One Messaging Comp 260 Technical Poster

Overview

The web is constantly seeking to modernise and adapt over its current and previous features. A key area of this adaptation is the methods in which clients communicate with one another. Therefore, the distributed system I propose features ,an increasingly popular [1], one to one messaging system inspired by 'live chat' (A term used to describe real time communication), the modern approach to previously used static contact forms. The skills I hope to achieve from this project are better expierence with both back-end data managment, through databases and realtime communication, provided by the web socket features.

Distributed system

This artifact relates to the distributed system specialism by communicating data across multiple systems to create the core messaging component whilst using mimicry of an external database (SQLite is internal to the source code and cannot be considered a true external database but does mimic the effects of one) too achieve the distributed portion of the specialism. With the additional rooms and persistent storage features to expand this further.

Project schedule

Firstly, (as presented above) the front-end interface will be created. A user study the design will then be issued and results implemented. Following this the registry system will be created along with the SQLite database. Finally the web socket functionallity will be implemented to establish the miniumum viable product. A secondary user study can then be issued for the first build and feedback applied. Additional out of scope features can then be implemented once this stage had been achieved.

Main user interface wireframe

Features and Structure

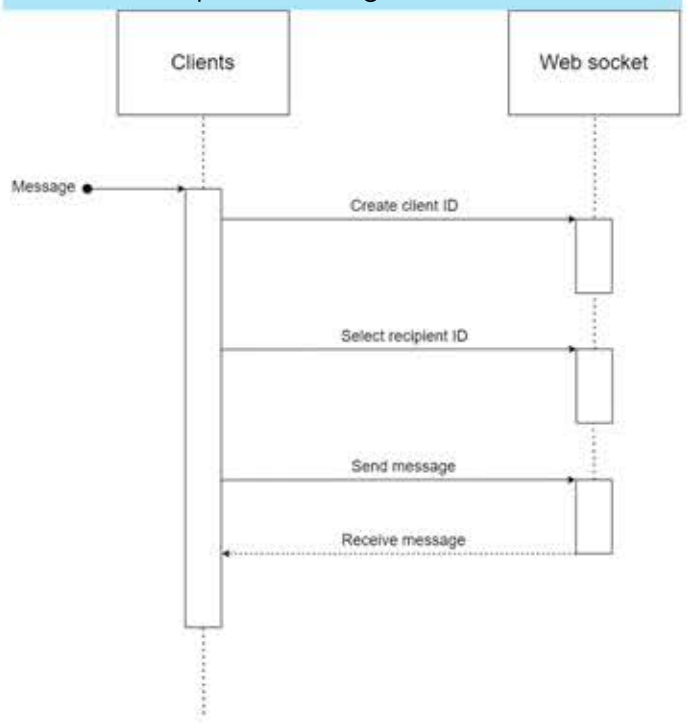
• A front-end user interface providing a input field for messages, chat box for viewing sent and received messages, time of message and client ID sent to and received from. With additional labels for displaying currently logged client IDs and which ID is currently in use.

• Realtime message sending and received between clients through web sockets.

• Client registering and logging system, stored to database.
With additional stretch features (as mentioned previously) being persistent storage of messages and client IDs even if the application has closed and session management for multiple rooms simultaneously.

The artifacts structure includes: Front end is comprised of HTML and CSS. With handlebars for its simplicity and templating use.
Back end uses express.js framework due to its versatility and compatibility with libraries. With an SQLite database for storing client information, SQL specifically as data stored will be structured and not unique. Finally, the socket.io library as it offers additional features above standard web sockets.

Artifact Sequence diagram

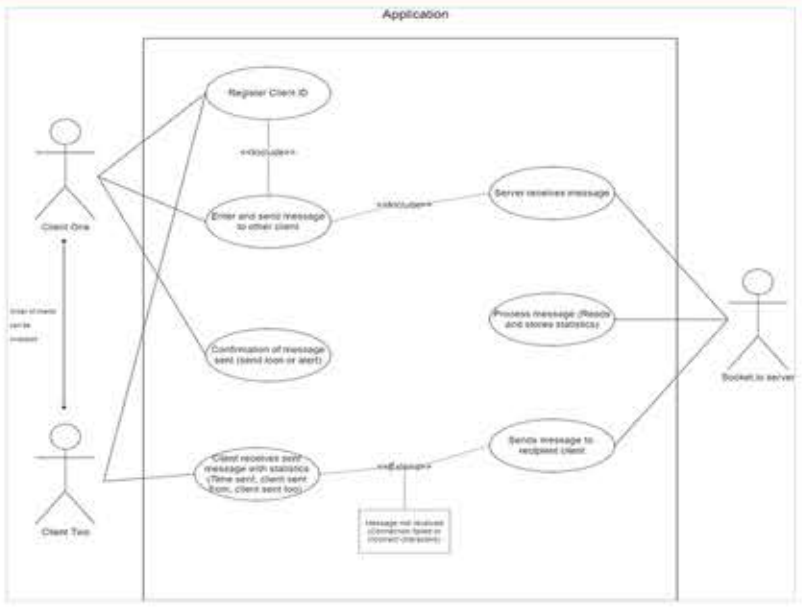


Requirements and Scope

The artifact's minimum viable product: One front-end page for the user interface. Two base registered clients for demonstrating the realtime messaging feature, including the SQL database to hold client data and a basic web server to control data transmission. The database functionality initially is where the majority of time will be taken to establish the required registry system (as mentioned in the project schedule).
These core features fit into the developments scope, with the future direction of the project to be catered towards a package service that can be easily implemented into already established web applications. This could be installed from a small amount of simplified source code and be a useful alternative to communicating with clients and easily expanded to add more clients or features.

Artifact use case diagram

This use case diagram shows the application's main interaction between the user and the web socket. Clients register so the socket has both a send and receive target. A client will send a message to another client which is passed through the server, providing it with statistics such as time sent or received and which client ID it was send from. The recipient client will then receive this message with the mentioned statistics.



Justification and Industry Application

The need for this artifact comes from the modernising web features, as mentioned in the overview. The core implemented features of this artifact can be easily applied to other sites in place of a contact form for direct and quicker communication between site service members and users. Generating a greater user experience [2] because of the speed, added interactivity and personal element from direct communication, rather than being forwarded to an email inbox and waiting longer to resolve issues. Small and personal sites benefit greatly from initial user experience to create site traffic and promote their site in index rankings [3].
'Live chat' already exists in industry in sites like 'Voxi' or 'H & M'. However commonly large sites get around the need for large twenty four seven customer service teams by implementing some form of artificial intelligence (AI), otherwise known as a chatbot [4]. This AI links to a prewritten question and answer form to mimic a somewhat realistic conversation without the need for a staff member. Therefore, this application not only provides additional jobs in the place of software it also allows truly real conversation outside of mimicry.

Bibliography
[1] Cleveland, B., 2015. The Stats Behind Chat: It's Popular and Growing. [online] Icmi.com. Available at: <https://www.icmi.com/resources/2015/the-stats-behind-chat-its-popular-and-growing> [Accessed 14 February 2022].
[2] Ruby. 2020. The Benefits of Live Chat on Website User Experience - Ruby Blog. [online] Available at: <https://www.ruby.com/blog/live-chat-user-experience/> [Accessed 14 February 2022].
[3] Shulzhenko, M., 2018. AI Chatbots Vs. Human Powered Live Chat: Which is Winning the Customer Service Game? | Provide Support. [online] Provide Support Blog. Available at: <https://www.providesupport.com/blog/ai-chatbots-vs-human-powered-live-chat/> [Accessed 14 February 2022].
[4] R.B, S., 2019. Why user experience is important for small businesses?. [online] Medium. Available at: <https://medium.com/uxahead/why-user-experience-is-important-for-small-businesses-6798dca531d9> [Accessed 14 February 2022].