

# Komunikator typu IRC - projekt zaliczeniowy

## 1. Opis struktury projektu

Projekt składa się z dwóch głównych katalogów: *client* i *server*, w których znajdują się źródła klienta i serwera. W obu podprojektach wykorzystałem CMake do generowania plików Makefile.

Na *Server* składają się następujące katalogi: *core* - zawierający podstawowe funkcjonalności i właściwą ich implementację, *models* - zawierający modele danych wykorzystywane w projekcie, *exceptions* - zawierający wyjątki generowane przez aplikacje w runtime'ie, *repository* - zawierający repozytoria, które przechowują dane serwera, jak konfiguracja czy lista użytkowników (pełnią w pewnym sensie funkcję takiej *in memory database*). W projekcie znajdziemy też katalog *utils* z parserami wykorzystywanymi do przetwarzania tekstu na modele.

Klient ma nieskomplikowaną strukturę, na którą składają się typowe dla QT pliki *.ui* z formularzami oraz implementacje w plikach źródłowych C++.

## 2. Opis procesu kompilacji

W katalogu serwera znajdują się 3 pliki *.ini* zawierające początkową konfigurację. W *config.ini* skonfigurujemy parametry serwera, jak port, na którym ma działać, czy rozmiary buforów. W pozostałych natomiast znajdują się dane o pokojach i użytkownikach.

Do kompilacji wymagana jest obecność w systemie narzędzia CMake. Po przejściu do głównego katalogu projektu wystarczy wykonać *bash ./build.sh*, a skrypt wygeneruje pliki wykonywalne znajdujące się odpowiednio w katalogach *client/build* oraz *server/build*. Każdy uruchomić można z terminala.

## 3. Opis protokołu komunikacyjnego

Dane przekazywane są za pomocą wiadomości oddzielonych od siebie średnikami. Każda przesłana do serwera wiadomość ma następujący format:

```
username;password;room;type;CONTENT
```

Username to unikalna nazwa użytkownika, password to hasło, room to unikalna nazwa pokoju, zaś type oznacza typ wiadomości, czyli, czy jest to standardowa wiadomość wysłana do innych użytkowników, czy odpowiednia interakcja z serwerem. Specjalne instrukcje interakcji z serwerem są następujące:

- create\_account - tworzenie nowego konta użytkownika
- join\_room - dołączanie do pokoju
- leave\_room - opuszczanie pokoju
- login - logowanie do serwera
- logout - wylogowywanie z serwera
- get\_rooms - prośba o listę wszystkich dostępnych pokoi
- get\_rooms\_for\_user - prośba o listę wszystkich pokoi, do których należy użytkownik
- get\_online\_users - prośba o listę wszystkich aktywnych użytkowników w podanym pokoju

Wiadomość zwrotna to albo zwykła wiadomość od użytkownika z parametrami również rozdzielonymi średnikami lub wiadomość od serwera rozpoczęta prefixem [srv]:.