

Functional Specification

Backend

- The Backend needs to be programmed in a language that supports high levels of concurrency, which through our design, we have selected Golang to do.
- The backend implementation should make use of the Communicating Sequential Processes concurrency paradigm, including message channels and goroutines.
- Take measures to reduce the latency as much as we possible without affecting the applications practicality.
- The level of concurrency provided by the platform should support the requirement that the vast number of messages passed are processed in real-time as opposed to single threaded queueing.
- The backend must communicate with the frontend using the Socket.IO library implementation, in order to implement WebSocket communications.
- The backend should be concurrent, conform to an efficient order of time complexity, and be as optimal as feasible without compromising usability.

AngularJs

- Design a clean, modern design using frameworks which support most modern browsers, desktops and mobile phones.
- Uphold an efficient, streamlined user interface.
- The functionality of creating QR codes dynamically on the front end while linking to the backend, should be implemented on Angular.JS.

- Create factories in Angular.Js to utilise other services such as the ability to add authentication and other services within a factory, without the need for repeating code.
- Create a Single Page Application with a contemporary design, efficient frontend scripting.
- The scripting should be easy to maintain, test and handle dependencies well.

Database

- Implement a database in MySQL that supports the other components within the platform.
- Implement security measures such as salting and hashing sensitive data stored within the database.