**Documentação Interfaces e Protocolos**

**Subscribe x Publish**

Um dispositivo pode ser “inscrito= subscribed” ou uma “publicação=publish”. Quando tal dispositivo é um Publish, ele publica informações pré-estabelecidas. Já o dispositivo Subscribe recebe todas essas informações que estão sendo publicadas.

No nosso caso a placa ESP32 Publicava para a aplicação informando o estado do semáforo em tempo real, e a aplicação inscrevia para a placa ESP32 atuando sobre o semáforo o tirando do seu funcionamento normal e o aplicando o estado de alerta.

Caso o dispositivo gere alguma informação que não está cadastrada, esta não será publicada. Por isso que a maior vantagem deste protocolo é a economia de banda de internet, já que não há tráfego de informações sem importância.

**• Broker**

Para que o protocolo MQTT aconteça existe o Broker que é um servidor que gere as informações aos inscritos e vice-versa, funcionando, então, como um intermediário entre Subscribers e Publishers. No nosso caso usamos o broker "broker.hivemq.com".

Além de armazenar, o Broker também filtra as informações. Os dados dentro deste servidor estão armazenados em tópicos e, desta forma, os Subscribers escolhem quais os tópicos querem se inscrever e receber apenas os que os convém. Em outras palavras, o Broker recebe todas as mensagens, filtra e decide quem está interessado e inscrito nela ou não.

**• Hivemq**

‎Utilizamos no nosso Painel MQTT o ‎‎brokerHiveMQ MQTT‎‎ onde você pode usar qualquer cliente ou biblioteca MQTT para publicar com o Broker, os testes e uso são gratuitos, mas existe a recomendação para que não os use para informações confidenciais, pois todos podem se inscrever em todos os tópicos, incluindo hakers. Sinta-se livre para usar com mqtt e o brokerhiveMQ.

Também se recomenda a usar uma lógica de reconexão ao seu cliente, conforme fizemos em nosso código, isso é recomendado pois pode haver a atualização a instância do HiveMQ em qualquer momento, ou seja, o HiveMQ não garante uma conexão com 100% de tempo de atividade. ‎

‎Ao usar ohivemq cloud ‎‎você tem uma plataforma mqtt totalmente gerenciado e gratuito, podendo criar clusters confiáveis, escaláveis e seguros de broker cloud MQTT.

**• Versões MQTT**

Desde abril de 2019, a atual versão do protocolo MQTT é a 5.0. Nela foram feitos alguns ajustes e melhorias. A versão 3.1.1 igualmente está em uso e segue os padrões OASIS, assim como a versão 5.0, e foi ratificada também pela ISO.

**• Segurança do MQTT**

O protocolo MQTT é o que tem o maior nível de segurança se comparado a outros protocolos de comunicação entre máquinas, ele suporta vários tipos de autenticações e mecanismos de segurança de dados sendo tudo isso é configurado no Broker e normalmente é feito pelo próprio cliente.

Também é possível fazer a criptografia ponta a ponta com a segurança SSLpara proteger os conteúdos das mensagens no protocolo MQTT,

Também existem algumas configurações de qualidade do serviço, conhecido como Qualityof Service (QoS). Este aumenta a confiabilidade, já que garante a entrega da mensagem.