

## Multi-Client Chat Application

1. Main features of the Application are presented in the following paragraphs:
  - It implements a server-side and a client-side runnable main method.
  - A client-side application can communicate with other clients connected to the server.
  - Communication between clients is not implemented through a direct link between clients, but through the server.
  - Server receives messages from the clients and broadcasts them to other clients connected to it.
  - Server sends information automatically to all connected clients whenever a new client connects to it or when an existing client quits the chat session.
  - Each client can see in its console the names of the users involved in chatting.
  - Client can quit the chat session by typing **“quit”** at the console.
  - Application can support an unlimited number of clients' connections.
  
2. Application implementation details are presented in the following paragraphs:
  - Server-side application is made up of two classes, Server and ClientHandler.
  - Client-side application is made up of only one class, Client.
  - Broadcasting of messages between clients is not implemented directly in the Server class, but through the ClientHandler class.
  - For each Socket connection created by a client, a ClientHandler object is created, and all the ClientHandler objects are stored in an ArrayList<ClientHandler>.
  - ArrayList<ClientHandler> is the key to broadcasting functionality of the chat application.
  - Client-side application is implementing two number of Threads, one is the main Thread and the second is implemented through listenForMessage method.
  - The two Threads design and implementation make it possible for the application to send and receive messages at the same time.
  - When a client quits the chat, the client is removed from the ArrayList<ClientHandler> and the application is always up to date with the clients that are connected to the chat.
  - Each client is identified by the username he/she enters at the beginning of a chat session. Broadcasting of messages is based on these usernames (i.e., a message sent by one user to the server is not broadcasted back in the user console)
  - ClientHandler object can see the message is sent by the client with a specific username and is doesn't broadcast it again at the same client's concole.)