

Daniels Komjago

PROZE

Protokołu sieciowy dla aplikacji Skoczek

Protokół ten jest protokołem tekstowym. Wykorzystuje standardowe klasy i interfejsy języka Java, gniazdka komunikacyjne oraz protokół TCP. Brak szyfrowania, przesyłane dane są w formacie String.

W zależności od liczby przesyłanych informacji pakiety mają różny rozmiar i liczbę znaków.

SCHEMAT PRZYKŁADOWEGO POŁĄCZENIA

Przykładowa informacja od klienta do serwera:

Klient: Wiadomość -> Serwer

- Nawiązanie połączenia

Po nawiązaniu połączenia przy pomocy protokołu TCP klient przesyła informację do serwera, że chce nawiązać z nim połączenie poprzez wysłanie informacji LOGIN

Klient: LOGIN\n -> Serwer

Serwer po otrzymaniu informacji, jeżeli akceptuje połączenie odpowiada klientowi, że został zalogowany oraz jaki jest jego numer porządkowy.

Serwer: LOGGED_IN 1\n -> Klient

W przypadku braku akceptacji na połączenie, serwer informuje klienta że chęć nawiązania połączenia została odrzucona.

Serwer: CONNECTION_REJECTED\n -> Klient

Po nawiązaniu połączenia możliwa jest dalsza wymiana informacji pomiędzy aplikacjami.

- Pobranie pliku konfiguracyjnego

W celu pobrania pliku konfiguracyjnego klient przesyła informację do serwera.

Klient: GET_SETTINGS\n -> Serwer

Serwer po otrzymaniu żądania, przesyła podstawowe ustawienia gry

Odebrane dane są odpowiednio obrabiane po stronie klienta.

- Pobranie pliku zasad gry

W celu pobrania pliku zasad gry klient przesyła informację do serwera.

Klient: GET_RULES\n -> Serwer

Serwer po otrzymaniu żądania, przesyła informacje o zasadach gry

Odebrane dane są odpowiednio obrabiane po stronie klienta.

- Pobranie listy najlepszych wyników

W celu pobrania listy najlepszych wyników klient przesyła informację do serwera.

Klient: GET_HIGH_SCORES\n -> Serwer

Serwer po otrzymaniu żądania, przesyła wyniki, które są potem przekształcone do postaci, którą wyświetla się.

Odebrane dane są odpowiednio obrabiane po stronie klienta.

- Pobranie poziomu

W celu pobrania pliku konfiguracyjnego klient przesyła informację do serwera.

Klient: GET_LEVEL: X\n -> Serwer

Serwer po otrzymaniu żądania, przesyła dane poziomu

Po otrzymaniu informacji, klient odpowiada informacją że dane zostały odebrane.

- Wysłanie wyniku na serwer

W celu wysłania wyniku na serwer, klient wysyła informację o nicku gracza i jego wyniku

Klient: GAME_SCORE:User@1000\n -> Serwer

Serwer po otrzymaniu żądania, umieszcza wynik w pliku.

Gdy wynik zapisanie wyniku powiodło się:

Serwer: GAME_SCORE_ACCEPTED\n-> Klient

Gdy nastąpił błąd:

Serwer: GAME_SCORE_REJECTED\n -> Klient

- Kończenie połączenia

Połączenie z serwerem zostaje zakończone tylko i wyłącznie na prośbę klienta. Aby tak się stało, musi on przesać informację od serwera

Klient: LOGOUT\n -> Serwer

Serwer po otrzymaniu informacji odpowiada klientowi że połączenie zostało zakończone i zamyka połączenie.

Serwer: LOGGEDOUT\n -> Klient

Klient po otrzymaniu informacji kończy połączenie.

W przypadku wystąpienia jakiegokolwiek błędu po stronie serwera przesyła on informację o awaryjnym zamknięciu połączenia

Serwer: CLOSE_CONNECTION\n ->Klient

Klient po otrzymaniu informacji odpowiada że połączenie zostało zakończone i zamyka połączenie z serwerem.

Klient: CONNECTION_CLOSED\n -> Serwer

Serwer po otrzymaniu tej informacji zamyka połączenie z klientem.