

Rilevatore PM10

Nutria LUG - nutria.lug@gmail.com

13 marzo 2017

Sommario

Il tema dell'inquinamento e dell'aria *respirabile*, è ogni giorno sempre più sentito. Diventa quindi interessante, se non addirittura necessario, per il cittadino rendersi conto di quanto sia qualitativamente buona l'aria che respira tutti i giorni. Rilevare questo tipo di informazioni richiede dispositivi adeguati che spesso risultano essere costosi e quindi non alla portata di tutti.

Con questo progetto si vuole mostrare come poter dare a ciascun cittadino questa possibilità a costi contenuti attraverso strumenti open source.

Introduzione

Con il presente documento si intende illustrare il progetto per la realizzazione di alcuni rilevatori di PM10, da posizionare all'interno della città di Pavia, al fine di monitorarne la quantità e di ricavarne dati statistici fruibili ai cittadini tramite rete internet.

L'obiettivo che si vuole raggiungere è quello di potenziare il sistema di rilevamento dell'inquinamento, introducendo quante più postazioni nella città di Pavia e zone limitrofe. Per raggiungerlo saranno costruiti dispositivi *low cost* completamente funzionanti, i quali saranno poi posizionati presso abitazioni private di cittadini. Questi saranno volontari che offriranno lo spazio per il collocamento dei rilevatori e metteranno a disposizione il proprio collegamento alla rete internet per la trasmissione dei dati.

Raccolta Dati

I dati saranno raccolti in via del tutto anonima con una frequenza di una rilevazione ogni 15 minuti, per un totale di 96 rilevazioni al giorno. I dati raccolti sono poi trasmessi a un elaboratore centrale che, una volta configurato e adeguatamente programmato, metterà a disposizione di tutti i cittadini le informazioni raccolte in forma statistica.

Sarà previsto un accesso *speciale* ai cittadini che si offriranno volontari per il posizionamento delle centraline, con il quale potranno accedere al proprio dispositivo per consultare dal proprio PC o SmartPhone i dati raccolti.

Struttura del Progetto

Il progetto comprende tre parti principali:

- Costruzione del dispositivo elettronico.
- Sviluppo e installazione del pacchetto software per il funzionamento del dispositivo e la raccolta dei dati.
- Sviluppo dell'interfaccia web per la visualizzazione dei dati raccolti.

Le prossime sezioni sono dedicate alla spiegazione dettagliata di come si intende realizzare ogni elemento, poi verrà illustrato il funzionamento completo dell'intero sistema e infine, l'ultima sezione sarà dedicata alla redazione del budget richiesto per fare una stima dei costi.

1 Rilevatore

Il dispositivo elettronico utilizzato per le rilevazioni è assemblato a partire da un micro computer che farà da microcontrollore. Esistono in commercio diverse schede di dimensioni ridotte e capaci di eseguire un intero sistema operativo Linux. Con questo è possibile interfacciarsi ai sensori per effettuare le rilevazioni, decodificare i segnali raccolti e mandarli tramite rete internet a un server centrale.

1.1 Microcontrollore

Il micro pc prescelto è una scheda Arietta G25-256 prodotta e distribuita dall'azienda italiana ACME System: <http://www.acmesystems.it>. Sul sito

del produttore è possibile consultare gli schemi della scheda, poichè rilasciati in maniera open source.

Le specifiche della scheda Arietta G25-256 sono riportate di seguito:

- Processore ARM9 CPU Atmel AT91SAM9G25 da 400Mh
- 256Mb di memoria RAM DDR2
- Fino a 3 porte USB 2.0
- 2 file di 20 pin per le connessioni hardware

La scheda base verrà dotata di modulo WiFi per la connessione a internet e scheda microSD per il sistema operativo e il salvataggio dei dati. Entrambi i moduli sono forniti dalla ACME System.

1.2 Sensori

La scheda sopra descritta verrà collegata a un sensore per la rilevazione di PM10. Tra i disponibili in commercio è stato scelto il sensore: Grove Dust Sensor prodotto e distribuito da <https://www.seeedstudio.com/>

L'azienda produce altri tipi di sensori per il rilevamento di gas atmosferici, quindi si progetterà il dispositivo in maniera tale da poter essere estendibile.

1.3 Contenitore

Tutta la parte elettronica dovrà essere contenuta in un apposito involucro adatto per essere posizionato all'esterno. In commercio esistono scatole di giunzione fatte per riparare componenti elettroniche dalle condizioni climatiche. Si è scelto di usare una di queste per contenere il dispositivo. La scatola dovrà essere fatta di materiale impermeabile e avere dimensioni opportune per contenere sia la scheda che il sensore: circa 200x120x55 mm.

2 Software

Il software necessario per il funzionamento del sistema verrà installato e rilasciato in maniera OpenSource. Questo significa che tutto il codice sorgente sarà disponibile per essere consultato, per eventuali controlli di correttezza, ed esteso da chiunque ne abbia le competenze.

Sono previste tre componenti principali:

- *Dispositivo*: Si occupa di fare le rilevazioni, di salvarle e di mandarle a un server principale.

- *Client Web*: Si tratta dell'applicazione web usata per consultare i dati raccolti tramite il proprio computer, attraverso il browser.

- *Server Web*: È la componente che si occupa di ricevere i dati dai sensori e di salvarli. Inoltre espone l'applicazione web per la fruizione di quanto raccolto.

Nelle prossime sezioni è descritto il funzionamento delle singole componenti del progetto.

2.1 Sul Dispositivo

Il dispositivo dovrà occuparsi di raccogliere i dati, salvarli e di mandarli a un server centrale tramite rete internet. Per poter fare queste operazioni non verrà utilizzato un unico programma, ma il problema sarà scomposto, in maniera tale da individuare una moltitudine di applicazioni più semplici che *collaborino* tra di loro. Alcune di queste esistono già e verranno opportunamente installate e configurate. Altre invece dovranno essere appositamente scritte da un team di programmatori.

Sistema Operativo

Per fare comunicare tra di loro i programmi del dispositivo, è necessario un sistema operativo. Per questo scopo è stato scelto Linux nella sua versione di Debian: famoso per la stabilità e la sicurezza. Inoltre il produttore della scheda Arietta mette a disposizione delle schede microSD con preinstallato questo sistema operativo.

Lettura dal Sensore

La lettura dal sensore è la parte principale dell'intero progetto. Questa dovrà essere quanto più accurata possibile e indipendente dalla precisione del sensore stesso.

Per fare ciò si provvederà a scrivere due programmi apposta: il primo è un driver che si interfacerà ai PIN della scheda per ricevere il dato del sensore, il secondo invece si occuperà di tarare il driver. Quest'ultimo verrà eseguito in fase di installazione e dovrà *scegliere* il giusto fattore moltiplicativo usato per perturbare i dati ottenuti dal sensore. Per fare questa scelta si posizionerà il dispositivo nella stessa zona dove sono attualmente presenti i sensori di PM10 dell'ANSA e si confronteranno i risultati ottenuti dal *Dust Sensor* con quelli pubblicati ufficialmente.

Le rilevazioni saranno effettuate a intervalli di tempo prestabiliti (quantificati in misura di ogni 15 minuti). Per gestire l'esecuzione intervallata si

utilizzerà un programma disponibile open source: *cron*.

Gestione Dati

I dati ottenuti dalla rilevazione dovranno essere sia salvati localmente sul dispositivo che mandati a un server centrale attraverso la rete internet. Il salvataggio in locale si rende necessario per due motivi:

- Fornire la possibilità ai volontari di visualizzarli
- Far fronte a eventuali cali di rete senza perdere informazioni. Se la rete non è disponibile nel momento in cui viene fatta la rilevazione, il dato non potrà essere mandato all'elaboratore centrale, quindi si rischia di perdere il dato. Salvando in locale si evita questo inconveniente.

Sia per l'invio dei dati che per il salvataggio in locale si svilupperà la parte applicativa apposta.

Per la comunicazione dei dati si utilizzerà la libreria *libcurl*, la quale permette di scrivere programmi in grado di spedire messaggi via web tramite il protocollo HTTP e HTTPS. I messaggi possono essere mandati sia in modalità GET che in POST. Questo agevola l'implementazione di un sistema basato su tecnologia *WebService*¹.

Il salvataggio dei dati deve poter avvenire su *DBMS*² in maniera tale da essere interrogato dal sistema di visualizzazione locale. Allo stesso tempo però bisogna tenere conto delle prestazioni limitate del micro PC utilizzato. Quindi la scelta che è stata fatta è quella di utilizzare il sistema *SQLite*: caratterizzato da semplicità d'uso e leggerezza in termini di requisiti hardware.

2.2 Server Web

Il server web è la componente principale dell'intero sistema. Si occupa infatti delle operazioni fondamentali messe a disposizione: ricevere le rilevazioni dei dispositivi per salvarle sul data base e mettere a disposizione un'interfaccia con la quale interrogare il data base per leggerne i dati.

Per realizzare queste due macrofunzionalità si è scelto di usare web service REST, scritti con il framework Django³. Questo permette di realizzare interfacce web interrogabili attraverso i protocolli HTTP e HTTPS in modalità GET o POST.

Le funzionalità messe a disposizione dal servizio web sono:

- Registrare una rilevazione. Per ogni rilevazione sarà tenuta traccia di un identificativo del dispositivo mittente, il valore e un riferimento temporale (giorno e ora).
- Richiedere il valore medio di PM10 per una dato intervallo temporale.
- Richiedere valore minimo e massimo rilevato con i relativi riferimenti temporali di registrazione.
- Richiedere una *tabella* di tutte le rilevazioni effettuate mostrando: data, ora, codice del dispositivo e valore. Questa può essere restituita per intero o paginata (indicando dimensione e numero della pagina richiesta).

Il servizio sarà progettato in maniera tale da essere estendibile, garantendo la possibilità di aggiungere funzionalità in base a eventuali richieste da parte degli utenti. Quanto elencato, ed eventuali operazioni aggiuntive, saranno fruibili attraverso un'interfaccia grafica descritta nel paragrafo successivo.

2.3 Client Web

Il client web è l'interfaccia grafica presentata all'utente finale, in pratica: l'applicativo attraverso il quale è possibile interagire con il sistema.

L'applicazione sarà totalmente sviluppata in HTML5, CSS3 e JavaScript, in questo modo sarà fruibile da qualsiasi dispositivo (fisso o mobile) dotato di browser web.

L'accesso all'applicativo sarà pubblico, in maniera tale che chiunque potrà accedere e usufruire delle funzionalità messe a disposizione:

- Visualizzare la tabella delle rilevazioni
- Visualizzare la media giornaliera
- Visualizzare i picchi di minimo e di massimo con i relativi riferimenti temporali
- Mostrare le rilevazioni tramite grafici a barre, a torta, a linee e a bolle. Questo scegliendo l'intervallo temporale di riferimento, che potrà essere: giorno odierno, ultima settimana, ultimo mese, ultimo anno e personalizzato (fornendo una data di inizio e una di fine).

¹Servizio web descritto nella sezione successiva

²Data Base Management System: Programma per la gestione di basi di dati

³È possibile avere maggiori informazioni a riguardo consultando il sito: <http://www.django-rest-framework.org/>

2.4 Sicurezza

In questa sezione si vuole chiarire quali sono le scelte adottate in termini di sicurezza.

Data la natura del sistema, ci sono diversi aspetti da prevedere:

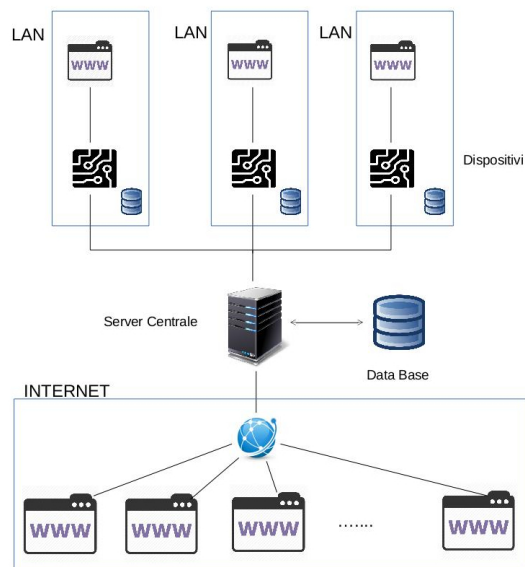
- **Dispositivo:** la scheda Arietta permette di eseguire un intero sistema operativo Linux connesso a internet, quindi è necessario proteggerne l'accesso non autorizzato.
- **Server:** La parte più critica è costituita da questa componente, infatti si occupa di immagazzinare i dati e di offrire le operazioni fondamentali per l'utilizzo del sistema da parte dell'utente. Ovviamente richiede un livello maggiore di sicurezza.
- **Autorizzazione in scrittura:** il dispositivo deve mandare al server delle richieste per la scrittura dei dati. Per evitare che chiunque possa fare altrettanto (e rischiare di superare la soglia massima di capacità del database) è necessario usare un sistema di autorizzazione robusto ed efficace.
- **Comunicazione:** dal momento in cui avviene uno scambio di messaggi tramite rete internet è necessario criptarli, allo scopo di prevenire eventuali tentativi di intrusione.

Per risolvere questi aspetti in termini di sicurezza sono previste le seguenti politiche:

- Il dispositivo userà un sistema operativo Debian Linux protetto da password robusta: almeno 8 caratteri contenenti almeno un numero, una lettera maiuscola, una minuscola e un carattere speciale. Inoltre non conterrà al suo interno alcuna parola presente in un dizionario.
- Il server sarà rilasciato su un servizio online reputato sicuro: Aruba. In questo modo la parte di sicurezza sarà delegata alla relativa azienda. Per maggiori informazioni sulle politiche di sicurezza adottate è possibile visitare la pagina: <https://datacenter.aruba.it/en/security.aspx>
- Per gestire le autorizzazioni per le richieste di scrittura sul data base, sarà utilizzato il protocollo oauth2⁴, il quale è completamente supportato dal framework scelto: <http://www.django-rest-framework.org/api-guide/authentication>
- Tutti i messaggi scambiati tra client e server useranno il protocollo HTTPS per garantirne la robustezza in termini di crittografia.

3 Sistema Completo

L'immagine mostra uno schema dell'intero sistema che si intende realizzare:



I dispositivi vengono programmati per fare una rilevazione ogni 15 minuti; il dato ottenuto viene salvato in un data base locale della scheda stessa e mandato, tramite rete internet, a un server centrale. Al termine dell'operazione, sul data base principale (quello del server) sono presenti tutti i dati raccolti di ogni dispositivo e su ogni scheda vengono salvati le singole rilevazioni.

Attraverso un'interfaccia web, esposta dal server centrale e accessibile a chiunque, è possibile consultare e analizzare le rilevazioni fatte da tutti i dispositivi. La stessa interfaccia verrà esposta dalla scheda Arietta su rete locale LAN. In questo modo, chi mette a disposizione la propria rete, può consultare le rilevazioni effettuate dal proprio dispositivo. In pratica la metodologia di navigazione dei dati risulta essere la stessa, cambia solo il dominio sul quale vengono effettuate le analisi.

4 Budget

Per realizzare il progetto presentato è stato redatto un budget, calcolato per la realizzazione di 30 dispositivi in funzione per la durata di due anni:

⁴Sito di riferimento: <https://oauth.net/2/>

Descrizione	Quantità	Prezzo Unitario	Prezzo Totale
Servizio Hosting Aruba (Pacchetto Easy): Spazio Web, Dominio, DataBase MySQL, Supporto Python e Django	2	26,50 €	53,00 €
Groove Dust Sensor	30	20,00 €	600,00 €
Scatola di Giunzione 200x120x55mm	30	15,00 €	450,00 €
Weller, Saldatore - SP25	1	25,00 €	25,00 €
Stagno 100g Diametro 0,56mm	1	12,00 €	12,00 €
Set di Cavi e Resistenze	1	12,00 €	12,00 €
Arietta G25-256 Evaluation Kit	30	81,30 €	2439,00 €
Spedizione Arietta	1	9,00 €	9,00 €
Sviluppo Software (Ore)	70	10,00 €	700,00 €
Montaggio Dispositivi (1h 30m cad)	45	10,00 €	450,00 €
TOTALE			5000 €