

# WeatherLink una piattaforma per l'integrazione e la visualizzazione dei dati meteo

Riccardo La Grassa<sup>1</sup>, Marco Alfano<sup>1,3</sup>, Biagio Lenzitti<sup>1</sup>, Davide Taibi<sup>2</sup>

<sup>1</sup>Università degli Studi di Palermo – Dipartimento di Matematica e Informatica, Palermo, Italy  
{marco.alfano, biagio.lenzitti}@unipa.it

<sup>2</sup>Consiglio Nazionale delle Ricerche - Istituto per le Tecnologie Didattiche, Palermo, Italy  
{davide.taibi}@itd.cnr.it

<sup>3</sup>Anghelos Centro Studi sulla Comunicazione, Palermo, Italy

## ABSTRACT

WeatherLink è una piattaforma per la raccolta, l'elaborazione e la pubblicazione di dati relativi a misure atmosferiche. WeatherLink si compone di una parte server che memorizza l'enorme quantità di dati provenienti dalle diverse board dislocate nel territorio che fungono da client. Ogni board tramite i suoi sensori raccoglie dati su temperatura, pressione atmosferica, qualità dell'aria e li invia al server. Il server si occupa della memorizzazione dei dati, e offre anche delle API che possono essere interrogate per estrarre i dati ed effettuare elaborazioni successive. I dati vengono esposti come dati aperti nel formato JSON e JSON-LD. Le API sono state utilizzate per la realizzazione di una Web Application che consente il monitoraggio delle board e la visualizzazione dei dati aggregati o per singola board attraverso grafici relativi ai diversi parametri meteorologici catturati dai sensori.

## Keywords

Open Data, IoT, data integration and visualization

## 1. INTRODUZIONE

WeatherLink è una piattaforma per la raccolta, l'elaborazione e la pubblicazione di dati di natura meteorologica. L'idea di WeatherLink è quella di creare una fitta rete di stazioni meteo amatoriali e non, al fine di integrare e rendere disponibili per successive elaborazioni i dati, in formato aperto, provenienti da diverse sorgenti. La piattaforma si compone di due elementi principali: i) il server che comunica con le stazioni per la raccolta delle informazioni meteo ed espone, tramite servizi implementati con metodi REST, i dati in formato aperto; ii) le stazioni meteo, realizzate tramite Board specifiche, sono dislocate nel territorio e raccolgono i dati mediante sensori di temperatura, umidità, pressione.

Il server è stato implementato attraverso diversi moduli Python per l'acquisizione dei dati e la loro memorizzazione. Per quanto riguarda le board, il codice è stato scritto interamente in C. Il codice è stato reso pubblico in modo da consentirne il riuso<sup>1</sup>. Gli utenti che vogliono contribuire alla rete di sensori WeatherLink possono registrare sul server la propria stazione meteo e installare

sulle board il codice necessario. L'utente, inoltre, può accedere sia ai dati della propria board in locale, per via dell'aggiuntivo sviluppo del front-end lato board, che ai dati storicizzati sul server. Uno dei punti di forza di WeatherLink è quello di offrire un pieno servizio a chiunque voglia utilizzare le API, esponendo dati in formato aperto (JSON e JSON-LD) e senza restrizioni temporali. Questo approccio rende WeatherLink differente da altre piattaforme disponibili in rete che limitano il riuso dei dati o ne restringono l'accesso solo a un ristretto arco temporale. Sulla base dei dati resi disponibili dal server WeatherLink è stata costruita una Web Application appositamente progettata per la visualizzazione dei dati meteo, descritta in dettaglio nella seguente sezione.

## 2. DATA VISUALIZATION

La Web Application è stata realizzata con l'utilizzo del framework Django<sup>2</sup>, ed è basata sul paradigma MVC (Model View Control) che ha permesso la creazione di modelli di dati efficienti utilizzati per lo storage su database. Una volta creati i modelli dei dati, specifici comandi estrapolano il codice SQL necessario per eseguire le query su un database di tipo SQLite. Inoltre si è preferito di avere due database indipendenti, uno gestito da Django per la memorizzazione dei dati provenienti dalle stazioni meteo e uno alimentato da moduli python in esecuzione periodica per il recupero dei dati da fonti di dati esterni come OpenWeatherMap (OWM)<sup>3</sup>. Nello specifico i moduli scritti in python, *Get\_data* e *Store\_it*, hanno il compito di recuperare i dati da OWM e di memorizzarli in *db\_weather.sqlite*. Una volta registrati su OpenWeatherMap e ottenuto un token, si sono utilizzate le API messe a disposizione della piattaforma OWM (con licenza CC-BY-SA), al fine di creare un proprio storico da utilizzare per successive elaborazioni e da integrare con i dati raccolti dalle stazioni meteo. La finalità di WeatherLink, è quella, di creare un proprio network di weather station, integrare i dati con altre fonti di dati aperti relativi alle misurazioni meteo e fornire, dopo successiva aggregazione ed elaborazione, nuove visualizzazioni dei dati. In fase di progettazione del database, si sono considerati diversi aspetti legati alla pertinenza dei dati nel

<sup>1</sup> <https://github.com/pulsar2468/IoT-project>

<sup>2</sup> <https://www.djangoproject.com>

<sup>3</sup> <https://openweathermap.org>

contesto meteorologico, come la gestione dei diversi tipi di dati, o le problematiche su possibili errori dovuti all'invio di dati parziali. Un utente ha la facoltà di creare più stazioni meteo legate al proprio account, registrando ogni stazione meteo sul server WeatherLink e ottenendo uno specifico token. Uno dei principali

servizi offerti da WeatherLink è la possibilità di visualizzare in tempo reale i dati raccolti dalla propria stazione meteo direttamente plottata su OpenStreetMap (OSM).

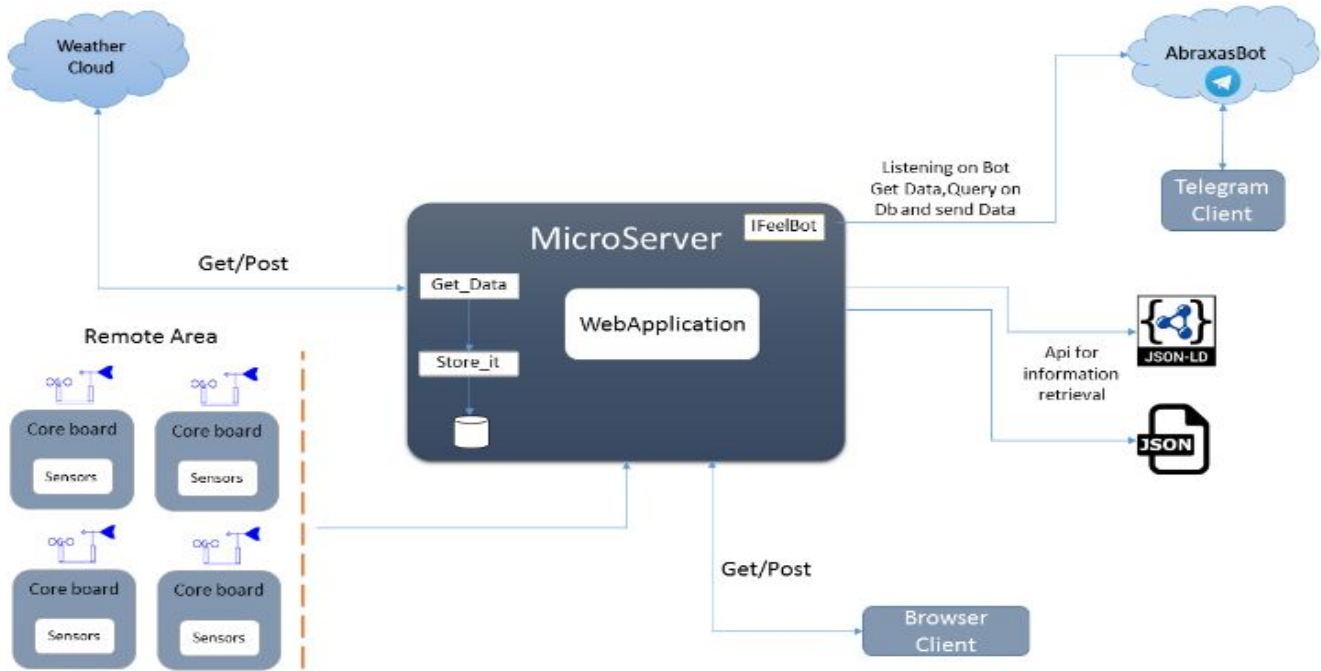
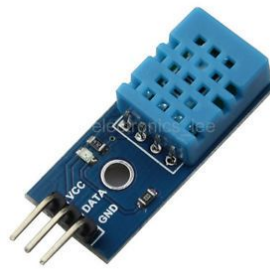


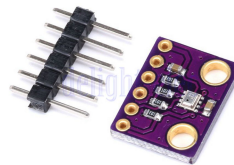
Figura 1: Architettura di WeatherLink



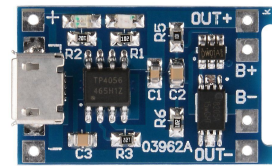
Board Wemos D1 mini Pro



Dht11 (Sensore Temperatura/Umidità)



BMP 280 (Sensore di Pressione)



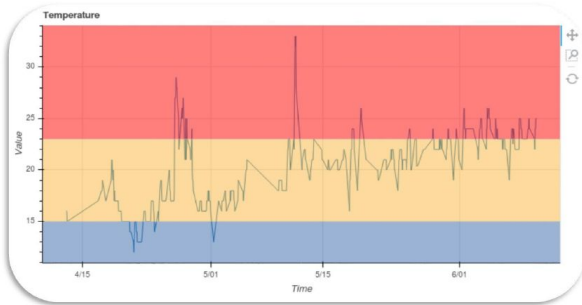
Tp4056 (Caricatore Lineare)



mini cella fotovoltaica (5v - 20/25 mAh)

Figura 2: Stazione meteo e sensori utilizzati.

La piattaforma presenta due tipologie di plotting: Mappa e Grafici. Per la visualizzazione Mappa è stata utilizzata la libreria folium (leaflet derivata per python) per plottare degli oggetti di tipo *circle* direttamente su mappa OpenStreetMap (OSM). Inizialmente, si effettua un recupero tramite query, nel database db\_weather.sqlite dei dati più recenti, ma l'utente può estendere l'arco temporale da visualizzare. La figura 3 mostra uno dei grafici visualizzati dalla Web Application di WeatherLink<sup>4</sup>.



**Figura 3: Andamento della temperatura.**

### 3. Realizzazione e dati tecnici

Come mostrato in Figura 1 l'architettura di WeatherLink è composta da un server centrale e numerose boards che compongono la rete di stazioni meteo. Al momento la rete è composta da board Wemos d1 mini pro con Esp8266 [1][2], sensore dht11 (Temperatura/Umidità) e Bmp 280 (Pressione). Inoltre le board sono dotate di Caricatore Lineare Tp4056, Mini pannello Solare (5v - 20/25 mAh) e Batteria al litio. I2C/OneWire/SPI sono i tre protocolli utilizzati per la comunicazione con i sensori. La board ha solo un canale analogico e 11 digitali, Flash 16 MB, 80KB di ram e 80/160 MHz di clocks.

Il flusso di esecuzione implementato su ciascuna board è il seguente:

- Setup del web Server e connessione alla rete.
- Inizializzazione dei vari sensori e della porta d'ascolto
- Acquisizione dati ogni 5 secondi e invio al server. (Funzionalità one shoot)

Inoltre in locale gli utenti possono accedere ai dati raccolti tramite normale protocollo HTTP (attraverso la funzionalità di web server locale). In base alla richiesta fatta dagli utenti (tramite HTTP get), la board risponde con l'informazione desiderata per ciascun sensore.

Recentemente, una nuova versione è in fase di testing per ridurre drasticamente i consumi energetici da parte della board, introducendo un pannello fotovoltaico, un caricatore lineare con modalità trickle, ed una batteria al litio. Il software è stato modificato, eliminando la funzionalità di web server locale, e decrementando il consumo di corrente ponendo la board in

<sup>4</sup> Un video dimostrativo della piattaforma WeatherLink è disponibile qui: <https://www.youtube.com/watch?v=pKFKAI9XQC4>

modalità deep-sleep ogni qualvolta si effettui un invio dei dati (one shoot). Il risveglio dalla modalità, viene effettuato per mezzo di un interrupt causato da un timer rtc interno. La board è stata pensata per operare sia in rete che in zone sprovviste di copertura wireless. In quest'ultimo caso la board implementa un file system e conserva i dati dei sensori in una microSD.

### 4. Conclusioni e sviluppi futuri

La piattaforma WeatherLink è stata progettata con l'obiettivo di integrare dati provenienti da stazioni meteo realizzate attraverso apparecchiature IoT, con altre sorgenti di dati pubblicati in formato aperto, in modo da creare una fonte completa di dati meteo su cui effettuare nuove elaborazioni e visualizzazioni. Al momento si è in procinto di costruire una rete di stazioni di rilevamento sparse per la città di Palermo che conterranno, oltre al sensore per il rilevamento della temperatura e umidità, un rilevatore del livello di qualità dell'aria in modo che si riesca a fornire una sorta di indice di "qualità ambientale" delle varie zone della città. Inoltre sono in via sviluppo dei connettori per i dati, disponibili in formato aperto, messi a disposizione da ARPA Sicilia<sup>5</sup>. In questo modo i dati raccolti dalle stazioni meteo verranno integrati non solo con i dati di OWM ma anche altre fonti di dati aperti spianando la strada ad ulteriori elaborazioni [3][4].

### 5. Riferimenti

- [1] Wemos technical Specification, disponibile in rete: [https://wiki.wemos.cc/\\_media/products:d1:mini\\_pro\\_v1.0.0.pdf](https://wiki.wemos.cc/_media/products:d1:mini_pro_v1.0.0.pdf)
- [2] Datasheet per ESP8266, disponibile in rete: [http://espressif.com/sites/default/files/documentation/0a-esp8266e\\_x\\_datasheet\\_en.pdf](http://espressif.com/sites/default/files/documentation/0a-esp8266e_x_datasheet_en.pdf)
- [3] M. Bannayan, G.Hoogenboom, (2008). Weather analogue: A tool for real-time prediction of daily weather data realizations based on a modified k-nearest neighbor approach, In Environmental Modelling & Software, 23, (6), 703-713, ISSN 1364-8152, <https://doi.org/10.1016/j.envsoft.2007.09.011>.
- [4] M. Holmstrom, D. Liu, and C. Vo. (2016).Machine Learning Applied to Weather Forecasting. Stanford University.

<sup>5</sup> <http://www.arpa.sicilia.it/temi-ambientali/monitoraggi-ambientali>