

Zadanie 13

Programowanie Java

Celem dzisiejszego laboratorium jest stworzenie aplikacji klient-serwer, umożliwiającej przesyłanie plików przez sieć. Aplikacja powinna wykorzystywać gniazda (sockets) do komunikacji i obsługiwać transmisję danych w uporządkowany sposób, aby zapewnić dokładność i niezawodność.

Wymagania funkcjonalne odnośnie serwera

1. Akceptowanie połączeń przychodzących od klientów za pomocą `ServerSocket`.
2. Po otrzymaniu żądania od klienta:
 - Odczytanie żądanego pliku z lokalnego systemu plików serwera.
 - Wysłanie rozmiaru pliku (jako liczba całkowita) do klienta.
 - Wysłanie nazwy pliku.
 - Wysłanie zawartości pliku.
3. Obsługa przypadków, gdy żądany plik nie istnieje, i odpowiednie informowanie klienta.
4. Wyświetlanie informacji dotyczącej transferu, w tym adresu IP klienta i szczegółów transferu plików.

Wymagania funkcjonalne odnośnie klienta

1. Połączenie z serwerem za pomocą `Socket`.
2. Żądanie konkretnego pliku według nazwy.
3. Odbieranie i przetwarzanie danych w następującej kolejności:
 - Rozmiar pliku (najpierw).
 - Nazwa pliku (druga).
 - Zawartość pliku (trzecia).
4. Zapisywanie otrzymanego pliku lokalnie z prefiksem (np. `received_`).
5. Wyświetlanie zawartości pliku na konsoli (zakładając, że plik jest tekstowy).
6. Wyświetlenie informacji dotyczącej transferu, w tym adresu IP serwera, adresu IP lokalnej maszyny, oczekiwanego rozmiaru pliku, odebranych bajtów i potwierdzenia zapisania pliku.

Wymagania нефункционалне

Zapewnienie odpowiedniej obsługi błędów w scenariuszach takich jak: plik nie znaleziony na serwerze; przerwanie połączenia, odebranie nieprawidłowego lub uszkodzonego rozmiaru pliku.

Użyj odpowiednich klas Javy do komunikacji przez gniazda, takich jak `InetAddress`, `ServerSocket` i `Socket`. Kolejność transmisji (rozmiar pliku, nazwa pliku, zawartość pliku) musi być ściśle przestrzegana, aby uniknąć niezgodności danych. Użyj bufora o rozmiarze 4096 bajtów dla transmisji danych. Użyj adresu `localhost` i portu 12345 do testowania, aby uprościć konfigurację i debugowanie.