# Technical Test - Chat Application

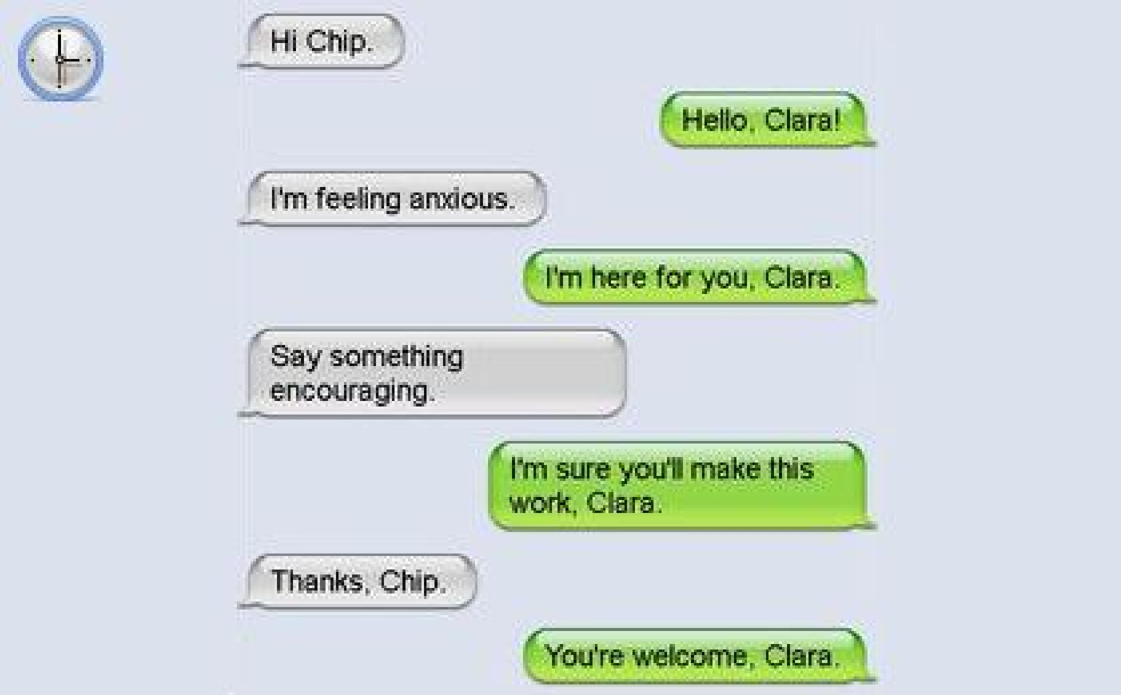
We would like you to build a web chat service using only the following components:

- HTML
- CSS
- Javascript
- React
- NodeJS
- NodeJS Libraries (That are not in themselves chat applications!)

The code should be made public in your GitHub account, you should do an initial commit stating the time you started the test, and put a time in your final commit message. While doing the task, you should commit your changes (with a description of what this commit includes), about every 5-10 minutes.

We would like to see some basic react components used, as well as components requiring composition.

The application should allow 2 users to chat using a web browser, and the messages should be styled using CSS so that your own messages are on the right hand side in a green bubble, and the other persons messages are on the left hand side in a grey bubble, so it looks something like this:



Hi Chip.

Hello, Clara!

I'm feeling anxious.

I'm here for you, Clara.

Say something encouraging.

I'm sure you'll make this work, Clara.

Thanks, Chip.

You're welcome, Clara.

You should have the option to issue the following commands:

- /nick <name> - sets your name for the chat
- /think <message> - makes the text appear in dark grey, instead of black
- /oops - removes the last message sent

If the user sets a nickname, their name should appear at the top of the chat window on the **other** person's browser.

The Node application you create for allowing communication should be as lightweight as possible, with most functionality happening on the front end. It would, however, be a bonus to show / use some type of webpack functionality for the javascript and CSS files.

Optionally, you can add the following extra features:

1. When the user is typing, indication that they are typing should be shown to the other user.

2. (smile) should produce a smiley face, (wink) should produce a winking face

3. When a new message arrives, it should slide in, and the messages above slide up

4. /fadelast - would fade out the last message to 10% visibility

5. /highlight <message> - would make the font of the message 10% bigger, and make the background 10% darker

6. /countdown <number> <url> - would start a visible countdown on the other persons browser, and at the end of the countdown redirect them to the URL specified.  e.g.:

    ○ typing ‘/countdown 5 http://www.test.com’

      The other person's browser would show 5..4..3..2..1.. then open a new window to www.test.com