**Coursework Report**

Scott Templeton

40317707@napier.ac.uk  
Edinburgh Napier University - Web Technology

1. Induction

In this section of the coursework I extended from coursework one’s requirements and implemented the new brief. These additions provided a network that allowed users to register accounts, sign-in and edit details. Create conversations between a group of users. Front-end I followed my over-arcing ‘branding’ using minimal shapes and colours.

Then after planning the structure of my system on paper and brain storming the best possible framework to structure the back-end for nodejs to enable the application to become fully functional. I allowed my application to have extended capability and allow users to edit user date like their name.

2. Software Design

I started with the front end section of assessment two so I had a clear understanding of the what functionality was needed later if I decided to add additional tools to make the user experience better. Using the brief I created a list of HTML pages that needed to be created with the corresponding features and functionality using websites like bechance and Instagram to find inspiration for layouts helped break the ice.

Once the templates were created I then progressed into the back-end area of the project. Initialising and setting up node I follow industry standard protocols for structuring my file structure to make an universally readable. I went about using MongoDB as followed in the lectures to setup. I chose MongoDB as I personally feel creating RESTful APIs it results in a quicker turn around and JSON formatted data is easier to be parsed in the front-end to populate the pages. Then using node I started to make the web-server actually render pages by using Expresse’s routing tools. to display the static pages.

Now the site actually loads to the user I need to using the a CRUD system to allow user registration and login capabilities. Using post requests to send the data provided by the user in the on register to the server. Creating a user object and saving them to the database if their username was available for use.

Once user creation was completed I then worked on the messaging service of the project. Rather than creation direct user to user chat. Multi-user group chat 2-Unlimited users made the application more useful. Each chat having an ADMIN which had the power to add and remove users from the app. If users want direct chat the same system was used but only have the two users in a chat room.

Critical Review:

Completing the back-end of this website I followed the brief and met the required points as found in the brief successfully. These included User Registration and login system, with account customisation and user group-chat with unlimited amount of users.

I decided to expand on the user to user messaging and create an application that allows group chats that could include 2-Unlimited users expanded the projects depth. This wasn’t more challenging to make than direct messages but was the same system just expanded. I was rather pleased with the final out-come as it could be pushed to be a fully functional secret messaging service.

Parts than could require more features if more time was given to allow for greater user experience would be more user customisable data. Profile pictures, user chat colours. Message requests to allow user privacy. I felt I would have been able to successfully make this with the aid of templating engines like Jade or Handlebars.