Variante du lab

- Introduction
- Materiel
- Programmation
- NodeRED
- Android
- Boitier
- Améliorations possibles

Introduction

Au fablab nous avons également connecté le portail de l'entrée ! En effet à notre installation nous avons demandé à notre hébergeur le tirage d'une commande à distance. Cela nous permet de piloter le portail avec un simple bouton (la ligne est dîte ***à contact sec*** c'est à dire qu'il suffit de coller les deux fils qui nous sont fournis pour ouvrir le portail, comme si on appuyait sur le bouton d'une télécommande). Evidemment on ne s'est pas limité à ça :) on a donc rajouté un relais et un esp8266 pour pouvoir piloter logiciellement cette ouverture et étendre considérablement les possibilités ...

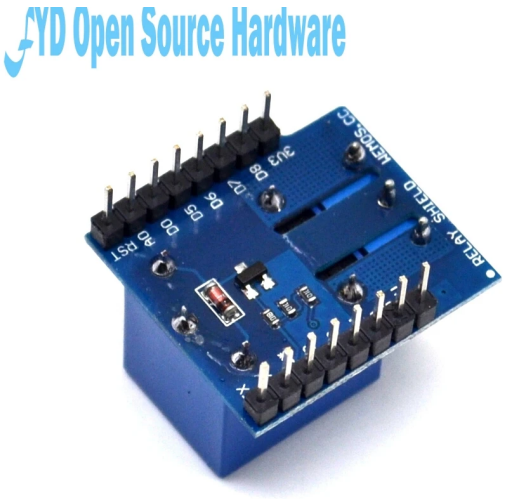


Materiel

On utilise l'écosystème Wemos mini, en empilant la carte de base (qui utilise le fameux ESP8266) avec le shield relais qui correspond :









Une fois empiler, on conecte au PC et on utilise l'IDE Arduino pour le programmer. Le relais est piloté via le pin D1. Pls d'info sur le shield :

https://wiki.wemos.cc/products:d1_mini_shields:relay_shield

Programmation

Je ne vais pas parler ici de l'installation et de l'utilisation de l'esp8266 dans l'univers Arduino, y'a des tonnes d'infos sur le net sur le sujet ... Je pars donc du principe que vous êtes déjà familier avec les basiques de programmation.

Concrètement qu'est-ce qu'on va faire ?

Le principe est très simple : on va connecter la carte au wifi, grâce à cette connexion on va se connecter à un serveur MQTT, et si on reçoit le bon message au bon endroit, alors on déclenche une ouverture du portail. A partir de la, on peut greffer toute sorte de chose pour piloter le relais, du moment que c'est capable de causer en MQTT. Ici nous verrons deux cas : NodeRED et Android.

Schéma fonctionnel

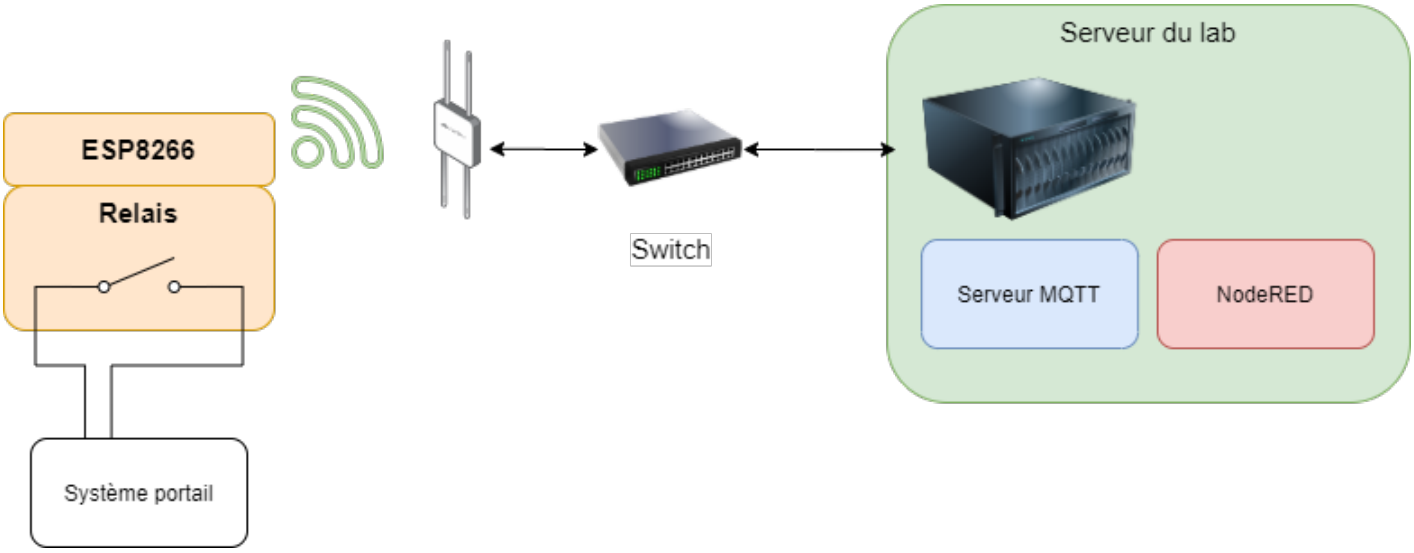
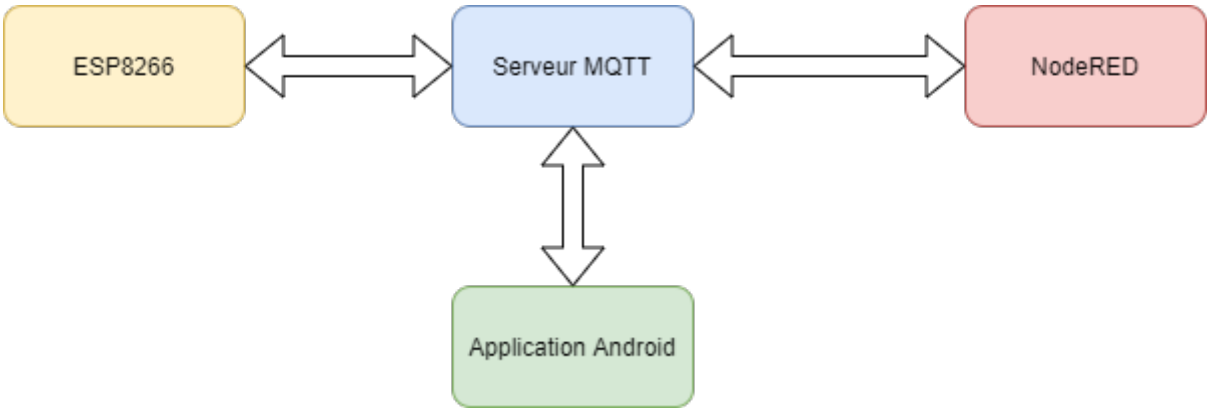


Schéma applicatif



Code

Alors voici le code :

https://gitlab.com/support210/labsud_portailconnecte

Comme vous le voyez il y a du monde mais il peut-être grandement réduit (et simplifier si on le voit ainsi) en enlevant des fonctionnalités. Par exemple, j'ai inclus deux mécanisme de mise à jour par distance : un qui passe directement par l'IDE Arduino (si vous êtes sur le même réseau, la carte apparaît dans *Outils>Port* et un autre qui passe par le navigateur (on arrive sur une page et on peut charger le fichier compilé). Du coup j'ai aussi ajouter un code raccourci pour que ça soit plus compréhensible.

Dans les grande lignes on a la séquence suivante (pour la version short) :

1. on se connecte au wifi
2. on se connecte au serveur MQTT
3. si un message est reçu sur le bon topic, on déclenche une impulsion sur le relais.

C'est vraiment un mix des exemples de base du Wifi et du MQTT.

NodeRED

TODO

Android

La où ça devient rigolo, c'est qu'on a même pas à coder une application (sur Android en tout cas).

TODO

Boitier

TODO

Améliorations possibles

Ce qu'il y a de pas mal dans cette histoire c'est qu'on peut assez facilement sécuriser l'ensemble. En effet comme on s'appuie sur le protocole MQTT, il suffit de basculer la connexion entre les parties en SSL ou en TLS. Ainsi les transactions deviennent illisibles pour une personne mal intentionnée.