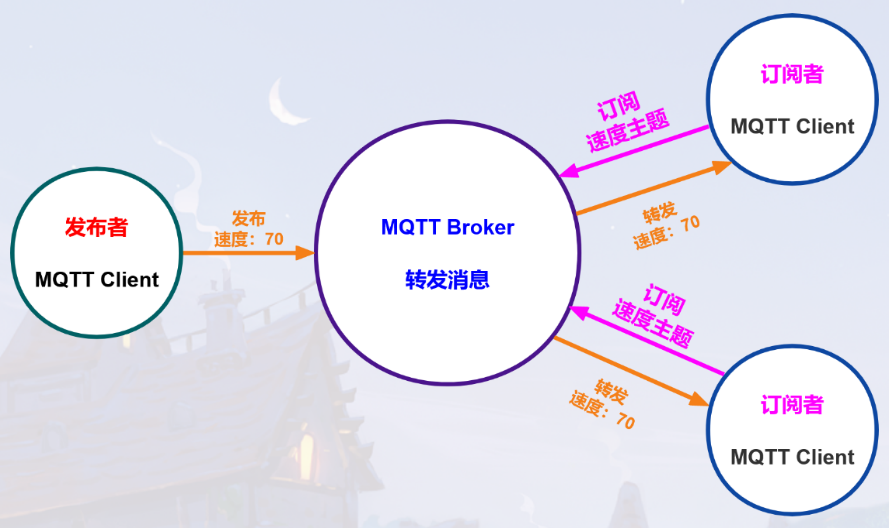
**一、SpringBoot+MQTT：**

MQTT（消息队列遥测传输）是一个基于**客户端-服务器**的消息**发布/订阅**传输协议。MQTT协议是**轻量、简单、开放**和**易于实现**的，这些特点使它适用范围非常广泛。在很多情况下，包括受限的环境中，如：机器与机器（M2M）通信和物联网（IoT）。

MQTT的发布/订阅模式如下：



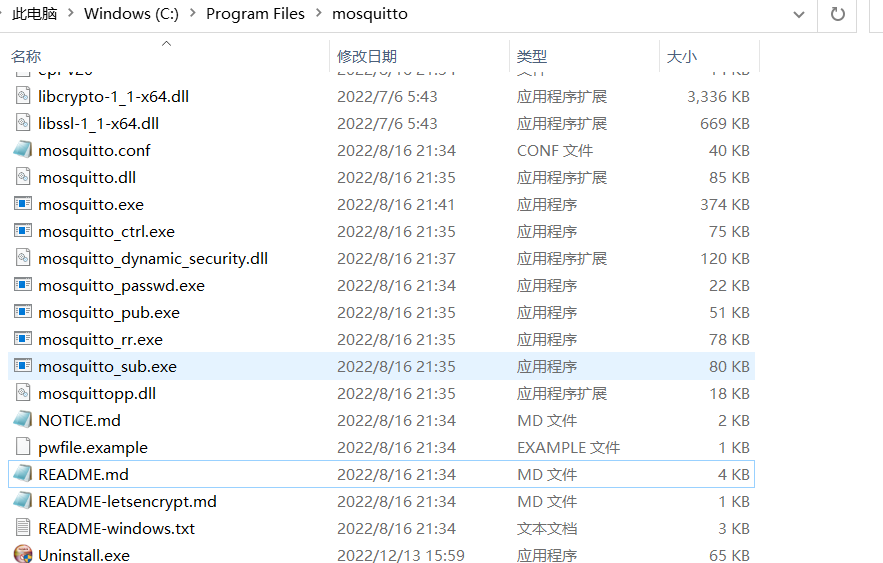
要具体实现MQTT服务，步骤如下：

**1. 服务端**

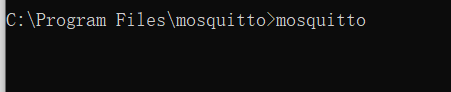
服务端使用mosquitto。

下载网址：<https://mosquitto.org/download/>

1.1 下载安装后进入mosquitto的文件夹，在这里打开cmd。



1.2 在cmd中输入命令mosquitto启动服务。



**2. 客户端 选择MATTX或者MQTT.fx**

2.1 MQTTX：下载页面：[https://mqttx.app/zh#download](https://mqttx.app/zh" \l "download" \t "_blank)

2.2 MQTT.fx：下载页面：<https://mqttx.app/zh#download>

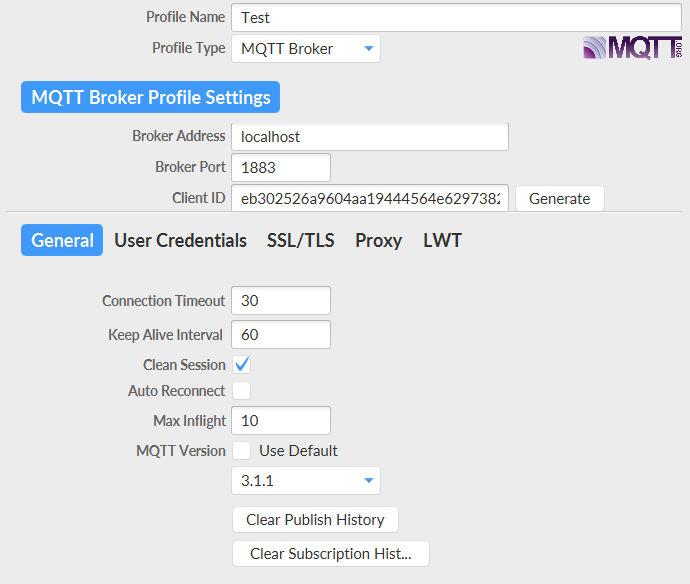
注：本文使用MQTT.fx

**3. MQTT过程演示。**

3.1 打开MQTT.fx，然后点击设置

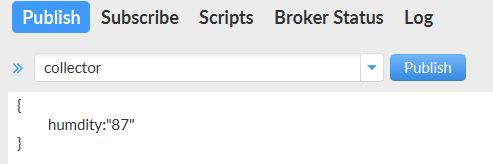


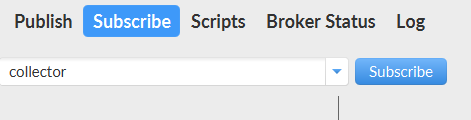
3.2 配置如下。其中Profile Name可以随便写，Clinet ID点击Generate生成。



3.3 完成上面配置后，点击apply和ok，然后点击connnet。

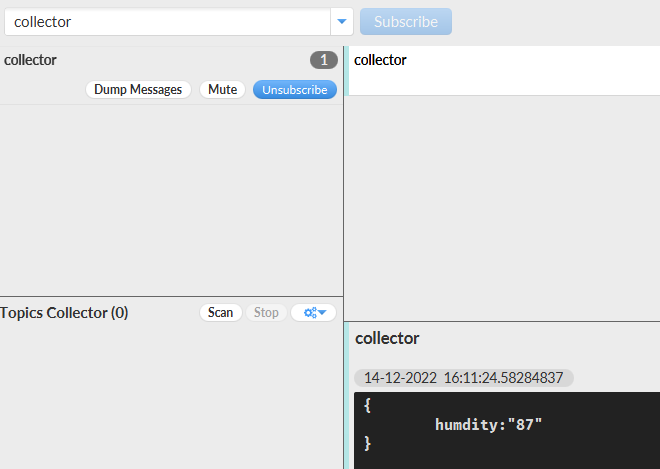
3.4 发布和订阅主题，主题名随便写，但是发布和订阅的主题需要一致。然后点击Publish和Subscribe。参考如下：





3.5 发布主题的内容已经获取主题内容。

点击上述的publish后，然后subscribe这边内容如下：



**4. SpringBoot的编程（开发环境：maven，IDEA，JDK17）**

4.1 pom.xml 中MQTT的jar包

<dependency>

<groupId>org.springframework.integration</groupId>

<artifactId>spring-integration-mqtt</artifactId>

</dependency>

4.2 MqttConfig.java

4.2.1 MqttPahoClientFactory获取一个工厂对象，其中要设置MqttConnectOptions对象。MqttConnectOptions对象里面设置代理端的一些配置，也就是mosquitto的配置消息。

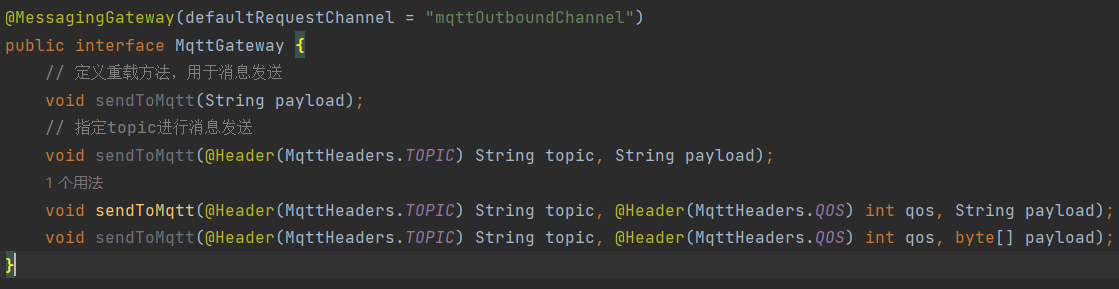
4.2.2入站和出站通道分别由一个MessageChannel对象维护。

4.2.3 入站消息配置。先是使用Paho客户端消息驱动通道适配器，主要用来订阅主题，然后配置Paho消息转换器，最后设置适配器的转换器，QoS和入站通道。

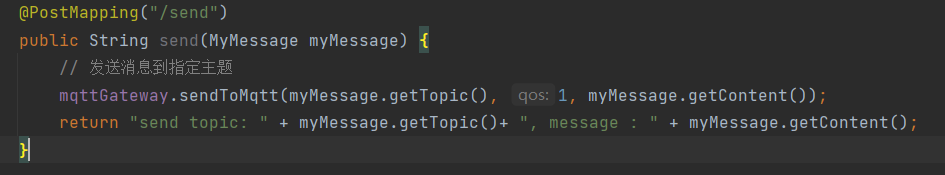
4.2.4 入站消息处理。使用消息Handler来处理消息，如打印消息、存储消息到数据库。

4.2.5 出站消息配置与入站消息配置类似。具体参考附件代码。

4.3 MQTT网关。直接写接口就行，SpringBoot的AOP会自动帮你处理。



4.4 Controller。实现与服务器交互。

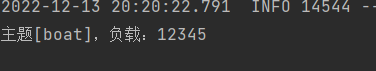


4.5 启动SpringBoot。

**5. 前后端交互展示。**

使用http传输数据。



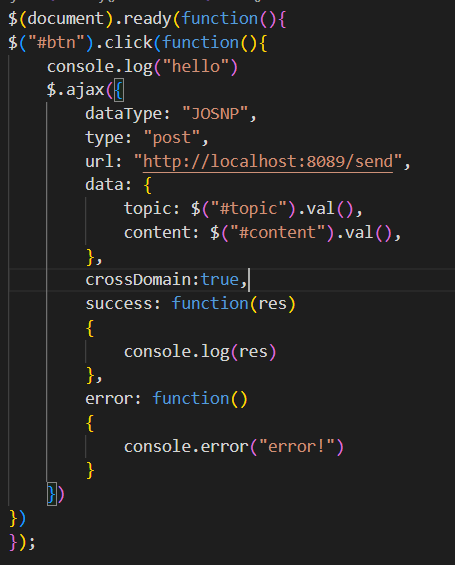


前端代码如下

HTML：



JS：



**二、花卉识别深度学习模型：（需要华为云账号）**

本模型为基于卷积神经网络实现的花卉识别实验，与传统图像分类方法不同，卷积神经网络无需人工提取特征，可以根据输入图像，自动学习包含丰富语义信息的特征，得到更为全面的花卉图像特征描述，可以很好地表达图像的不同类别信息。

训练步骤如下：

**1.OBS**

登录华为云，点击控制台，搜索OBS，进入OBS，点击创建桶。配置参考如下：



然后点击创建。

**2. 下载数据集并上传OBS。**（可选，如果在代码中下载数据集时超时，需要手动下载）

在浏览器中输入

<Https://ascend-professional-construction-dataset.obs.myhuaweicloud.com/deep-learning/flower_photos.zip>

然后下载数据集

下载完成后，点击创建好的桶，点击上传对象，即上传此数据集。



**3. ModelArt中训练模型**

点击控制台，搜索modelArt，选择开发环境中的notebook，然后创建一个notebook



选择的镜像参考如下。



使用默认存储即可。



其他保持不变。

点击开发环境的notebook，点击启动。

手动加一个窗口然后执行以下命令。

!pip install easydict

代码中修改如下



然后就是按照顺序进行，如果你下载数据集超时，你就注释掉!wget部分，手动上传数据集，步骤如下：



然后继续按照顺序进行即可。

最后获得一个模型。然后这个模型可以部署到服务器上。

**三、结合**

思路：在Atlas硬件平台上使用camera获取花卉图片，然后上传到部署了花卉深度学习模型的服务器上，服务器返回花卉信息到SpringBoot的服务上。使用Arduino配置传感器，然后使用MQTT协议上传数据到SpringBoot端。最后，SpringBoot整合所有数据，与前端交互，就可以实现一个智能花卉管理系统。

Arduino部分可以参考附件Arduino进行配置。