

# NTUEE Embedded System Lab, Spring 2018

## Homework 1

電機四 b03901014 張雅量

April 13, 2018

### Note

This document is for <https://github.com/amjltc295/Chatroom>. For my teammate (電機三 b04901014 陳力維)'s work, please see <https://github.com/b04901014/chatroom>.

### Features

This is a Facebook Messenger style chatroom based on ReactJS.

#### (a) Main Features

There are three main features: public room, Google translation and private conversation.

##### (1) Public Room

When users come to this page, they would see the public room with welcome message first. In this public room, users could talk to others. Everyone could see the message. Users joining / leaving this room would be reported by the Manager. Icons are the same (manager's) for all users because it could not yet be set, but their names are different.

NTUEE ESLab HW1

Name: 1TSRHeO9cmAuyckPAAAH Online users: 2

Public Room

I'm User B

15:48:49 GMT+0800 (CST)

Miss Google

Leave a message and I'll translate it for you in a

00:00am

66BLq\_xXd1wAr40-AAAJ

hi

15:33:55 GMT+0800 (CST)

Public Room

Manager - Allen

Hello, I am Ya-Liang Chang (Allen).

Manager - Allen

Welcome to NTUEE ESLab HW1 Demo!! Test

Manager - Allen

1TSRHeO9cmAuyckPAAAH joined this room

Manager - Allen

RqLMk5C0sv1d7V7PAAAF left this room

Manager - Allen

66BLq\_xXd1wAr40-AAAJ joined this room

66BLq\_xXd1wAr40-AAAJ

hi

66BLq\_xXd1wAr40-AAAJ

I'm User B

hello

I'm User A

NTUEE ESLab HW1

Name: 66BLq\_xXd1wAr40-AAAJ Online users: 2

Public Room

I'm User B

15:48:49 GMT+0800 (CST)

Miss Google

お元気ですか

15:34:10 GMT+0800 (CST)

1TSRHeO9cmAuyckPAAAH

hi

15:33:55 GMT+0800 (CST)

Public Room

Manager - Allen

Hello, I am Ya-Liang Chang (Allen).

Manager - Allen

Welcome to NTUEE ESLab HW1 Demo!! Test

Manager - Allen

66BLq\_xXd1wAr40-AAAJ joined this room

1TSRHeO9cmAuyckPAAAH

hello

1TSRHeO9cmAuyckPAAAH

I'm User A

hi

I'm User B

## (2) Google translation

By clicking Miss Google in the Messagers list on the left, users could talk to Miss Google. Miss Google is a kind woman that would translate users' words to a random language. Japanese, Chinese, French, Spain ... you may find a interesting response.

NTUEE ESLab HW1
Name: 66BLq\_xXd1wAr40-AAAJ
Online users: 2

**Msgagers**

Public Room
I'm User B
15:48:49 GMT+0800 (CST)

Miss Google
我很好
15:50:19 GMT+0800 (CST)

1TSRHeO9cmAuyckPAAAH
hi
15:33:55 GMT+0800 (CST)

Miss Google

Welcome, dear User1, I'm Miss Google.

Leave a message and I'll translate it for you in a random language!

Miss Google
你好

Miss Google
hi

Miss Google
Hola

Miss Google
お元気ですか

Miss Google
我很好

hello

hi

hi

how are you

I'm fine

### (3) Private Conversation

By clicking other users' icon in the public room, users could start a private conversation with a specific user. This conversation is private, which means other users could not see those messages. The one be clicked would see a pop-up thread from that user, and could reply him / her by that thread.

NTUEE ESLab HW1
Name: 1TSRHeO9cmAuyckPAAAH
Online users: 2

**Msgagers**

Public Room
I'm User B
15:48:49 GMT+0800 (CST)

Miss Google
Leave a message and I'll translate it for you in a
00:00am

66BLq\_xXd1wAr40-AAAJ
I'm User B
15:50:49 GMT+0800 (CST)

66BLq\_xXd1wAr40-AAAJ

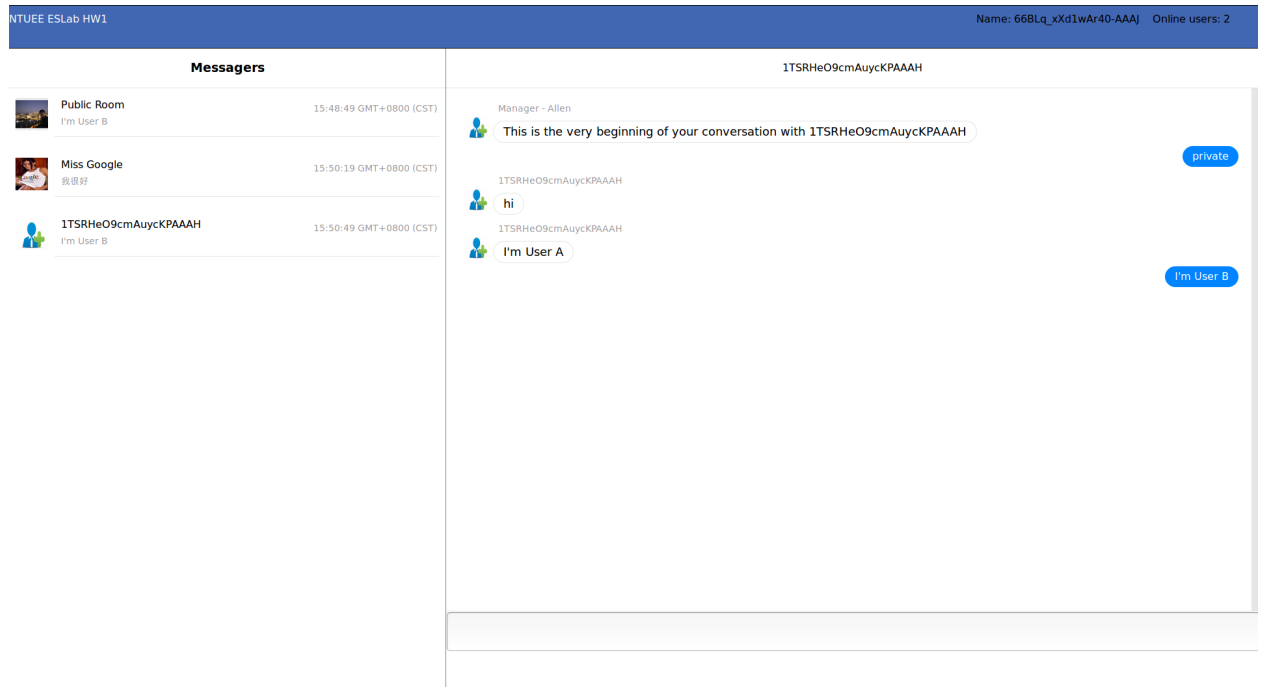
Manager - Allen
This is the very beginning of your conversation with 66BLq\_xXd1wAr40-AAAJ

66BLq\_xXd1wAr40-AAAJ
private

66BLq\_xXd1wAr40-AAAJ
I'm User B

hi

I'm User A



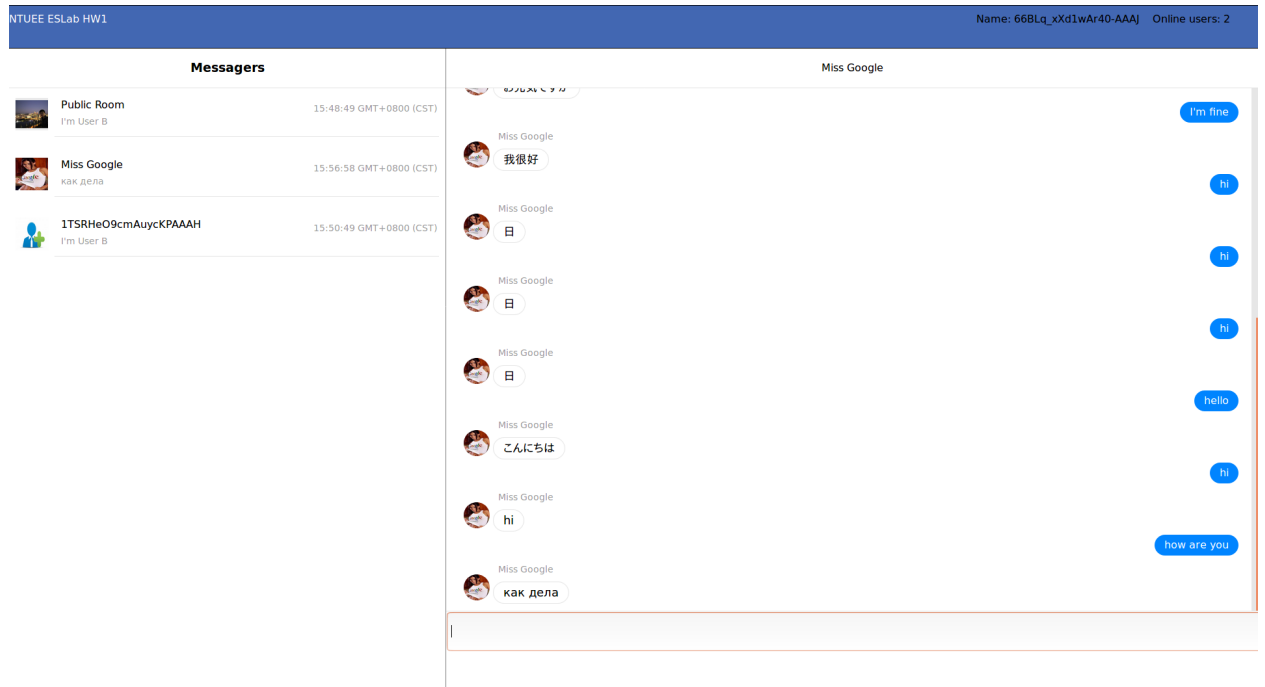
(b) Other Details

(1) User ID

Each user has a unique user ID, shown on the top-right corner.

(2) Automatic Scrolling

When new messages appear or input message exceeds the bottom, the program will automatically scroll to the latest message.



(3) Online User Number

On the top-right corner, users could see how many users are online.

(4) Message Time Meesage time is recorded and shown on the messenger list.

(5) User Icons User icons are displayed with their messages. However, users could not set their own icons; it is a feature to be developed. For now only Manager, Miss Google and New User icons are shown.

(6) Deployment

This program is deployed on Heroku (<https://b03901014eslabhw1.herokuapp.com/messenger>) Everyone could play with it!

## Program Structure

There are two main parts for this chat room: server and clients. It's a many (clients, chat room users) to one (server, chat room itself) system. Clients would send messages to the server, and the server would deal with these messages by sending them to other clinets or the client themselves after proper processing (translation for example).

(a) Communications

The communication between the server and clients is based on different events of SocketIO, which is a Javascript Library specifically for communication protocol between server / clients.

The server itself has only one component that deals with different socket events. The

server listens to these events from clients and do corresponding reponse.

In the client part, App.js would control other components. For example, if there is a Enter event with new message, App.js would react immediately and send the new message to the server. If there are any changed, React would re-render the page (update all components). By doing so, new messages or events could be shown instantly.

(b) Process

After "npm start", server.js would be launched. Users could access the chat room by going to corresponding page (localhost:5000/messenger for local and <https://b03901014eslabhw1.herokuapp.com/messenger> for deployment.) All logic and rendering process are in the same page, because it's written in ReactJS!

(c) Files

```
→ Chatroom git:(develop) tree
.
├── assets
│   └── css
│       ├── normalize.css
│       ├── reset.css
│       └── style.css
├── images
│   ├── example2.png
│   └── example.png
├── index.html
├── package.json
├── README.md
├── server.js
├── src
│   ├── components
│   │   ├── App.js
│   │   ├── MessageItem.js
│   │   ├── MessageList.js
│   │   ├── Messenger.js
│   │   ├── Navbar.js
│   │   └── UserInput.js
│   └── index.js
├── webpack.config.js
└── webpack.config.production.js

5 directories, 18 files
```

- (1) assets/  
CSS files for html style.
- (2) images/  
Images for README.md.
- (3) index.html

Entry point for the chat room.

- (4) package.json  
Dependencies for npm.
- (5) README.md  
Readme file for GitHub.
- (6) server.js  
Server to deal with all message sending from clients (chat room users). SocketIO is used to handle those message. Google-translate-api is used for Miss Google. Other packages are used for API and web settings.
- (7) src/  
Provides components for client user interface. App.js deal with main logic and rendering structure. Other .js files are for rendering as their names.
- (8) webpack.config  
Configuration for webpack.
- (9) node\_modules/  
After running "npm install", npm modules would be put here.