**Protokół sieciowy**

***Opis ogólny protokołu:***

Zaimplementowany przez nas protokół sieciowy będzie protokołem tekstowym o modelu klient-serwer. Serwer oczekuje na zgłoszenia, obsługuje je(przygotowuje odpowiedź i odsyła). Klient nawiązuje połączenie, wysyła żądanie, czeka na odpowiedź, a później ją przetwarza. Serwer będzie miał możliwość obsługi wielu klientów (każdy klient obsługiwany jest w innym wątku). Jednostką przesyłu informacji będzie linia tekstu, kończąca się znakiem końca linii. Parametry dla poszczególnych zdań protokołu. Poszczególne parametry oddzielane spacjami. Serwer wysyła do klienta dane konfiguracyjne gry, dane konkretnych poziomów oraz listę najlepszych wyników. Natomiast klient wysyła do serwera wynik gracza po zakończeniu gry.

***Opis szczegółowy protokołu:***

Oznaczenia:

C – Klient S – Server

*Pobieranie danych gry:*

Klient wysyła do serwera informację o pobraniu danych konfiguracyjnych gry.

C: Get game config => S

Serwer odpowiada wysłaniem danych, a dokładnie liczbę poziomów(int), szerokość(int) i wysokość(int) okna :

S: Level\_number Width Height => C

*Pobieranie danych konkretnego poziomu:*

Klient w celu pobrania danych konkretnego poziomu wysyła do serwera informację, że chce je uzyskać: C: Get level config(numer poziomu)=> S

Serwer odpowiada wysłaniem danych nazwy poziomu(string), liczby pól w szerokości(int), liczby pól w wysokości(int) i wygląd mapy(ciąg cyfr oddzielony średnikami i przecinkami)

S: Level\_name Map\_width Map\_height Map\_state => C

*Pobieranie listy najlepszych wyników:*

W celu pobrania listy najlepszych wyników klient wysyła do serwera informację, że chce ją uzyskać:

C: Get ranking list=> S

Serwer odpowiada wysłaniem listy najlepszych wyników składającej się z nicku(string) i wyniku(int) konkretnego użytkownika.

S: Nick Wynik => C

*Przesłanie wyniku do serwera:*

Najpierw klient wysyła do serwera informację o chęci przesłania go na serwer (w celu umożliwienia odebrania go przez serwer):

C: Sending score => S

Po tym klient wysyła dane użytkownika (nick i wynik)

C: Nick Wynik => S

Po odebraniu tych danych, do klienta wysyłana jest informacja o odebraniu danych:

S: Score received => C