Università di Verona

Dipartimento di Informatica

Corso Prototipizzazione con Arduino

Gestione e controllo accessi da remoto

Autori:
Matteo Esposito Marroccella VR451657
Nicola Speri VR437023

A.A. 2021/2022

Indice

Elenco delle figure

1 Introduzione

È stato realizzato un sistema per la gestione e il controllo degli accessi tramite **RFID**. Le chiavi per l'accesso e il relativo nominativo del proprietario sono custodite in un database **MySql** che viene interrogato dopo ogni richiesta di entrata. Nell'eventualità che un utente sia sprovvisto di tag è disponibile l'accesso anche tramite PIN numerico su tastierino. Una volta effettuato l'accesso, esso verrà segnalato e registrato su una lista online su **Google Sheets** che potrà essere visionata e scaricata dagli appositi operatori.

L'obbiettivo è quindi quello di presentare un semplice sistema di sicurezza. Il sistema pur nella sua semplicità può essere molto efficace. I permessi di accesso possono essere dati e tolti da remoto accedendo al database.
L'idea che sta alla base è quello di un suo utilizzo in una struttura in cui non sono presenti operatori umani ed è necessario gestire, modificare e verificare gli accessi da remoto.

Il link della repository *Github* è il seguente:

https://github.com/matteoespo/ElaboratoPrototipizzazioneArduino.git

2 Materiali

Di seguito tutti i materiali utilizzati per il progetto.

- **ESP32**: è il cuore del progetto. 1)Gestisce la parte di connessione al database per verificare la presenza della chiave rfid, 2)invia i dati degli accessi attraverso Arduino Cloud e compila il foglio degli accessi, 3)gestisce il modulo Rfid (RC522) e 4)Consente o rifiuta l'accesso sia tramite rfid che tramite Pin da tastierino numerico.

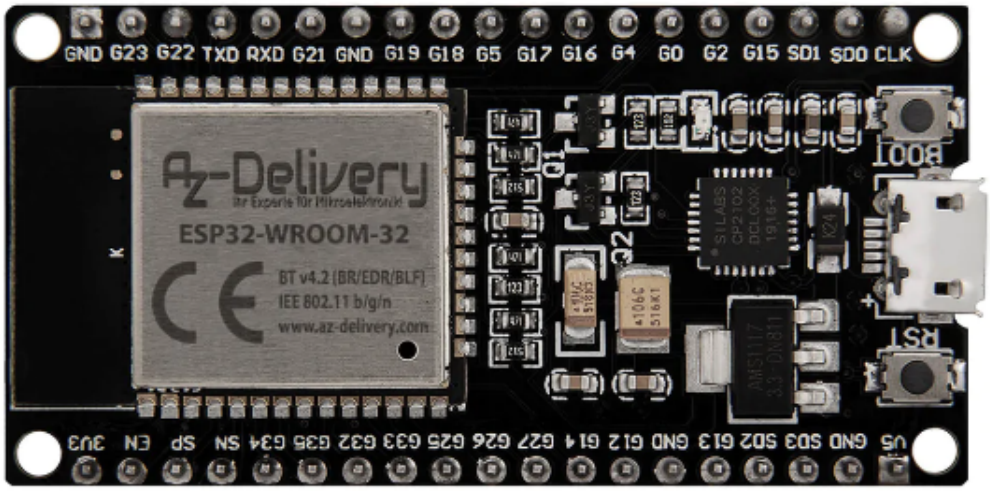

Figura 1: Microcontrollore ESP32

- **RC522**: questo modulo è di tipo rfid e funziona sia in lettura che in scrittura, il modulo rileva la chiave univoca appartenente al tag rfid.

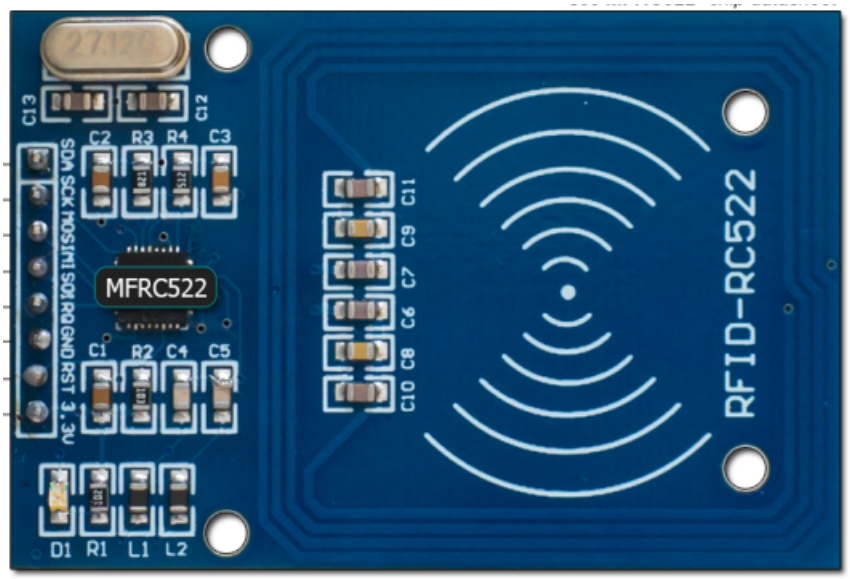

Figura 2: Modulo Rfid RC522

- **Keypad a membrana 4x4**: questo modulo consente l'eventuale accesso classico senza connessione internet

Figura 3: Modulo keypad 4x4

- **Led**: sono stati usati due led, per semplificare una ipotetica serratura: uno verde che indica che l'accesso è consentito, uno rosso che indica che l'accesso è negato.

Figura 4: Led

- Resistori: i resistori sono necessari per i led e hanno una resistenza di 200Ω

Figura 5: resistore

3 Circuito

3.1 Schema

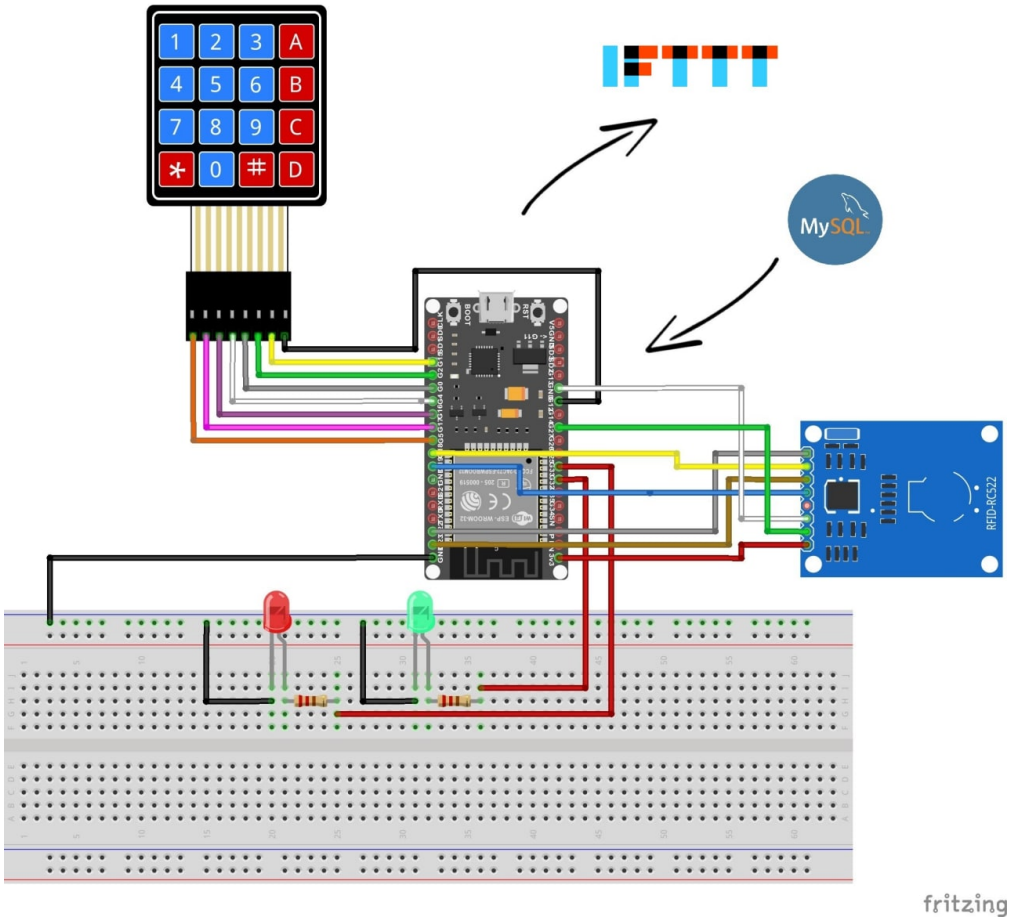

Figura 6: Schema del circuito con Fritzing

3.2 Funzionamento

3.2.1 Accesso con Password

Ogni persona in possesso del Pin lo digiterà nel tastierino numerico e se il Pin è corretto il led verde si accenderà, quello rosso altrimenti.

3.2.2 Accesso con RFID

Avvicinando il tag rfid al modulo RC522 verrà effettuata la lettura della chiave e sarà passata al nostro ESP32.
Il microcontrollore andrà ad interrogare tramite uno script PHP il nostro database MySql e verificherà la presenza della chiave. Se la chiave non sarà presente, l'accesso verrà negato. Se la chiave sarà presente, il database restituirà la chiave e il nome della persona a cui appartiene.
Una volta ricevute tali informazioni il nostro microcontrollore andrà ad acconsentire l'accesso. Nel frattempo si sarà anche connesso ad Arduino Cloud e tramite il servizio IFTT avrà inserito una nuova riga sul foglio Google Sheets con i dati della persona che ha effettuato l'accesso e data e ora dell'accesso.

4 Database e IFTT + Google Sheets

Di seguito alcuni dettagli della parte implementativa remota.

4.0.1 MySQL

Per la parte di verifica della chiave abbiamo creato un database MySQL con una tabella contenente due colonne: uid della chiave e nome del proprietario.
L'ESP32 effettuerà dunque una chiamata HTTP GET con url ip del pc HOST + script php + id da ricercare nel database.
Un esempio di url è la seguente: http://192.168.1.9/test.php?uid=153724 .
Una volta effettuata la richiesta GET verrà eseguito lo script php. Lo script eseguirà una query select sul database con l'uid passato e restituirà la riga con uid e nome se è presente, 0 altrimenti. Ricevuto il contenuto della risposta HTTP l'ESP32 saprà se consentire l'accesso o meno.

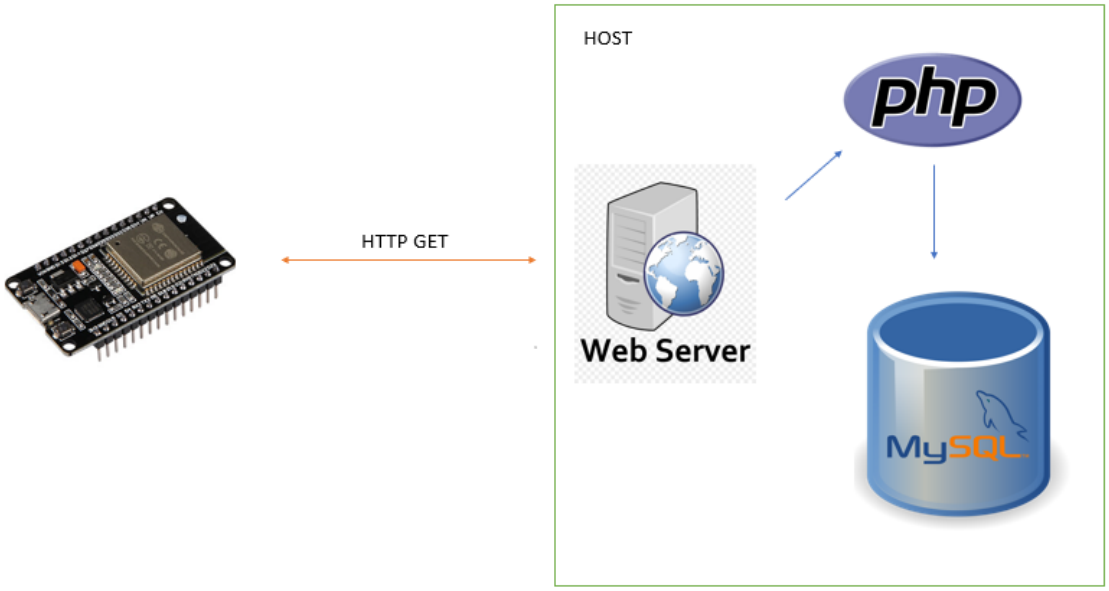

Figura 7: Schema parte database

4.0.2 IFTT + Google Sheets

La parte di salvataggio degli accessi utilizza due piattaforme:

- **IFTT** che è un servizio web gratuito che permette la creazione di semplici catene di condizioni, chiamate *applet*.
- **Google Sheets** che è una suite gratuita di google su web che consente di usare fogli di calcolo.

L'ESP32 una volta che ha verificato che una chiave o un pin sia valido per l'accesso,oltre ad aprire il cancello/serratura (nel nostro caso per comodità accendiamo un led verde), invia su un foglio di google un log degli eventi.Questo accade indipendentemente se l'accesso è stato autorizzato o meno e sia che venga autorizzato mediante chiave rfid che nel caso in cui l'accesso viene autorizzato tramite combinazione numerica.In particolare attraverso un url specifico e una applet che integra webhooks ogni qualvolta che viene scatenato l'evento nuova chiave uid e nuovo nome, viene salvata una nuova

entries sul foglio di google contenente oltre questi due dati anche la data e l'ora come primary key della entries.In particolare se l'accesso è garantito da chiave rfid viene memorizzato l'uid e il nome del proprietario della chiave. Se il tentato accesso tramite chiave rfid viene negato(questo perchè l'uid di quella chiave non era presente in database) viene salvato sul foglio l'uid della chiave e al posto del nome viene scritto 'unknownuid'.Se l'accesso invece viene effettuato correttamente tramite tastierino numerico,al posto dell'uid non troveremo nulla mentre al posto del nome troveremo la scritta 'rightpassword', al contrario se l'accesso verrà negato per password errata troveremo la scritta: 'wrongpassword'.

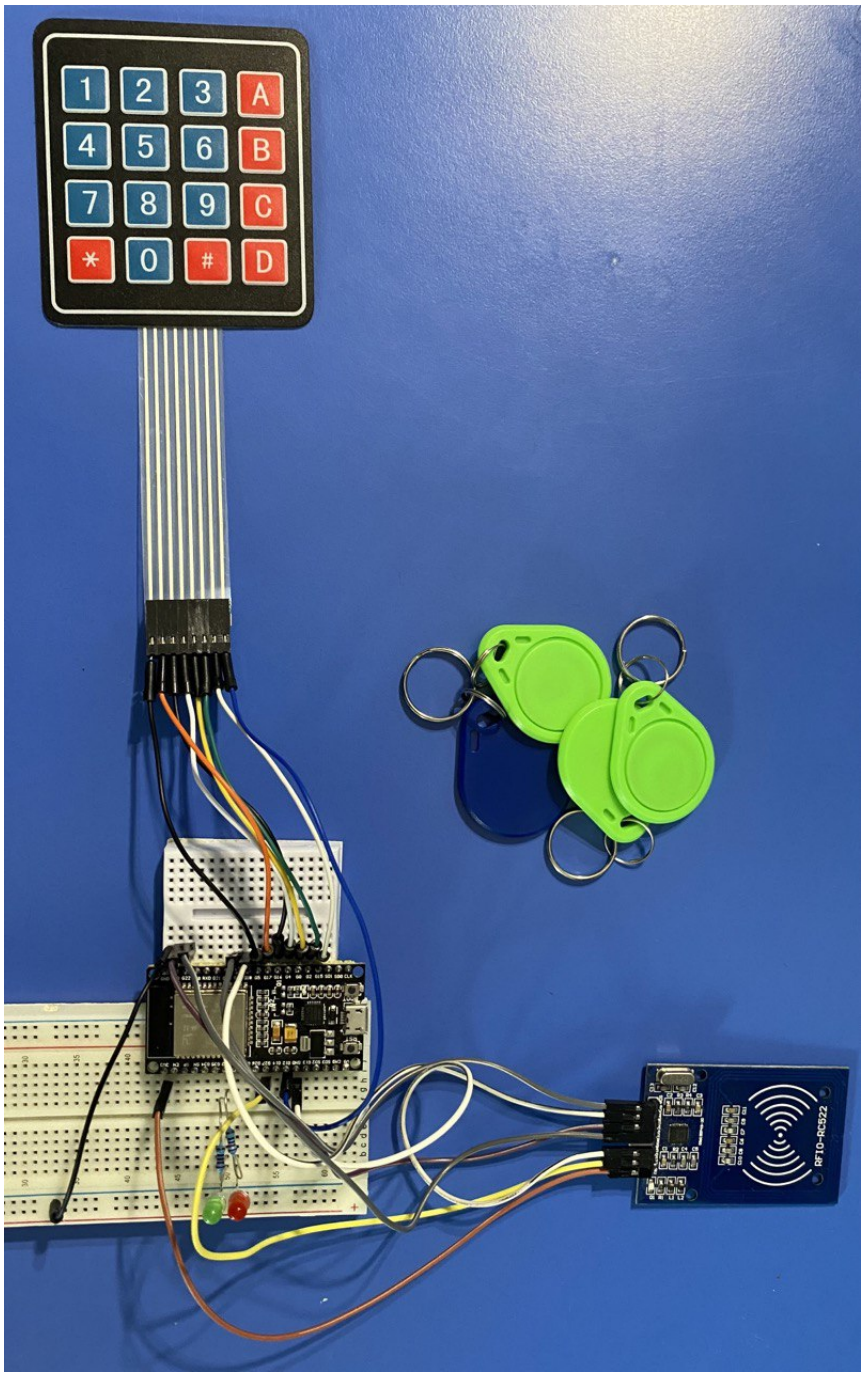

Figura 8: Foto progetto