

Działanie serwera

Funkcjonalność sieciowa obejmuje serwer przechowujący i udostępniający informacje konfiguracyjne: listę najlepszych wyników, definicje wyglądu plansz, poziomów gry, scenariusza gry oraz wszelkie pozostałe parametry aplikacji. Serwer obsługuje dowolną liczbę klientów. Serwer sieciowy jest niezależną, bezobsługową, parametryzowaną aplikacją. Aplikacja w wersji sieciowej działa także bez serwera – z wykorzystaniem lokalnych plików konfiguracyjnych (niezależnych od serwera). Adres serwera i numer portu są parametrami aplikacji klienta.

Działanie klienta

Uruchomienie klienta wiąże się z dwiema opcjami: aplikacja zostaje uruchomiona z użyciem plików lokalnych albo zostaje umożliwione logowanie się poprzez podanie adresu hosta i numeru portu. Następnie zostaje wysłane żądanie przez klienta do serwera w celu uzyskania plików konfiguracyjnych. Na końcu gry zostaje wysłane żądanie aby lista wyników została zaktualizowana.

Podstawowe informacje o protokole

Protokół jest typu tekstowego. Dane zostają przesłane w jednej linii a znak nowej linii \n informuje, że jest to koniec przesyłanych danych.

Działanie protokołu

Zawsze w sytuacji gdy klient wysyła żądanie do serwera, ustanawiane jest połączenie, po otrzymaniu odpowiedzi na zadane żądanie, następuje przerwanie połączenia.

- Klient kieruje prośbę do serwera o wysłanie plików konfiguracyjnych C: `get_configuration => S`
- Serwer wysyła odpowiedź w postaci przesłania klientowi parametrów konfiguracyjnych globalnych np. Wymiary okna) oraz poziomu (np. Numer poziomu) S: `give_configuration maps => C`
- Klient wysyła do serwera listę najlepszych wyników, wyniki (`ArrayList<Intiger>`), oraz nazwy-nicki (`ArrayList<stringr>`), C: `save_score score nick => S`
- Serwer wysyła odpowiedź, informując użytkownika czy wyniki dotarły został wpisany na listę (boolean) `if_saved => C`
- Klient wysyła prośbę do serwera o udostępnienie informacji o pozycji gracza w rankingu (`intiger`) `score` oraz (`string`) `nick` do udostępnienia C: `get_rank place => S`
- Serwer wysyła odpowiedź do klienta, wysyłając wynik rankingowy (`intiger`) oraz nazwę (nick) gracza (`string`) na danej pozycji. S: `give_score score nick => C`
- Klient wysyła prośbę do serwera o definicję wyglądu mapy o podanym numerze C: `get_config_level map_index => S`
- Serwer wysyła odpowiedź wysyłając parametry(`intiger[]`) na podstawie których program narysuje ściany mapy (parametr `Wall_width` i `Wall_height`), gracza(parametr `Player_height`, `player width`) oraz nada mu prędkość (parametr `Player_Velocityx` i `Player_Velocityy`), narysuje piłki(`Ball_Height`, `Ball_Width`), oraz nada im prędkość(`Ball_Speedx`, `Ball_Speedy`), narysuje pociski(`Bullet_Height`, `Bullet_Width`) oraz nada im prędkość(`Bullet_Speedy`). . S: `give_map_configuration_parameterts=> C`.