

# Relazione Progetto

## “Assignment #03 - Smart Room”

Gianluca Consoli  
0001001882  
gianluca.consoli@studio.unibo.it

Alessandro Stefani  
0000971571  
alessandro.stefani10@studio.unibo.it

## Funzionamento del progetto

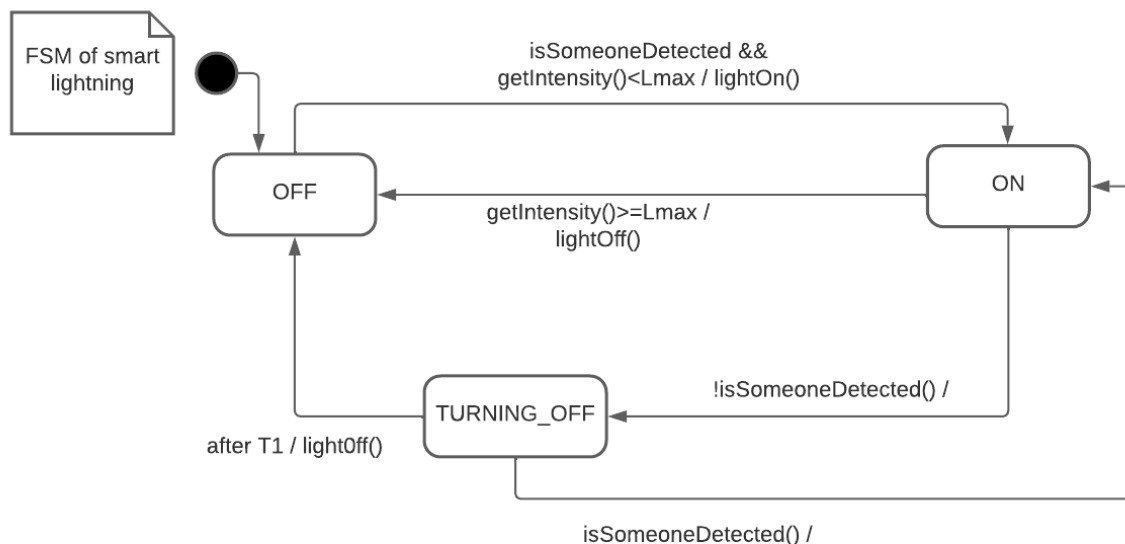
Il progetto consiste nella realizzazione di una smart room, che monitora e controlla lo stato della stanza; per essere precisi in questo sistema siamo in grado di:

- vedere per quanto è stata accesa la luce nella stanza;
- controllare sia tramite Dashboard (sito web) che tramite applicazione android il sistema della luce e le taparelle;
- determinare se qualcuno si sta muovendo dentro la stanza con un sensore (PIR).

Per rendere più semplice la gestione del dominio di questo sistema, esso è stato suddiviso in 5 sottosistemi, ognuno di loro indipendente uno dall'altro, l'unica cosa che gli permette di comunicare tra loro sono i protocolli di comunicazione che hanno in comune certi sottosistemi.

### Room Sensor-Board (esp) - Gianluca Consoli

Sottosistema che monitora tramite dei sensori, se qualcuno è presente nella stanza; ciò viene fatto con un sensore che rileva il movimento di un corpo nello spazio circostante. Se qualcuno si è mosso e l'illuminazione nella stanza è abbastanza bassa la luce verde si accende e rimane accesa finché c'è qualcuno che si muove, se invece non rileva nessun movimento dopo un certo **T1** (max) la luce si spegne; mentre se l'illuminazione nella stanza raggiunge una certa soglia si spegne subito. Per la creazione di questo Smart Light System è stata utilizzata una FSM:



Il Room Sensor-Board dovrà comunicare che la luce è accesa al Room Service oppure se qualcuno è entrato o uscito dalla stanza, tutto tramite protocollo MQTT. L'MQTT è un protocollo di comunicazione che è composto da tre attori:

- **Publisher (ESP)**: ha il compito di connettersi al message broker e pubblicare i contenuti;
- **Subscriber (Room-Service - PC)**: anche lui si connette al message broker e si sottoscrive al contenuto a cui è interessato;

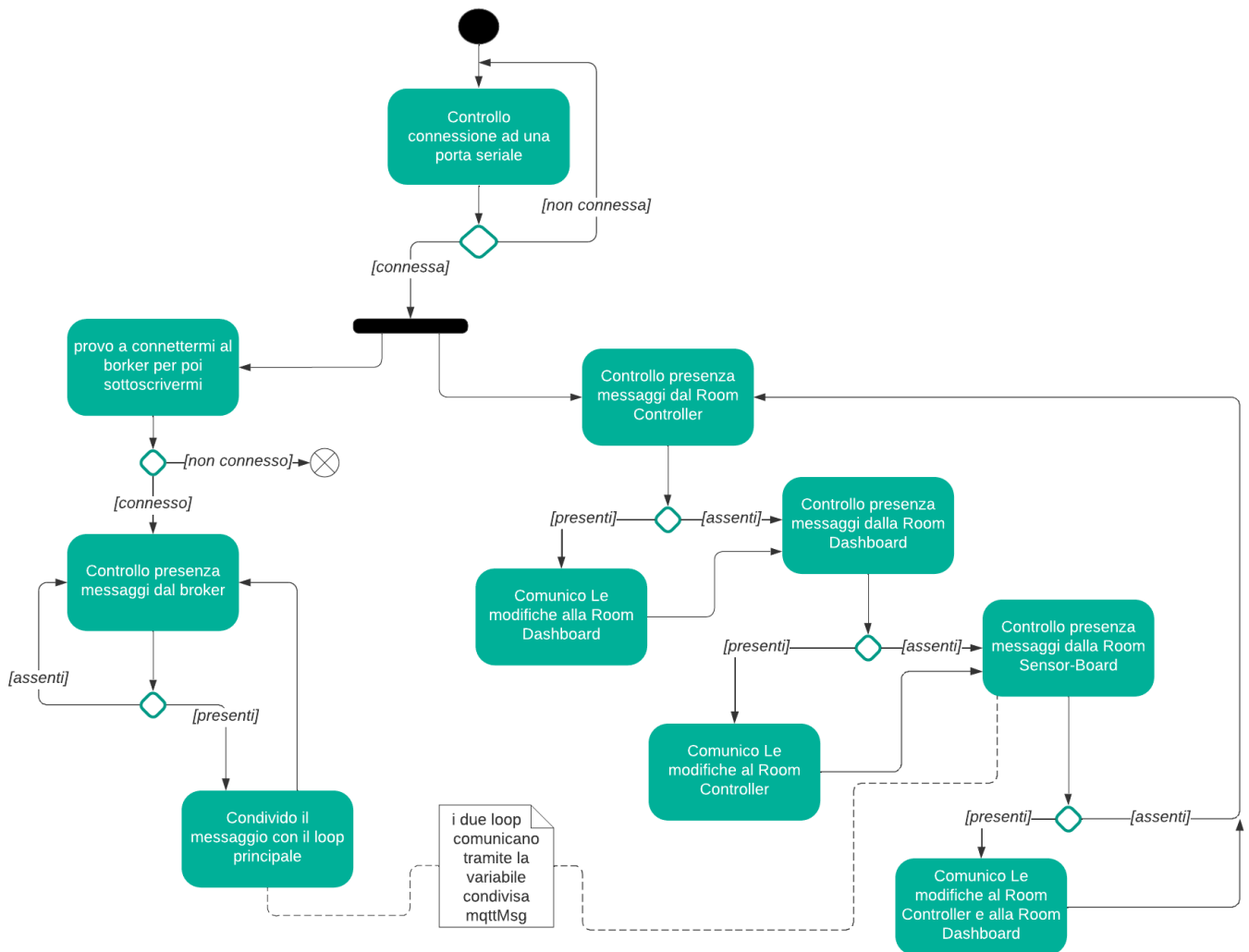
- **Message Broker (HiveMQ - server/site):** fa da intermediario tra publisher e subscriber, nel senso che non appena il publisher pubblica un nuovo contenuto, andrà a notificare a tutti quelli sottoscritti a quel publisher che è stato pubblicato qualcosa di nuovo.

Riassumendo come è stato integrato MQTT nella parte ESP è che dopo che l'ESP stesso si è collegato come Publisher con un certo topic al message broker, nel momento in cui avviene in cambiamento di stato nella stanza (es. rivelazione di movimento), il Publisher provvederà a pubblicare il messaggio e a spedirlo al message broker.

## **Room Service (backend - pc) - Alessandro Stefani**

Il Room Service (serialComm.py), implementato con Python, riceve informazioni dal Room Sensor-Board tramite MQTT e comunica con la Room Dashboard e il Room Controller rispettivamente tramite HTTP e porta seriale.

Dopo essersi connesso alla porta seriale, il RS lancia due loop: il primo su un thread separato dove rimane in ascolto di eventuali messaggi in arrivo dal Room Sensor-Board; il secondo controlla e agisce di conseguenza se riceve messaggi dal Room Controller, dalla Dashboard o dal Room Sensor-Board (comunicati dal primo loop se presenti).

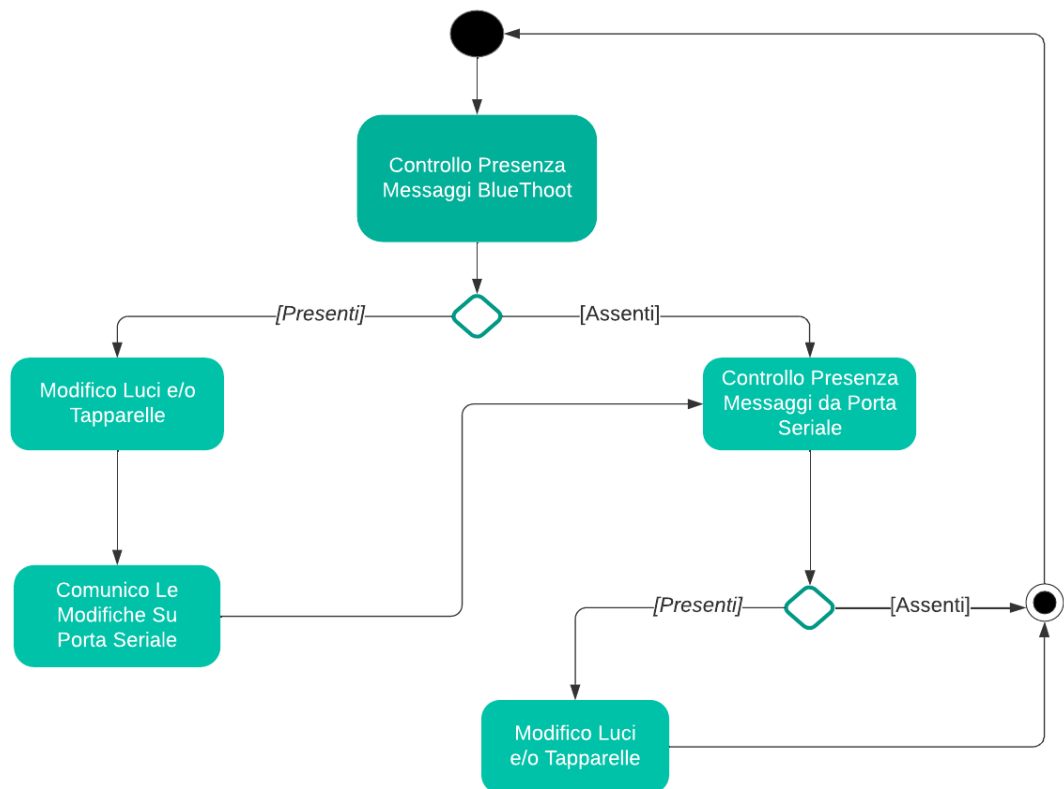


Nota: il programma quando lanciato dal mio pc non riesce a connettersi al broker tramite il Wi-Fi in sede a Cesena, altrimenti funziona correttamente.

Per avviare il room service lanciare serialComm.py

## Room Controller (Arduino) - Alessandro Stefani

Il Room Controller consiste in un semplice loop dove controlla se riceve comandi dalla Room App tramite bluetooth (questi comandi sono inoltrati al Room Service così da mantenere aggiornato il sistema) o dal Room Service tramite porta seriale e agisce di conseguenza.



## Room App (Android - smartphone) - Gianluca Consoli

Questo sottosistema ha il compito di mandare messaggi attraverso Bluetooth, per dare comandi a due oggetti della stanza: luce e taparelle; più specificatamente è possibile spegnere / accendere la luce oppure alzare / abbassare le taparelle.

### Come avviene la comunicazione tramite bluetooth

La comunicazione (bluetooth) è composta da tre fasi:

- **Ricerca dispositivi:** nel momento in cui uno apre l'applicazione Android, se il cellulare ha il bluetooth attivo avverrà una ricerca (di un certo range di metri) automatica su tutti i dispositivi che ci sono nella stanza; cliccando sul menu a tendina è possibile vedere la lista dei dispositivi a cui ci si vuole collegare;
- **Pairing:** avviene quando si cerca di accoppiare 2 dispositivi bluetooth, appena uno clicca sul dispositivo su cui si vuole collegare avverrà il pairing;
- **Scambio di dati:** dopo che è avvenuto il pairing con successo è possibile mandare messaggi tramite un canale di comunicazione (RFCComm Socket) tra i due dispositivi.

## **Room Dashboard (Frontend/web app on the PC) - Alessandro Stefani**

La Room Dashboard è implementata sfruttando le conoscenze acquisite nel corso di Tecnologie Web. Nella pagina principale (dashboard-page.html) è possibile visualizzare tutte le informazioni relative alla stanza, specialmente l'uso delle luci ,visualizzabile in una tabella, ricevute tramite HTTP da Room Service, e inviare comandi alle luci (tramite un pulsante) e alle tapparelle (tramite uno slider) sempre tramite HTTP attraverso Room Service.

Nota: per come è stato specificato l'url per la connessione http all'interno di room-service/msgHttp.py la cartella assignment-03 deve essere all'interno della cartella htdocs di xampp e il modulo Apache deve essere avviato (tramite XAMPP Control Panel). A questo punto la Room Dashboard sarà accessibile al link "http://localhost/assignment-03/room-dashboard/dashboard-page.html"