

Cloud Data Logger

Cernat Constantin

Rezumat

Titlul temei și scopul aplicației

Tema propusă este realizarea unei aplicații care ne va permite preluarea valorilor în timp real a unor senzori și a stocării lor pe resurse de tip Cloud, și a unui tool pentru afișarea valorilor pe o pagină web.

Scopul aplicației este colectarea datelor în sensul monitorizării ambianței (temperatura, umiditatea, prezența gazelor și substanțelor toxice din aer) a unui mediu precum încăperea unei clădiri, campus studentesc, un parc sau o zonă locală din oraș cu posibilitatea globală de accesare a datelor .

Descrierea aplicației

O astfel de aplicație, poate fi pusă în funcționare datorită flexibilității sale și a consumului redus de resurse hardware, unica constrângere fiind capacitatea de stocare a datelor. Clientul web stocat pe placa de dezvoltare Raspberry Pi preia valorile de la senzori la fiecare minut, le stochează în fișiere cu extensia .sql titlul fișierelor va corespunde cu ora la care s-a făcut preluarea datelor. Următorul pas fiind încărcarea periodică a fișierelor pe un Cloud(în cazul meu fiind Google Drive). Pagina web pe care vor fi afișate datele în timp real, precum și interpretarea datelor de pe Cloud va fi stocată pe Raspberry Pi. O astfel de mobilitate ne permite observarea și măsurarea condițiilor de mediu din anumite zone (observare care se face cu ajutorul sateliților) , însă aplicația de față fiind accesibilă unui număr mai larg de consumatori.

Tehnologii folosite și structura proiectului

Cea mai bună alegere a fost folosirea serviciilor Google datorită diversității de opțiuni și posibilitățile pe care le oferă gratis pentru utilizatori noncomerciali. Inițial am pornit de la crearea unui proiect în Google Developer Console, care ne oferă accesul către API-urile necesare aplicației (Application Programming Interface) mai exact : serviciile REST și accesul clientului web către resursele Google Drive pentru a face sincronizarea și stocarea datelor).

Clientul web care va fi stocat pe placa de dezvoltare Raspberry Pi este scris în limbajul de programare Python, folosind librăria Pydrive 1.0.1, OAuth 2.0. Sistemul de operare instalat pe Raspberry fiind Raspbian Jeesy(Linux kernel), oferă posibilitatea automatizării taskurilor precum preluarea periodică a datelor și încărcarea lor pe cloud fără intervenția utilizatorului. Mai exact necesitatea de a crea un cron job sau a unui daemon care va face funcționarea autonomă a întregului sistem .

Protocoalele care pot fi folosite pentru comunicarea dintre Raspberry Pi și senzori sunt: SPI, 1Wire și I2C. Aplicația fiind una Object-Oriented, va folosi utilitățile Drive REST API pentru crearea fișierelor obiect și încărcarea lor pe Cloud cu posibilitatea folosirii lor ulterioare pe pagina Web.

Pentru a crea pagina Web va fi folosită o librărie din Javascript (folosind limbajele PHP, Javascript).