



POLITECHNIKA WARSZAWSKA

Projekt techniczny

Aplikacja do gry w szachy z wykorzystaniem komunikacji webowej oraz interfejsu graficznego 3D

Projekt Indywidualny

Sebastian Kurpios

Spis treści

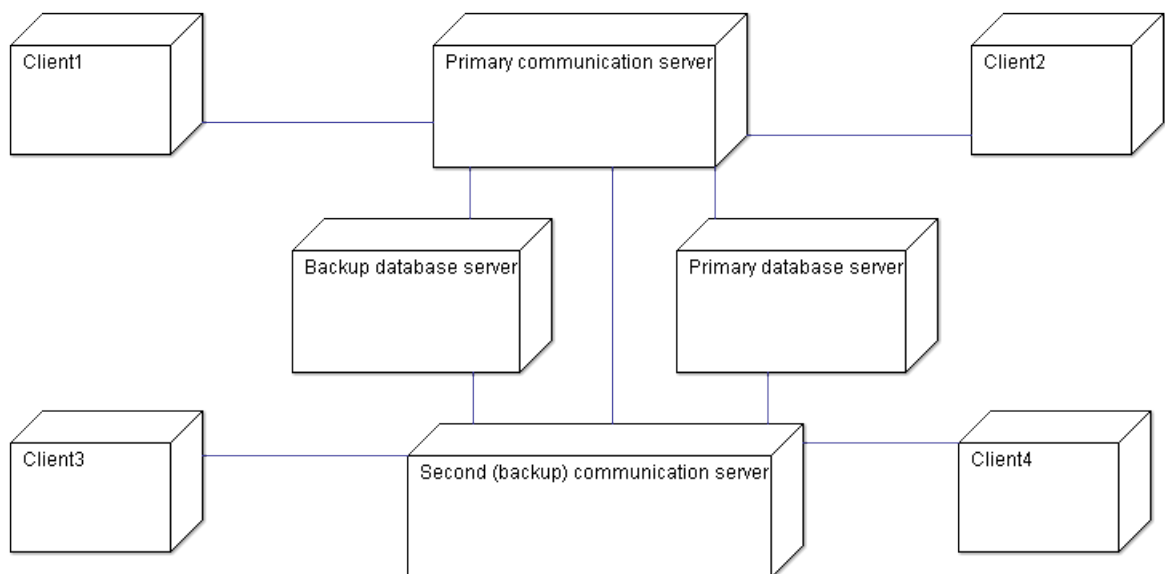
1	Opis aplikacji	2
2	Komponenty	2
3	Dane wejściowe i wyjściowe	3
3.1	Dane wejściowe	3
3.2	Dane wyjściowe	4
4	Wymagania	5
4.1	Diagram użycia	5
4.2	Wymagania funkcjonalne	7
4.3	Wymagania нефункционалне	7
5	Schematy blokowe	9
6	Diagram klas	11
7	Diagram związków encji	13
8	Technologia	13
9	Algorytmy	14
9.1	Algorytm min-max	14
9.2	Algorytm alfa-beta	14
9.3	Analiza istniejących rozwiązań	15
9.3.1	Przykłady istniejących programów	15

1 Opis aplikacji

Celem projektu jest wykonanie aplikacji do gry w szachy z interfejsem 3D. Program powinien pozwalać na grę jednoosobową z możliwością wyboru stopnia trudności oraz wieloosobową. Dodatkowo wersja wieloosobowa powinna umożliwić komunikację pomiędzy graczami oraz wybór gracza według jego umiejętności. Ponadto użytkownik aplikacji będzie mógł zarejestrować się w serwisie, a wyniki jego potyczek będą zapisywane w bazie danych.

2 Komponenty

1. Aplikacja desktopowa - klient
2. Serwer komunikacyjny
3. Serwer bazy danych



Rysunek 1: Diagram komponentów

3 Dane wejściowe i wyjściowe

3.1 Dane wejściowe

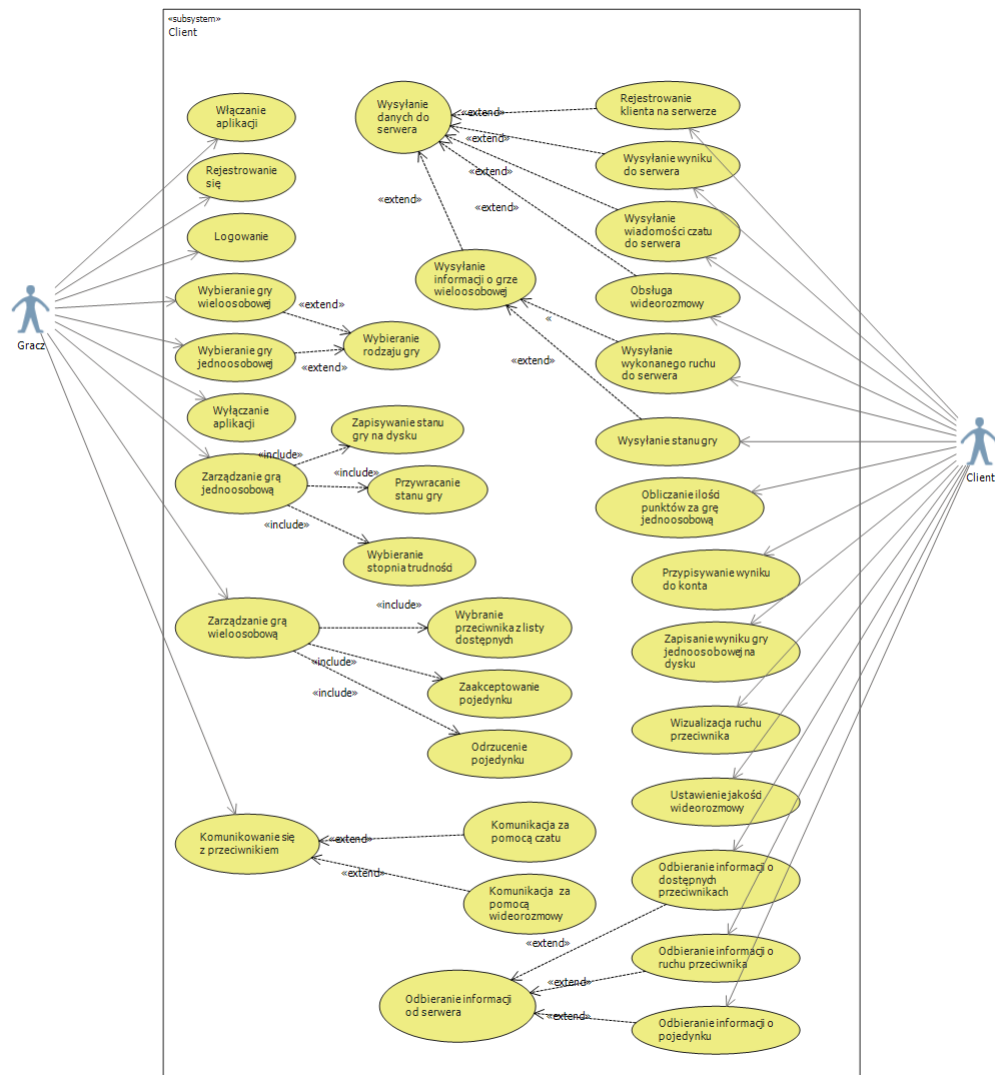
1. Dane wejściowe (konfiguracyjne) serwera
 - Port nasłuchiwania
 - Nazwa bazy danych
 - Nazwa użytkownika bazy danych
 - Hasło bazy danych
2. Dane wejściowe aplikacji klienckiej
 - Dane początkowe i konfiguracyjne
 - (a) Adres IP serwera
 - (b) Port serwera
 - (c) Tryb gry
 - (d) Dane kamery
 - Dane logowania
 - (a) Login
 - (b) Hasło
 - Dane rejestracji
 - (a) Login
 - (b) Hasło
 - (c) Adres E-mail
 - (d) Imię
 - (e) Nazwisko
 - Dane aplikacji w trybie jednoosobowym
 - (a) Algorytm działania
 - (b) Poziom trudności
 - Dane aplikacji w trybie wieloosobowym
 - (a) Identyfikator wybranego przeciwnika
 - (b) Tryb komunikacji pomiędzy użytkownikami (kamera czy komunikator tekstowy)

3.2 Dane wyjściowe

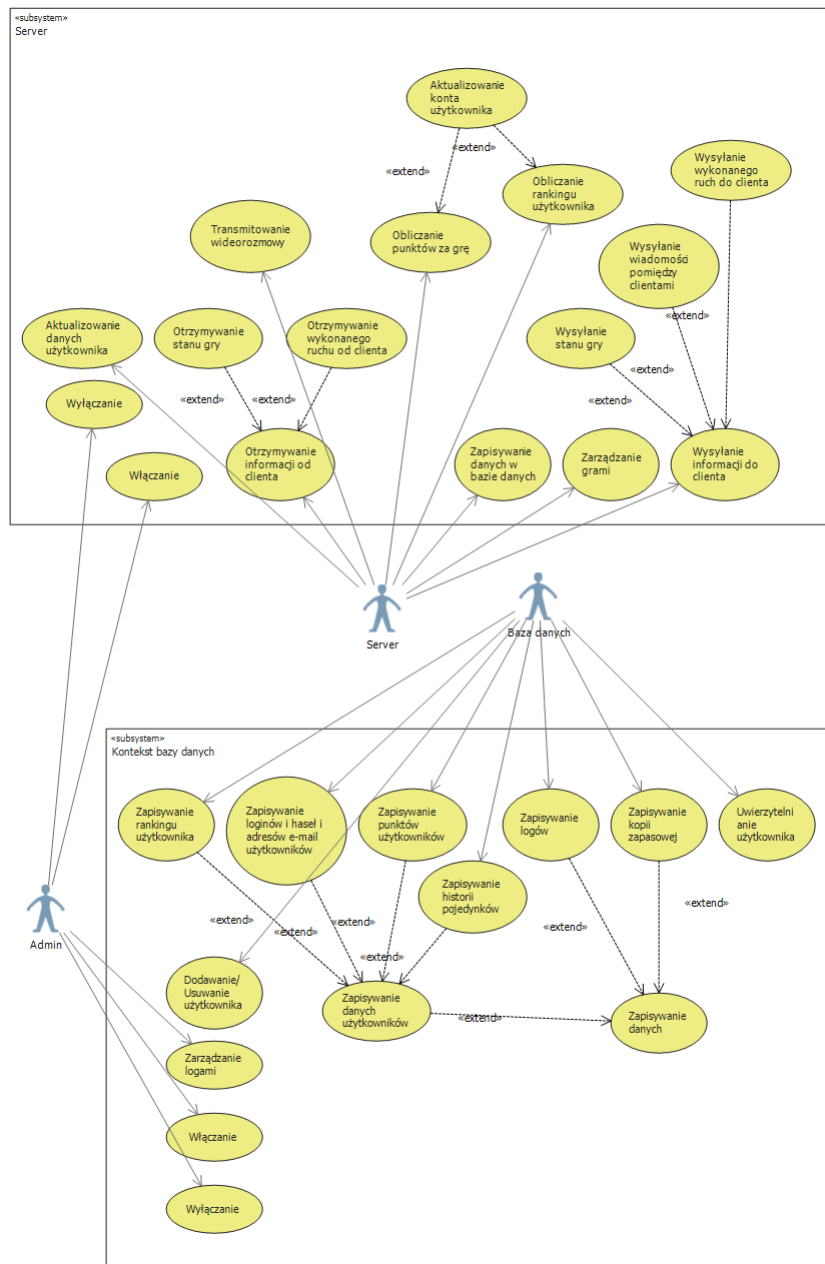
1. Dane wyjściowe serwera - brak, serwer nie kończy działania
2. Dane wyjściowe aplikacji klienckiej
 - Wynik gry
 - Liczba punktów zdobytych za grę
 - Uaktualniony ranking

4 Wymagania

4.1 Diagram użycia



Rysunek 2: Diagram użycia dla aplikacji klienckiej



Rysunek 3: Diagram użycia dla serwera i bazy danych

4.2 Wymagania funkcjonalne

1. Gra jednoosobowa:

- wybieranie stopnia trudności wirtualnego przeciwnika (głębokość dla algorytmu min-max lub alfa-beta)
- zapisywanie stanu gry i przywracanie go w dowolnym momencie
- obliczanie ilości punktów za grę ze względu na trudność przeciwnika i wynik
- zapisywanie na dysku wyniku gry oraz liczby otrzymanych punktów
- przypisywanie liczby otrzymanych punktów do danego konta

2. Gra wieloosobowa:

- wybieranie przeciwnika z listy dostępnych użytkowników
- możliwość akceptacji lub odrzucenia zaproponowanego pojedynku przez innego użytkownika
- możliwość komunikacji z przeciwnikiem za pomocą czatu lub wideorozmowy
- obliczanie ilości punktów za grę ze względu na ranking przeciwnika i wynik
- możliwość rozgrywania gry anonimowo lub jako zalogowany użytkownik
- możliwość identyfikacji użytkownika za pomocą konta w serwisie facebook

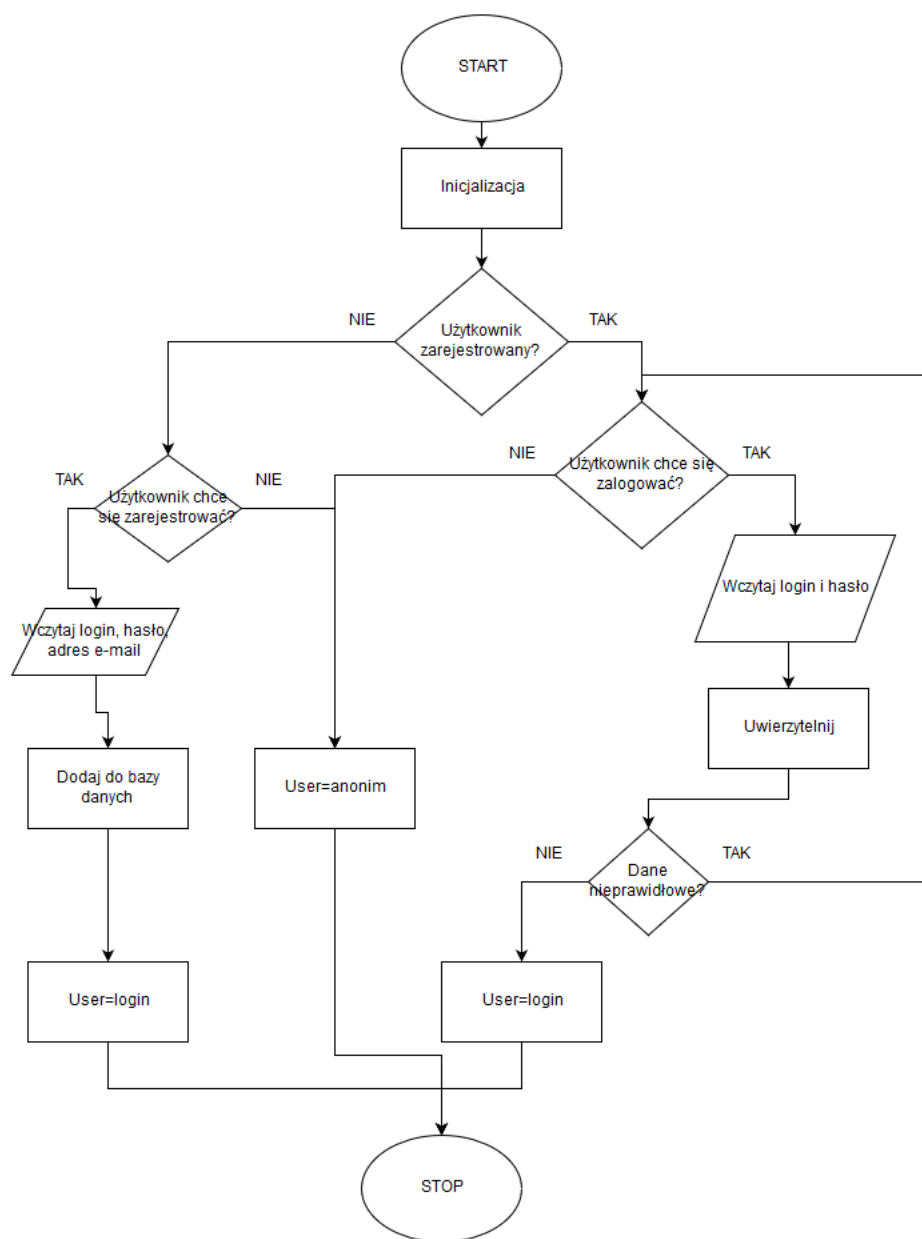
4.3 Wymagania niefunkcjonalne

1. Reakcja na niedostępność serwera z powodu awarii lub braku odpowiedzi na zapytanie w wyznaczonym czasie:

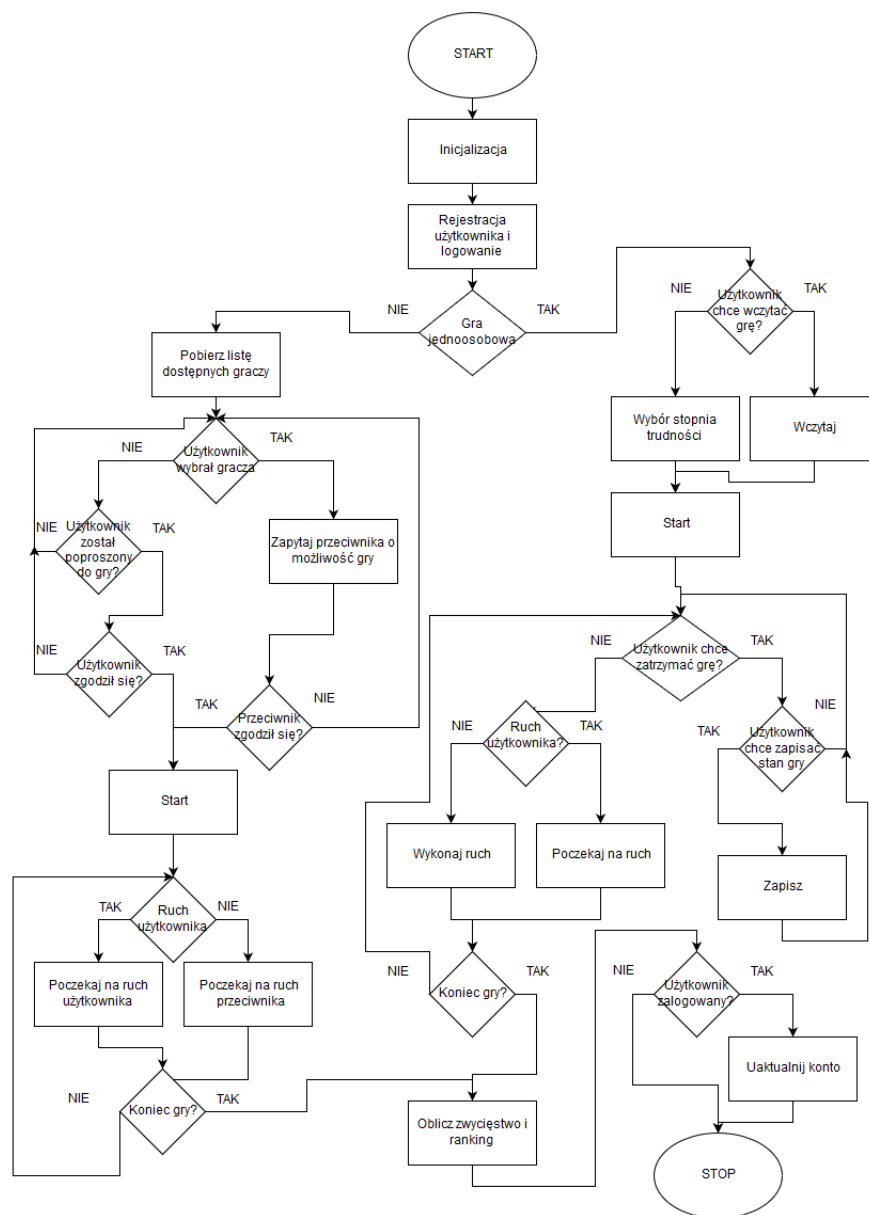
- możliwość rozegrania gry wyłącznie w trybie z komputerem
- brak możliwości dopisania liczby otrzymanych punktów do konta gracza
- zapisanie stanu gry wyłącznie na dysku

- w razie przerwania gry automatyczne zakończenie pojedynku bez przyznania punktów
2. Reakcja na przerwanie połączenia z klientem w czasie gry
 - oczekiwanie przez wyznaczony czas na użytkownika
 - przyznanie przegranej niedostępnemu użytkownikowi po odczekaniu
 3. Reakcja na brak urządzeń audio-video w urządzeniu użytkownika
 - brak możliwości komunikacji za pomocą wideorozmowy

5 Schematy blokowe

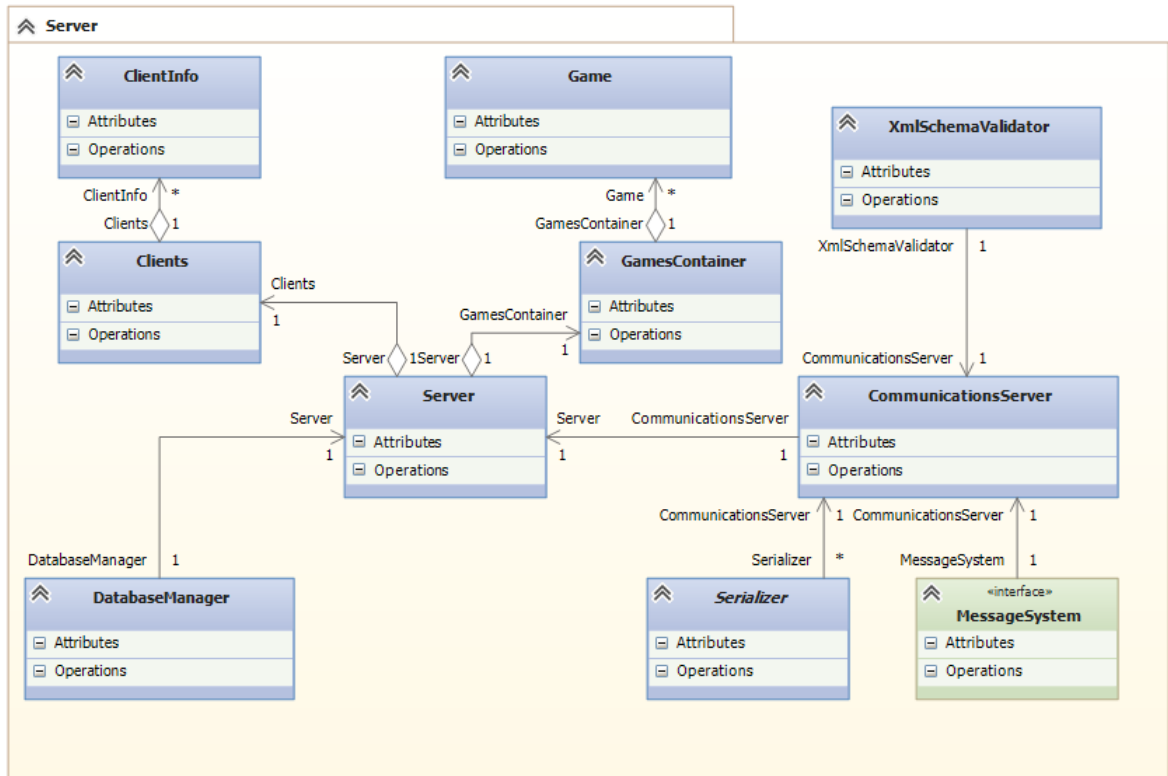


Rysunek 4: Schemat blokowy logowanie i rejestracji

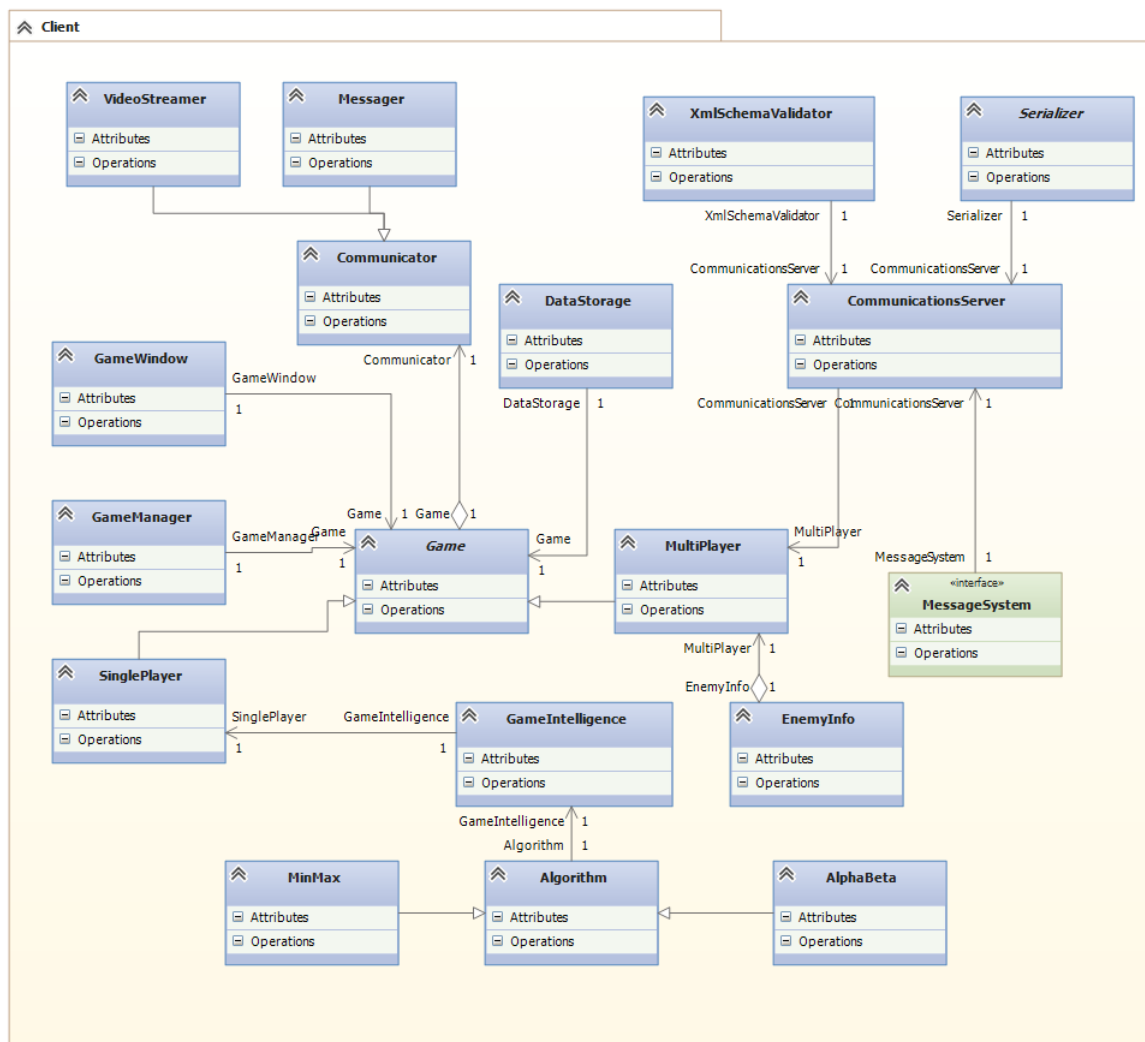


Rysunek 5: Uproszczony schemat blokowy aplikacji klienckiej

6 Diagram klas

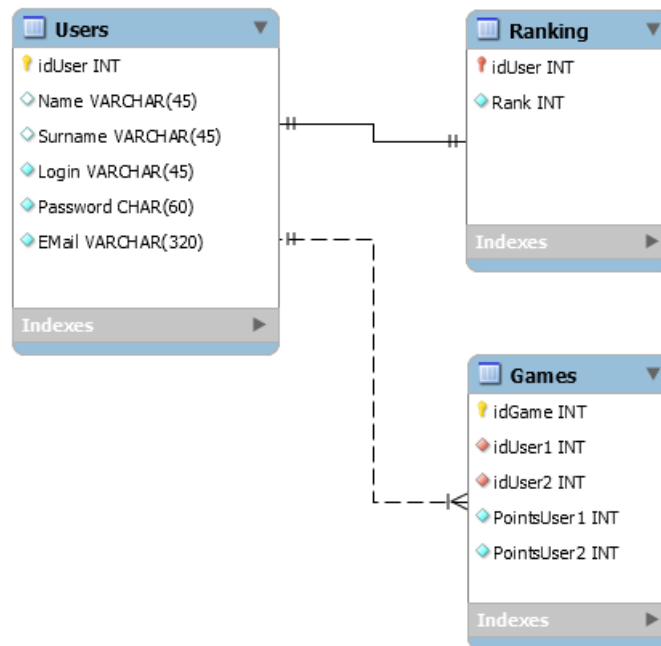


Rysunek 6: Diagram klas dla serwera



Rysunek 7: Diagram klas dla aplikacji klienckiej

7 Diagram związków encji



Rysunek 8: Diagram związków encji bazy danych

8 Technologia

1. Język programowania - C#
2. Platforma - .Net Framework
 - Dostęp do bazy danych - ADO.NET
 - Komunikacja - Gniazda sieciowe
 - Interfejs graficzny (GUI) - Windows Forms
3. Silnik graficzny - OpenTK
4. Baza danych - Microsoft SQL Server

9 Algorytmy

9.1 Algorytm min-max

```
function minimax(wezel, glebokosc, maxGracz)
  if glebokosc = 0 or wezel jest terminalny
    return heurystyke w wezle

  if maxGracz
    najlepszaWartosc = -inf
    for each dziecka wezla
      v = minimax(dziecko, glebokosc-1, FALSE)
      najlepszaWartosc = max(najlepszaWartosc, v)
    return najlepszaWartosc

  else (* minGracz *)
    najlepszaWartosc = -inf
    for each dziecka wezla
      v = minimax(dziecko, glebokosc-1, TRUE)
      najlepszaWartosc = min(najlepszaWartosc, v)
    return najlepszaWartosc
```

9.2 Algorytm alfa-beta

```
funkcja alfabeta(wezel, glebokosc, alfa, beta)
  if wezel jest koncowy lub glebokosc = 0
    return wartosc heurystyczna wezla

  if przeciwnik ma zagrac w wezle
    for each potomka wezla
      beta = min(beta, alfabeta(potomek, glebokosc-1, alfa,
        beta))
      if alfa >= beta
        przerwij przeszukiwanie {odcinamy galaz Alfa}
    return beta
  else {my mamy zagrac w wezle}
    foreach potomka wezla
```

```
    alfa = max(alfa, alfabeta(potomek, glebokosc-1, alfa,
                             beta))
    if alfa >= beta
        przerwij przeszukiwanie {odcinamy galaz Beta}
return beta
```

9.3 Analiza istniejących rozwiązań

9.3.1 Przykłady istniejących programów

- Arena ¹
- WinBoard ²
- Chessmaster 11th Edition ³

Większość aplikacji do gry w szachy posiada albo rozbudowany interfejs graficzny albo rozbudowany dostęp do silników graficznych, a aplikację, które mają te dwie cechy są zazwyczaj płatne. Moja aplikacja do gry w szachy będzie łączyć te dwie cechy wraz z rozbudowaną komunikacją pomiędzy użytkownikami, a ponadto będzie opensource.

¹<http://www.playwitharena.com/>

²<http://hgm.nubati.net/>

³<http://chessmaster.uk.ubi.com/xi/index.php>