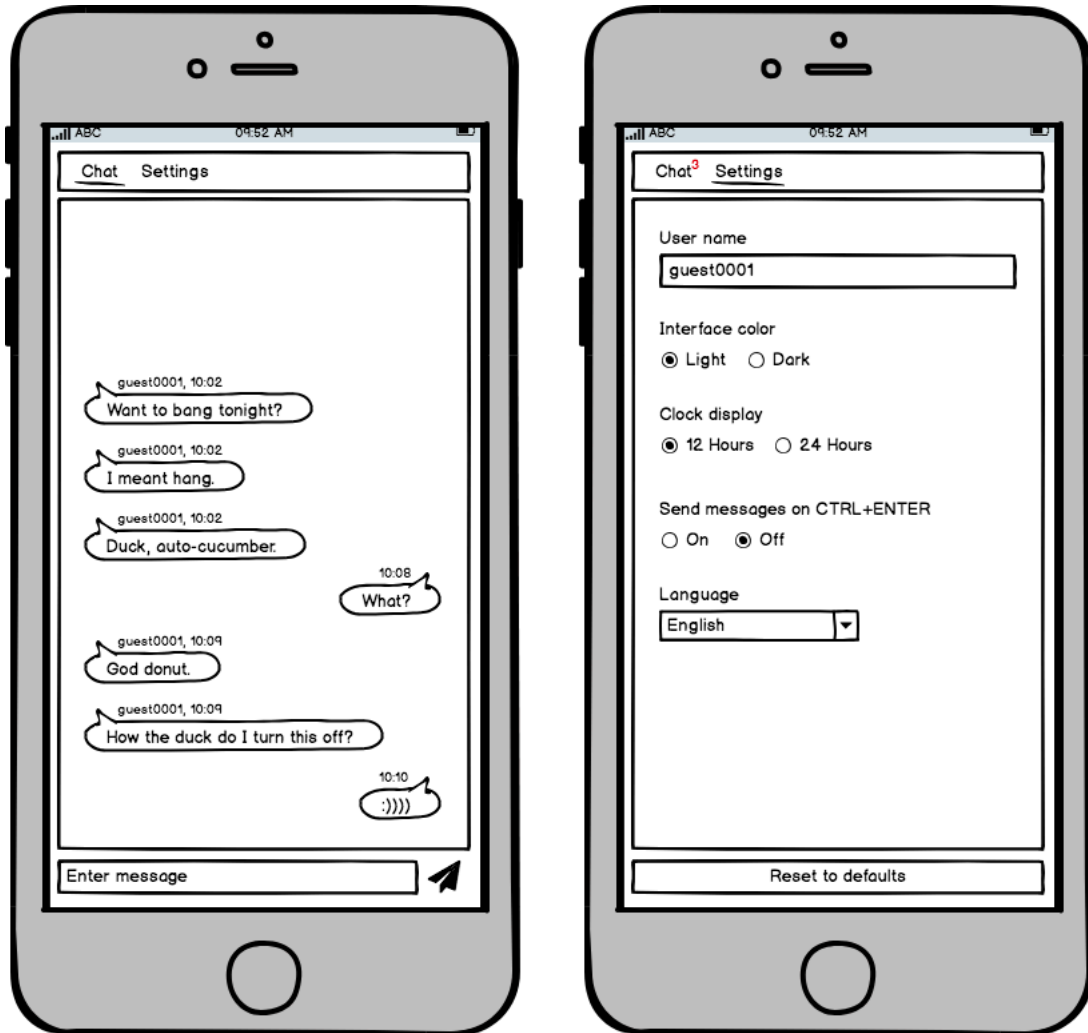


# Frontend homework

Create a single page application based on following mockups:



Application should contain **2 pages**:

1. Chat page
2. Settings page

---

*\* It is expected you implement your own chat server in node.js.  
Please, use socker.io.*

---

Example: <https://socket.io/get-started/chat>.

## Chat page

It's a [socket.io](https://socket.io) based chat interface. When the user opens the application, he/she meets this view.

It contains a chat message box, where the sent messages are floating right and the received messages are floating left. The message contains the text of the message, date time and the user name if it is not the current user.

There is a text input field and a send button at the bottom of the page.

If the user is on another tab and he/she gets a message, the chat tab will blink, until he doesn't read the message.

Optional:

- Smiles support;
- Unread messages count in the chat tab;
- Link parser;
  - YouTube link (embedded video should appear);
  - Link to an image (embedded image should appear);
  - All other links should appear as anchor;
- Any your own ideas are welcome;

## Settings page

User can modify the following settings:

1. User name;
2. Interface color;
  - a. Light;
  - b. Dark;
3. Clock Display;
  - a. 12 hours;
  - b. 24 hours;
4. Send messages on CTRL+ENTER;
  - a. On;
  - b. Off;

Optional:

- Internationalization (It's enough to have just one additional language);
- Add support to unread chat notifications;
- Any your own ideas are welcome;

All settings should be saved in local storage and there must be a button "Reset to defaults", by clicking on it application should fallback to default settings.

## Requirements

1. You have to use React as your framework;
2. You have to use CSS preprocessors;
3. You have to write the app in TypeScript;
4. It should work on every desktop and phone, so you have to make responsive design. And it has to work both portrait and landscape orientation;
5. It should work on desktop/tablet/phone at least on the following browsers: Chrome, Firefox and Safari. Consider the latest versions for each browser;
6. Please, **do not** use any tool like or similar to create-react-app;
7. Write at least some tests that would cover the main functionality of your app. We **do not** expect to be 100% test covered;
8. Write clean, commented, small and modularized code;
9. Working code, that works if we serve it with the http server and open in a browser;
10. README file that contains:
  - a. What is it;
  - b. How does it work;
  - c. How could we setup and run it;
  - d. Create a checkbox list of all the features required by this homework and check the ones you were able to accomplish;

### Optional:

- JSdoc markdown.
- CSS Modules or any other css solution.
- If your application needs any state management tool, prefer one of the following:
  - Apollo (apollo-link-state);
  - React Context API;
  - Redux;

---

*\* Nice and intuitive design with good user experience is an extra point.*

*\* Please remember to test your homework on Chrome, Firefox and Safari.*

*\* Remember there are Mac, Linux, and Window users out here that will check your application.*

---