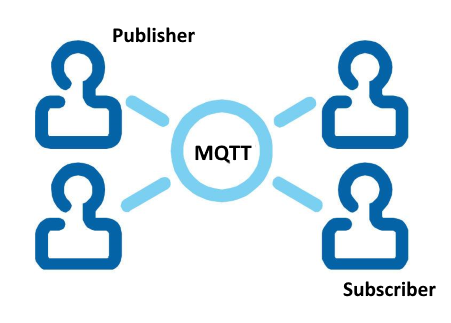
MQTT client application for Android

The Android Broker test is a Delphi-based application designed using the IDE RAD Studio Tokyo 10.2. It has been compiled for an Android environment. Basically, it is a client application which communicates with an IoT board through a MQTT broker server (see http://mqtt.org). With this application, IoT (internet of things) projects can be easily designed. The IoT board can be anyone based on the chip ESP8266, including the Wemos D1 R2 and the nodeMCU 1.0 boards.

The IoT module must have the sketch test\_eclipse.ino previously stored in it. This sketch is based on the libraries esp8266wifi.h and PubSubClient.h (<https://pubsubclient.knolleary.net/>). The last one provides a client for doing simple publish/subscribe messaging with a server that supports MQTT protocol. The sketch test\_eclipse.ino uses the broker test.mosquitto.org which listens on the port 1883. It subscribes to the topic “MessFromClient71” and publishes in the topics “Estado/Led”, “Estado/Botao” and "TopicIoTboard71". Every request received from the Android Broker test application is printed on the serial monitor. If the mobile application publishes in the topic “MessFromClient71” the commands /LED=ON or /LED=OFF, the IoT board put the builtin led in the appropriate state and publishes this status in the topic “Estado/Led”. If the mobile application publishes in the topic “MessFromClient71” the command /STATEBUTTON, the IoT board publishes the state of the builtin button (pressed or unpressed) in the topic “Estado/Botao”. Finally, every message introduced via the serial monitor is published by the IoT board in the topic “TopicIoTboard71”. Thus, the board not only can work showing or setting the states of the devices connected on it, but it also works as a chat room with the mobile application, exchanging messages received from the application or introduced on the serial monitor.



The Broker Test application for Android has been designed using the TMS MQTT components (<https://www.tmssoftware.com/site/tmsmqtt.asp>) for Rad Studio. It uses the TMSMQTTClient component which communicates the application with the MQTT free broker servers test.mosquitto.org, broker.hivemq.com, iot.eclipse.com, mqtt.fluux.io, test.mosca.io, broker.mqttdashboard.com (see <http://moxd.io/2015/10/17/public-mqtt-brokers/>). The server to be used must be selected from a listbox. The mobile application subscribes to the topics Estado/Led, Estado/Botao and TopicIoTboard71 and publishes in the topic MessFromClient71. The received payloads are written on a memo log window and the commands or sentences to be published can be introduced through an edit box and sent to the broker server. Finally, two buttons allows the application to connect and disconnect to the broker server. The connection status, as well as the received payloads sent by the IoT board are written on the memo log window.

The following picture shows the screen of the mobile application:

