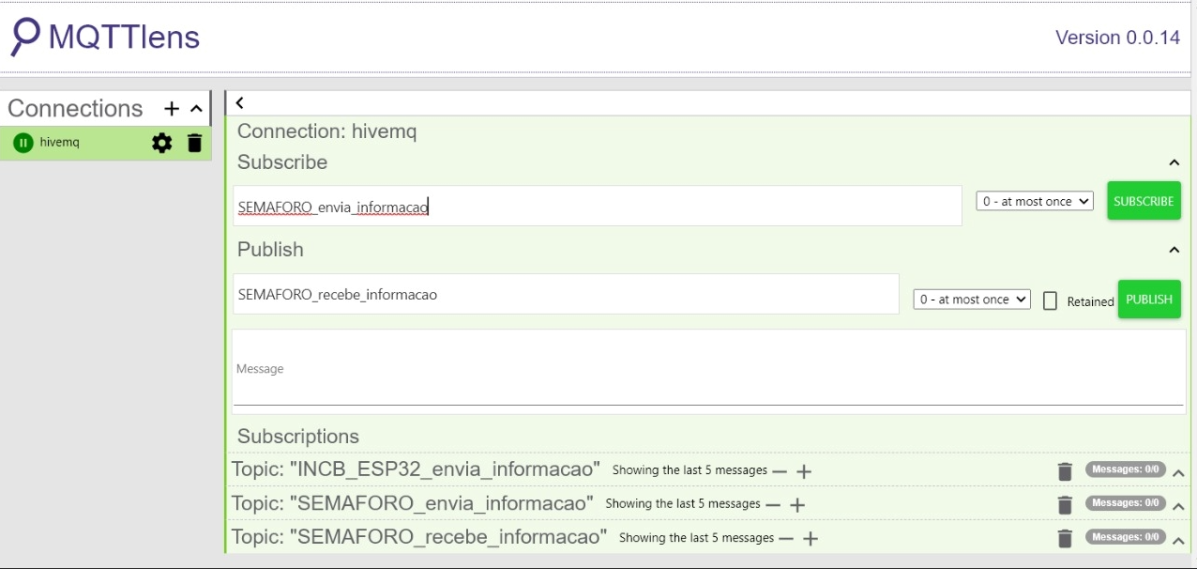
**Descrição do Funcionamento**

1. **Funcionamento**

O projeto do semáforo é conectado remotamente via protocolo MQTT, funcionando com suas luzes **(vermelho, verde e amarelo).** Utilizando o **MQTTIens** conseguimos verificar em tempo real o estado em que o semáforo se encontra **(amarelo, verde e vermelho).**

****

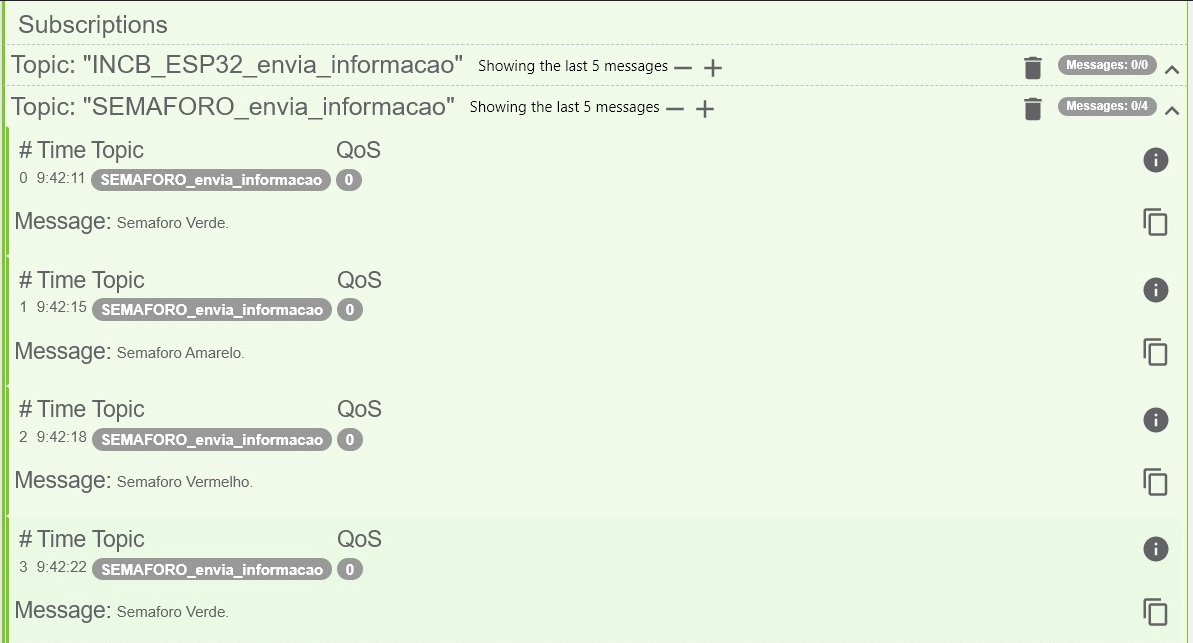
**Figura 9. MQTTIens**

fonte: [autor]

Com o funcionamento padrão do semáforo, podemos observar que no nosso Publish temos um tópico chamado **SEMAFORO\_recebe\_informacao,** que já está identificada na nossa IDE do Arduino e temos também o subscripe **SEMAFORO\_envia\_informacao**.

Se formos em “Message” dentro do MQTTIens e digitar “alerta” (em caso de algum acidente) o operador do semáforo remotamente após publicar, vai fazer com que o semáforo fique em sinal de alerta (luz amarela) sendo enviado essa ordem via protocolo MQTT.

Se alterarmos a “Message” pra “normal”, após o operador publicar, o semáforo voltara para seu estagio normal, acendendo suas luzes padrão. (amarelo, vermelho e verde)

****

**Figura 9. Informações MQTTIens**

fonte – [autor]

Na situação acabamos por cumprir o requisito da placa ESP32 enviar para o broken MQTT e também com o protocolo MQTT do serviço para a placa ESP32.

Por ultimo, temos o sensor infravermelho que ao ser acionado o led amarelo (Atenção) começa a piscar automaticamente, e isso é enviado via MQTT para o operador de forma remota avisando que esta em estado de alerta, no caso de acidente, engarrafamento ou algo nesse sentido.