

1. Ente presso cui è stato svolto il lavoro di stage

Il lavoro di stage è stato svolto presso *Yatta*, il primo makerspace in centro a Milano. Yatta è uno spazio attrezzato con tecnologie e macchinari digitali destinato alla prototipazione e gestito da professionisti del settore 3D, programmazione, elettronica e grafica, in una visione ampliata del concetto di artigiano orientato verso la Manifattura e Industria 4.0.

2. Contesto iniziale

Sono stato inserito all'interno dell'azienda per lavorare su una versione iniziale già esistente ma incompleta e, secondo me, abbastanza incasinata a livello di codice scritto, perciò si è deciso di iniziare il tutto da capo utilizzando quasi tutti gli stessi componenti.

3. Obiettivi del lavoro

L'attività principale del tirocinio è stata la realizzazione di un sistema di controllo accessi basato su Arduino o MCU analogo, da applicare presso varchi. I requisiti principali sono stati:

- diverse modalità di riconoscimento: tessere RFID, tastierino numerico, telecomando;
- a seguito di ogni accesso avvenuto con successo, deve aprire la serratura di una porta;
- a seguito di ogni accesso avvenuto con successo, deve salvare il log su scheda SD o su sistema remoto (LAN o Cloud) connesso via WiFi.

4. Descrizione del lavoro svolto

Il frutto di questo progetto è stato il prototipo di un sistema di controllo accessi IoT [1, 2, 3] per l'azienda ospitante del tirocinio. Durante lo sviluppo del progetto, l'idea è evoluta: oltre ai requisiti iniziali, è stato sviluppato anche un server/database ospitato nella LAN di una rete privata accessibile solo dall'amministratore. Il database è stato creato per memorizzare i log, i soci e le tessere dell'azienda. Inoltre è stata sviluppata un'interfaccia grafica (in questo caso un sito web in locale) per gestire il database tramite semplici operazioni CRUD.

5. Tecnologie coinvolte

Sistema embedded:

- componenti: Arduino UNO[4], Raspberry Pi, Ricevitore RF, Keypad, RTC, Servomotore, Lettore RFID, ESP8266[5];
- linguaggi di programmazione: Arduino Programming Language.

Sito web:

- design patterns: MVC, REST;
- tecnologie: Node.js e alcune sue librerie, MongoDB[6, 7], jQuery, W3CSS, AJAX;
- linguaggi di programmazione: HTML5, CSS, JavaScript (client-side e server-side).

6. Competenze e risultati raggiunti

Il prototipo è funzionante e abbastanza completo, ma prima di essere messo in funzione potrebbero essere necessari alcuni ritocchi e miglioramenti per renderlo più moderno secondo gli standard dei sistemi commerciali attualmente in uso.

L'esperienza di tirocinio è stata, secondo me, un successo e mi ha insegnato come lavorare in autonomia ma allo stesso tempo tenendo conto anche del teamworking (discussione degli specifici e delle varie tecnologie da usare, creazione del modello 3D del prototipo assieme alla collega).

Problemi affrontati:

- incompatibilità tra Raspbian (sistema operativo su 32 bit) e le ultime versioni di MongoDB → installazione del sistema operativo openSUSE (64 bit) sulla Raspberry Pi
- problema di sicurezza all'interno della rete locale dovuta alla mancanza di un sistema di login → rete LAN privata accessibile solo dall'amministratore
- disattivazione automatica delle tessere dei soci non più iscritti → implementazione di uno scheduler
- problema dei timestamp nei log dovuto al fatto che la Raspberry Pi senza connessione Internet non è in grado di tenere traccia del tempo → RTC

Riferimenti bibliografici

- [1] M. Rouse, Internet of Things (IoT), ultimo aggiornamento Marzo 2019. <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>
- [2] A. Tumino, Internet of Things: gli oggetti intelligenti prima di ogni “cosa”, 24 Gennaio, 2018. https://blog.osservatori.net/it_it/internet-of-things-oggetti-intelligenti-prima-di-ogni-cosa
- [3] J. Allen, Opening new doors with IP access control, 16 Marzo, 2018. <https://www.axis.com/blog/secure-insights/opening-new-doors-with-ip-access-control/>

- [4] A. Carraturo, A. Trentini, Sistemi Embedded: Teoria e Pratica, prima edizione: Settembre 2017. <http://www.ledizioni.it/prodotto/a-carraturo-a-trentini-sistemi-embedded-teoria-pratica/>
- [5] Ai-Thinker Team, ESP-01 WiFi Module (Version1.0). <http://www.microchip.ua/wireless/esp01.pdf>
- [6] MongoDB, MongoDB Node.js Driver Documentation v3.2 (<https://mongodb.github.io/node-mongodb-native/3.2/>)
- [7] Mongoose, elegant mongodb object modeling for node.js. <https://mongoosejs.com/>