

Assignment 1

Individual Assignment

Due: 5pm Mon 2 September 2019

Introduction

You are required to build the server and front end for a chat system. For assignment 1, chat functionality is not required, you will be building the dashboard for the chat system, and in assignment 2 you will add the chat functionality along with other features. The chat system will allow users to communicate with each other in real-time within different groups and channels. Some users will have admin permission to add users to channels and groups, whilst a super admin has access to the entire site. The solution must be implemented using Node.js, Angular, and sockets. No user authentication is required for assignment 1.

Architecture

The system will have users which can have the following additional roles:

- Group Admin
- Super Admin
- Group Assis

A Group Admin has the ability to create groups. A Group Admin also has the ability to create channels or subgroups within groups. A Group Admin has the ability to create/invite users to a channel (if the user has already been created, they will simply be added to the channel). A Group Admin can remove groups, channels, and users from channels. A Group Admin can also allow a user to become a Group Assis of the group.

A Group Assis of a group can add or remove users in the group from channels within the group. A Group Assis of a group can also create channels within the group.

A Super Admin can create users with Group Admin role. A Super Admin can also remove users. A Super Admin can also provide another user with Super Admin role. A Super Admin also has Group Admin role.

A user is identified by their username. Initially there is one user called 'super' who is also a Super Admin. A user also has an email address (no emails are sent to the email address).

The first page of the website requires a user to enter their username, which is remembered in local storage. A user may 'logout' which also clears the username out of local storage. Once a user enters their username the page should display the groups they have been added to and the channels for each group.

For Super Admin, Group Admin and Group Assis, the page should also display the input forms according the role abilities for adding or deleting users, groups and channels, respectively.

Selecting a channel should display the channel history (which will be empty in assignment 1).

A text box should allow for new messages to be sent to the channel. New messages are broadcast to all users currently viewing the channel and added to the history (not required in assignment 1).

Data is stored by serialising JavaScript objects into JSON strings and stored in the file system.

Git

Git must be used during the development of the chat system. We recommend that you use GitHub and share the repository with your marker. You will be marked on frequent updates to the repository and the usage of git features.

Documentation

Documentation of your implementation is required. You will need to provide the following:

- Describe the organisation of your Git repository and how you used it during the development of your solution
- Description of data structures used in both the client and server sides to represent the various entities, e.g.: users, groups, channels, etc.
- Angular architecture: components, services, models, routes.
- Node server architecture: modules, functions, files, global variables.
- A description of how you divided the responsibilities between client and server (you are encouraged to have the server provide a REST API which returns JSON in addition to a static directory)
- A list of routes, parameters, return values, and purpose in the sever side
- Describe the details of the interaction between client and server by indicating how the files and global vars in server side will be changed and how the display of each angular component page will be updated.