

## EXERCISE 3: RPC-BASED CHAT

### Learning Objectives

By the end of this exercise, you should be able to:

- Implement a basic RPC service

### Exercise statement

You are asked to develop a chat application based on client-server architecture. The system specifications are detailed below:

- Both client and server source code must be written in C or C++.
- The software must be designed to implement *Remote Procedure Calls*. Specifically, the tool *rpcgen* must be used for the calls that clients make to server.
- The server (1) must maintain a local file that saves all chat messages, and (2) must publish two methods to clients, namely:

```
1 | void write(char *string) // This method inserts a new line into the server's local file.  
2 | char* getChat(void) // This method returns all lines from the server's local file.
```

Note that these two methods are indicative and can be improved, for example, *write* procedure could return an integer to ensure that the operation has performed correctly; Fetching only retrieves the new lines written, etc...

- The client **cannot** maintain any local files with the chat lines. For this reason, the client will have to consult the server every second, using the *getChat* method to retrieve new information.
- Obviously, every time the client wants to write a new line in the chat, it will have to use the write<sup>1</sup> method.
- The client will receive the user's nickname as an argument, so in order to run the client you will need to run the following command:

---

<sup>1</sup>Therefore, it seems that on the client side there will be two processes that will need to be harmonized using *ncurses*: one that will be responsible for refreshing the chat screen and the other one that listens to the keyboard. My goodness definitely knows no bounds :).

```
1 | ./clientChat <chatServerIP> <nickname>
```

- A possible client execution would be:

```
1 | ./clientChat localhost Benito
2 |
3 | ..... Welcome Benito!
4 | Zipi: Hi there! How is it going?
5 | Zape: So far, so good bro...
6 |
7 | Benito --> _
```