Sistemi Embedded e Iot - a.a. 2019/20

Relazione Progetto #3 Smart Dumpster

Stefanini Andrea, Canali Mattia, Mingaroni Anna

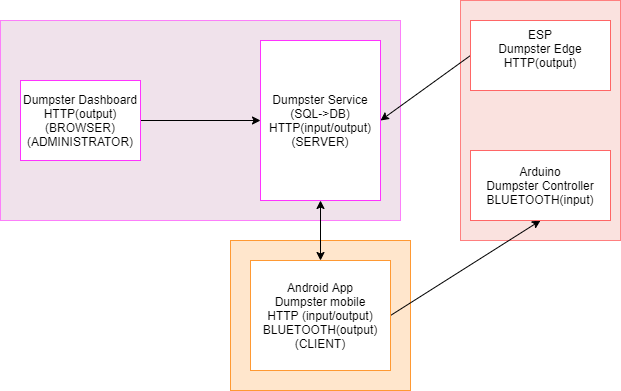
1. ANALISI E PROGETTAZIONE

La richiesta è quella di realizzare uno Smart Dumpster, ovvero un sistema IOT che gestisca e simuli la funzione di un cassonetto dei rifiuti smart realizzato tramite l’utilizzo di **Arduino** (BLUETOOTH/SERIALE), un **ESP** (HTTP/SERIALE), un **Server Web** (a libera scelta) una **Applicazione Android** (BLUETOOTH/HTTP) e un’interfaccia grafica, **Dashboard**, (HTTP) dove monitorare l’andamento dei rifiuti.

Abbiamo deciso di implementarlo considerando l’applicazione Android e la web dashboard come CLIENT, il dumpster service scritto in PHP funzionerà come SERVER e comunicherà con il dumpster fisico tramite (l’ESP) e con l’applicazione android per mezzo di richieste e risposte HTTP.

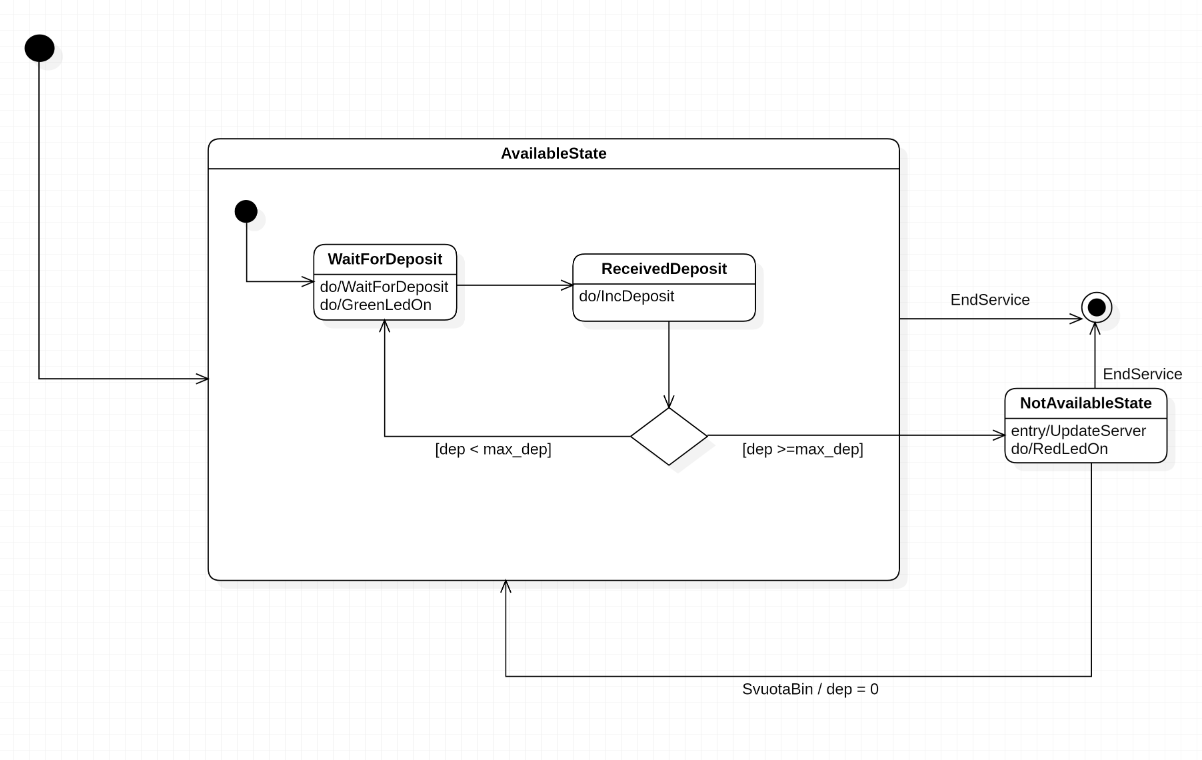
I sottosistemi fondamentali del sistema finale sono:

1. **Dumpster Dashboard**, riguardo la visualizzazione analisi e controllo dei livelli dei depositi (front end).
2. **Dumpster Service** (Server), riguardo al continuo tracciamento dello stato del cassonetto (REST).
3. **Dumpster App** (Client), affinché l’utente possa interagire con il cassonetto e procedere con l’iniziazione del deposito rifiuti.
4. **Physical Dumpster**, suddiviso in **Dumpster Controller**, riguardante il controllo del cassonetto e l’interazione con gli Utenti, e **Dumpster Edge**, riguardante il monitoraggio della quantità dei rifiuti all’interno del cassonetto e l’invio di informazioni, via HTTP, alla parte Service.
5. IMPLEMENTAZIONE

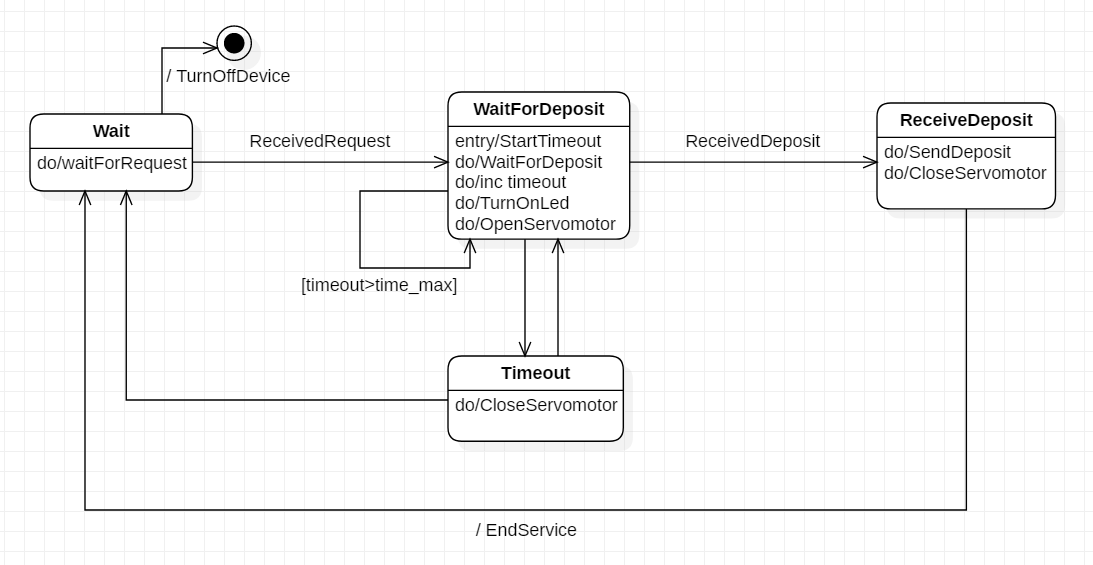
 Viola = Rete, Rosso = Fisico, Arancione = App

1. DIAGRAMMI A STATI FINITI

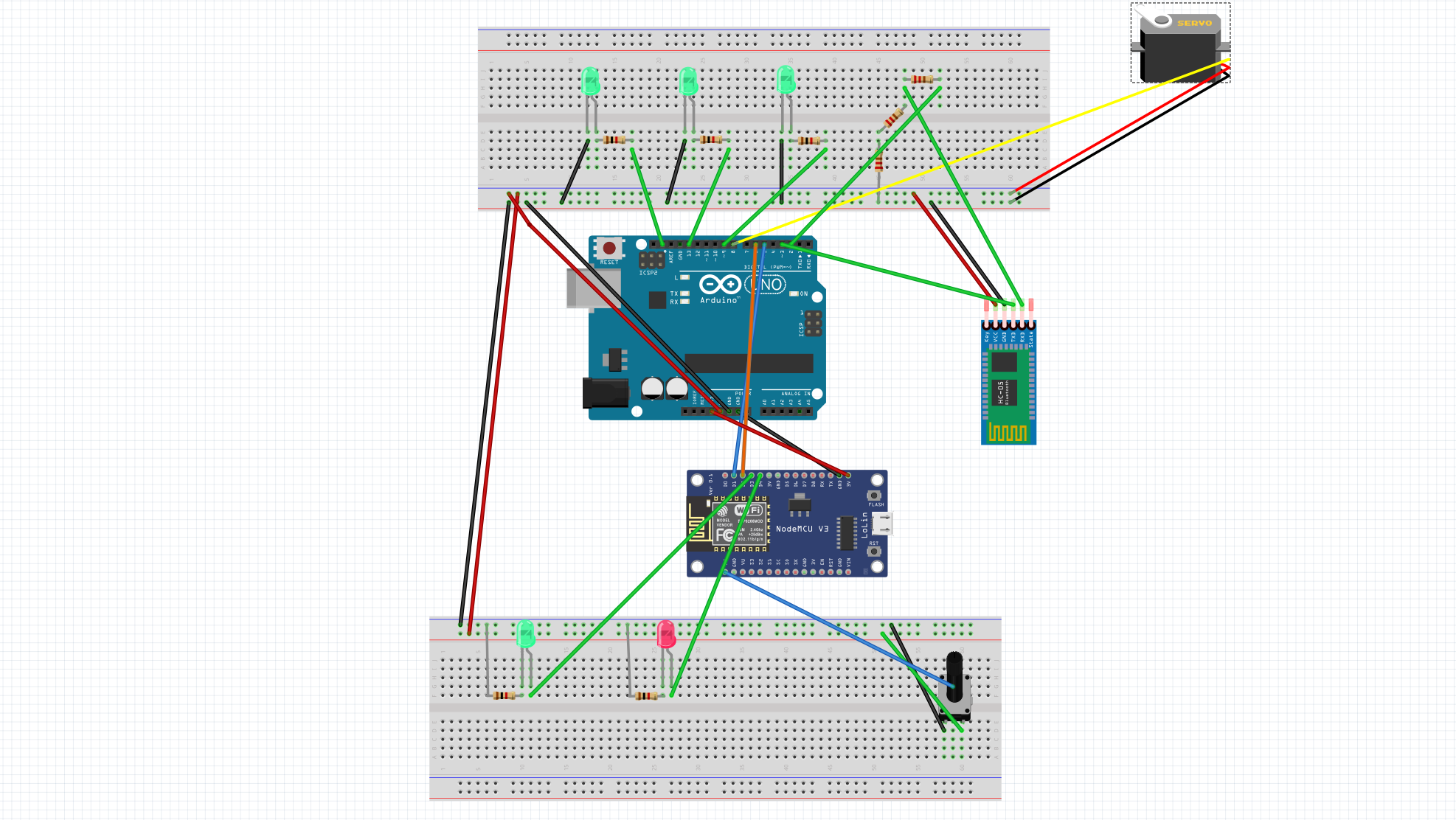
EDGE



CONTROLLER



1. ARDUINO COLLEGAMENTI E PERIFERICHE



1. STRUTTURA E SVILUPPO
2. **Microcontrollore e Esp:**

Arduino (microcontrollore) tramite il modulo HC-05 si interfaccia e scambia messaggi bluetooth con l’applicazione android, inoltre sono presenti 3 led su arduino, per definire il tipo di rifiuto che deve essere buttato **La Lb Lc**, e un servo motore **M** che simula l’apertura e la chiusura dello sportello.

L’ESP si occupa di monitorare la quantità di rifiuti presenti nel cestino, con 2 led che mostrano lo stato del sistema **L\_avaliable** o **L\_notavaliable**, comunicando con il server (tramite rete wifi) il peso e la tipologia di rifiuto, inoltre se i rifiuti superano una certa **Wmax** decide e trasmette al server lo stato che esso deve tenere **available or** **not available** aggiornando di conseguenza i led.

Questo rappresenta la parte fisica del vero e proprio sistema IoT, ovvero lo smart dumpster.

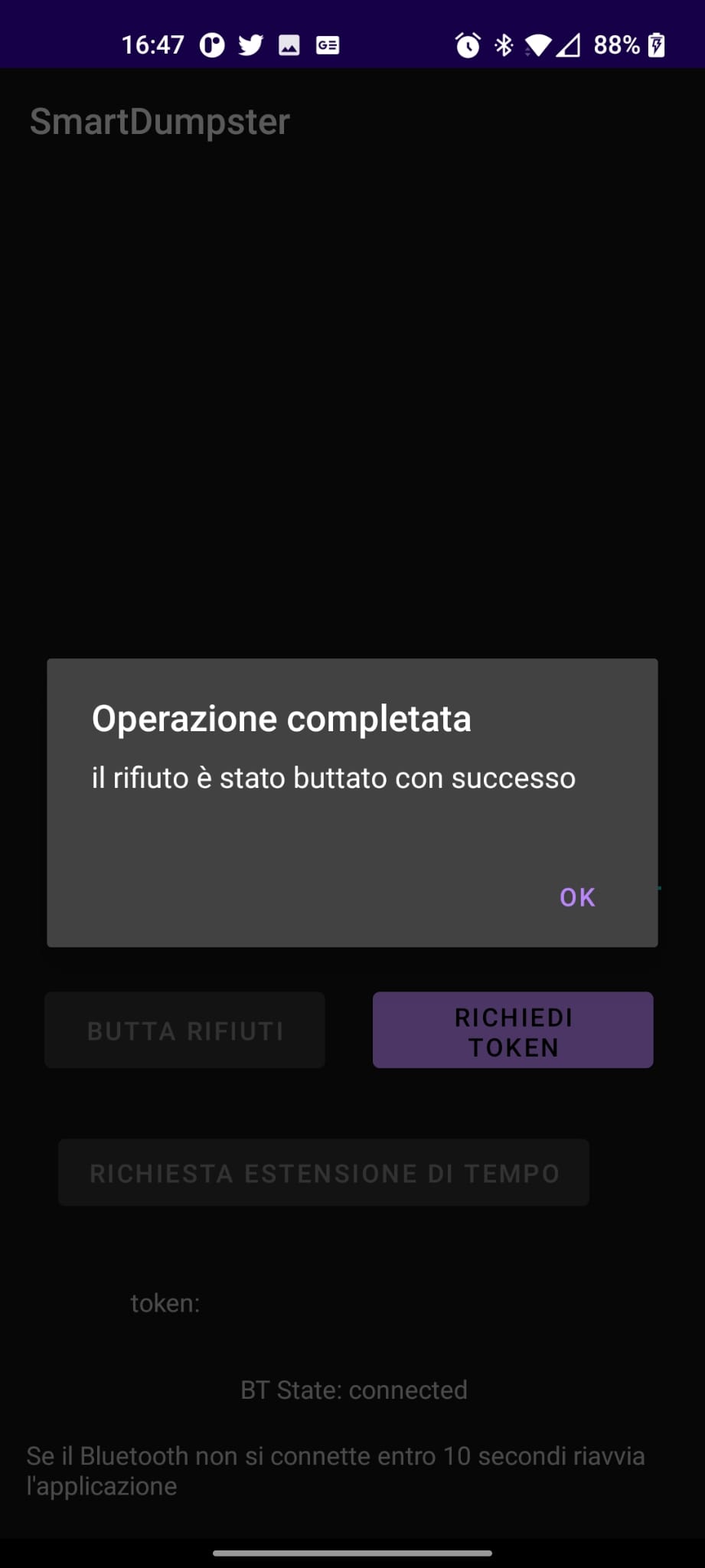
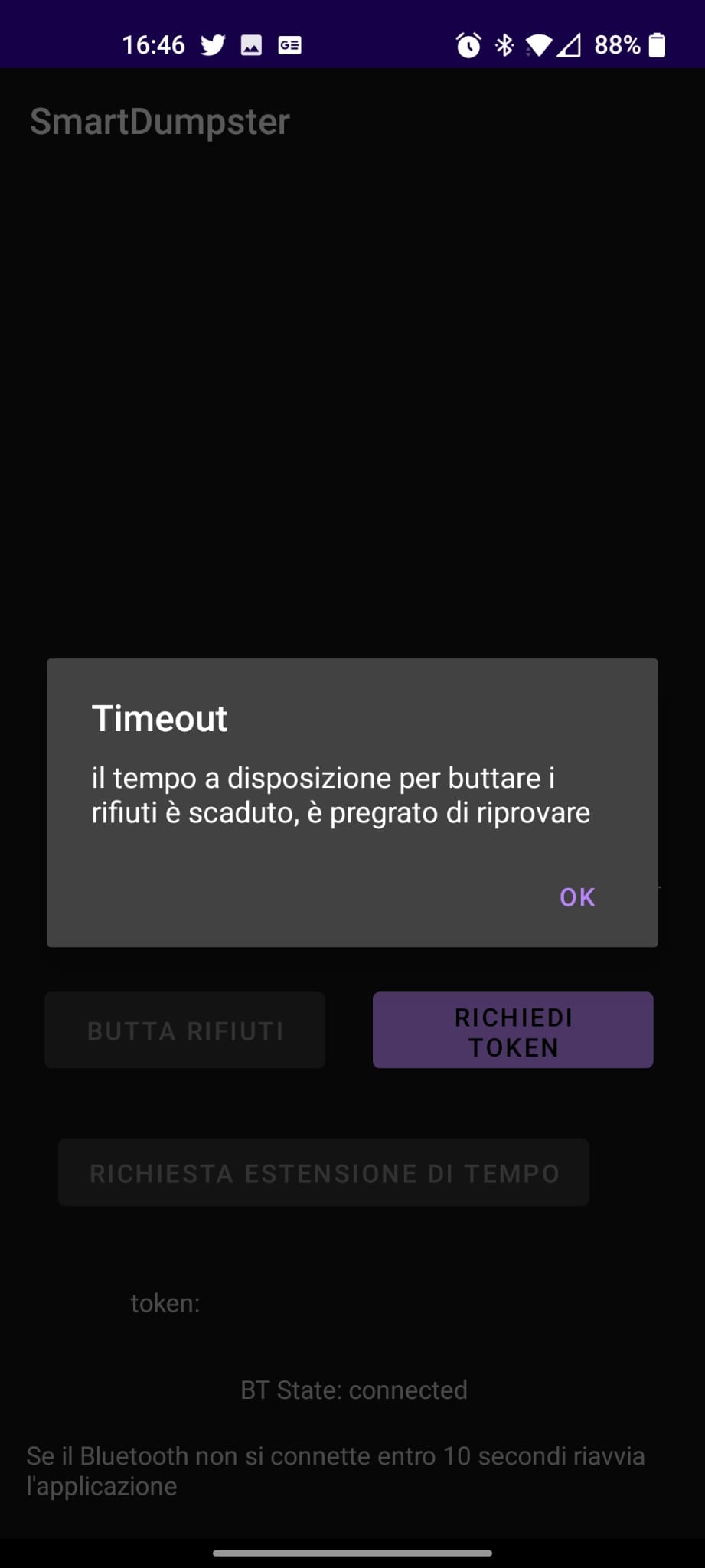
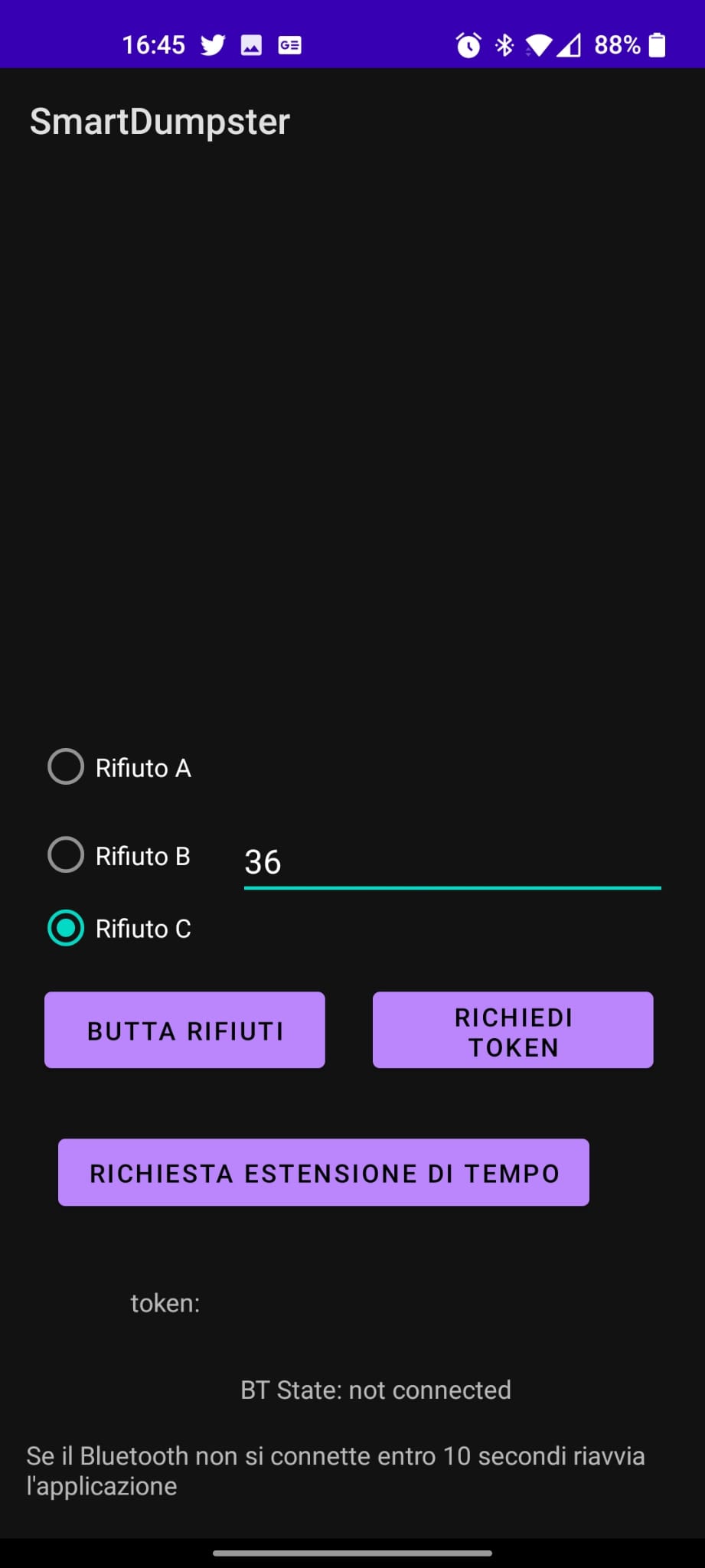
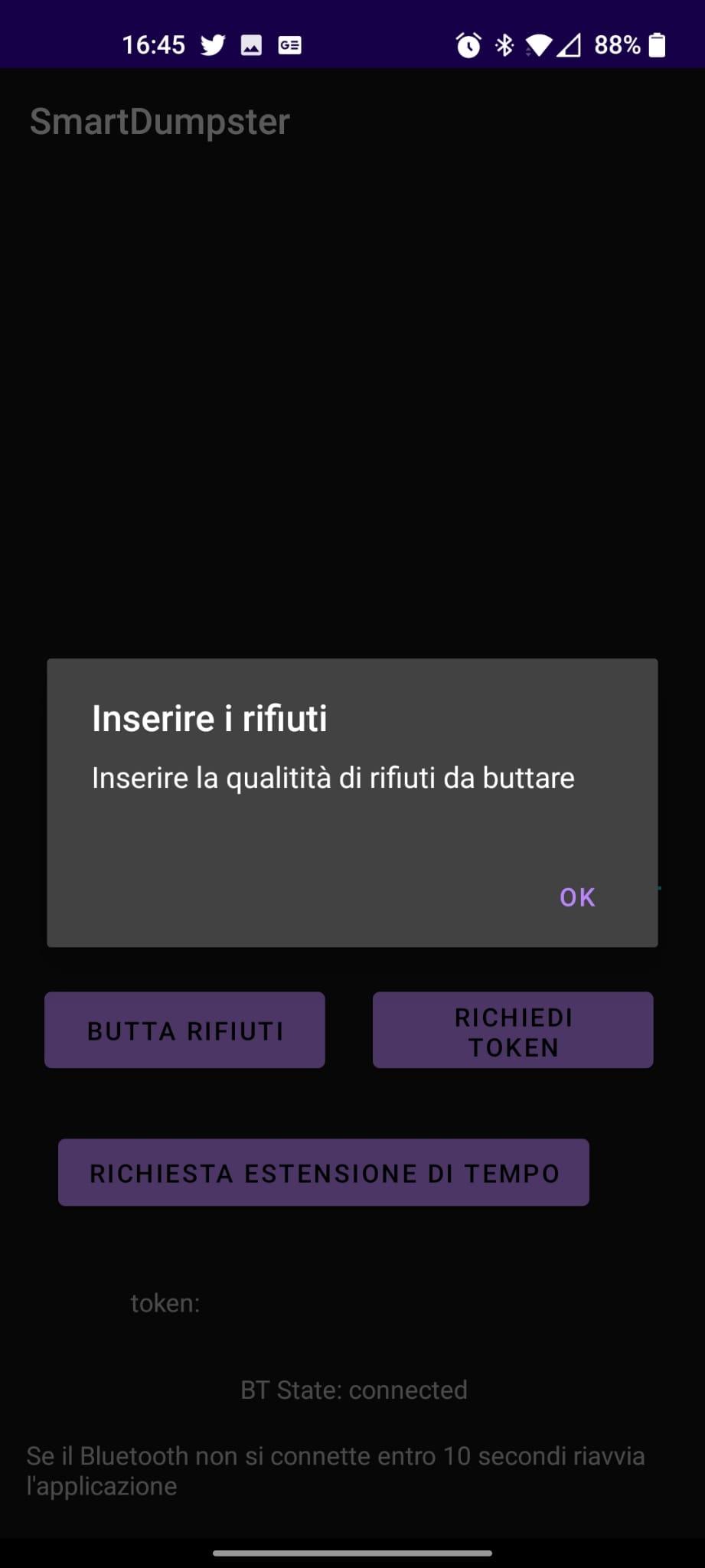
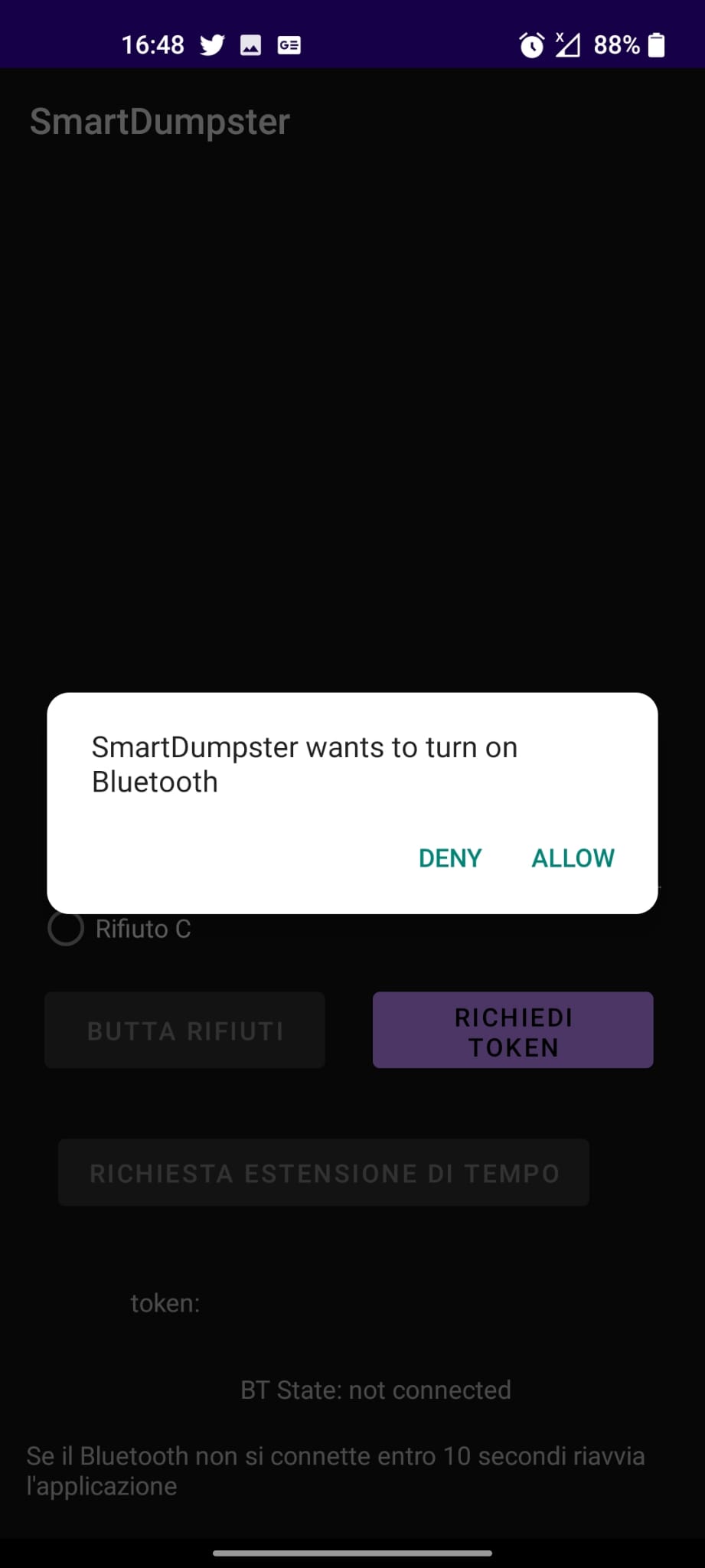
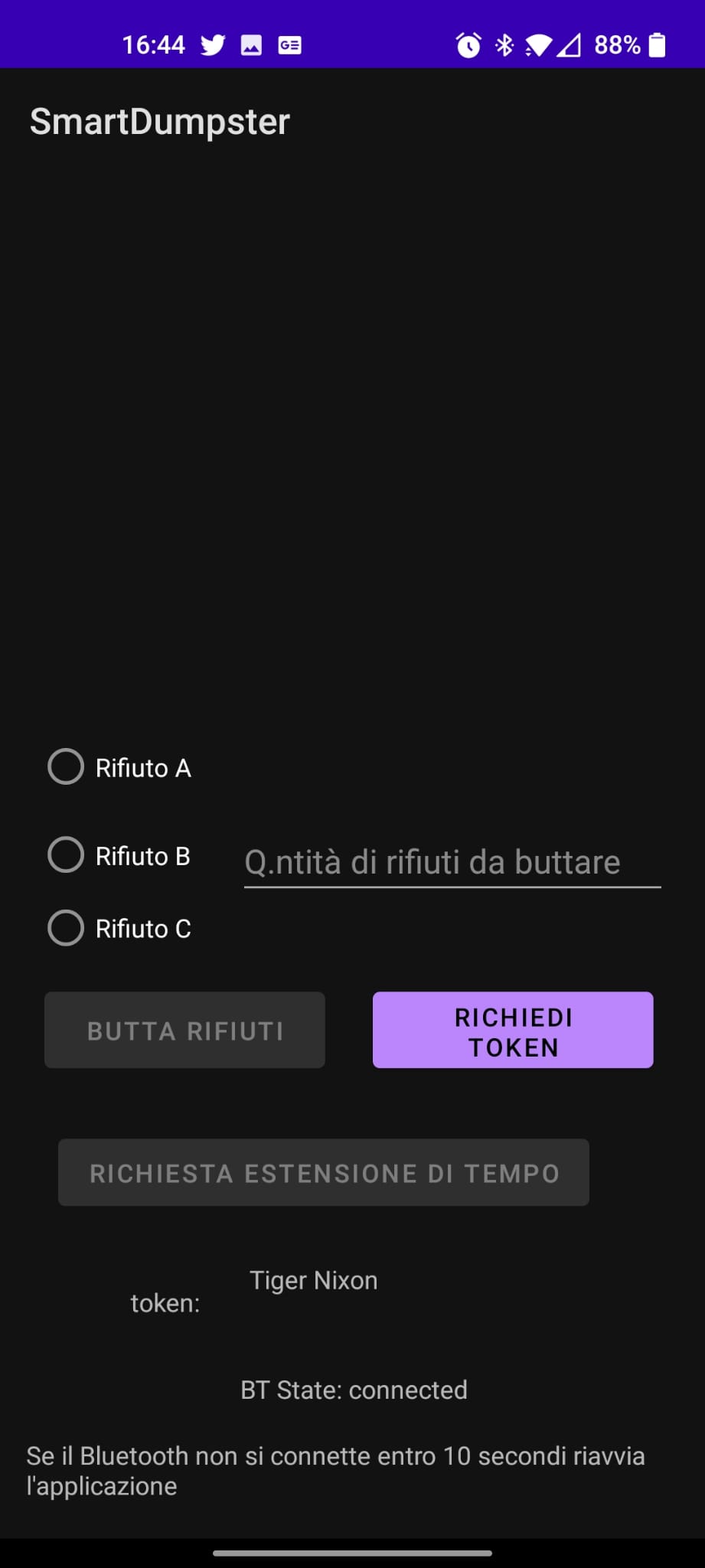
1. **Applicazione Android:**

Il compito dell'applicazione android è quello di fornire l'accesso al cliente finale, che tramite l’applicazione potrà scegliere che tipologia di rifiuto buttare all’interno del cestino e la quantità, precedentemente viene controllato un token che serve a dimostrare che l’utente possiede l’autorizzazione per gettare rifiuti nel cestino a seconda della disponibilità del server.

Abbiamo sviluppato l’app android con l’ausilio del ide android studio, e tramite l’utilizzo di **Volley** e **bt-lib**, utilizzate rispettivamente per la comunicazione HTTP e BLUETOOTH.

La comunicazione **HTTP** viene sfruttata per la consegna del token e controllo/aggiornamento dello stato del server, mentre quella **BLUETOOTH** comunica il peso e la tipologia del rifiuto.

Di seguito alcuni screenshot del funzionamento dell’App SmartDumpster

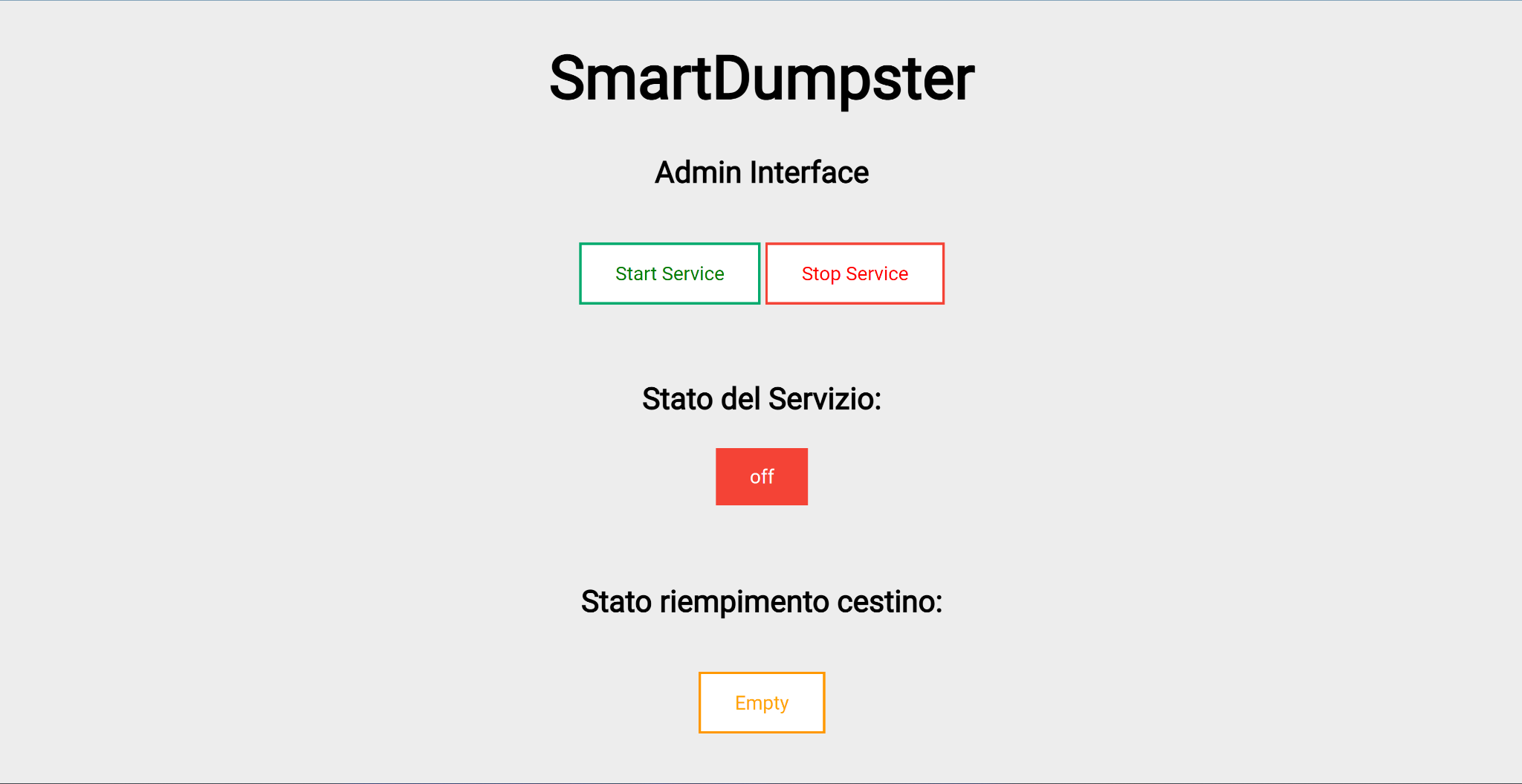


1. **Server e Dashboard:**

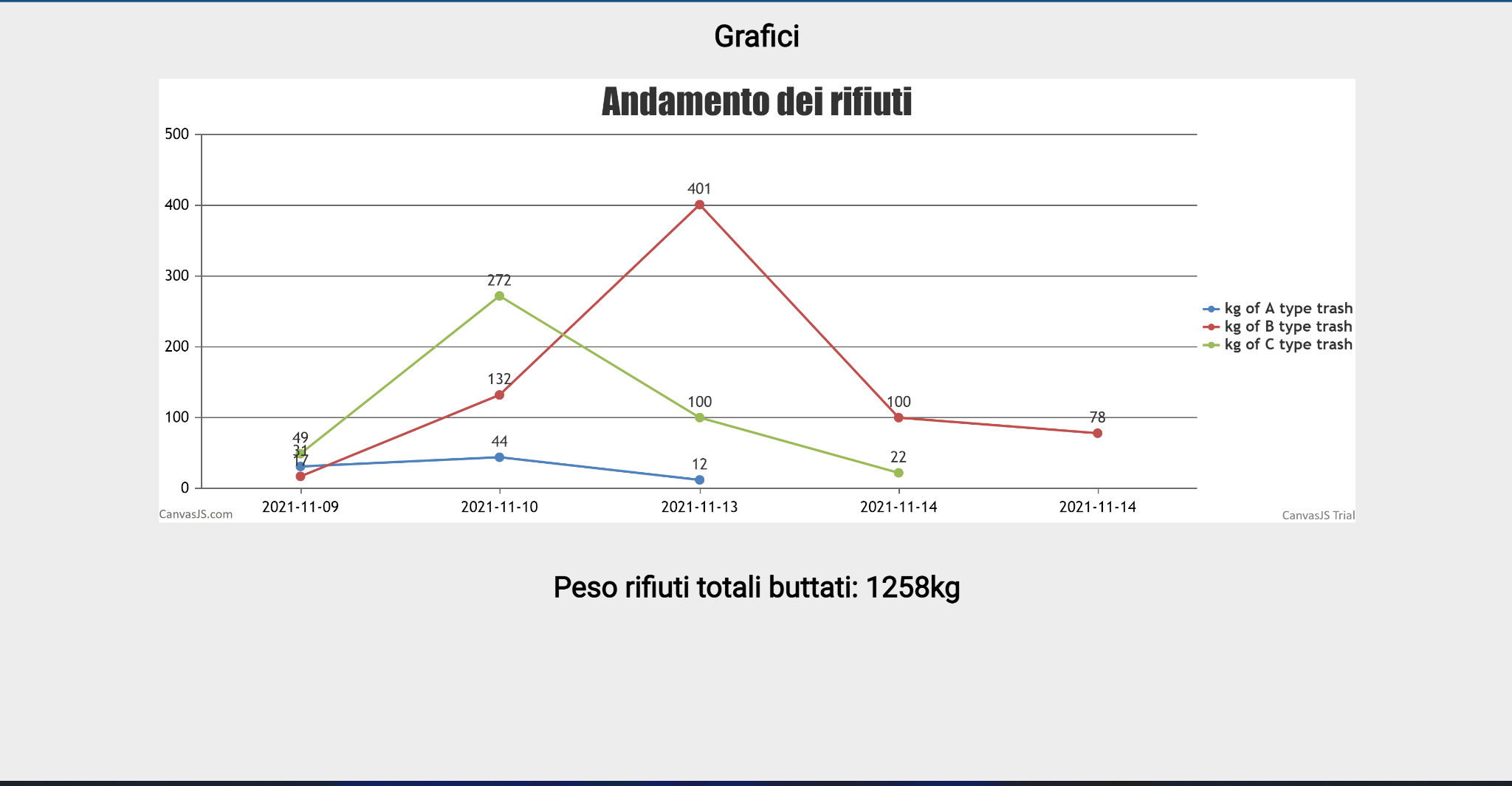
Abbiamo sviluppato il server in **PHP**, che tramite query comunica con un database sql per il salvataggio dati e l’aggiornamento dello stato del servizio.

Utilizzando **HTML** e **CSS** abbiamo creato una dashboard minimale che permette all'amministratore la gestione dello smart dumpster e la visualizzazione dei dati contenuti nel database**.**

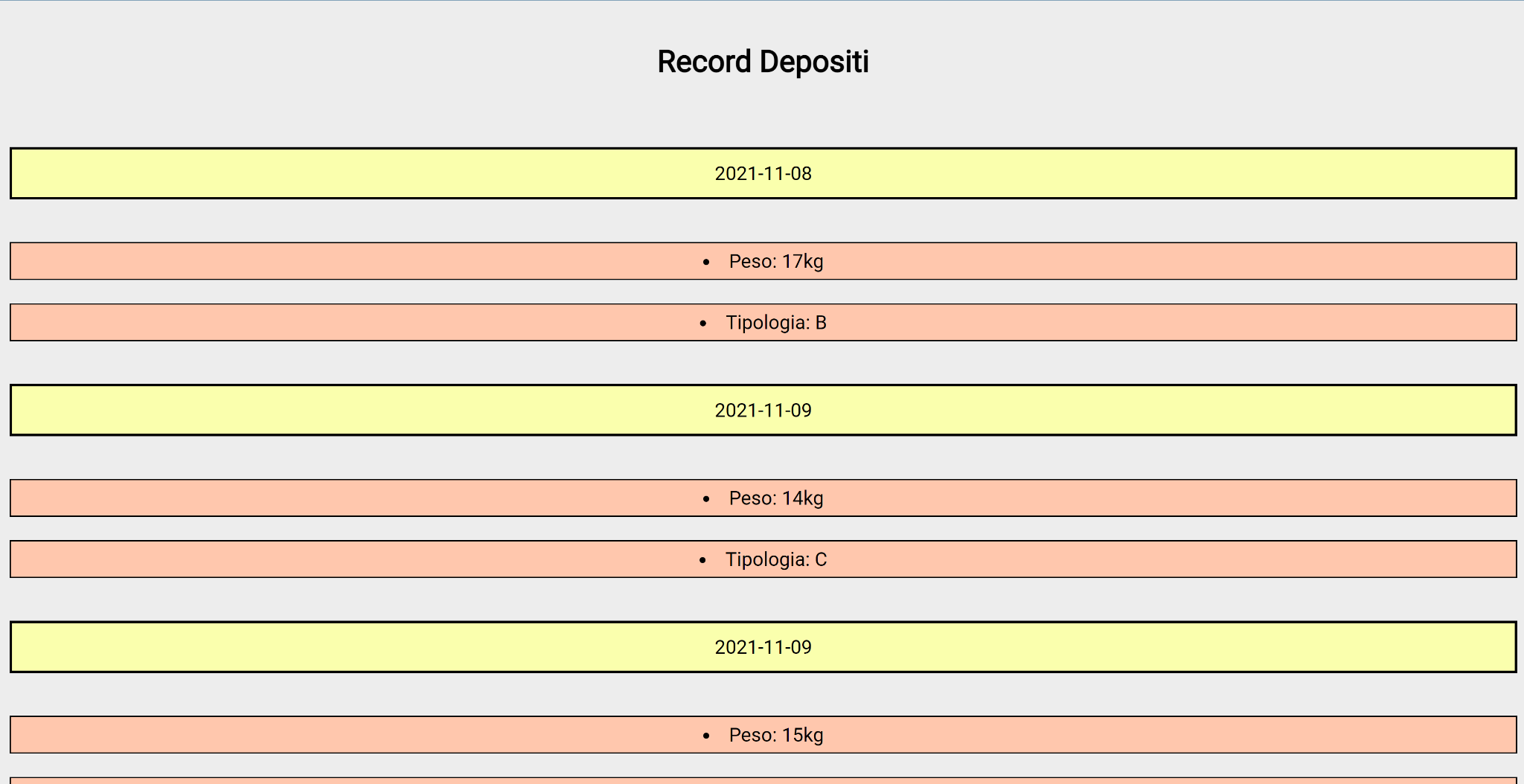
l’immagine sopra riportata rappresenta l’Homepage della dashboard service:



Qui invece viene riportata la schermata offerta per avere un’idea dell’andamento del bidone nel tempo:



ed infine viene offerto il record dei rifiuti buttati:



Inoltre tramite gli appositi bottoni, l’admin può decidere se stoppare il servizio, attivarlo e svuotare quindi il bidone

**Migliorie applicabili al progetto**:

* Sistema con registrazione, login e validazione utente, il token in questo caso diventerebbe superfluo
* Servo motore aggiuntivo per differenziare automaticamente i rifiuti
* Aggiunta di pulsanti fisici (sullo smart dumpster) per la selezione dei rifiuti
* Display per la visualizzazione del peso dovo aver gettato un rifiuto, e per la conferma del login effettuato
* Lettore rfid/nfc per la validazione utente tramite carta e senza dispositivo
* Lato applicazione utente sarebbe utile far visualizzare il peso dei rifiuti e possibili sconti collegati per incentivare il riciclaggio
* Sistema centralizzato con dashboard web per la gestione di molteplici SmartDumpster con ottimizzazioni e previsioni per generare i percorsi migliori da tenere per svuotare i dumpster