SCENARIO 1 – IL CLIENTE RICERCA UN DETERMINATO LIBRO DA TELEGRAM:

* L’utente accede a telegram previa autenticazione
* Ricerca un libro
* Il server ritorna l’elenco dei totem che lo contengono o il più vicino. (RICHIESTA 1)

SCENARIO 2 – IL CLIENTE PRENOTA UN LIBRO

* L’utente seleziona il totem dalla lista appena ritornata
* Conferma la prenotazione, viene creato un prestito sul db e viene ritornato all’utente un codice univoco di prenotazione da utilizzare per il ritiro e per la riconsegna del libro
* Il server fa una publish MQTT al topic del totem selezionato, specificando l’id dello scompartimento e l’id della prenotazione (PACCHETTO 1, CODICE=1)
* Il bridge legge e comunica al microcontrollore di accendere il led rosso dello scompartimento corrispondente. (PACCHETTO 3, CODICE=1)

SCENARIO 3 – IL CLIENTE RITIRA IL LIBRO

* L’utente va presso il totem (applicazione telegram), scrive \ritira e inserisce il codice prenotazione
* Il server invia al bridge l’id dello scompartimento e il codice di prenotazione (PACCHETTO 1, CODICE=2)
* Il bridge comunica al microcontrollore l’id dello scomparto da aprire (PACCHETTO 3, CODICE=2)
* Il led lampeggia finchè l’NFC lo riconosce, dopodichè si spegne e l’arduino conferma al bridge che il ritiro è avvenuto per quello scomparto (PACCHETTO 2, INTERO=1)
* Il bridge risale al codice prenotazione associato a quello scompartimento e fa una richiesta HTTP per aggiornare lo stato del libro (RICHIESTA 5, PACCHETTO 4).

SCENARIO 4 – IL CLIENTE RICERCA UN LIBRO DAL TOTEM

* Il cliente è davanti al totem, inserisce il suo codice utente (autenticazione automatica http al server)
* Ricerca un libro.
* Se il libro è presente all’interno di quel totem viene mostrato a video un opportuno messaggio e viene chiesto se si vuole ritirare subito il libro. In caso di risposta positiva il server crea il prestito e si avvia lo SCENARIO 3 dal punto 2.
* Se il libro è su un altro totem: vuoi ricercare su un altro totem? In caso di risposta positiva inizia lo SCENARIO 1 dal punto 3.

SCENARIO 5 – RICONSEGNA DEL LIBRO

* L’utente scrive /consegna su telegram
* Ritorna i totem che hanno spazio disponibile.
* L’utente va al totem, inserisce il codice di prenotazione e gli viene mostrato il numero dello scompartimento in cui lasciare il libro
* Il server comunica al bridge l’id della prenotazione e l’id dello scompartimento (PACCHETTO 1, CODICE=3)
* L’utente lascia il libro nello scompartimento
* L’arduino legge l’NFC, comunica al bridge il numero dello scomparto (PACCHETTO 2, INTERO=2) e accende il led verde
* Il bridge riconosce l’id della prenotazione e fa una richiesta http al server per confermare la riconsegna del libro. (RICHIESTA 5)

ESEMPIO RICHIESTE HTTP

1. GET localhost:5000/totems/IDTOTEM

ritorna il totem (dati del totem, posizione…) e i libri contenuti in quel totem

1. GET localhost:5000/totems?nomeLibro=NOMELIBRO

ritorna i totem che contengono quel libro

1. POST localhost:5000/prenotazioni

Inserisce una nuova prenotazione di un libro su un totem (parametri nel body)

1. POST localhost:5000/login e signup
2. PUT localhost:5000/prenotazioni/IDPRENOTAZIONE

Aggiorna lo stato della prenotazione (parametro stato prelevato o consegnato)

Il messaggio inviato è un json:

{

‘totemID’: 1,

‘codicePrenotazione’: 1234,

‘idScompartimento’: 5,

‘codice’: ‘ritirato’ (o ‘riconsegnato’)

}

CONVENZIONE PACCHETTI

1. Pacchetto MQTT da server a bridge: IDSCOMPARTIMENTO/CODICE\*/IDPRENOTAZIONE
2. Pacchetto Seriale da arduino a bridge: BYTE\_INIZIO(ff)/IDSCOMPARTO(2 byte)/INTERO\*(1 byte)/BYTE\_FINE(fe)
3. Pacchetto Seriale da bridge ad arduino: BYTE\_INIZIO(ff)/IDSCOMPARTO(2 BYTE)/CODICE\*(1 BYTE)/BYTE\_FINE(fe)
4. Pacchetto HTTP da bridge a server:
   1. RICHIESTA 5

\*CODICE (1 = libro prenotato, 2 = libro pronto per il ritiro, 3 = scompartimento in cui mettere il libro)

\*INTERO (1 = libro ritirato, 2 = libro riconsegnato)

NOTE