak dostępu do niektórych obiektów, np. obiektów prywatnych.

## 6. Dekorator

Dekorator jest to strukturalny wzorec projektowy pozwalający dodawać nowe zadania obiektom przy pomocy innych obiektów które zawierają odpowiednie zachowania. Pozwala on na przypisanie dodatkowych obowiązków obiektom w trakcie działania programu bez niszczenia kodu z którego korzystają obiekty. Jest to wzorec projektowy znany również pod nazwą “nakładka” oznacza to że może być połączony z innym docelowym obiektem. Nakładka zawiera ten sam zestaw metod jak element docelowy i przydziela mu odpowiednio otrzymywane żądania.

## 7. Fasada

Fasada umożliwia uproszczenie interfejsu ze złożonych klas, bibliotek oraz frameworku. Fasada to klasa stanowiąca prosty interfejs dla złożonego podsystemu, zawierającego mnóstwo ruchomych części. Fasada ogranicza funkcjonalność, ale za to eksponuje tylko te elementy, których klient naprawdę potrzebuje. Stworzenie fasady jest wygodnym sposobem integracji swojej aplikacji ze skomplikowaną biblioteką posiadającą wiele funkcji, gdy potrzebujemy tylko wąskiego zakresu jej funkcji.

## 8. Kompozyt

Jest to strukturalny wzorzec projektowy który pozwala na dzielenie obiektów w struktury drzewiaste i traktowanie ich jako osobne obiekty. Jest stosowany tylko w przypadku gdy główny model aplikacji może być przedstawiony w formie drzewa. Kompozyt określa dwa typy elementów które współdzielą ten sam interfejs: proste liście i złożone kontenery. Kontener może składać się z liści oraz z innych kontenerów co pozwala na skonstruowanie zagnieżdżonej, rekurencyjnej struktury obiektów przypominającej drzewo.

## 9. Obserwator

Obserwator jest behawioralnym wzorcem projektowym, który umożliwia na poinformowanie innych obiektów o wydarzeniach mających miejsce w innym, obserwowanym obiekcie. Po każdej zmianie w monitorowanym obiekcie do innych obiektów zostają wysyłane powiadomienia zawierające informacje o zmianach w obiekcie. Wadą jest powiadamianie “subskrybentów” w nieokreślonej, przypadkowej kolejności.

## Spis rysunków

6.1. Diagram przypadków użycia . . . . .	8
8.1. Diagram ERD . . . . .	16
9.1. Diagram klas . . . . .	17
10.1. Diagram sekwencji . . . . .	18
11.1. Diagram aktywności . . . . .	19

## **Spis tabel**