**Documentation MQTT**

The language of choice for our team to use as a base to code the MQTT protocol in Julia was the implementation in embedded C. The biggest initial issue the team faced was figuring out exactly what the protocol was and where to begin.

The way the program is intended to work is based on modules. By using modules we allow the program to have new elements to be easily added and altered rather then risking the need to change the entire program to implement or change a single feature of the protocol.

The most useful thing we used to test the code was the inclusion of the load.jl file. This allowed us to run the Julia program in command line which we found to be much more helpful. It also allowed us to put in print statements both before and in some cases after the methods that were called for certain things. Adding these print statements allowed us to more accurately find where the program fell down. if the program failed to enter a particular method then the print statements allowed us to more find that the method was the reason the program crashed as we could see it ran the previous method and then crashed when the new method was called.

The basis of the connect packet is to establish a connection to the server with a socket assigned to the connection to be able to keep the connection open and to be able to use the connection. If there is already a connection to the server then the new attempt at a connection will simply be rejected. In the event that the client is connected to the server the program must have a way in which to keep the connection open(alive). If the connection is left idle for too long then it will terminate. Normally the keep alive call just send a ping to the server at certain increments.