

PROGETTO TOO(L)SMART

OUTPUT AZIONE 2 - O.2.g

Codice Output	O.2.g
Denominazione	Elenco degli strumenti a supporto delle attività di formazione degli utenti dell'ente riusante: per auto-analisi delle competenze, Piano formativo per ambiti e per livelli, Format e contenuti attività di capacity building
Unità di Misura	Numero
Valore Target	1
Enti coinvolti	Ente Responsabile: Comune di Torino Enti Partecipanti: tutti
Descrittivo:	
<p>L'output fa parte del pacchetto di strumenti del kit di riuso, volto a facilitare la diffusione della BP abilitando l'attivazione di un completo e autonomo trasferimento di soluzioni tra Amministrazioni e supportando le differenti fasi che compongono tali processi, e in particolare:</p> <ol style="list-style-type: none"> Fase "Ricerca e Selezione della buona pratica": // Fase "Trasferimento e adozione della BP": <i>Componente formativa interna al partenariato:</i> Per ciò che riguarda i rapporti all'interno del partenariato di progetto, gli enti cedenti hanno affiancato gli enti riusanti nell'adattare la BP ai loro contesti locali. Per dare modo ai comuni riusanti di scegliere la modalità di installazione del middleware più confacente è stato infatti elaborato, illustrato e condiviso un documento realizzato da UNIME che illustra le differenti possibili opzioni di installazione dei stack4things (https://docs.google.com/document/d/15o4EpxjzBulvpim6k_2OKF3Ax9K_-skJs1vk_xw7WK4/edit) In secondo luogo, anche il comune capofila ha svolto un'attività di coordinamento mirata sul supporto ai partner nell'illustrazione delle differenti opzioni di installazione al fine di individuare quella più idonea nel rispettivo comune. Effettuata la scelta installativa, UNIME insieme a smartme.io ha fornito supporto completo al riuso del middleware in tutte le Città partner, in modalità "training on the job" con collegamenti a distanza. Per supportare le attività di installazione dei sensori, grazie all'affidamento a CSP sono state fornite guide e strumenti formativi/informativi utili riguardo: <ul style="list-style-type: none"> - posizionamento - modalità di installazione - manutenzione di base - informativa per i referenti di plesso - monografia del sensore - informativa per occupanti. Infine, per fornire un supporto specifico e mirato nella verifica di funzionamento dei sensori in 	

campo ai partner è stato attivato un servizio di ticketing attraverso l'istituzione di un indirizzo email dedicato (toolsmart-support@smartme.io) a cui i partner riusanti potevano rivolgersi per qualsiasi questione tecnica e tecnologica relativa sia all'installazione che al funzionamento della stazione ambientale, ottenendo risposte celeri e dedicate da smart.me e UNIME.

Componente formativa esterna - rivolta ad altre PA interessate al riuso:

Per ciò che attiene la formazione volta a favorire il trasferimento e l'adozione della BP al di là del partenariato, si è provveduto a organizzare una serie di webinar dedicati a tutte le PA interessate a replicare la soluzione sul proprio territorio.

- 1) Il primo webinar, tenutosi in data 2 marzo 2020 e intitolato *"DA SMARTME A TOOLSMART: IL RIUSO DELLA BUONA PRATICA"*, ha inteso presentare l'infrastruttura integrata per il monitoraggio ambientale diffuso realizzata nell'ambito del progetto grazie al paradigma dell'Internet of Things, per creare un sistema di raccolta e utilizzo di dati che rafforzi la capacità degli enti locali di rispondere alle sempre più complesse criticità insite nella dimensione urbana e che, al contempo, generi opportunità di innovazione e sviluppo economico. Sono stati analizzati gli aspetti tecnici (sia hardware che software) dell'infrastruttura, basata su cloud computing e open software e hardware per l'implementazione di reti di sensori e servizi digitali, e sono poi stati raccontati il caso d'uso della Città di Torino, esaminando tutti gli elementi utili al riuso della buona pratica, siano essi tecnici, formativi, amministrativi, organizzativi.

Relatori: Comune di Torino (Elena Deambrogio) *"Introduzione al webinar e Inquadramento di Toolsmart: progetto, obiettivi, BP, il concetto di riuso, il monitoraggio partecipativo, partner"*; UNIME *"La Buona Pratica: illustrazione tecnica della BP e della piattaforma (requisiti, componenti, utilizzi, installazione)"*; UNIME *"Il riuso della BP: l'esperienza presso gli altri territori partner (macchine virtuali)"*; Comune di Torino *"Il riuso della BP: l'esperienza di Torino"*; Smartme.io *"Presentazione caso di studio su stazione monitoraggio ambientale"*; CSP *"L'installazione delle centraline e l'esperienza formativa con le scuole"*; UNIME *"Possibili estensioni (nuovi servizi che è possibile attivare)"*.

Il primo webinar ha visto la partecipazione di 20 rappresentanti di 11 enti.

- 2) Il secondo webinar, tenutosi in due date diverse, 15 e 21 aprile 2020, verteva sul tema del "monitoraggio ambientale".

La prima parte, a cura di CSP, aveva ad oggetto il "monitoraggio ambientale partecipativo" e ha toccato i temi dell'IoT, dell'inquinamento ambientale, dell'inquinamento atmosferico e della conseguente importanza del monitoraggio ambientale partecipativo, elemento chiave delle sperimentazioni di Too(I)smart (cfr. O.3.f).

La seconda sessione, a cura della Direzione Ambiente della Città di Torino, ha riguardato invece le Politiche ambientali della Città di Torino e il ruolo del monitoraggio ambientale, tema su cui si innesta Too(I)smart (cfr. O.3.f).

- 3) E' stato inoltre messo a disposizione materiale informativo relativo alle attività immateriali di supporto riguardo a i temi dell'open lab e del crowdfunding.
- 4) Ulteriori webinar sono stati, infine, registrati per raccontare le esperienze locali presso i vari Enti partner, in sostituzione di quelli che sarebbero dovuti essere gli "eventi locali" inizialmente previsti da progetto e poi annullati a causa dell'emergenza sanitaria COVID19, che ha reso impossibile realizzare eventi aperti al pubblico. Tali webinar hanno riguardato le

esperienze di Padova, Siracusa e Lecce. Anche Torino ha registrato un video descrittivo di quanto svolto dall'Ente, video registrato durante l'evento locale organizzato dalla Città, l'unico che si è potuto svolgere in presenza in quanto tenutosi a novembre 2019 in periodo ante-emergenza sanitaria. Per gli approfondimenti sugli eventi locali, si rimanda all'output O.5.d.

I webinar, volti a diffondere conoscenza sul progetto TOO(L)SMART e aperti alla partecipazione di tutte le PA interessate ad approfondire la buona pratica al fine di stimolarne e facilitarne il riuso, sono stati realizzati tramite la piattaforma GoToMeeting, secondo la modalità "a distanza".

Per assicurare un'ampia diffusione e formazione sulla BP, tutti gli interventi dei vari relatori e le loro presentazioni sono state registrate e messe a disposizione anche sul canale Youtube di progetto <https://www.youtube.com/playlist?list=PL0f9hTMMy2Yp4Pms-w0yLQuHmUZbOz3eNI> in modalità "offline", in modo da poter essere veicolate non solo a tutti gli Enti iscritti e partecipanti all'evento (qualora l'evento prevedesse la partecipazione di un pubblico virtuale "in diretta", come nel caso del primo webinar) ma anche diffusi ulteriormente, a evento concluso e registrato.

I webinar nel loro insieme costituiscono il Piano di Accompagnamento al Riuso, elaborato per condividere l'esperienza e la conoscenza maturata nei mesi di progetto sui quattro ambiti succitati. Il Piano è completato dal materiale documentale messo a disposizione sul sito (<https://www.torinocitylab.it/it/toolsmart>) e rappresenta uno strumento di approfondimento essenziale per l'attività di formazione rivolta alle PA, nell'ottica di diffondere conoscenza sul progetto e sulla buona pratica che ne è alla base al fine di favorirne la replicabilità.

La diffusione del Piano di Accompagnamento al Riuso (e documentazione collegata) per raggiungere i target individuati è avvenuta tramite la pubblicazione di una news dedicata sul sito (<https://www.torinocitylab.it/it/news/444-toolsmart-piano-di-accompagnamento-al-riuso>), rimbalzata poi attraverso i canali di comunicazione del progetto, quali gli account social di Facebook, Twitter e LinkedIn dell'Ente capofila e dei partner e, più miratamente, attraverso le mailing list dedicate di ANCI, PON METRO e CINI nonché le mailing list di alcuni partner di progetto (cfr. O.5.e).

Analisi dei fabbisogni formativi dei dipendenti pubblici delle PA coinvolte

Tra le attività condotte in tema di formazione e capacity building, vale la pena citare il confronto e l'approfondimento, svolto in stretta sinergia con UNIME, del fabbisogno formativo di contenuti e percorsi formativi per le pubbliche amministrazioni per eventuali corsi futuri.

I temi individuati sono i seguenti:

- sistemi a microcontrollore: studio dei principali sistemi a microcontrollore. Introduzione all'hardware. Schede di prototipizzazione (Arduino e similari). Principi di programmazione. Ambienti di sviluppo;
- Cyber Physical Systems: concetto di Cyber Physical System. Sistemi analogici e digitali. Sensori (temperatura, umidità, pressione, accelerometri, qualità dell'aria.). Attuatori (Led, Buzzer, Motori). Il concetto di GPIO. Interfacciamento con sensori e attuatori;
- programmazione Flow-based: il concetto di algoritmo. I diagrammi di flusso. L'ambiente Node-Red per la programmazione visuale di Cyber Physical Systems;
- protocolli di comunicazione: lo scambio di dati tra dispositivi. Sistemi wired e wireless. Il concetto di protocollo. Protocolli intra-device e inter-device. I protocolli seriali (UART, SPI, I2C). Il protocollo Wi-Fi. Il protocollo Bluetooth. Protocolli per le Smart City (LoRa);

- sistemi Cloud: concetti di base. Virtualizzazione. Il sistema OpenStack. Gestione di un sistema OpenStack. Attivazione dei principali servizi.
- gestione di flotte di dispositivi IoT: integrazione tra Cloud e IoT. Il concetto di Front-end e Back-end. Gestione dei dati da remoto. Gestione dei dispositivi da remoto. Il concetto di plug-in. Le principali soluzioni Open Source (Stack4Things con il relativo servizio OpenStack IoTonic);
- Data repository e Big Data: il concetto di Big Data. I database relazionali. I database non relazionali. Analisi di flussi di dati. Il concetto di Open Data. Il CMS CKAN. Sistemi di data analytics, data transformations e data visualization (Grafana);
- Applicazioni per Smart City: progettazione di un'applicazione. La fase di testing. Il deployment. Applicazione sul monitoraggio ambientale.
- sicurezza in ambito Smart City: il concetto di privacy. La cybersecurity. Normative di riferimento. Principali meccanismi di sicurezza. Le blockchain.

Attività formative/informative rivolte alla Comunità scolastica per i nuovi casi d'uso #Smart Citizenship e #Smart Skills

Al di là delle attività formative rivolte alla PA, si segnala, infine, l'attività formativa svolta nei confronti della comunità scolastica.

A Torino, l'installazione dei sensori nei plessi scolastici coinvolti è stata accompagnata, infatti, da un primo momento di coinvolgimento con il Comitato Scuole a Mirafiori Sud, durante il quale sono stati presentati i progetti Too(l)smart e Progireg e scelti i punti di installazione (cfr O.3.f).

#Smart Skills - Formazione finalizzata alla creazione di competenze professionalizzanti e all'orientamento scolastico e sul lavoro (target: Istituti tecnici superiori; Licei)

In data 7 novembre 2019 è stato avviato un percorso formativo professionalizzante presso l'Istituto Levi, organizzato in più tappe:

- 1) Lezione introduttiva, tenuta da CSP sui temi dei cambiamenti climatici e sull'uso strategico dei dati grazie all'IoT e ai Big Data per studiare, prevenire e mitigarne gli effetti.
- 2) Attività pratica di assemblaggio, da parte dei ragazzi, dei kit di autocostruzione del sensore e formazione multidisciplinare da parte dei docenti dell'Istituto.

L'Istituto Tecnico Levi è stato infatti dotato di n. 6 kit di autocostruzione di stazioni di monitoraggio a fini didattici: all'Istituto, infatti, sono stati dati in comodato d'uso gratuito 6 starter kit, ovvero stazioni meteo da montare a fini di formazione tecnica.

I ragazzi delle classi individuate (2 classi di IV superiore), seguiti dai loro professori, hanno utilizzato tali "Kit di autocostruzione di sensori" per sviluppare un percorso formativo dedicato.

L'esperienza formativa messa in campo, che ha portato alla costruzione da parte dei ragazzi di centraline di monitoraggio proprio tramite i kit forniti, potrà trasformarsi, in futuro, in un processo di "peer education" (inserita in un'attività di alternanza scuola lavoro) offerto ad altri ordini di scuole in materia di "tecnologie per l'ambiente".

- 3) Un ulteriore evento di inquadramento del progetto Too(l)smart, dei suoi obiettivi, del concetto di monitoraggio ambientale partecipato e degli strumenti utilizzati (centraline, kit di autocostruzione) è stato organizzato il 29 novembre 2019 presso l'Istituto Levi. L'evento, dedicato agli Studenti dell'Istituto e alle classi coinvolte ma aperto anche ad altre scuole del quartiere, ha visto la partecipazione, come relatori, dei coordinatori di progetto della Città di Torino, di CSP, di INRIM.

#Smart Citizenship - Strumenti formativi/informativi per abilitare il coinvolgimento attivo della comunità scolastica nel processo di monitoraggio ambientale partecipativo

Sempre nei confronti della Comunità scolastica, CSP – che ha installato le centraline presso i plessi scolastici torinesi coinvolti – ha messo a disposizione un documento di istruzioni per la manutenzione delle centraline medesime, qualora la stessa si rendesse necessaria.

Nel caso si notassero dei malfunzionamenti di una centralina (mancanza di dati o arrivo di dati «fuori range atteso»), infatti, il personale in forze nel sito ospitante la centralina stessa (referenti tecnici, docenti o altre figure incaricate e formate) potrebbe essere chiamato a svolgere semplici azioni quali controllo dell'alimentazione, controllo visivo degli elementi della centralina, (verificare se non vi siano parti mancanti o fuori posto), collaborare a eseguire semplici test, quali ad esempio far ruotare la banderuola o il segnamento, versare dell'acqua del pluviometro.

Sulla base dell'esperienza Torinese, i Comuni di Siracusa e Messina hanno modellizzato le attività formative in fase di progetto e hanno ricevuto gli starter kit per gli Istituti coinvolti. Tuttavia, le attività in presenza a scuola non sono state svolte per via dell'emergenza sanitaria in corso. Le attività sono state programmate per l'a.s. 2020-2021 negli istituti tecnici superiori selezionati (es. I.T. Fermi di Siracusa).

3) Fase Gestione a regime della BP:

Quanto scritto nella precedente sezione è valido anche per la fase di gestione a regime della BP, per cui si rimanda a quanto scritto sopra.

Allegati:

Formazione intra-partenariale:

All. 1 – Dettagli tecnici Stack4Things

All. 2 – About Stack4Things

Formazione extra-partenariale – PA:

Piano di Accompagnamento al Riuso:

cfr. video <https://www.youtube.com/playlist?list=PL0f9hTMy2Yp4Pms-w0yLQuHmUZbOz3eNI>

e cfr. documentazione ppt correlata messa a disposizione su <https://www.torinocitylab.it/it/toolsmart>

Installazione dashboard, centraline e monografie:

Cfr. Allegati 5 (a, b, c) e 6 (a, b, c, d, e, f, g) dell'O.3.f

Possibili modalità di installazione di Stack4Things

Stack4Things è disponibile in due versioni: la versione OpenStack che si basa sul famoso framework Cloud e la versione standalone che è invece indipendente da quest'ultimo. Consigliamo di installare la versione standalone se il numero di nodi da gestire non supera il migliaio. Questo per facilitare tutto il processo di installazione e gestione del framework. Se invece si pianifica di utilizzare Stack4Things per un numero maggiore di nodi si consiglia di installare la versione OpenStack. L'installazione di quest'ultima consente ovviamente anche di conoscere e acquisire esperienza con questo importante framework.

1 - Installazione autonoma

Ogni partner del progetto potrà decidere di installare autonomamente la propria istanza di Stack4Things. In questo caso è necessaria un'ottima conoscenza del framework OpenStack (nel caso di versione basata su OpenStack) e buone competenze di amministrazione di sistemi Linux (qualunque sia la versione). Sono disponibili delle guide dettagliate sui repository Github del progetto sia per l'installazione che per la configurazione e personalizzazione. Ogni partner può scegliere di installare la propria istanza di Stack4Things nel proprio data center (su una o più macchine fisiche o virtuali) oppure in hosting su un provider Cloud (affittando una o più macchine fisiche o virtuali). Vantaggi: massima personalizzazione e controllo della propria istanza, si acquisisce esperienza nell'installazione del framework. Svantaggi: non è possibile stimare il tempo di installazione in quanto dipende dall'esperienza di chi la effettua.

Tempi: possono essere molto lunghi (almeno 2 settimane).

Costi: nessuno a meno dell'hosting su Cloud.

2 - Installazione tramite macchina virtuale

Ogni partner del progetto avrà la possibilità di scaricare le immagini del disco di alcune macchine virtuali sulle quale è già installata una istanza di Stack4Things. L'istanza andrà poi solo configurata e personalizzata. In questo caso è necessaria solo una certa competenza nella creazione di macchine virtuali a partire da immagini del disco e buone competenze di amministrazione di sistemi Linux per la parte di configurazione e personalizzazione. Sono disponibili delle guide dettagliate sui repository Github del progetto per configurazione e personalizzazione. Ogni partner può scegliere di creare la propria istanza nel proprio data center (su un certo numero di macchine virtuali) oppure in hosting su un provider Cloud (affittando una o più macchine virtuali). Sia nel primo che nel secondo caso sarà necessario accordarsi con UniMe sul formato delle immagini del disco da utilizzare. Vantaggi: sono richieste meno competenze, i tempi sono più brevi.

Svantaggi: minore competenza acquisita sul framework e minore personalizzazione della propria istanza.

Tempi: molto più brevi di una installazione manuale, certamente nell'ordine di pochi giorni.

Costi: nessuno a meno di hosting su Cloud.

3 - Installazione tramite macchina virtuale e setup da remoto

Come l'opzione 2 ma con la possibilità di usufruire di un servizio di configurazione e personalizzazione da remoto fornito da UniMe tramite il proprio spinoff SmartMe.IO.

Vantaggi: sono richieste meno competenze, i tempi sono molto più brevi. Svantaggi: minore competenza acquisita sul framework, minore personalizzazione della propria istanza.

Tempi: molto più brevi di una installazione manuale, certamente nell'ordine di pochi giorni.

Costi: 1200€ per la configurazione remota, 1000€ per eventuale supporto base annuale, nessun altro costo a meno di hosting su Cloud.

4 - Installazione tramite appliance

UniMe tramite il proprio spinoff SmartMe.IO può fornire direttamente ai partner dei server fisici da installare nei propri data center. Su tali macchine è già installata una istanza di Stack4Things che andrà ad essere personalizzata direttamente da remoto dai nostri tecnici. In questo caso non è necessaria alcuna competenza da parte dei tecnici dei comuni. E' necessario solo fornire al server alimentazione e rete (possibilmente tramite DHCP in modo da rendere il tutto ancora più automatico). Vantaggi: nessuna competenza richiesta, tempi brevissimi. Svantaggi: costo più elevato.

Tempi: solo quelli della spedizione e dell'installazione fisica del server nel data center.

Costi: 3000€ per il costo dell'appliance e della configurazione remota, 1000€ per eventuale supporto base annuale.

Dettagli tecnici sul processo di installazione di Stack4Things e sul suo funzionamento

Stack4Things è implementato seguendo i più restrittivi requisiti di sicurezza. In particolare:

- La comunicazione da un qualunque browser verso l'interfaccia Web avviene tramite protocollo HTTPS con certificato lato server. Per questo motivo è necessario ottenere da una qualunque autorità di certificazione riconosciuta dai più comuni browser un certificato per la macchina sulla quale verrà installata l'interfaccia Web.
- La comunicazione da un qualunque nodo IoT verso il servizio IoTronic avviene tramite protocollo WAMP in modalità sicura, sfruttando Web Socket Secure (WSS). Per questo motivo è necessario ottenere da una qualunque autorità di certificazione riconosciuta un certificato per la macchina sulla quale verrà installato il servizio IoTronic (e in particolare il router WAMP Crossbar.io).
- La comunicazione da un qualunque nodo IoT verso il servizio IoTronic per l'accesso a servizi attivi sul nodo IoT (funzionalità denominata Services) avviene tramite tunnel Web Socket Secure (WSS). Per questo motivo è necessario ottenere da una qualunque autorità di certificazione riconosciuta un certificato per la macchina sulla quale verrà installato il servizio IoTronic (e in particolare il componente WStun).

Ogni componente lato Cloud di Stack4Things si comporta da server rimanendo in ascolto su una determinata porta TCP. Sarà quindi necessario configurare la rete del proprio datacenter in modo da fornire alla/e macchina/e su cui verrà installato Stack4Things un/più indirizzo/i pubblico/i accertandosi che le seguenti porte non siano protette da firewall:

- IoTronic: 8443
- Interfaccia Web: 443
- Crossbar.io: 8181
- WStun: 8080

Infine, per l'apertura dei tunnel necessari a raggiungere i servizi attivi sui nodi IoT (ad esempio SSH) sarà necessario accertarsi che il range di porte 50001-50100 sia non protetto da firewall. Da notare che tutte le porte suddette sono configurabili tramite i file di configurazione di Stack4Things.

Al livello IoT, i requisiti di sicurezza prevedono che ogni nodo IoT sia identificato da un codice alfanumerico univoco e da una password che vengono comunicati dal nodo all'atto della connessione verso il router WAMP. Solo i nodi dotati di identificativo univoco registrato e di password valida sono accettati sul sistema. Tale modalità è solitamente sufficiente ad evitare attacchi di "impersonification" in cui lo stesso codice identificativo e la stessa password vengono utilizzati per accedere tramite altri nodi IoT non autorizzati. La comunicazione tra il nodo IoT e il servizio Cloud IoTronic è infatti sicura (WSS) per cui non è possibile "sniffare" le credenziali quando queste vengono trasmesse sulla rete. **L'attacco è possibile solo se è disponibile accesso fisico al dispositivo da parte di un malintenzionato che può leggere direttamente dal file system del nodo IoT le**

credenziali e può utilizzarle per connettersi con un altro dispositivo (l'accesso al nodo fisico potrebbe consentire anche manipolazioni di dati e alterazione delle funzionalità del nodo stesso). Da notare che se anche solo per un istante due nodi IoT provano a connettersi con le stesse credenziali, tale evento viene segnalato agli amministratori che possono disattivare il codice e verificare fisicamente se il nodo è stato spostato o danneggiato. Una modalità di autenticazione simile, ma che offre un livello di sicurezza maggiore, prevede l'utilizzo di certificati X.509 generati da una specifica autorità di certificazione (non necessariamente riconosciuta a livello globale) che vengono utilizzati in modo simile alla modalità precedente. In particolare, all'atto della connessione il nodo IoT invia a Iotronic sia il proprio codice identificativo che un hash del codice firmato con la propria chiave privata. Tale modalità ha le stesse vulnerabilità nel caso di accesso fisico al dispositivo da parte di un malintenzionato. Infine, un'ultima modalità di autenticazione del nodo prevede l'utilizzo di dispositivi hardware specifici di tipo Trusted Platform Module (TPM) che possono essere utilizzati per immagazzinare un certificato X.509 in modo che questo non sia accessibile anche in caso di accesso fisico al dispositivo. In questo modo l'attacco di "impersonificazione" è praticamente impossibile perché richiederebbe modifiche hardware complesse se si avesse l'intenzione di utilizzare lo stesso certificato per connettersi con un nodo IoT differente.

La parte successiva di questo documento riassume le caratteristiche tecniche di Stack4Things e sarà tradotto in italiano a breve.

Stack4Things

An OpenStack-based Internet of Things Framework

Overview

Stack4Things is an OpenStack-based Internet of Things framework developed by the Mobile and Distributed Systems Lab (MDSLab) at the University of Messina, Italy. Stack4Things is an open source project that helps you in managing IoT device fleets without caring about their physical location, their network configuration, their underlying technology. It is a Cloud-oriented horizontal solution providing IoT object virtualization, customization, and orchestration. Stack4Things provides you with an out-of-the-box experience on several of the most popular embedded and mobile systems.

High level functionalities

It provides the following high level functionalities.

- **Object virtualization:** Interact with your IoT devices as entities in the Cloud. Access all their hardware and software properties with a uniform REST-based interface.
- **Remote control and customization:** Fully customize your devices from low level firmware/operating system configuration to your business logic. Operate on your nodes through the Cloud interface or remotely access them wherever they are.
- **Fleet management:** Organize your IoT objects in fleets and control them hassle-free, whichever the scale. Trigger operations on an entire fleet with a simple click.
- **Overlay networks of things:** Create and manage Cloud-mediated virtual networks among your objects. Deploy standard applications on your IoT devices as they were on the same LAN, without caring about NAT/firewall issues.

The REST API

Stack4Things exposes a REST API which is based on the following main resources.

- **Boards:** Boards are the virtual counterpart of physical IoT nodes. You can register a new board and let the corresponding IoT node appear on the Cloud with a geographical position and a presence status (online, offline, under maintenance). Each board has metadata associated and a list of attached sensors. This is useful for your applications to automatically discover the IoT nodes that can be used for a certain goal. You can also trigger actions on your IoT nodes, e.g., rebooting them or executing shell commands.
- **Projects:** Projects are our way to organize IoT nodes into fleets and manage them as an ensemble. You can add a board to a project, remove it or list the boards in a project. Once your project is ready, you can trigger batch operations on the entire fleet with a single command, monitor the status of the triggered operations, and retrieve the corresponding results.
- **Services:** Stack4Things allows you to access any TCP/UDP port on your IoT nodes from the Cloud even if they are behind NATs or strict firewalls. See it as a Cloud-based port forwarding mechanisms that, e.g., enables the possibility to SSH into any IoT node whatever its geographical position and network configuration. You can add a new service by specifying the port and the protocol and you can then expose it on the Cloud on any of your IoT nodes. When the service is not needed anymore you can unexpose it.
- **Virtual networks:** As virtual machines in the Cloud are usually connected through virtual networks, Stack4Things allows you to create Cloud-mediated overlay networks that put boards into direct communication. NATs and strict firewalls are punched and boards can interact as if they were on the same LAN, simplifying the implementation of IoT applications at the scale of metropolitan/wide/global area networks. You can create a network by specifying the range of IP addresses that

you need and then dynamically add/remove boards to/from it. You can then destroy the network if you do not need it anymore.

- **Plugins:** Stack4Things provides contextualization functionalities in the form of self-contained piece of application logic that can be stored in the Cloud and then injected on boards when needed. Node.js and Python are supported as programming languages. Asynchronous and synchronous plugins can be implemented. Asynchronous plugins can be used to inject long-running application logic on boards that can be started/stopped as needed. Synchronous plugins are remote procedures that can be invoked on the board and can return data to the caller. Plugins are versioned on the Cloud so that the developer has full control over the logic running on the IoT nodes.



Stack4Things - IoT platform

Index

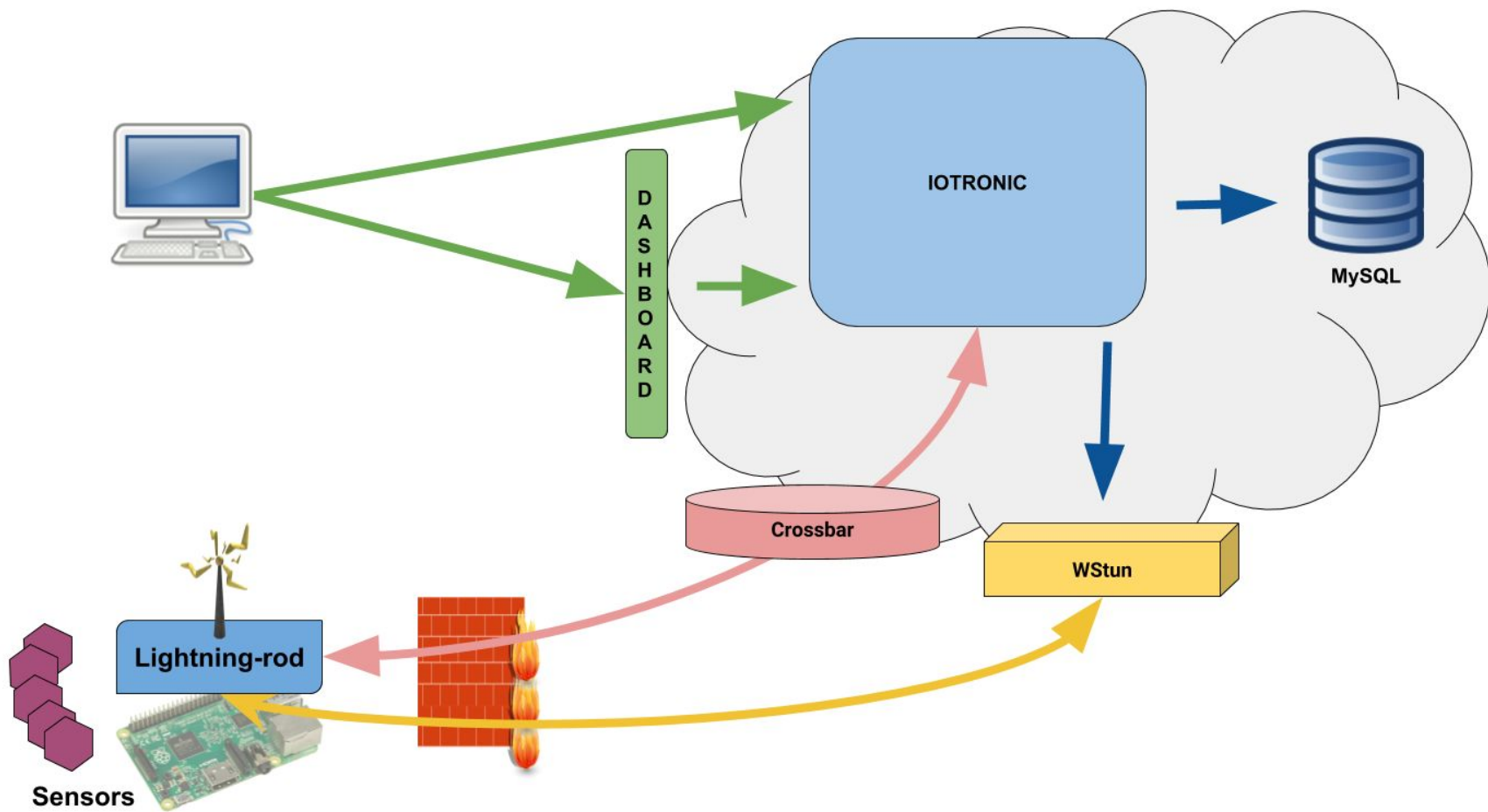


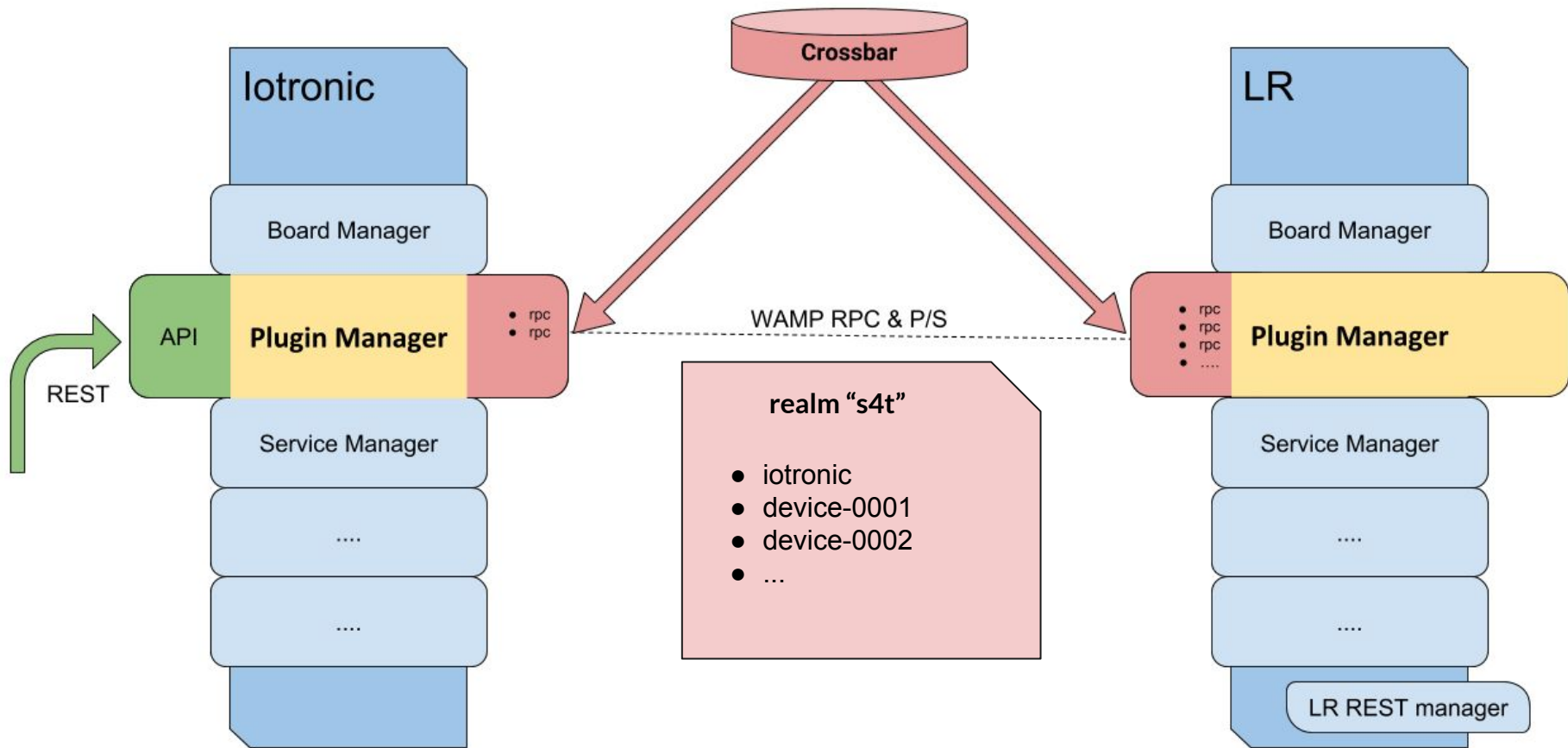
- Stack4Things overview:
 - Iotronic e Lightning-rod
 - WAMP protocol (Autobahn - Crossbar.io)
- Plugin Manager:
 - overview
 - plugin management: create, execution, update, batch operation
- Device management:
 - upgrade Lightning-rod
 - device package management
 - Service Manager (WSTUN: websocket tunnels)
 - LR logging management

Stack4Things



- Stack4Things è un progetto **open-source** che permette il **management** di device IOT remoti senza la necessità di aver conoscenza della loro configurazione di rete (con o senza firewall) e della tecnologia hardware usata.
- Fornisce un out-of-the-box experience per i più popolari **sistemi embedded** e **sistemi mobile**.
- Sviluppato in **NodeJS**.
- Si compone di una parte server (cloud-side), **lotronic**, una parte client (device-side) **Lightning-rod (LR)** e di un'interfaccia web di management.
- Il canale di comunicazione tra lotronic e le varie istanze di LR è basato sul protocollo **WAMP**, gestito lato cloud da un'istanza **Crossbar.IO** (autobahn library).





Plugin Manager



Definizione di “plugin”: *una porzione di codice (Python o NodeJS) innestata in un template “Iotronic-compliant”, iniettato e messo in esecuzione sui device.*

- I template dei plugin disponibili permettono di eseguire il codice utente in maniera:
 - **asincrona** (esecuzione in long-running):
 - azioni del Plugin Manager: **start - stop - restart**
 - **sincrona** (esecuzione e ritorno di un risultato):
 - azioni del Plugin Manager: **call**

NB: Ognuna delle azioni sui plugin può essere eseguita sul singolo device o in batch (flotta di dispositivi).

- Plugin logging/exceptions management

Plugin Manager



Procedura di deploy di un plugin:

1. Sviluppo di un plugin partendo da un template
2. Creazione del plugin in lotronic
 - a. Validazione/Rilascio del plugin da parte dell'Iotronic Admin
3. Inject del plugin su uno o più device
4. Eseguire azioni sul plugin su uno o più device:
 - a. ***start - stop - restart - call***

Async python plugin



```
# User imports
import time
from datetime import datetime

#PARAMS: {"name": "S4T"}

def main(plugin_name, params, api):

    logging = api.getLogger(plugin_name);

    extra = api.getExtraInfo()
    logging.info("Board extra-info: " + str(extra))

    while(True):
        now = datetime.now().strftime( "%-d %b %Y %H:%M:%S.%f" )
        logging.info("Hello "+str(params['name'])+" @ "+now)
        time.sleep(3)
```

Async nodejs plugin



```
//#PARAMS: {"name": "S4T"}

exports.main = function (plugin_name, arguments, api){

    var name = arguments.name;

    var logger = api.getLogger(plugin_name, 'debug');
    extra = api.getExtraInfo();

    logger.info("Board extra-info: ",extra);

    setInterval(function(){
        logger.info('Hello '+name+'!');

    }, 3000);

};
```

Device management

- Aggiornamento di LR da remoto
- Gestione dei software del device (install, remove, update)

Package Management

Package Manager

Command

pip

apt

opkg

pip

install

Insert the parameters/tags

List of packages

Insert here the packages

Enable / Disable per project action

Show

10

Search:

entries

	label	board_id
<input type="checkbox"/>	st-artik	af01f0fe-dc15-1a08-d27f-3602f6419744

Showing 1 to 1 of 1 entries

Previous

1

Next

Execute

Output

Lightning-rod Management

Enable / Disable per project action

Show

10

Search:

entries

	label	board_id	distro	lr_version
<input type="checkbox"/>	0123181022007	0123181022007	openwrt	2.3.3
<input type="checkbox"/>	kitra-dev-mimmo	bc1d3f61-5aab-0c0f-9a67-02e17792b84c	debian	2.3.3
<input type="checkbox"/>	lede-p0	lede-p0	openwrt	2.3.3
<input type="checkbox"/>	smartme-messina	smartme-messina-ba3830dc4e6e	openwrt	2.3.3
<input type="checkbox"/>	st-dev-mimmo	1181fd23-958c-4669-397e-4fd9e3c43ab1	openwrt	2.3.0
<input type="checkbox"/>	st-dev-smartme	473a3576-ef4b-ab91-51b5-ba3830dc4e6e	openwrt	2.3.3

Showing 1 to 6 of 6 entries

Previous

1

Next

Version

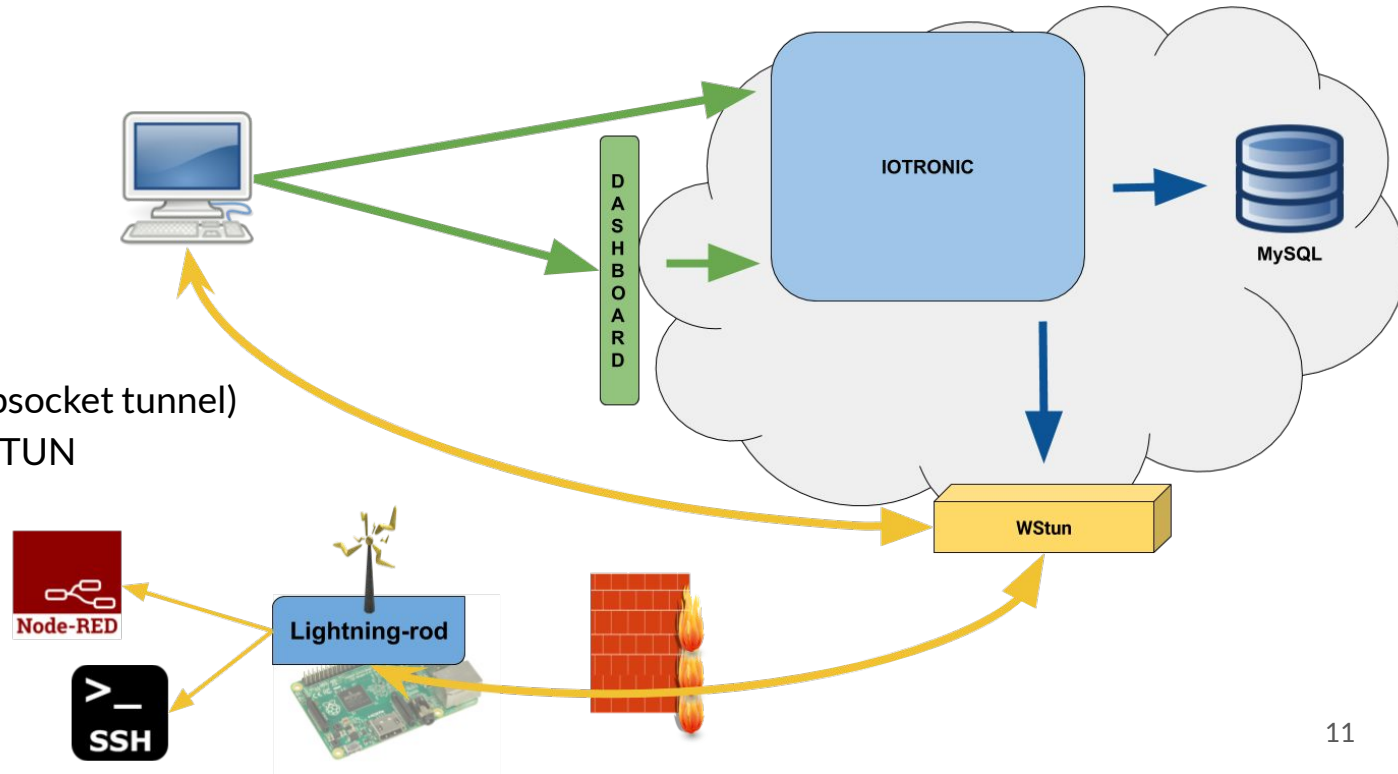
Insert version

(only if debian)

Update

Device management

- Interazione remota (websocket tunnel)
 - Service Manager & WSTUN
 - SSH
 - HTTP/S
 - ...



Device management

- Request result logging

Message for board st-dev-smartme

Result: SUCCESS

```
Downloading http://download.arancino.cc/distro/LEDE/releases/17.01.3-1-
glibc/targets/brcm2708/bcm2710/packages/Packages.gz
Updated list of available packages in /var/opkg-lists/reboot_core
Downloading http://download.arancino.cc/distro/LEDE/releases/17.01.3-1-
glibc/targets/brcm2708/bcm2710/packages/Packages.sig
Signature check passed.
Installing nano (2.9.6-1) to root...
Downloading http://download.arancino.cc/distro/LEDE/releases/17.01.3-1-
glibc/targets/brcm2708/bcm2710/packages/nano_2.9.6-1_arm_cortex-a53_neon-vfpv4.ipk
Configuring nano.
```

Previous

Requests

Show

10

Search:

entries

	id_request	subject	timestamp	result
	e8ed259e-a673-4d6f-b949-c9114d411ba8	remove plugin: 175	2019-03-20T10:45:46.000Z	completed
	ad393be5-2b6d-4050-abde-3b625536b07e	install package nano	2019-03-20T10:33:04.000Z	completed
	6ff478db-e545-468f-9232-117a8d85a9d6	remove package nano	2019-03-20T10:32:19.000Z	completed
	59aab99b-f16c-43b0-a4ac-9bba0d215ca7	get plugin logs: 175	2019-03-20T10:26:15.000Z	completed
	64dda719-68e4-4037-9ab6-d144ac785514	get plugin logs: 175	2019-03-20T10:26:04.000Z	completed
	d3b0f892-cb81-4c1a-9394-29abc9376737	get plugin logs: 175	2019-03-20T10:25:41.000Z	completed
	c225b80-d75f-412d-b227-63e0983e3077	get plugin logs: 135	2019-03-20T10:25:21.000Z	completed
	92e440b5-b1e6-4195-9f25-d98a7e0e76c8	get plugin logs: 175	2019-03-20T10:24:56.000Z	completed
	ed31c186-eacd-4e52-8dd8-de0ec73e695b	plugin action: kill 175	2019-03-20T10:24:16.000Z	completed
	8a6ac2f5-fa33-4c8e-8688-5a050c25b5c6	plugin action: run 175	2019-03-20T10:23:47.000Z	completed

Showing 1 to 10 of 101 entries

Previous

1

2

3

4

5

...

11

Next

Select All



Remove

lotronic-UI

Stack4Things

Support

ALL

Boards (8 / 20)

dev_sme1

dev_sme2

dev_st1

laquila5gb

lede-enel

sme-arancino

st-artik

st-dev-smartme

0123181022007

LEDERO

Santa Lucia Logger

carlentini_rpi0w

carlentini_rpi3

dev_ct1

dev_st2

kitra-dev-mimmo

laquila5g

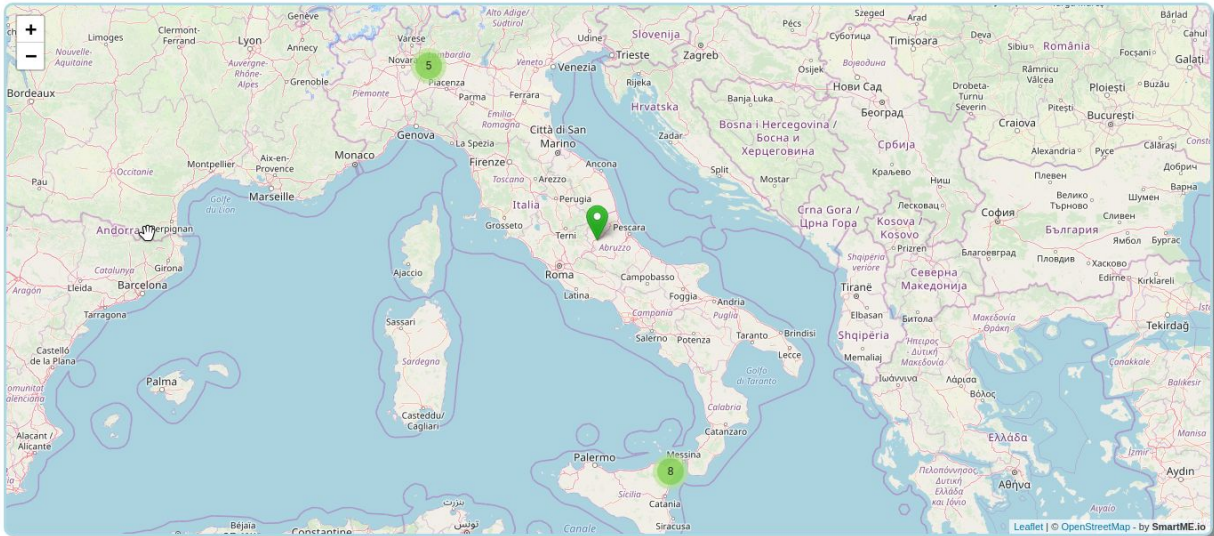
lede-flash

lede-mctx

lede-p0

lede-test-00

rasp-mimmo



Services

Plugin

Riferimenti



- Iotronic:
 - <https://github.com/MDSLlab/s4t-iotronic-standalone>
 - <https://www.npmjs.com/package/@mdslab/iotronic-standalone>
 - API docs: <https://iotronic-dev.sacertis-technology.eu-de.containers.appdomain.cloud/iotronic-api-docs/>
- Lightning-rod:
 - <https://github.com/MDSLlab/s4t-lightning-rod>
 - <https://www.npmjs.com/package/@mdslab/iotronic-lightning-rod>
- WSTUN:
 - <https://github.com/MDSLlab/wstun>
 - <https://www.npmjs.com/package/@mdslab/wstun>
- Iotronic-ui:
 - <https://github.com/MDSLlab/s4t-iotronic-webinterface>