Karolina Pawluk, PROZE

Gra Asteroids

Protokół sieciowy

Opisany protokół sieciowy ma za zadanie umożliwić nawiązanie połączenia pomiędzy klientem a serwerem oraz pozwolić na wymianę informacji (przesyłanie danych) pomiędzy nimi. Protokół ten będzie umożliwiał przesyłanie plików konfiguracyjnych potrzebnych do uruchomienia programu, przesyłanie listy rankingowej, dodawanie do niej użytkowników pobieranie liczby asteroid (parametru poziomu), przesyłanie informacji o stanie zakupu produktu, ilości punktów, imienia oraz wybranego statku i poziomu z/do serwera.

Rodzaj protokołu: Tekstowy

Typ pakietu: Struktura o zmiennej długości

Separator: Spacja

Koniec pakietu: Znak nowej linii

**Odczyt aktualnej liczby rekordów (HighScores):**

C: GetHighScores => S – żądanie wysłania listy rekordów

S: HighScoresResponse {id} {score} … {id\_N} {score\_N}

=> C – serwer zwraca listę N=10 rekordów, odpowiedź zbudowana jest na zasadzie {id} {score}, gdzie {id} oznacza nick gracza (string), zaś {score} jest liczbą całkowitą reprezentującą wynik punktowy danego gracza

**Dodanie/Odrzucenie nowego rekordu (GameScore):**

C: GameScore {id} {score} => S – nowy wynik do dodania, gdzie {id} oznacza nazwę gracza (string), zaś {score} jest liczbą całkowitą reprezentującą wynik punktowy danego gracza

S: UserScoreAccepted => C – potwierdzenie przyjęcia rekordu, serwer zapisuje wynik w pliku konfiguracyjnym i uaktualnia ranking

S: UserScoreRejected => C – odrzucenie wyniku, zawiera informację, że wynik nie zakwalifikował się do listy najlepszych wyników

**Wczytanie poziomu (LoadLevel):**

C: GetLevel => S – żądanie wysłania parametrów poziomu gry

S: LevelEnemies {N} {M} => C – liczba asteroid do zniszczenia, gdzie N jest liczbą małych asteroid, a M liczbą dużych asteroid N, M – liczby całkowite

**Wczytanie plików konfiguracyjnych (GetSettings)**

C: GetSettings => S – żądanie wysłania ustawień konfiguracyjnych aplikacji

S: ReturnSettings => C – zwraca cały plik konfiguracyjny

C: GetShopSettings => S – żądanie wysłania ustawień konfiguracyjnych sklepu

S: ReturnShopSettings => C – zwraca cały plik konfiguracyjny

C: GetGameplaySettings => S – żądanie wysłania ustawień konfiguracyjnych gry

S: ReturnGameplaySettings => C – zwraca cały plik konfiguracyjny

C: GetHelp => S – żądanie wysłania pliku tekstowego pomocy

S: ReturnHelp => C – zwraca cały plik tekstowy

**Przesyłanie nowych danych i zapis po stronie serwera (Save)**

C: ChangeShip {id} => S – żądanie zapisu wybranego statku, id to nr statku

S: ShipChanged => C – statek został zapisany do pliku

S: ShipNotChanged => C – statek nie mógł zostać wpisany do pliku na skutek błędu

C: ChangeName {Name} => S – żądanie zapisu wybranego imienia, name to nick gracza

S: NameChanged => C – imię zostało zapisane do pliku

S: NameNotChanged => C – imię nie mogło zostać wpisane do pliku na skutek błędu

C: ChangeScore {id} => S – żądanie zapisu wybranego wyniku, id to ilość punktów

S: ScoreChanged => C – wynik został zapisany do pliku

S: ScoreNotChanged => C – wynik nie mógł zostać wpisany do pliku na skutek błędu

C: SaveBoughtThings {score} {id}=> S – żądanie zapisu ilości punktów po dokonaniu zakupów i  
 ilości życia, score to ilość punktów, id to liczba żyć

S: BoughtThingsSaved => C – dane zostały zapisane do pliku

S: BoughtThingsNotSaved => C – dane nie mogły zostać wpisane do pliku na skutek błędu

C: SaveShop {id} => C – żądanie zapisu zakupu wybranego produktu, id to nr  
 produktu

S: ShopSaved => C – produkt został zapisany do pliku

S: ShopNotSaved => C – produkt nie mógł zostać wpisany do pliku na skutek błędu

**Zestawienie połączeń**

C: Login => S – prośba o połączenie z serwerem

S: LoggedIn => C – odpowiedź serwera, klient zalogowany

S: ConnectionRejected => C – klient nie został zalogowany

C: Logout => S – prośba o wylogowanie

S: LoggedOut => C – wylogowano klienta

C: CloseConnection => S – prośba o zamknięcie połączenia

S: ConnectionClosed => C – zamknięcie połączenia

S: InvalidCommand => C – błędne polecenie klienta