

Michał Malinowski

Tomasz Czaplicki

PROZE20L

Projekt protokołu sieciowego dla gry „Dyna Blaster”

Poniższy projekt protokołu sieciowego zostanie zaimplementowany w aplikacji „Dyna Plaster” autorstwa wyżej wymienionych. Będzie odpowiadał za wymianę informacji między klientem gry, a serwerem. Lokalny klient będzie pobierał z serwera następują informacje: pliki konfiguracyjne, rozmiar i schemat poziomów oraz aktualny ranking najlepszych wyników.

Implementowany protokół jest protokołem tekstowym, wykorzystującym klasy języka Java oraz protokół TCP. Pakiety mogą różnić się długością i rozmiarem (wynika to z różnej ilości przesyłanych informacji). Dane nie są szyfrowane. Będą przesyłane w jednej linii, znak nowej linii informuje o końcu przysyłania danych.

Schematy wymiany informacji pomiędzy klientem a serwerem:

1) Pobieranie plików konfiguracyjnych

K: GET_CONFIG\n => S

W odpowiedzi serwer przesyła do klienta parametry wraz z ich wartościami

S: wartosc#wartosc#...#wartosc# => K

Dziewięć kolejnych wartości typu INT oznaczające po kolei:

- Punkty za zabicie Wirusa
- Punkty za zabicie Złego Kota
- Punkty za zabicie Ducha
- Punkty za bonus Zdrowia
- Punkty za bonus Bomby
- Punkty za bonus Nieśmiertelności
- Maksymalna liczba żyć
- Współczynnik punktów czasu
- Współczynnik punktów zdrowia

2) Pobieranie danych poziomów

Odpowiada za wczytanie budowy poziomów w grze

K: GET_LEVELS\n => S

S: wartosc#wartosc#...#wartosc# => K

Kolejne wartości typu INT oznaczające po kolei:

- Liczba poziomów
- Wielkość poziomów (format: #wartosc1 wartosc2#)
- Rozdzielczość bloków
- Parametry poziomów
(format: #X11 X12 X13 ... X1n#X21 X22 X23 ... X2n#...#Xm1 Xm2 Xm3 ... Xmn#)

Xmn – określa, co znajduje się na polu

- 0 - pusto
- 1 - kamień
- 2 - drewno
- 3 - portal nieodkryty
- 4 - portal odkryty
- 5 - Wirus
- 6 - Serce
- 7 - Zły Kot
- 8 - Duch
- 9 - Złota Bomba
- 10 - Nieśmiertelność
- 11 - Bomber

3) Pobieranie rankingu najlepszych graczy

Lista będzie pobierana automatycznie po wybraniu w menu głównym zakładki Ranking

K: GET_RESULTS\n => S

W odpowiedzi serwer zwraca listę najlepszych graczy poczynając od pierwszej pozycji:

S: nick@score#nick@score#...#nick@score# => K

Zmienna nick jest typu STRING a zmienna score typu INT.

4) Wczytywanie zapisu z serwera

Odpowiada za wczytanie zapisu do gry

K: GET_SAVE\n => S

S: wartosc#wartosc#wartosc#wartosc# => K

Kolejne wartości typu INT oznaczają po kolei:

- Poziom, na którym skończono
- Liczba żyć
- Punkty
- Czas

5) Wysyłanie wyniku na serwer

Wynik będzie przesyłany na serwer automatycznie po ukończeniu wszystkich poziomów. Na serwerze będzie przechowywane 10 najlepszych wyników.

K: GAME_SCORE:nick@score => S

6) Wysyłanie zapisu na serwer

Odpowiada za zapisanie stanu rozgrywki na serwerze

K:GAME_SAVE:wartosc#wartosc#wartosc#wartosc# => S

Kolejne wartości typu INT oznaczają po kolei:

- Poziom, na którym skończono
- Liczba żyć
- Punkty
- Czas

W przypadku, gdy aplikacja nie ma możliwość korzystania z zasobów z serwera to wtedy używane są dane lokalne poprzedzone informacją o niemożności pobrania danych z serwera.