

Gra Dyna Blaster - etap 4

W naszym projekcie zaimplementujemy protokół sieciowy modelu klient-serwer. Serwer po otrzymaniu zgłoszenia sprawdza zgłoszenie przygotowuje odpowiedź i ją wysyła do klienta. Po czym wraca do oczekiwania na nowe zgłoszenie od użytkownika. Klient nawiązuje połączenie z serwerem. Klient wysyła żądanie, następnie oczekuje na odpowiedź i przetwarza otrzymaną odpowiedź.

W naszym modelu sieciowym wykorzystamy typ przesyłania danych tekstowych. Porcją przesyłu będzie linia tekstu kończąca się znakiem końca linii. parametry dla poszczególnych zdań protokołu, parametry separowane są zazwyczaj spacjami. Serwer odczytuje całe linijki z pliku konfiguracyjnego i wysyła go do Klienta. **Klient po otrzymaniu odpowiedzi na żądanie zapisuje odebrany String w odpowiednim pliku.**

Klient C prosi o wysłanie (int)liczby poziomów,(int) szerokość i wysokość okna :

C : **GET_SETTINGS** => S

Serwer S wysyła (int)liczby poziomów,(int) szerokość i wysokość okna(oddzielone spacjami):

S:number_of_levels width height =>C

Klient C stwierdza przejście do stanu systemowego RESULTS :

C : **GET_BEST_SCORES** => S

Serwer S wysyła (String) placeX=nick,score (przesyła 10 najlepszych wyników każdy wynik oddzielony spacją,X - numer miejsca)

S:placeX=nick,score =>C

Klient C stwierdza przejście do stanu systemowego NEWGAME lub NEXTLEVEL żąda (odpowiedniego levelu X - numer levelu):

C : **GET_LEVEL-X** => S

Serwer S wysyła (String) number_of_boxs_width, number_of_boxs_height (String) name_level, (String) looks_of_map (number_of_boxs_width number_of_boxs_height - ilość obiektów w szerokości i wysokości np skrzynek lub ścian, looks_of_map-ciąg cyfr oznaczający wygląd mapy kolumny oddzielone średnikami a wiersze przecinkami) (każdy z parametrow oddzielony spacjami)

S:number_of_boxs_width number_of_boxs_height name_level looks_of_map =>C

Klient C prosi o wysłanie atrybutów gry ilości zdrowia, liczby dostępnych bomb,dostępny czas:

C : **GET_GAME_ATTRIBUTES** => S

Serwer S wysyła atrybuty gry(int) ilości zdrowia,(int) liczby dostępnych bomb,(int)dostępny czas (wszystko oddzielone spacjami):

S: health max_amount_of_bomb max_game_time=>C

Klient C stwierdza przejście do stanu systemowego GAMEEND wysyła nick(int) i score(int):

C : **SEND_SCORE** -nick-score=> S

(Serwer wywołuje funkcję która sortuje 10 najlepszych wyników i wynik przesłany od użytkownika i zapisuje do pliku na serwerze)

Serwer S wysyła odpowiedź potwierdzającą zapisanie wyniku

S:Your score is save =>C

GET_BOMB_CONFIG

Klient C prosi o wysłanie atrybutów bomby czas do wybuchu

C : **GET_BOMB_CONFIG** => S

Serwer S wysyła atrybuty bomby (int) czas do wybuchu

S: time_to_explode=>C

Klient C prosi o wysłanie atrybutów ognia czas do zabrania hp

C : **GET_FIRE_CONFIG** => S

Serwer S wysyła atrybuty ognia (int) czas do zabrania hp

S: time_to_damage=>C

Klient C prosi o wysłanie zdań do wyświetlenia w sekcji HELP

C : **GET_HELP** => S

Serwer S wysyła treść pliku txt z opisem gry (każda linijka została oddzielona @,ze względu na występowanie spacji został użyty znak @)

S: helptext=>C