Software Design Specification

For

Project 3

5/19/2019

[Version Number: 1.1.0]

Prepared by:

Adam Weesner

Joel Sanchez

**Table of Contents**

[**Abstract**](#_vm7qyean4v2) **3**

[**Introduction**](#_omw9g1l03hv9) **3**

[**Configuration**](#_8st1cjt6h51j) **4**

[**Security Protocols**](#_ni4ey4v1mcvq) **4**

[**Testing and Field Evaluation**](#_70e9t0wu1ykt) **5**

# **Abstract**

In this project we will have the using TCP/IP protocol to set up a server that will be the host of the clients to connect to. The server will have an open port to where new clients can use that port to connect; as a result the server will create a new socket with the client. This is to keep the open port to anyone new wanting to join. This is an encrypted chat system where there is is only one chat room and the user must know the password in order to log in that chat.

# **Introduction**

The use of an encrypted chat room can be very useful in any environment the internet was not built to have security inmind; as a result, we must come up with encryption algorithms and protocols to keep our information safe from others. This program can be used a number of ways; one example, that it can be used over WAN (wide area network) or in a LAN network. The configuration will be needed to work will with NAT router; furthermore, no configuration will be needed over a LAN. The server will be able to run multiple times on a single machine with available port numbers, as long as the password is changed from application to application.

1. **Configuration**

The server will be one instant of it running for our test. It will normally be using port that is greater than 1023 to not conflict with any purpose port numbers, also it can not pass the the maximum port number of roughly 65000. A secret key will be used as a symmetric encryption easier use for clients and server to compute; furthermore, it will be easy to have the server to keep up with the constant messages being set by multiple parties. The user will have to identify them with a user/login credentials to keep track of the users. The IP address we can easily connect to will be the local host connection which is 127.0.0.1. This will used to keep other sources from having any source trouble; for example, a router firewall could block that port.

1. **Security Protocols**

The use of the security protocol will be used is AES encryption. The server will be having a password that will be the key for the AES encryption; as a result to have the others have use of this encryption session will need to know the password. When the server gets a message from the client, it will have to decrypt the message and vise versa; the server will encrypt everything it will send to the clients. The use of having another security in the TCP/IP is the removal of the sockets, this will keep anyone else or program from hijacking the port. It is good to have the list of the socket removed after ending a session or if the user lost the connection.

1. **Testing and Field Evaluation**

The project will have to be tested without the encryption at the beginning of the project. Getting the communication between the socket and application must be set in stone so that when we start the encryption process of the project, there can not be any offset of the application data or an offset of packets since this will have the decryption to fail. The number of trials and port number will also have to keep in check; for example, if the application or server crashes, the port number might be considered to be open in operating system eyes. The application will fail to bind to that socket, and the only solution to fix that is to have a reboot of the system. This can be time consuming if the port does not get closed correctly.