

## *Projekt protokołu sieciowego dla aplikacji Sokoban*

- **Zarys zadania:**  
Zgodnie z założeniem, celem IV etapu zadania programistycznego w semestrze 2020Z z przedmiotu Programowanie Zdarzeniowe jest implementacja oraz opis działania protokołu sieciowego dla wybranej aplikacji. Zadaniem protokołu sieciowego jest komunikacja na linii klient-serwer, zaimplementowana przy użyciu języka Java oraz dostępnych w nim komponentów, np. gniazda komunikacyjne.

Komunikacja pomiędzy klientem, a serwerem umożliwia wzajemną wymianę informacji w obrębie dostępu do plików konfiguracyjnych, listy map, tekstur, awatarów, czy też listy najlepszych wyników. Pomiędzy klientem i serwerem są wymieniane, więc zarówno linie tekstu, jak i bardziej skomplikowane obrazy png.

- **Sposób uruchomienia:**  
W momencie uruchamiania programu należy:
  1. Skompilować i uruchomić serwer (do tych czynności mogą posłużyć specjalnie przygotowane programy zawarte w podkatalogu Serwer).
  2. Z uwagi na połączenie serwera i klienta za pomocą numeru portu oraz hosta serwer na samym początku automatycznie generuje plik, gdzie są zawarte wszystkie niezbędne informacje do zaczącia gry poprzez serwer. Wygenerowany plik (DaneWyjscioveSerwera.txt, który znajduje się w tym samym miejscu co programy do kompilacji i uruchomienia serwera) należy przenieść do folderu Klient->src->com->sokoban->config w celu możliwości odczytania go przez serwer.
  3. Uruchomić klienta i wybrać interesującą nas opcję. Zgodnie z założeniami gra jest również możliwa w trybie offline, bez udziału serwera (wtedy w menu głównym gry należy wybrać opcję Game).
- **Schemat przykładowego połączenia oraz możliwych opcji:**  
Na samym początku następuje połączenie klienta z serwerem poprzez zestaw unikalnych danych: host i port.  
W przypadku wybrania opcji gry serwerowej jest nawiązywane w/w połączenie.
  1. W przypadku zdecydowania się w podmenu Server Game na rozpoczęcie gry klient korzysta z modelu awatarów otrzymanych poprzez komunikację z serwerem. Aby otrzymać te dane:
    - I. Klient wysyła prośbę do serwera o treści „Load\_avatars!n”, gdzie n to numer awataru, który ma zostać wczytany.
    - II. Serwer otrzymując żądanie przetwarza je, a następnie przesyła odpowiedni awatar z dostępnej mu listy awatarów.
    - III. Klient otrzymuje odpowiedź, po czym ten zestaw danych jest zapisywany do odpowiedniej tablicy wszystkich dostępnych postaci w grze.Po wybraniu odpowiedniego awataru użytkownik wpisuje swoją nazwę gracza, która potem będzie przetwarzana w celu możliwości prowadzenia spisu najlepszych graczy.
  2. Klient przechodzi do mapy, która również została otrzymana w wyniku odpowiedzi serwera na żądanie klienta.

- I. Klient wysyła żądanie do serwera o treści „Load\_levels!n”, gdzie n to numer mapy, która ma zostać wczytana.
  - II. Serwer otrzymując żądanie przetwarza je, generuje odpowiednią linię znaków (wszystkie rzędy mapy są oddzielone znakiem „/”) symbolizującą mapę, a następnie przesyła odpowiedź.
  - III. Klient otrzymuje odpowiedź, po czym ta linia jest rozkodowywana, a następnie tworzona jest z niej mapa zapisywana do zbioru wszystkich map w grze.
3. Zarówno liczba map, jak i pierwsza i ostatnia mapa (a więc i wybór) map, a także ilość żyć, początkowa liczba żyć i kroków są pobierane z serwera za pomocą żądania „Load\_config” z pliku konfiguracyjnego. Serwer dostając żądanie przetwarza je, odpowiednio koduje dane (jak wyżej) oraz odsyła je klientowi. Klient otrzymując dane, obrabia je, a następnie używa podczas korzystania z programu.
4. Na mapie znajdują się również inne elementy graficzne pobrane z serwera oprócz awatarów. Są to wszelkiego rodzaju tekstury.
  - I. Klient wysyła żądanie do serwera o treści „Load\_textures!n”, gdzie n to numer tekstury, która ma zostać wczytana.
  - II. Serwer otrzymując żądanie przetwarza je, a następnie przesyła odpowiednią teksturę z dostępnej mu listy tekstur.
  - III. Klient otrzymuje odpowiedź, po czym ten zestaw danych jest zapisywany do odpowiedniej tablicy wszystkich dostępnych tekstur w grze (ściany, punkty docelowe, bloki).
5. Podczas gry na serwerze klient ma również możliwość podglądu tabeli z najlepszymi wynikami. Załadowanie tabeli opiera się na przesłaniu prośby od klienta „Load\_ranking”, odczytaniu tej wiadomości przez serwer, przetworzeniu jej oraz odesłaniu listy najlepszych wyników w postaci ciągu znaków. Klient ma również możliwość przesłania swojego wyniku na koniec gry do listy najlepszych wyników na serwerze. W tym celu wysyła wiadomość „Save\_ranking!Imię/Punkty”, wiadomość jest odbierana przez serwer oraz porównywana z listą najlepszych wyników. W przypadku przesłania wyniku lepszego jest zwracana klientowi wiadomość potwierdzająca ten zapis, a w przypadku wyniku gorszego informująca o konieczności poprawy.