



## 1 Project goal

The goal of this project is to allow the student to demonstrate their ability to create pairs of applications in the client-server architecture. The project puts much emphasis on the input-output (I/O) streams aspect — not only regarding sockets, but also in the context of text and binary files.

## 2 Project content

As a result of the project's implementation, two applications should be created — a *client* and a *server*. The server should act as a communication layer between the clients.

### 2.1 Server's behavior

The server application, when run, should load its configuration from a file specified in the server class' constructor. This file should specify:

- Server's port
- Server's name
- List of banned phrases

Once the server is up and running, it should enable an arbitrary number of clients to connect to it. When a client is connected, the server registers it by remembering the client's name and port.

When a client sends a message to the server, that message should be broadcast to every *recipient*. It is the client who determines who is the recipient of the message they have sent.

The only exception to that would be if the client's message contains any of the banned phrases (they are specified by the server's configuration file). In such cases, the server should inform the client that their message will not be sent due to it containing a specific banned phrase.

## 2.2 Client's behavior

Once the server is up and running, a client application shall be started in order to establish a connection with the server.

If a connection to the server cannot be established, a suitable error message should be printed. If such connection succeeds, the client should receive from the server a list of all currently connected clients (in the form of their usernames), accompanied by instructions on how to use the messaging facility.

The client should be able to:

- Send a message to every other connected client. This should be the default behavior of sending messages.
- Send a message to a specific person using their username.
- Send a message to multiple specific people.
- Send a message to every other connected client, with exception to some people (specified by the client).
- Query the server for the list of banned phrases.

Whenever someone disconnects themselves from the server, every client should receive a notification about that event.