

Progetto di IOT SmartBin

Scorza Edoardo 0001077424
Giorgini Matteo 0001136576
Giuseppe Argenti 0001089431



1 dicembre 2024

0 Indice

1	Hardware	2
1.0.1	Componenti extra	3
2	Software	4
2.1	Sheduler	4
2.2	Task	4
2.3	FSM	4
2.4	Struttura	5
2.5	GUI	5
2.5.1	Comunicazione	6

1 Hardware

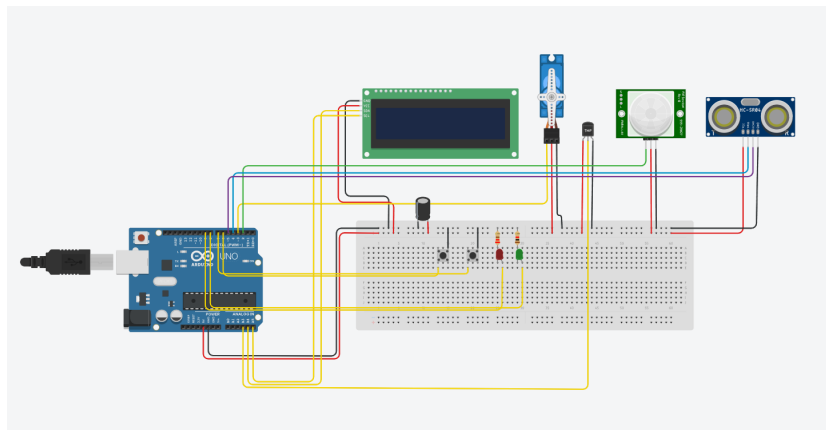


Figura 1.1: Rappresentazione del circuito in Thinkercad.

La implementazione fisica del circuito è realizzata con i componenti richiesti dalla specifica:

- **Display LCD I2C**
 - SDA: A4
 - SCL: A5
- **Button OPEN**
 - Pin: 6
- **Button CLOSE**
 - Pin: 7

- **Passive Infrared (PIR Sensor)**
 - Pin: 2
- **Sonar**
 - Trig: 4
 - Echo: 5
- **Red LED**
 - Pin: 8
- **Green LED**
 - Pin: 9
- **Temperature Sensor (LM35)**
 - Pin: A3
- **Servo Motor**
 - Pin: 3

1.0.1 Componenti extra

La scelta di usare il condensatore è dato dalla presenza di diversi componenti e dal servo che causa picchi dovuti all'improvviso azionamento di esso, mentre Arduino NANO è per semplificare il cablaggio del circuito.

2 Software

Per la realizzazione del software abbiamo optato per un sistema di Task e FSM, inizialmente prevedavamo l'uso di una libreria, ma, il mancato supporto di Functional ci avrebbe impedito di realizzare un oggetto con dentro la FSM.

2.1 Scheduler

Per lo Scheduler abbiamo usato la base trovata nel codice del corso e lo abbiamo modificato per supportare la comunicazione di Task.

2.2 Task

La classe Task è una versione modificata di quella base, con un riferimento alla propria variabile condivisa e un accesso a quelle delle altre Task.

2.3 FSM

Per la struttura e le specifiche richieste, la macchina a stati finiti è stata realizzata con uno switch, nella quale sono definite transizioni, chiamate in entrata, uscita e Timeout.

2.4 Struttura

Il progetto è separato in varie Task, suddivise in due categorie:

- **Task di Report**

Queste Task si occupano di leggere e/o passare dati:

- **ButtonTask**
- **TemperatureTask** (FSM)
- **GuiTask**
- **WasteDetectorTask**
- **UserDetectorTask** (FSM)

- **Task Decisionali**

Queste Task operano sull'hardware:

- **BinTask** (FSM)

2.5 GUI

Per la implementazione della GUI abbiamo optato per Python, per la sua praticità e semplicità.

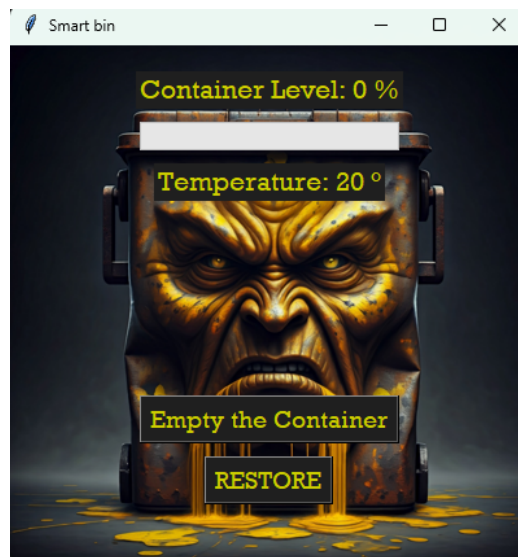


Figura 2.1: Screenshot della GUI in Python.

2.5.1 Comunicazione

Lo scambio di dati avviene mediante la **COM**. La **ricezione** della temperatura e del livello del bidone avviene tramite testo scritto nel monitor seriale da Arduino. Mentre la codifica della pressione dei bottoni **Empty** e **Restore** avviene tramite i caratteri **E** ed **R** stampati nel monitor seriale dalla GUI.