**Candidato:**  Filippo Zinetti

**Azienda:** Scuola Arti e Mestieri Trevano

**Periodo:** 03.05.2021 – 27.05.2021

**Presentazione:** 07.06.2021, 10:30

## Situazione iniziale

Per migliorare le procedure di evacuazione e la comunicazione con il personale in caso di danni strutturali come gli incendi, è stato precedentemente sviluppato il sistema di notifiche Butler. Si tratta di un software composto da una parte server e una client: quella server dispone di un centro di controllo web dal quale un addetto della sicurezza può creare e inviare ai clients delle notifiche come messaggi pop-up. Lato client, invece, ci sono dei piccoli programmi che lavorano in background su molteplici PC e sono definiti "Butlers" (maggiordomo) poiché servono le notifiche agli utenti. Al fine di migliorare il monitoraggio dei computers, i due programmi sono ora estesi con nuove funzionalità. Sono quindi sviluppate un’estensione per eseguire un inventario delle componenti dei PC che usano l’agent Butler e una per l’analisi delle connessioni che questi hanno verso l’esterno. L’azione di analisi è eseguita da ogni client su sé stesso e i risultati sono poi inviati al server per essere mostrati nel centro di controllo, dove l'amministratore potrà visualizzarli e decidere come agire.

## Attuazione

I due moduli sono aggiunti usando le stesse tecnologie e gli stessi approcci del sistema Butler sul quale il progetto si basa. Il linguaggio usato è Python 3.7 e l’interfaccia web del server è una SPA (Single Page Application) scritta unicamente in JavaScript e jQuery che è aggiornata con più funzionalità. Tutte le connessioni da e verso i due programmi rimangono stateless per risultare leggere sulla rete e la sicurezza è assicurata dai JWT (JSON Web Tokens). I dati sono ancora salvati in MongoDB in modo da unificare il formato (JSON) con il resto del progetto.

I nuovi moduli usano la libreria multipiattaforma psutil per ricavare tutte le informazioni necessarie sul sistema. Questi dati sono salvati sul database e identificati dall’indirizzo MAC di ogni computer. Il carico di lavoro è diviso tra server e client, dunque più dati sono presenti in punti diversi e vanno mantenuti consistenti tra loro senza tralasciare la pulizia, la leggerezza computazionale e quella sulla rete.

## Risultati

Il programma risultante è fedele ai requisiti minimi richiesti. I moduli sono stati implementati correttamente e le due parti comunicano tra loro solo le informazioni strettamente necessarie, continuando a risultare efficienti. La libreria psutil si è dimostrata pratica e ha permesso ai programmi di rimanere completamente multipiattaforma. La logica precedentemente implementata è risultata adatta anche per l’aggiunta di nuovi componenti. La memorizzazione dei dati ha causato alcuni problemi a causa della difficoltà nella gestione delle connessioni simili e duplicate. Il centro di controllo ha visto alcune aggiunte per la visualizzazione e la modifica delle informazioni, ma manca una parte di gestione del modello standard delle connessioni, che rimane presente e modificabile solo dal database. Si tratta dell’unica richiesta non completamente soddisfatta. Alcune scelte d’implementazione andrebbero adattate ai cambiamenti apportati, ma non è stato possibile farlo a causa del tempo limitato. I programmi funzionano comunque senza alcun problema.