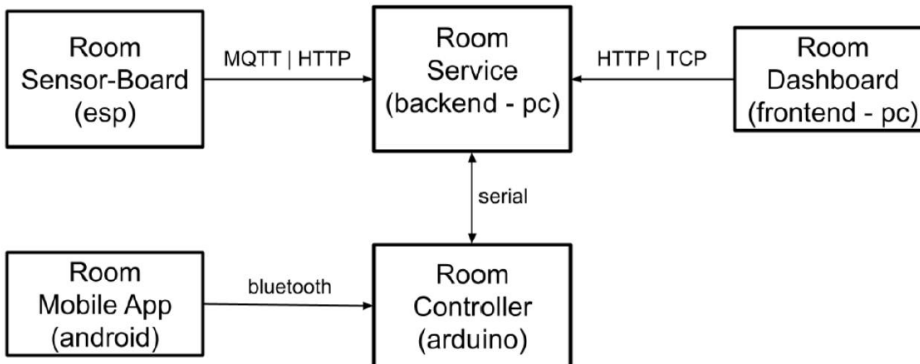


SMART-ROOM-IOT

Daniel Capannini

Mail: daniel.capannini@studio.unibo.it



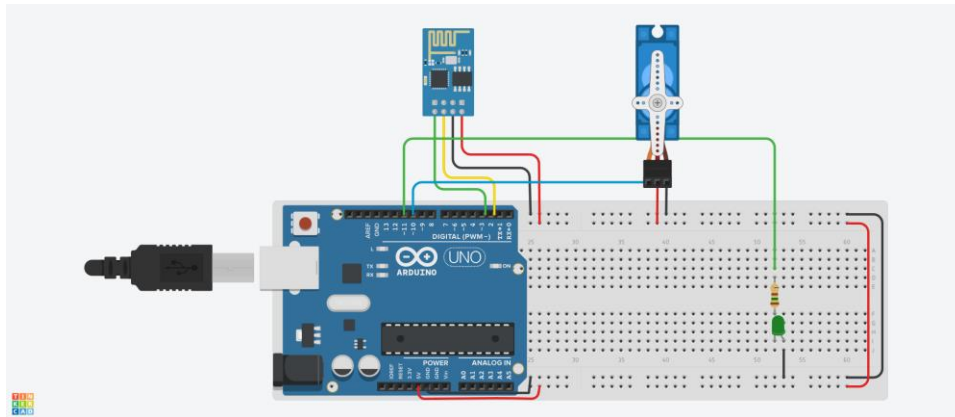
RoomService

Questa è la parte che contiene tutta la logica della dell'applicazione, è formata da un unico file python. Vi è tramite la libreria `paho.mqtt.client` in collegamento mqtt con il dispositivo esp, dal quale vengono letti messaggi che invia che contengono la luminosità dell'ambiente e se è presente del movimento e questi parametri vengono salvati in variabili locali. Vi è poi un ciclo `while true` nel quale da seriale vengono letti i dati di arduino che se sono diversi dalle aspettative modificano i dati locali e quelli del server online che vengono inviati tramite una richiesta post, in caso contrario si leggono i dati i dati presenti nel server tramite una get e poi gestite le varie casistica a seconda della situazione, in fine viene scritto in seriale ad arduino i nuovi dati per la luce e la tapparella.

RoomDashboard

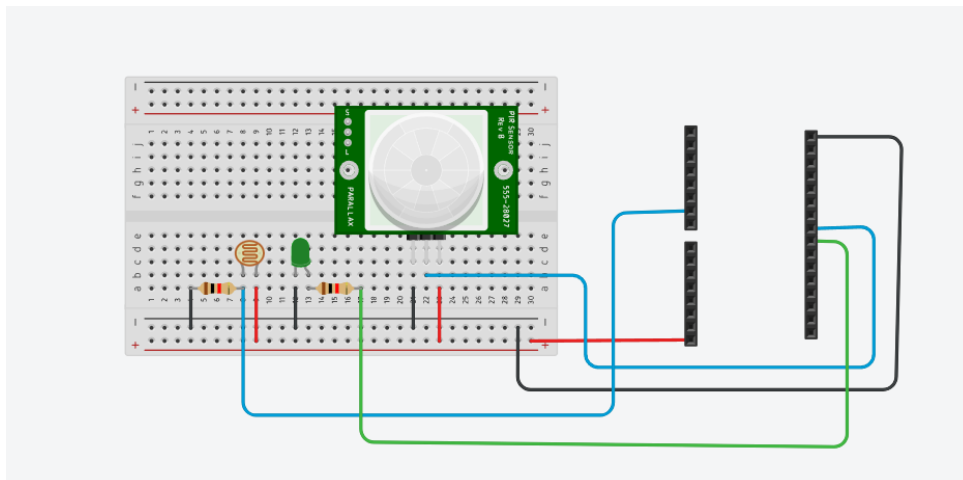
Questa parte è formata da una pagina web dalla quale è possibile accendere la luce e regolare l'apertura delle tapparelle. Sono presenti due file json "lights.json" e "window.json" nei quali viene salvato lo storico delle luci e della tapparella, e questi dati poi servono per poter creare i grafici presenti nella pagina.

RoomController



Questa parte sviluppata su arduino accende il led e apre il servo motore a seconda dei dati che gli vengono inviati tramite seriale e quelli tramite bluetooth, questo viene gestito da una scheduler che contiene 2 task, il BluetoothTask e il SerialTask. BluetoothTask si occupa di leggere i dati che arrivano via Bluetooth e di inviare tramite seriale lo stato attuale grazie alla classe MsgService, mentre il SerialTask fa la stessa cosa ma controlla i dati che vengono inviati da seriale.

RoomSensor-Board



Questa parte sviluppata su un esp32 si occupa di leggere i dati del pir relativi al movimento, di accendere il proprio led in caso vi sia movimento e leggere i dati del photoresistore, i dati relativi al pir e al photoresistore vengono inviati al server "broker.mqtt-dash-board.com" dal quale poi saranno letti da RoomService. Vi è uno scheduler che controlla lo svolgimento di un task, mainTask, il quale svolge tutti i compiti precedentemente indicati.