

# Group Chat Application

**Authors**

Derek McCarthy                    B00007439

Christopher Slattery              B00092939

Joseph Tierney B00092923

Eoghan De Bhal B00092942

Contents

[Group Chat Application with Integrated Commands and Custom Protocol 1](#_Toc511382569)

[Abstract 3](#_Toc511382570)

[Planning 4](#_Toc511382571)

[Functionality 5](#_Toc511382572)

[Protocol 6](#_Toc511382573)

[Conclusion 7](#_Toc511382574)

# 

# Abstract

This paper describes the technologies and approach used to create and maintain a group chat application that implements integrated commands and a custom protocol. The paper will also show the research and analysis required to make such an application. Another thing the paper will describe is the planning and functionality of the application and how the protocol we designed actually works.

# Planning

When planning for the implementation of this project, we decided it would be a good idea to write as much python code as we could before we started. We thought this would be a good idea as none of us had much experience with the python language before this project. We also had a meeting between the group to decide how we would implement the project to the best of our knowledge and also to decide the timeline of the project and who would be doing what part of the project. The next step we did was install python and all the relevant technologies needed for doing this project. We also thought it would be a good idea to write out all of the commands we possibly could and how the server should respond to them. This would also help us with future work with python projects. Another thing we did was study the different data structures that python has so we would have a better idea of how the server handles information like users. Another thing we studied before we started was the low-level string manipulation that python provides. This helped with helping the server to find out where a command starts and ends.

# Functionality

The program functions by having a server and clients. The server listens for connections from clients and opens a socket for each client. If a client connects to the server, they will be shown messages that were sent before they connected. This is a good feature of a chat system as it gives the newcomer(client) a chance to catch up with what the chat room is currently chatting about. The program also shows all messages that are sent to the server to every client user. In the program the server is also in charge of keeping note of all the different users that are currently connected to the server. It is also responsible for sending out different messages to different clients. The program also uses a hash of the payload which enables the server to be able to verify if all of the data being sent is arriving successfully to its destination. The payload data is hashed and compared to the one it received and verifies if it is the correct data. The payload is the data inside of the tags that are sent to the server. A custom ping is implemented into the server to acknowledge that it received the ping with a message sent back that says pong.

The system also allows other users to know who sent what message by having the name of the user who sent the message in the message also. We also have a feature in the system that allows a user to ask the server for the correct time and it is then given the correct time. We implemented this using EPOCH time. Lastly, we have some functionality that tells the user who has joined the chat session and how many messages are currently in the chat room.

# Protocol

Our protocol works by sending different data methods(functions) from various clients to the server. The server then interprets the functions and sends back data depending on what the data it received is and what the it is set up to send back. The server has many functions that send back data depending on what data they receive. One function it has is the pong function. This function is used to send back the word pong when the server receives the word ping. It also sends back the time (Round Trip Time) for the ping to send, receive and send back. It also has a function called introduceNewUser. This function is used to tell all the users that are currently in the chat room to receive a message on the screen telling them that a new user has joined. Another function we have is the hello function. This function is used to send a hello message back to the user after the user has sent a hello message to the server form a client. Our server also has a parseMessage function. This function stores messages sent from clients into a buffer and removes the command tags that are sent form the client. The next function it has is a parseInput function and this quite like the last one is used to check if a command is present in the text. The last function we have is the manageConnection function. this function is used to take input from a client and print it out on the server. This happens when a new thread is started from an incoming connection. This data that comes in is also added to a buffer.

# Conclusion

A structured layout, good visualisation, good navigation and a good user interface are all very important parts of having a functional and standardised application. We feel when designing the application, we adhered to these principles and we feel we benefited from this approach. We also feel we learned very much from this project. We learned very much as we expanded our knowledge by using the python programming language that we had never previously used. We also expanded our knowledge about internet protocols especially from a development point of view.