

## Assignment 03 - Smart Room

Marco Raggini

marco.raggini2@studio.unibo.it

Franceesco Carlucci

francesco.carlucci6@studio.unibo.it

Dilaver Shtini

dilaver.shtini@studio.unibo.it

Daniele Pancottini

daniele.pancottini@studio.unibo.it

13 settembre 2023

# Capitolo 1

## Architettura

La soluzione proposta si basa sulla comunicazione tra cinque ambienti principali: Arduino, Java, modulo ESP, una mobile App e una dashboard. Il sistema si divide in due stati principali: "automatico" e "manuale". In questo progetto consideriamo come stato di partenza la modalità automatica, che viene interrotta quando avvengono operazioni dalla dashboard o dall'app. Si ritorna nella modalità automatica quando non sono rilevate operazioni nell'app o nella dashboard per un determinato periodo di tempo.

### 1.1 Arduino

Arduino svolge il ruolo di Room Controller ed è l'unico componente che comunica direttamente con l'app, via bluetooth. Il collegamento è monodirezionale, ovvero, arduino riceve solo informazioni dall'app, senza rispondere. Quando viene ricevuto un comando dall'app, si entra in modalità manuale, qualsiasi sia lo stato precedente. Questa informazione viene mandata successivamente al Room Service, via seriale, gestito da java(backend), che informa la dashboard(frontend), attraverso una comunicazione HTTP. Se arduino non riceve più operazioni dall'app per un determinato periodo di tempo il sistema ritorna in stato automatico. Il funzionamento è simile per la dashboard, anche essa infatti può inviare informazioni riguardo lo stato della luce e della tenda ad arduino comunicando con java. Quando si entra in modalità automatica le informazioni vengono rilevate e inviate dal modulo ESP, passandole tramite una comunicazione MQTT a java, che successivamente informa arduino via seriale.

### 1.1.1 Light

Il task che si occupa della luce ha 3 stati principali, il Light off, ovvero il caso base, nella quale viene spenta la luce della stanza, il Light on, dove viene accesa la luce della stanza, ma solo se la modalità è in automatico, viene rilevato un movimento e la stanza è buia. Si ritorna al caso base se siamo sempre in modalità automatica e non viene rilevato nessun movimento. Se non siamo in automatico entriamo nel terzo stato Manual, stato in cui la luce viene gestita direttamente dall'utente, per poi tornare al caso base se non viene eseguita nessuna operazione dall'utente per un certo periodo di tempo.

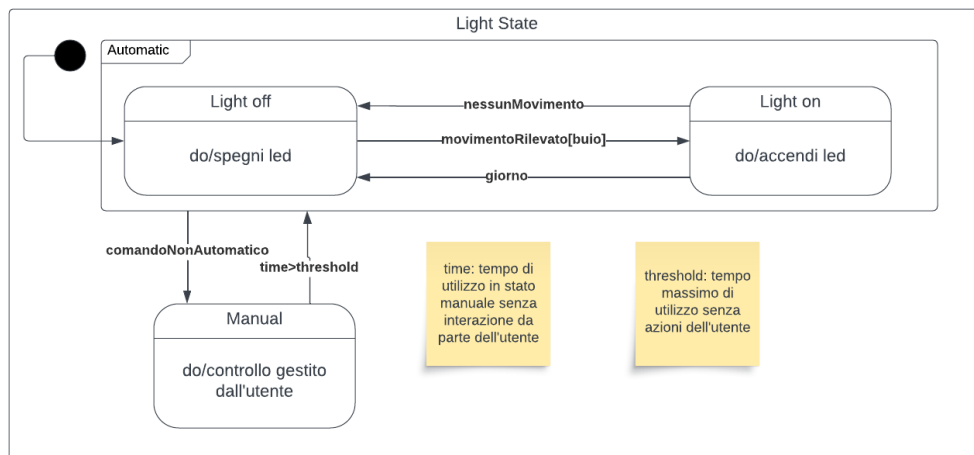


Figura 1.1: FSM of the light

### 1.1.2 Roller-Blinds

Il task per la gestione della tapparella ha tre stati principali, Rolled, ovvero il caso base in cui la tapparella viene abbassata, lo stato Unrolled, in cui ci si entra quando è giorno e viene rilevato un movimento, si esce dallo stato Unrolled quando è buio e non viene rilevato più nessun movimento. Questi stati rientrano nel macro-stato automatic, quando si esce da questo stato, per via di un comando manuale, si entra nell'ultimo stato, lo stato manual, dove il controllo viene affidato all'utente. Superato un certo limite di tempo di inutilizzo si ritorna alla macro-stato automatic.

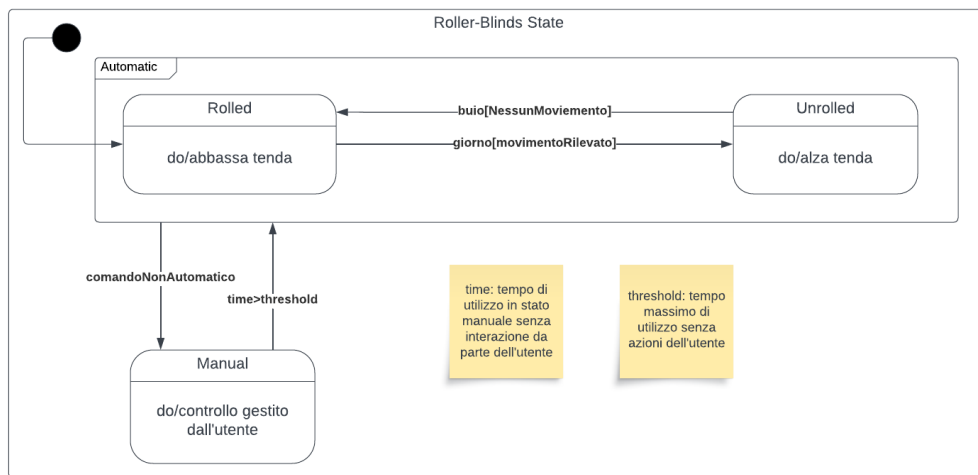


Figura 1.2: FSM of the roller-blinds

## 1.2 Java

Viene utilizzato come backend di questo progetto. Comunica con il modulo ESP tramite MQTT, che lo informa, tramite l'utilizzo di un PIR, se viene rilevato un movimento, e tramite un fotoreistore della quantità di luce presente. Queste informazioni vengono poi impacchettate per essere spedite ad arduino tramite seriale. Ricevuta successivamente una risposta da arduino, viene informata la dashboard, via HTTP, dei possibili cambiamenti dello stato della stanza, e viene aggiornato di conseguenza lo storico della luce e della tenda. Se non avviene nessun cambiamento, vengono comunque inviati i pacchetti, con gli stessi dati, e vengono rappresentati sulla dashboard.

## 1.3 ESP32

Principale responsabile della modalità automatica della stanza, comunica con java in maniera monodirezionale tramite MQTT. Si occupa di inviare i dati rilevati dai suoi sensori, ovvero il PIR, per il movimento, e il fotoreistore, per la luce presente, al backend.

## 1.4 Mobile App

Sviluppata come android app, viene utilizzata per regolare l'altezza della tenda, anche parzialmente, e per impostare lo stato della luce. Appena invia una delle due operazioni, si entra automaticamente in modalità manule, battendo anche la concorrenza della dashboard. Invia queste informazioni utilizzando il canale bluetooth ad arduino che di conseguenza aggiornerà fisicamente lo stato delle due componenti chiamate in causa.

## 1.5 Dashboard

La web-dashboard ci permette di visualizzare lo storico dello stato delle luci e consente di gestire, come l'app, l'altezza della tapparella e lo stato della luce nella stanza, inviando le informazioni al backend, tramite HTTP.

### 1.5.1 Breadboard

Abbiamo utilizzato l'applicazione web gratuita tinkercad per ricreare la progettazione dei vari componenti utilizzati. Purtroppo sono stati rimossi il modulo bluetooth e l'ESP quindi mancano nella rappresentazione. Il modulo ESP sarebbe collegato con il cavo azzurro e il cavo giallo, relativi al PIR e al fotoresistore, mentre il modulo bluetooth sarebbe stato collegato con i pin nell'arduino

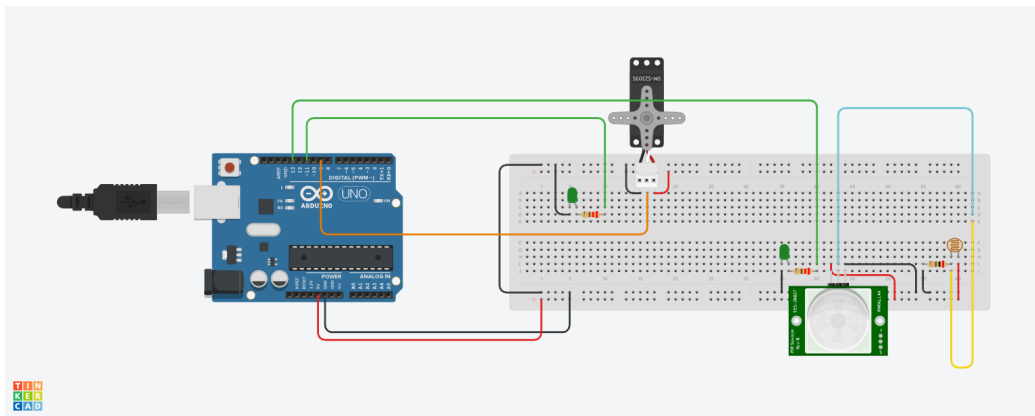


Figura 1.3: Breadboard