

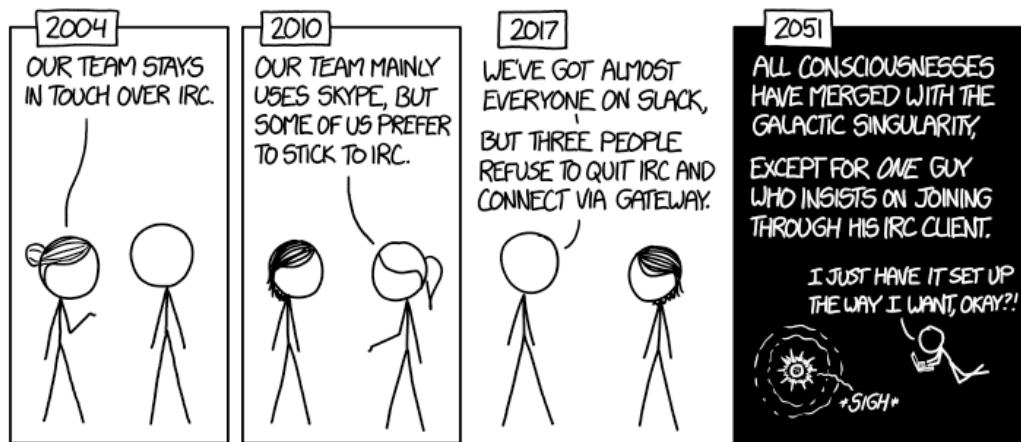
# IRC

AN INTERNET RELAY CHAT CLIENT AND SERVER



# IRC

You need to create an IRC server with **NodeJS** + **ExpressJS**, and a client in **ReactJS**.



Your server accepts **multiple simultaneous connections** and implements *channels*:

- ✓ it must be possible to join several *channels* simultaneously ;
- ✓ must be able to create, rename and delete *channels* ;
- ✓ a *message* must be displayed when a user joins or leaves a *channel* ;
- ✓ users must, of course, be able to speak in the *channels* they have joined.



Socket.IO

*Channels* and *messages* must be persistently preserved.



The persistence can be done with the method that you think is best: file, database, ...

Each user must give a *nickname* before they can use the application.



No authentication system is required, however it would be a welcome bonus.

The client and the server must communicate with each other, with the protocol of your choice.

## Commands

On the client side, each user should be able to do the following actions (using the specified command in chat, and using the interface):

- ✓ `/nick nickname`: define the nickname of the user on the server.
- ✓ `/list [string]`: list the available channels from the server. If **string** is specified, only displays those whose name contains the string.
- ✓ `/create channel`: create a channel with the specified name.
- ✓ `/delete channel`: delete the channel with the specified name.
- ✓ `/join channel`: join the specified channel.
- ✓ `/quit channel`: quit the specified channel.
- ✓ `/users`: list the users currently in the channel
- ✓ `/msg nickname message`: send a private message to the specified nickname.
- ✓ `message`: send a message to all the users on the channel



{EPITECH}  
LEARN DIFFERENT\*

\* apprendre autrement