BSc in Computing in Software Development

Year 4

Applied Project and Minor Dissertation

*Npm Chat Api*

*G00262708 Cormac McHale*

Contents

[Introduction 3](#_Toc464117570)

[Reason for Choosing Project 3](#_Toc464117571)

[Technologies you plan to use 3](#_Toc464117572)

[Architecture 3](#_Toc464117573)

[Work Allocation 3](#_Toc464117574)

Student Number(s): G00262708

Student Name(s): Cormac McHale

Supervisor: Gerrard Harrison

GitHub Link: <https://github.com/cormacmchale/npmChatApi>

# Introduction

My Project will be in three distinct Parts.

1. A basic angular App Gui – to display the functionality developed
2. An npm library that can be downloaded to an angular app to install chat functionality
3. A chat server with a Java Implementation of a chat service

# Reason for Choosing Project

I’m choosing this project for a for the following reasons.

1. To develop a deeper understanding of Networking communications.
2. To learn how to build a functional back end Java Program.
3. To learn how to publish my own API’s as an independent software developer.

# Technologies you plan to use

1. Node.js
2. Database (to be implemented/decided upon later) probably MySql as Java has functionality for this already
3. Angular (to build the simple Gui)
4. Javascript (to write the library)
5. IDE’s – Eclipse (Java), VS Code (Typescript-Angular App)
6. Some VM software/hosting for the chat server

# Architecture

The Angular Web app will download and install the npm package that I will develop. It will contain the methods to send messages through http requests to a server that will have a Java Program running listening for these on port 80.. it will then process the information however needed.

# Work Allocation

**Develop an npm Package that can allow two clients to communicate with each other through my server (VM).**

Step 1.

Develop a simple package that I can publish and import to an angular or ionic project

Step 2.

Deploy server and allow it to receive messages through web sockets/httpRequests (preferably Java)

Step 3.

Develop a package that has methods to send a message from the client to the server through a web socket/http request?? (needs to send all info required for a message (sender/receiver data))

Step 3.

Configure Server to handle Messages correctly (i.e.. pass onto another client, or initially just store message) store client info in database.. allow them to connect with other clients

Step 5.

If there is time add complexity (Authentication, users, etc….)

# Things to note and that can change.

The initial milestone that I’m aiming to hit will be just have two clients writing to a shared message board on my server… basically build the app, and have it deployed on two sperate machines and have them posting messages to a message board. The major obstacle here won’t be the communication or sending info to the database Via http request… it will be getting the info back from the database from the server to the clients as they post messages. I’m thinking that if I write a PHP script to sit on the server then I can display everything that’s on the server through the web app.