

```
1 // Variáveis
2 var num_vagas_disponiveis = "0";
3 var marker;
4
5 // Instancia um cliente Paho MQTT
6 HOST = "broker.hivemq.com"
7 PORT = 8000
8 CLIENT_ID = "frontend_test"
9 client = new Paho.MQTT.Client(HOST, PORT, CLIENT_ID)
10
11 // Define as configurações de conexão
12 var connectionOptions = {
13     keepAliveInterval: 20,
14     timeout: 3,
15     onSuccess: onConnect
16 }
17 client.connect(connectionOptions);
18
19 // Método chamado quando o cliente se conectar ao broker
20 function onConnect() {
21     // Conexão realizada, então se inscreve no tópico desejado
22     console.log("Iniciando...");
23     marker.setLabel("?");
24
25     var topic = "vagas/cpd/"
26     client.subscribe(topic);
27     console.log("Inscrito em: " + topic)
28 }
29
30 // Método chamado quando chegam novas informações sobre as vagas monitoradas
31 client.onMessageArrived = function (message) {
32     console.log("onMessageArrived:" + message.payloadString);
33     marker.setLabel(message.payloadString);
34 }
35
36 // Método chamado quando o cliente perde conexão com o broker
37 client.onConnectionLost = function (responseObject) {
38     marker.setLabel("OFF");
39     if (responseObject.errorCode !== 0) {
40         console.log("onConnectionLost:" + responseObject.errorMessage);
41     }
42 }
43
44 // Inicializa mapa
45 function initMap() {
46     var coords = { lat: -29.720966, lng: -53.714431};
47     var map = new google.maps.Map(document.getElementById('map'), {
48         zoom: 18,
49         center: coords,
50         scrollwheel: false
51     });
52     marker = new google.maps.Marker({
53         title: "Vagas disponíveis",
54         position: coords,
55         animation: google.maps.Animation.DROP,
56         label: num_vagas_disponiveis,
57         map: map
58     });
59 }
60
```