

Treball final de grau

Estudi: Grau en Enginyeria Informàtica

Títol: Implementació d'un equip per a la captació i emmagatzematge de dades d'una xarxa de sensors sense fils

Document: Memòria

Alumne: Charly Repullo Lauks

Director/tutor: Carles Pous Sabadí

Departament: Enginyeria Elèctrica, Electrònica i Automàtica

Àrea: Enginyeria de Sistemes i Automàtica

Convocatòria (febrer/2015)

ÍNDEX

1. Introducció, motivacions, propòsit i objectius del projecte	4
2. Estudi de viabilitat	6
3. Metodologia	7
4. Planificació.....	9
5. Marc de treball i conceptes previs	12
5.1 ZigBee.....	12
5.2 Waspnotes i mòduls xbee.....	13
5.3 Raspberry PI	16
5.4 LAMP	17
5.4.1 Linux:	18
5.4.2 Apache:.....	18
5.4.3 MySQL.....	19
5.4.4 PHP:	19
5.5 Python	20
5.6 Highcharts	20
5.7 Visió global	21
6. Requisits del sistema	22
7. Estudis i decisions.....	24
7.1 Raspberry Pi.....	24
7.2 Python:	25
7.3 MySQL	25
7.4 Highcharts	25
7.5 Protocol de trames:	26
8. Anàlisi i disseny del sistema.....	27
8.1 Recollir dades.....	27
8.2 Emmagatzemar Dades	27
8.3 Visualitzar dades	28
8.4 Tractament de Dades	31
9. Implementació i proves	32
9.1 Aprenentatge i configuració dels Waspnotes	32
9.2 Parsejar les trames.....	32

9.3 Gràfics de dades	32
9.4 Tractament posterior de les dades.....	33
9.5 Reemplaçar Dispositiu.....	33
10. Implantació i resultats	34
10.1 Legislació i normativa vigent.....	34
10.2 Resultats	34
10.3 Arxius del projecte	36
11. Conclusions	37
11.1 Objectius	37
11.2 Terminis Dels Objectius.....	37
11.3 Opinions Personals	37
12. Treball futur.....	38
12.1 Avisos.....	38
12.2 Identificació de Waspnotes	38
12.3 Intervals d'enviament.....	38
13. Bibliografia	39
14. Manual d'usuari i/o instal·lació	40
14.1 Instal·lar sistema operatiu.....	40
14.2. Actualitzar i preparar la Raspberry.....	41
14.3. Instal·lar Software.....	42
14.4. Configurar entorn.....	46
14.5. Seguretat.....	50
14.6. Carregar el projecte	51

1. Introducció, motivacions, propòsit i objectius del projecte

En els últims anys hi ha hagut un gran creixement i expansió de la tecnologia en tots els àmbits. S'intenta informatitzar tot o aplicar noves tecnologies per millorar-ne l'eficiència, utilitat i connectivitat. Els cotxes, mòbils, televisions i fins i tot cases intel·ligents es comuniquen per compartir informació per millorar el nostre entorn i facilitar-nos el dia a dia.

És per això, que es va iniciar aquest projecte. Mitjançant uns dispositius amb sensors sense fils podem obtenir dades com temperatura, humitat i lluminositat que permetran fer estudis i càlculs per millorar l'eficiència energètica de l'entorn que monitoritzen.

L'Eficiència Energètica consisteix en la reducció de consum d'energia, mantenint els mateixos serveis energètics, sense disminuir el servei, confort ni la qualitat de vida, assegurant el proveïment, protegint el medi ambient i fomentant la sostenibilitat.

Seguidament veurem la importància de controlar els consums energètics tant per estalviar com per protegir el medi ambient.

El Codi Tècnic de l'Edificació (Código Técnico de la Edificación (CTE)) és el conjunt principal de normatives que regulen la construcció d'edificis a Espanya des de 2006. És el codi d'edificació en vigor al país. En ell s'estableixen els requisits bàsics de seguretat i habitabilitat de les construccions, definits per la Llei d'Ordenació de l'Edificació (LOE).

La secció HE-3 del Codi Tècnic de l'Edificació estableix com a exigència bàsica que els edificis, tant els nous com els que es reformin, disposin d'instal·lacions d'il·luminació adequades a les necessitats dels seus usuaris i alhora eficaços energèticament. Per a això, l'eficiència energètica del sistema d'il·luminació no ha de superar un valor límit i haurà de comptar també amb un sistema de control que permeti ajustar l'encesa a l'ocupació real de la zona, així com un sistema de regulació que optimitzi l'aprofitament de la llum natural.

Els recursos energètics són limitats i costosos, i per tant, cal limitar-ne l'ús.

Per això, és necessari tenir un control tant dels consums com de la temperatura, il·luminació, humitat i altres factors de l'espai que es vol monitoritzar.

Les dades són recollides mitjançant una xarxa de sensors sense fil, amb l'objectiu de generar un model de comportament que permeti predir-ne el consum, a més d'extreure conclusions per augmentar-ne l'eficiència elèctrica. El cas que ens ocupa fa referència al mòdul PIV de la Universitat de Girona.

Aquest projecte, pretén complementar un projecte previ que consisteix en fer estudis amb les dades recollides per aquests dispositius.

Les dades obtingudes serviran per predir consums, detectar despeses innecessàries, patrons ocults i altres dades que podrien ajudar a millorar l'eficiència energètica de l'edifici.

Avui en dia moltes empreses es dediquen a reduir les despeses energètiques, sigui en altres empreses o a particulars. El que fan és primer analitzar els consums i estat de les instal·lacions i posteriorment intentar reduir al màxim els seus consums.

Molt sovint es pot estalviar força analitzant els consums i reprogramant les hores en què s'han d'encendre les llums i calefacció o aire condicionat. Un altre punt de millora és un bon aïllament i bons materials per evitar el fred o calor.

Per tant, les dades proporcionades per la xarxa de sensors ens seran molt útils, ja que ens permetran tenir informació de l'estat de l'edifici i també fer-ne estudis i anàlisi.

En aquest cas, es vol recollir dades de l'edifici PIV de la Universitat de Girona però més enllà d'un únic edifici, aquest sistema també es pot aplicar a un conjunt d'edificis o tota una ciutat, és per això ara es parla molt de "Smart City".

La «ciutat intel·ligent» de vegades també anomenada «ciutat eficient», es refereix a un tipus de desenvolupament urbà basat en la sostenibilitat que és capaç de respondre adequadament a les necessitats bàsiques d'institucions, empreses, i dels mateixos habitants, tant en el pla econòmic, com en els aspectes operatius, socials i ambientals. Una ciutat o complex urbà podrà ser qualificat d'intel·ligent en la mesura que les inversions que es realitzin en capital humà (educació permanent, ensenyament inicial, ensenyament mitjà i superior, educació d'adults ...), en aspectes socials, en infraestructures d'energia (electricitat, gas), tecnologies de comunicació (electrònica, Internet) i infraestructures de transport, contemplin i promoguin una qualitat de vida elevada, un desenvolupament econòmic-ambiental durable i sostenible, una governança participativa, una gestió prudent i reflexiva dels recursos naturals, i un bon aprofitament del temps dels ciutadans.

Les ciutats modernes, basades en infraestructures eficients i durables d'aigua, electricitat, telecomunicacions, gas, transports, serveis d'urgència i seguretat, equipaments públics, edificacions intel·ligents d'oficines i de residències, etc., s'han d'orientar a millorar el confort dels ciutadans, sent cada vegada més eficaços i brindant nous serveis de qualitat, mentre que es respecten al màxim els aspectes ambientals i l'ús prudent dels recursos naturals no renovables.

Per aquest projecte, es disposa d'un concentrador (Meshlium) que recull totes les dades d'una xarxa de sensors sense fils. Els nodes sensors (Waspnotes), estan constituïts per una placa amb microprocessador que porta incorporats els sensors i un mòdul de comunicacions ZigBee. Aquests sensors capturen dades de presència, temperatura, humitat relativa i lluminositat dins l'edifici PIV, que periòdicament van enviant al Meshlium. El Meshlium no és res més que un processador, un disc dur i un mòdul Zigbee amb un programari que té per objectiu recollir i emmagatzemar les trames d'informació que li arriben dels diferents Waspnotes en una base de dades. L'objectiu principal del projecte és el de poder disposar d'un equip que realitzi unes funcions similars a les del Meshlium, amb la finalitat de superar algunes mancances de l'equip actual i per tal de gaudir d'un mòdul de reserva a bon preu.

2. Estudi de viabilitat

La infraestructura inicial per aquest projecte tindrà un cost molt reduït, ja que només necessitarem un ordinador qualsevol amb Linux i una Raspberry PI.

L'ordinador és opcional si disposem d'una pantalla amb connexió HDMI, ja que podem utilitzar directament la Raspberry per desenvolupar el projecte. Per tant, els costos de desenvolupament són molt baixos.

El cost de la Raspberry PI més els accessoris necessaris és d'aproximadament 50€.

La resta d'eines per desenvolupar el projecte no han tingut cost, ja que són eines gratuïtes com Netbeans, Atom, Highcharts, Apache, Mysql ...

L'únic cost afegit que podríem tenir en compte és el cost del programador, però que en aquest cas és de 0€, ja que es tracta del Treball de Final de Grau.

Cal destacar que no és necessari utilitzar la Raspberry PI. Tot el projecte s'ha desenvolupat amb un portàtil amb Linux. El producte final funciona amb qualsevol ordinador amb Linux i el programari necessari (LAMP + Python + Highcharts). Es va escollir utilitzar una Raspberry per reduir l'espai necessari i sobretot per reduir consums, ja que només consumeix 5W/h.

Per tant, si suposem que ja disposem d'un ordinador, podem dir que el cost inicial és de 0€ i finalment 50€ per la Raspberry.

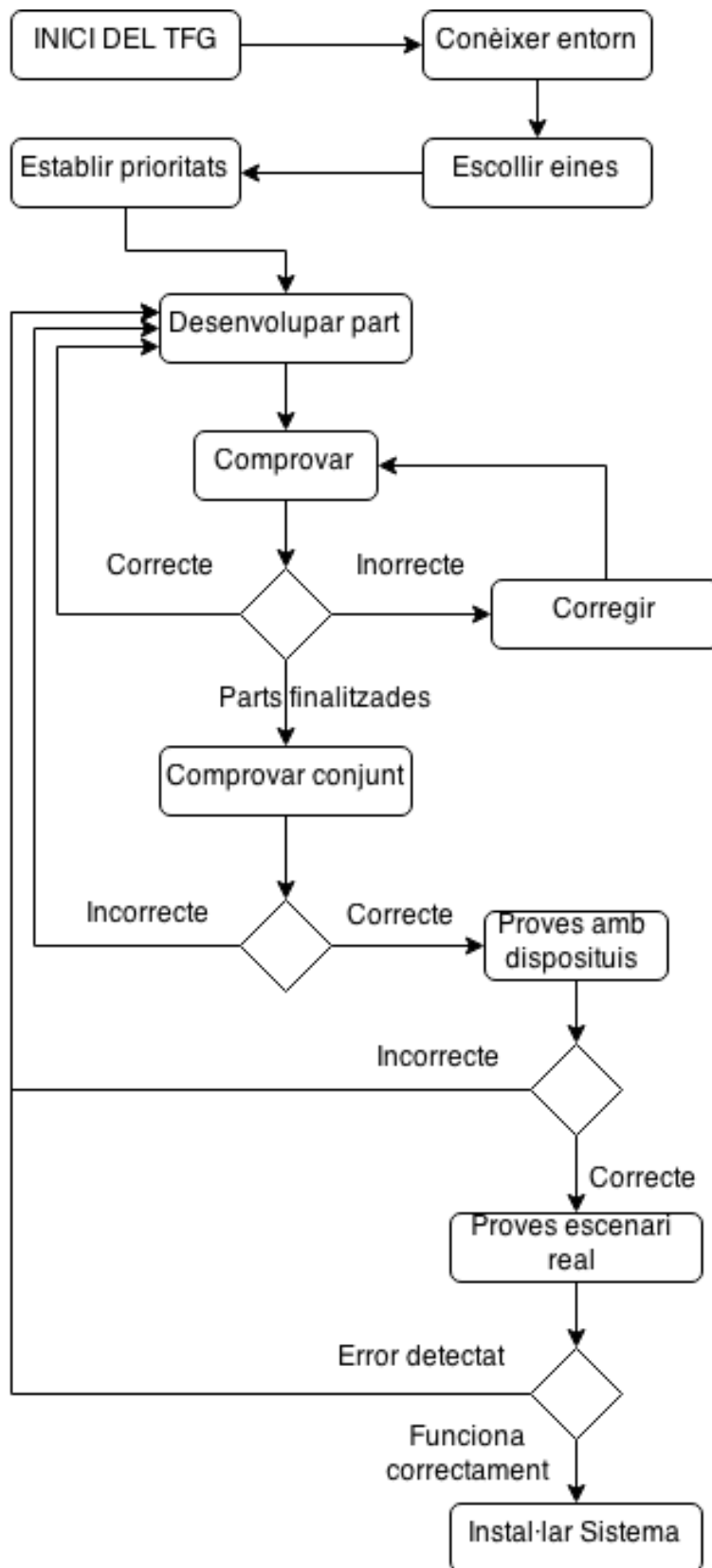
3. Metodologia

Per la realització del projecte no s'ha seguit una metodologia de treball estàndard, sinó que s'ha usat una metodologia personalitzada que consistia a proposar objectius basats en els requeriments més importants del projecte.

Es van prioritzar els requeriments més importants i un cop aquests ja s'aven assolit, ens vam centrar en altres funcionalitats que podien facilitar el posterior processament i visualització de les dades.

Els passos d'aquesta metodologia són els següents:

0. Conèixer l'entorn
1. Decidir el llenguatge de programació i eines a utilitzar.
2. Repassar i estudiar aquestes eines escollides.
3. Estructurar el treball segons les prioritats de les funcionalitats que s'han de realitzar.
4. Desenvolupar la part corresponent seguint l'ordre de prioritats.
5. Fer les comprovacions per confirmar que el funcionament és correcte en acabar cada part.
 - Si en fer les comprovacions els resultats no són els desitjats, es tornarà al punt 4 per a realitzar els canvis oportuns a l'última part desenvolupada o a les anteriors, si és necessari.
 - Si després de fer les comprovacions el resultat és el desitjat, es desenvoluparà la següent part tornant al punt 4. Una vegada que s'hagin finalitzat totes les parts amb les respectives comprovacions, s'iniciarà el punt 6.
6. Unir totes les parts desenvolupades i comprovar que el funcionament és correcte.
 - Si en fer les comprovacions el resultat no és el desitjat, es tornarà al punt 4 per realitzar els canvis oportuns en l'última part desenvolupada o en els punts necessaris, si així és necessari.
 - Si en realitzar les comprovacions el resultat és l'esperat, s'iniciarà el punt 7.
7. Fer proves amb diversos dispositius per a comprovar que el funcionament és el correcte.
 - Si en fer les comprovacions el resultat no és el desitjat, es tornarà al punt 4 per a realitzar els canvis oportuns a l'última part desenvolupada o en les anteriors si així és convenient.
 - Si en realitzar les comprovacions el resultat és l'esperat, s'iniciarà el punt 8.
8. Simular l'escenari real per comprovar que no hi hagi errors.
 - Si es detecta algun error o defecte, tornar al punt 4.
 - Si tot és correcte passar al punt 9.
9. Substituir el sistema existent (Meshlium) per la Raspberry pi.
10. Finalitzar la documentació desenvolupada al llarg del projecte.



4. Planificació

Com s'ha especificat en el punt 3, no s'ha seguit una metodologia estàndard, ja que disposàvem de força temps. La planificació s'ha fet sobre la marxa depenent dels punts que es consideraven més o menys importants.

El projecte es va iniciar a mitjans d'abril fins a mitjans de juny, i es va continuar des de principis d'octubre fins al gener.

El procediment seguit des del principi per avançar el projecte ha estat el següent; primer s'exposaven les necessitats o problemes més importants, després es decidia en equip com solucionar-ho i es posava una data per veure si s'havia aconseguit l'objectiu.

En tots els casos s'han complert els terminis i objectius proposats, excepte en algun cas concret en el qual s'ha replantejat un punt anterior i s'ha tornat a fer.

Cal remarcar que pel fet de disposar de suficient temps per desenvolupar el projecte, en cap moment s'han establert terminis d'entrega estrictes. S'han intentat respectar al màxim però no hi havia problema en allargar unes setmanes en cas d'haver-hi problemes.

A continuació exposarem tots els punts i objectius plantejats per l'equip durant el projecte.

Els punts estan en ordre cronològic, encara que en alguns casos s'ha hagut de replantejar alguna part del projecte a causa de noves decisions o requeriments.

- **Conèixer l'entorn**

El primer pas va ser aprendre quins eren els dispositius amb els quals s'havia de treballar.

Disposàvem de varis Wasmotes un Meshlium i dispositius Xbee amb USB.

- **Escollir entorn**

Un cop après quin era el funcionament bàsic dels waspmotes, es va escollir quines eines (software, llibreries, programes...) serien les més adequades pel projecte.

- **Aprendre com funcionen els dispositius Wasmote**

El següent pas va ser aprendre com configurar i carregar codi als Wasmotes.

Als aparells, se'ls ha de carregar un codi per definir les dades que recullen i envien per wifi.

Els Wasmotes sovint donen problemes a l'hora de carregar el codi i no és una tasca fàcil de realitzar si no es té gaire experiència.

Al principi es va decidir que no era necessari que jo aprenguéss la part de carregar el codi, però més endavant vam haver de dedicar-hi més temps.

- **Proves al despatx**

Les primeres proves sempre es feien al despatx per facilitar i agilitzar les comprovacions que s'havien de fer.

- **Estructurar l'enviament de dades (donar format o establir protocol)**
Per facilitar el tractament de les dades, es va decidir estandarditzar la cadena d'enviament de dades.
- **Llegir dades d'un dispositiu Xbee per USB**
El següent punt, va ser per les proves necessàries per poder llegir des del USB del mòdul Xbee les trames enviades pels Waspnotes.
- **Parsejar les dades**
Un cop vàrem aconseguir llegir les dades des de l'USB, el següent pas era parsejar correctament les dades.
Inicialment no s'havia establert cap protocol per les trames i era força complicat separar cada variable correctament dins de la trama enviada. És per això que es va decidir una forma estàndard d'enviar les trames.
- **Desar a la Base de Dades**
Seguidament, havíem de desar cada variable en una base de dades. Es va discutir quin seria el funcionament i característiques d'aquesta part del projecte.
Al principi només es treballava amb unes variables fixes, però després es va haver d'adaptar per poder afegir noves variables en temps d'execució.
- **Proves al despatx**
Regularment es feien proves al despatx per comprovar que cada part funcionés bé i si hi havia nous requeriments o necessitats.
- **Com visualitzar les**
Una funcionalitat que es va voler afegir posteriorment, va ser la de poder visualitzar algunes dades en gràfics.
Per facilitar la detecció de certs problemes com ara la bateria dels Waspnotes i altres, se'm va demanar de fer uns gràfics amb les dades.
- **Fer una web**
La forma més fàcil de poder visualitzar les dades era mitjançant una web accessible des de la mateixa xarxa a la qual estigués connectada la Raspberry.
Per tant es va decidir fer una web amb els gràfics.
- **Proves al despatx**
Es va quedar per mostrar a l'equip la web i aclarir quins eren els requisits de la web i si era necessari afegir alguna funcionalitat.
- **Arrodonir hores**
Un cop les funcionalitats principals ja estaven llestes, es van afegir altres requeriments com ara arrodonir les hores d'arribada de les trames a l'hora més propera.
Això es va demanar per facilitar el tractament de les dades en futurs estudis.

- **Proves amb diversos dispositius**

Fins ara totes les proves sempre s'havien fet amb 1 o 2 Waspmotes, i es va decidir de crear una xarxa més gran i enviar un gran nombre de dades per comprovar si tot funcionava correctament.

Les proves van resultar satisfactòries i van permetre detectar algun problema menor.

- **Nou requeriment**

En cas que falti una trama d'un dispositiu el programa haurà d'inserir una trama fictícia amb totes les variables amb valor 0 o Null amb hora corresponent.

En aquest punt van sorgir alguns problemes i es va endarrerir una mica el termini proposat inicialment. En concret hi havia un problema amb l'ordre d'inserció de les dades i amb els triggers utilitzats per arrodonir les hores. Una solució era deshabilitant el trigger i després tornar-lo a activar però no es podien deshabilitar triggers amb MySQL i això ens va fer replantejar el mètode per afegir aquestes dades.

- **Extreure les dades en un Excel o csv.**

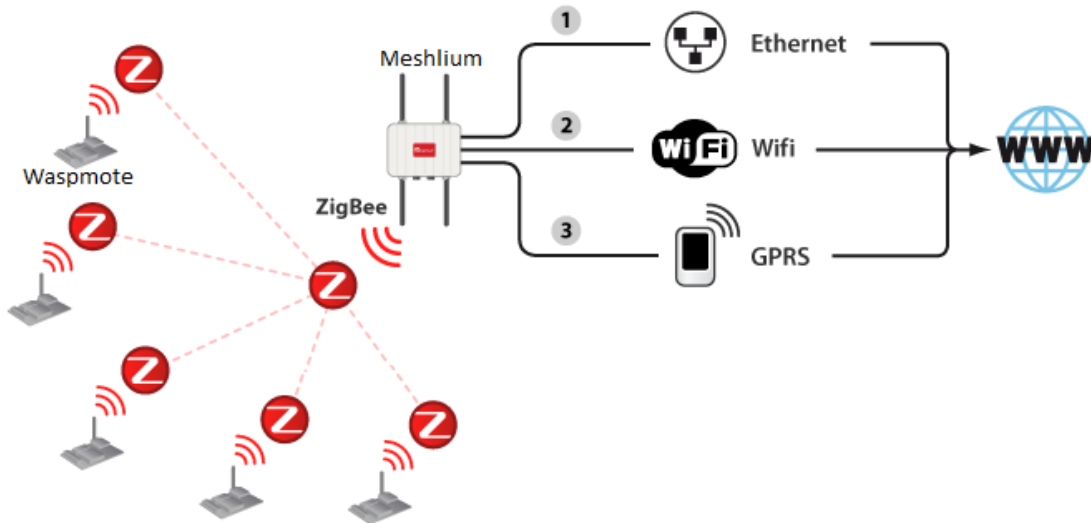
Per facilitar la feina a les persones que han de treballar amb les dades, es va decidir afegir una opció a la web per poder descarregar les dades de la base de dades en format Excel.

- **Proves al despatx**

Prova final amb tot el sistema acabat, per comprovar que tot funciona correctament i que ja es pot utilitzar.

5. Marc de treball i conceptes previs

5.1 ZigBee



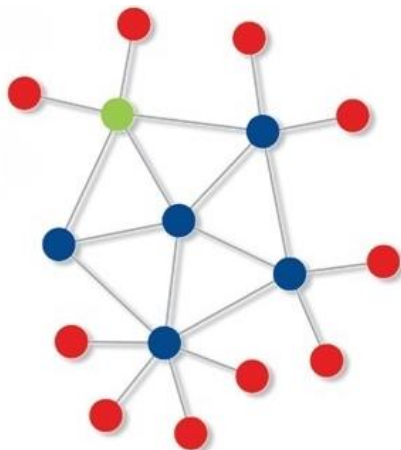
ZigBee®

ZigBee és el nom de l'especificació d'un conjunt de protocols d'alt nivell de comunicació sense fils per a la seva utilització amb radiodifusió digital de baix consum, basada en l'estàndard IEEE 802.15.4 de xarxes sense fils d'àrea personal (wireless personal area network, WPAN). El seu objectiu són les aplicacions que requereixen comunicacions segures amb un baix volum d'enviament de dades i maximització de la vida útil de les seves bateries.

En principi, l'àmbit on es preveu que aquesta tecnologia tingui més força és en domòtica. La raó d'això són diverses característiques que el diferencien d'altres tecnologies com el seu baix consum, la seva topologia de xarxa en malla i la seva fàcil integració (es poden fabricar nodes amb molt poca electrònica).

ZigBee està destinat a ser més simple i menys costosa que altres xarxes sense fils com ara Bluetooth o Wi-Fi.

- Coordinador
- Node Enllaç
- Node Sensor



Les xarxes ZigBee estan formades per un coordinador i els seus nodes que poden ser nodes amb sensors o simplement nodes per enllaçar amb altres nodes que permetran abastar més distància. El Meshlium és el coordinador de la xarxa i els Waspnotes són els nodes (en aquest cas no disposem de nodes enllaç) i el que pretén el projecte és reemplaçar el Meshlium.

5.2 Wasmotes i mòduls xbee



Els esforços d'investigació al llarg dels anys en xarxes de sensors sense fils s'han centrat a proporcionar un consum mínim i al mateix temps, el màxim rendiment i capacitats. Wasmote compta amb un consum de 0.7uA en la manera d'hibernació i set models diferents de ràdios de comunicació que poden ser elegits en funció de:

* Freqüència: 2.4GHz, 900MHz, 868MHz* Protocol: 802.15.4, ZigBee

* Potència: 1mW, 100mW

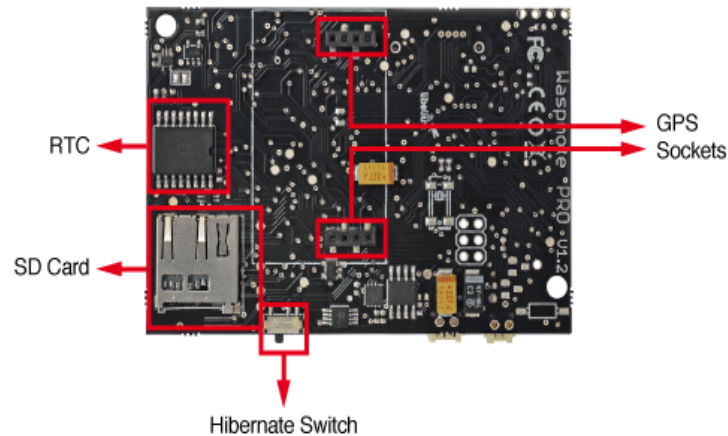


Aquests senyals de ràdio compten amb una alta sensibilitat de RX i una potència de transmissió TX que permeten al Wasmote aconseguir enllaços de llarg abast:

- 7km – 2,4 GHz
- 24Km a 900 MHz
- 40 Km a 868 MHz

Això ens permetrà monitoritzar gairebé qualsevol lloc. La plataforma està basada en una arquitectura modular, això vol dir que podem implementar mòduls addicionals en funció de les nostres necessitats, com els següents:

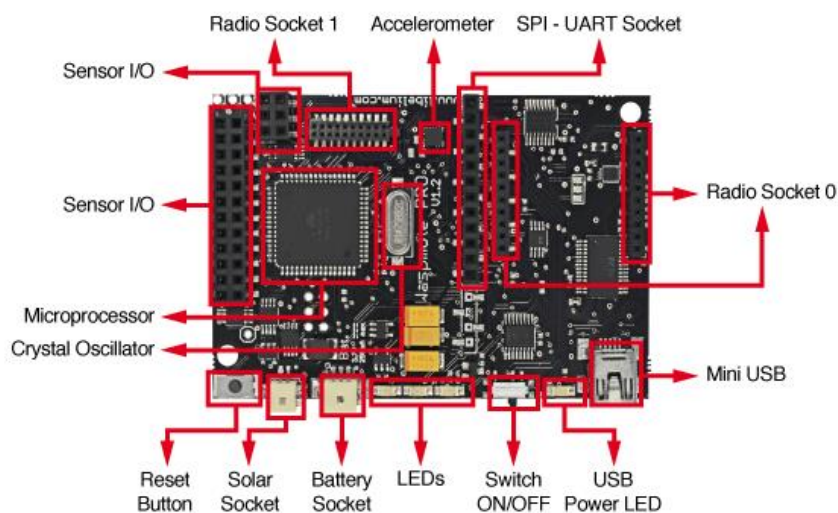
- GPS: latitud, longitud, altitud, velocitat, ...
- GPRS: sms, trucades, TCP / UDP sockets, ...
- SD card: fins a 2GB de memòria



Seguint la mateixa filosofia modular, podem connectar diverses plaques de sensors a Wasp mote com:

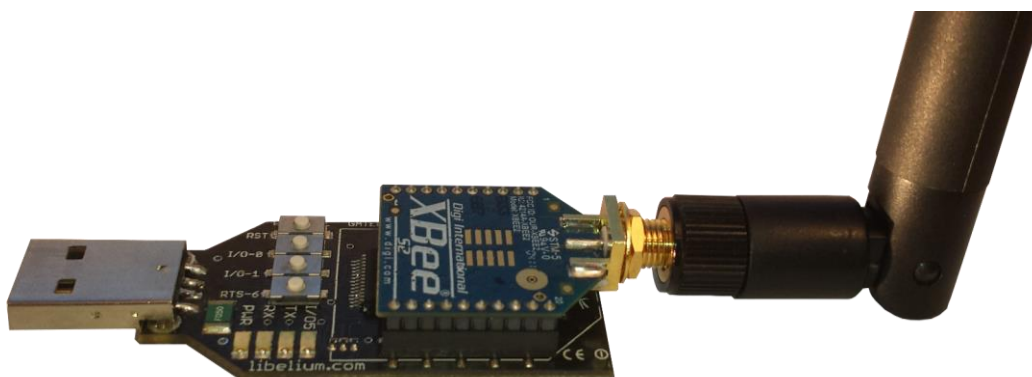
- Gasos: CO, CO₂, CH₄, SH₂, NH₃, ...
- Esdeveniments: pes, lluminositat, tilt, vibració, PIR, nivell de líquids ...
- Prototipatge: Llest per integrar nous sensors, inclosos ADC, etapes d'amplificació ...
- Agricultura, Agricultura Pro
- Smart Cities ...

A més, Wasp mote porta integrat un acceleròmetre de 3 eixos per obtenir la màxima precisió i estabilitat en tots dos rangs (+ -2g, + -6 g), el que li permet a Wasp mote controlar en temps real qualsevol tipus de moviment o vibració.



S'alimenta amb una bateria de liti que pot recarregar a través d'un connector preparat per panell solar. Aquesta opció és especialment interessant per als desplegaments en entorns naturals com els boscos.

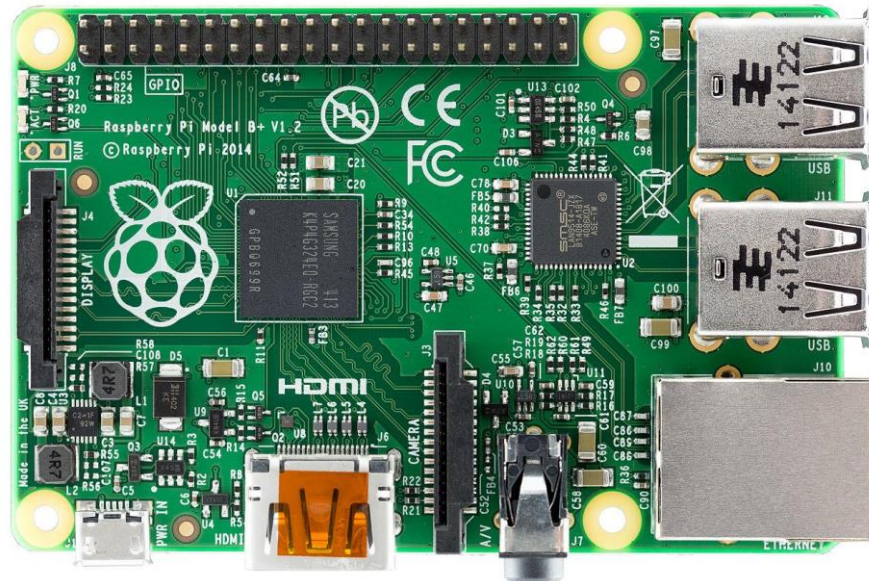
Per treure el màxim partit de totes les possibilitats i característiques de la plataforma, tenim disponible un entorn de programació. Tant l'API de Wasmote i el compilador són de codi obert.



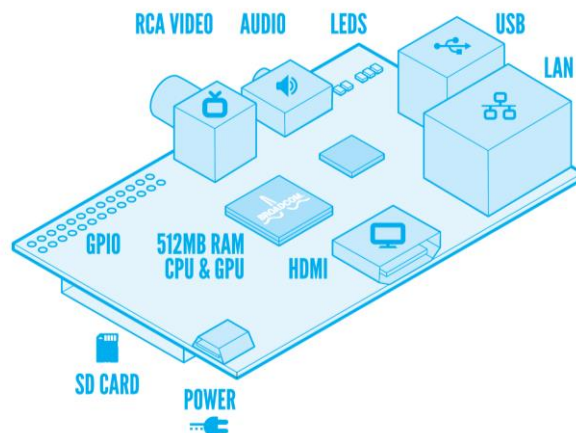
Aquest és el mòdul Xbee connectat a una placa amb USB que ens permetrà captar les dades enviades pels Wasmotes. Aquest mòdul ens facilitarà molt poder llegir les dades, ja que disposem d'un USB que podem connectar directament a la Raspberry Pi.



5.3 Raspberry Pi



El Raspberry Pi és un ordinador mono placa o SBC (acrònim en anglès de Single-Board Computer) de baix cost desenvolupat al Regne Unit per la Fundació Raspberry Pi. L'objectiu principal d'aquest disseny és estimular l'ensenyança de les ciències de la computació, però també s'ha popularitzat com a plataforma per a dissenys d'aficionats i per a usos informàtics generals. Els primers models es van començar a comercialitzar el febrer de 2012. La darrera versió, Raspberry B+, s'ha anunciat el juliol de 2014 tot mantenint el mateix processador però amb diverses millores en els seus connectors i un menor consum energètic. El disseny inclou un processador ARM1176JZF-S a 700 MHz i 512 MiB de memòria RAM amb l'objectiu d'executar GNU/Linux o RISC OS. El disseny no inclou un disc dur o una unitat d'estat sòlid, utilitza una targeta SD per a l'emmagatzematge permanent.



5.4 LAMP



El terme LAMP és un acrònim que s'origina a finals del 2000 a Alemanya per descriure la plataforma sobre la que funcionen les aplicacions web creades utilitzant la següent combinació d'eines:

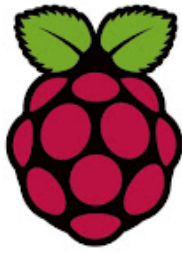
- Linux, el sistema operatiu;
- Apache, el servidor web;
- MySQL, el servidor de bases de dades;
- PHP, llenguatge de programació.

L'altra opció que sovint apareix en la publicitat dels proveïdors d'allotjament web és WMAP, que és la plataforma equivalent sobre Windows.

5.4.1 Linux:



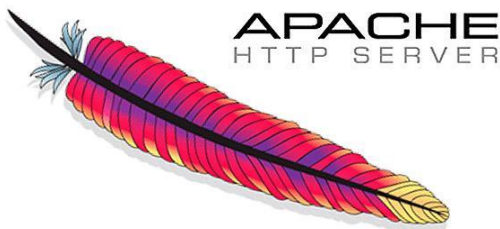
Linux



Raspbian

Raspbian és una distribució del sistema operatiu GNU / Linux i per tant lliure basat en Debian Wheezy (Debian 7.0) per a la placa ordinador (SBC) Raspberry Pi, orientat a l'ensenyament d'informàtica. El llançament inicial va ser al juny de 2012. Tècnicament el sistema operatiu és un port no oficial de Debian Wheezy armhf per al processador (CPU) de Raspberry Pi, amb suport optimitzat per a càlculs en coma flotant per maquinari, el que permet donar més rendiment en segons quins casos. El port va ser necessari en no haver-hi Debian Wheezy armhf per a la CPU ARMv6 que conté el Raspberry Pi.

5.4.2 Apache:



El servidor HTTP Apache és un servidor web HTTP de codi obert, per a plataformes Unix (BSD, GNU / Linux, etc.), Microsoft Windows, Macintosh i altres, que implementa el protocol HTTP / 1.12 i la noció de lloc virtual. Quan va començar el seu desenvolupament en 1995 es va basar inicialment en codi del popular NCSA HTTPd 1.3, però més

tard va ser reescrit per complet.

El servidor Apache es desenvolupa dins del projecte HTTP Server (httpd) de l'Apache Software Foundation.

Apache presenta entre altres característiques altament configurables, bases de dades d'autenticació i negociat de contingut, però va ser criticat per la falta d'una interfície gràfica que ajudi en la seva configuració.

Apache té àmplia acceptació a la xarxa: des de 1996, Apache, és el servidor HTTP més usat. Va aconseguir la seva màxima quota de mercat en 2005 sent el servidor emprat en el 70% dels llocs web al món, però ha patit un descens en la seva quota de mercat en els últims anys.

5.4.3 MySQL



MySQL és un sistema de gestió de bases de dades relacional (anglès RDBMS - Relational DataBase Management System) multi-fil (multithread) multiusuari, que usa el llenguatge SQL (Structured Query Language).

MySQL ha esdevingut molt popular gràcies a la seva velocitat en executar consultes i el seu suport de forma nativa per part del llenguatge PHP (fins a la versió 4.X d'aquest llenguatge, ja que a partir de la versió 5 deixa d'estar-ho), en l'elaboració d'aplicacions web, en l'entorn del programari lliure.

Es pot fer ús de MySQL en aplicacions de tota mena (web, d'escriptori o d'altres) de forma lliure i gratuïta sota les condicions de la llicència GPL. Si es vol integrar MySQL com a part d'un producte privatiu cal adquirir una llicència d'ús específica per a aquest propòsit.

MySQL és un dels components de l'arquitectura LAMP (de Linux - sistema operatiu) i la WAMP (de Windows - sistema operatiu). Les tres sigles que segueixen (A,M i P) fan referència al conjunt format per Apache (servidor web), MySQL (base de dades) i PHP (llenguatge de programació). Una plataforma per a la construcció de llocs web utilitzant programari lliure.

5.4.4 PHP:



PHP és un llenguatge de programació d'ús general originalment dissenyat per al desenvolupament web de contingut dinàmic. Va ser un dels primers llenguatges de programació per la part del servidor que es podien incorporar directament en el document HTML en lloc de cridar a un arxiu extern que processés les dades. El codi és interpretat per un servidor web amb un mòdul de processador de PHP que

genera la pàgina web resultant. PHP ha evolucionat de manera que ara inclou també una interfície de línia d'ordres que pot ser usada en aplicacions gràfiques independents. Pot ser usat en la majoria dels servidors web de la mateixa manera que en gairebé tots els sistemes operatius i plataformes sense cap cost.

Es considera un dels llenguatges més flexibles, potents i d'alt rendiment coneguts fins al dia d'avui. El que ha atret l'interès de múltiples llocs amb gran demanda de trànsit com Facebook, per optar per PHP com a tecnologia de servidor.

5.5 Python

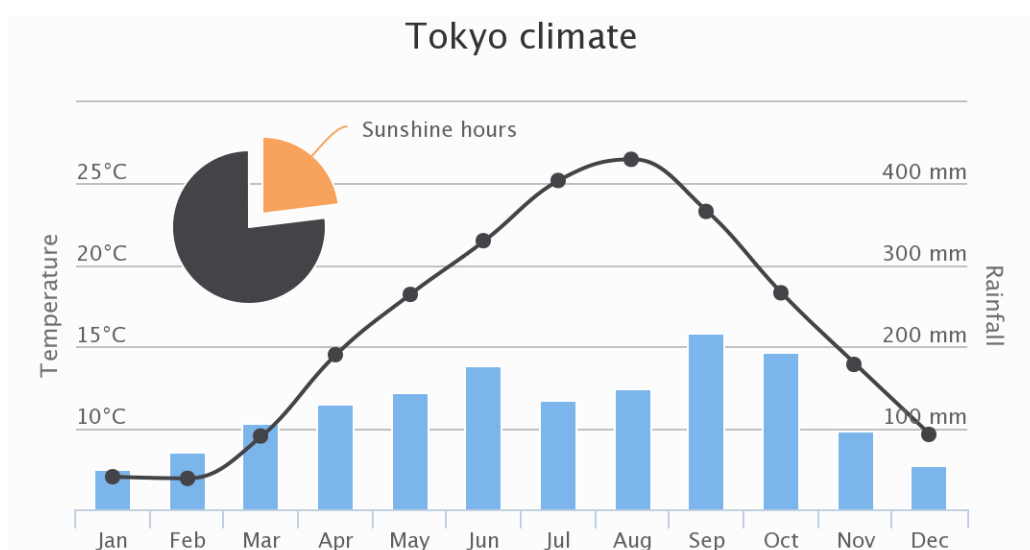


Python és un llenguatge de programació d'alt nivell àmpliament usat per propòsits generals creat per Guido van Rossum l'any 1991. La seva filosofia de disseny busca llegibilitat en el codi i la seva sintaxi permet als programadors expressar

conceptes en menys línies de codi del qual seria possible en llenguatges com C. També proveeix estructures per permetre programes més entenedors tant en petita com a gran escala. Python suporta diversos paradigmes de programació, incloent-hi programació orientada a objectes, imperativa i també funcional o procedimental. Presenta un sistema dinàmic i una gestió de la memòria automàtica i té una gran i exhaustiva biblioteca estàndard.

Com altres llenguatges de programació dinàmics, Python és usat sovint com a un llenguatge script, però també es fa servir en una àmplia gamma de contextos no-script. Utilitzant eines desenvolupades per tercers com Py2exe o Pyinstaller, el codi Python pot ser reduït a programes executables independents. Existeixen intèrprets de Python per molts sistemes operatius diferents.

5.6 Highcharts



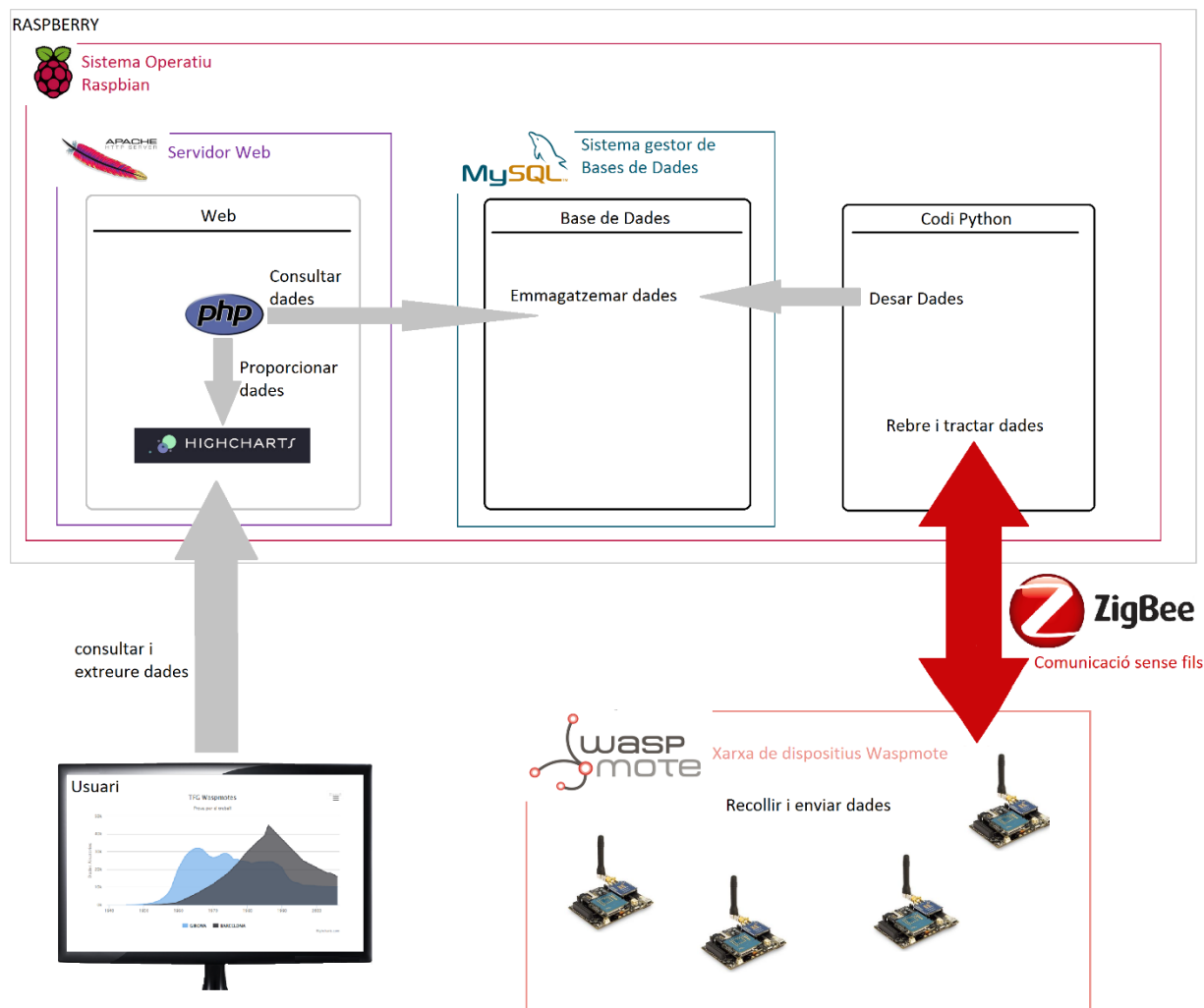
Highcharts és una llibreria gràfica escrita en JavaScript que ofereix una manera fàcil d'afegir gràfics interactius a un lloc web o aplicació web. Amb Highcharts podem fer gràfics d' àrea, columna, barres, circulars, de dispersió, bombolla ...

Funciona en tots els navegadors mòbils i d'escriptori moderns i iOS i Android, suporta multitouch i ofereix una bona experiència d'usuari.

Highcharts es basa únicament en les tecnologies de navegació natives i no requereix client plugins secundaris com Flash o Java. A més no cal instal·lar res al servidor. Highcharts només necessita dos arxius JS a executar: El nucli Highcharts.js i jQuery.

5.7 Visió global

A continuació mostrem un esquema amb totes les parts del projecte per tenir una visió global del sistema.



6. Requisits del sistema

En aquest apartat explicarem quins són els requisits que ha de complir el sistema per tal de satisfer les necessitats de l'equip que treballa amb les dades.

El punt més important del projecte era tenir un dispositiu capaç de rebre, tractar i emmagatzemar les dades enviades pels Waspnotes. Poder disposar de les dades, és el punt crític del projecte, ja que és el que permetrà poder fer bons estudis i previsions de consums energètics.

El sistema haurà d'estar sempre en funcionament, és a dir que el seu consum d'electricitat ha de ser el més baix possible. Un ordinador de sobretaula normal, consumeix entre 100W/h i 200W/h depenent de les seves característiques, en canvi el Raspberry Pi té un consum màxim de 5W/h el que suposa entre 20 i 40 vegades menys consum elèctric.

Tot seguit, exposarem quins han estat els requisits que se m'han demanat durant el desenvolupament del projecte.

- **Analitzar entorn**

Al principi simplement es va plantejar fer un script per llegir les dades des de l'USB. També havia de parsejar les dades, ja que n'arriben en una cadena de caràcters i posteriorment desar cada variable a la base de dades.

- **Estructurar l'enviament de dades**

Donar format o establir un protocol per facilitar i estandarditzar la forma d'enviar cada variable dins d'una trama.

- **Llegir dades d'un dispositiu Xbee per usb**

Per captar les dades enviades pels Waspnotes, disposàvem d'un mòdul Xbee amb usb. Aquest mòdul es configura per estar en la mateixa xarxa que els Waspnotes i llavors cada vegada que escolta una trama l'envia pel port sèrie(USB) i el programa la pot llegir.

- **Parsejar les dades**

Les dades arriben en una trama, una línia amb totes les variables. Per tant el següent pas va ser tractar aquesta trama per tal de separa cada dada en una variable independent.

- **Desar a la Base de Dades**

Les variables obtingudes s'han d'emmagatzemar correctament en una base de dades per tal que siguin fàcilment accessibles i facilitar-ne el tractament.

- **Arrodonir hores**

L'equip que treballa amb les dades per fer estudis de consums, necessita tenir una trama per hora. Un requisit del sistema és arrodonir les hores d'arribada de cada trama. Com expliquem en l'apartat 9.4, si una trama arriba a les 15:28, s'haurà d'arrodonir a les 15:00.

- **Afegir trames perdudes**

Com en l'apartat anterior, per facilitar l'estudi de les dades, en cas que falti una trama d'un dispositiu, s'haurà d'inserir una trama fictícia amb dades a 0 o NULL.

- **Crear gràfics**

Per facilitar la visualització de certes dades, es va demanar la implementació d'una pàgina web on es poguessin veure dades com temperatura i bateria de cada dispositiu.

- **Extreure les dades en un Excel o .csv**

Un membre de l'equip necessita exportar les dades en un Excel per poder treballar més còmodament així que es va decidir automatitzar aquest procés, i des de la web es poden descarregar les dades en format Excel.

- **Manual**

Un punt important és disposar d'un manual detallat de tot el que s'ha fet i com replicar-ho en cas de necessitat, com reemplaçar la Raspberry o la targeta SD en cas d'avaries o errors.

7. Estudis i decisions

Un punt molt important a tenir en compte a l'hora d'escollir una eina, era la simplicitat, usabilitat i documentació disponible.

S'ha preferit utilitzar eines que disposin d'una bona documentació i/o que siguin conegudes i simples d'utilitzar per facilitar possibles modificacions o ampliacions futures.

7.1 Raspberry Pi

L'objectiu del projecte era Reemplaçar el Meshlium. Els requisits principals són un baix consum d'electricitat per estalviar costos i una mida reduïda per qüestions d'espai.

Les propostes inicials eren dispositius tipus android stick o android pc. Personalment ja havia treballat i fet alguns projectes amb Raspberry pi i coneixent-ne les propietats, flexibilitat i possibilitats d'ús, es va decidir que seria la millor opció.

La Raspberry pi és un petit ordinador amb un processador ARM sobre el que podem instal·lar diferents distribucions Linux.

Raspbian és el sistema operatiu més utilitzat amb la Raspi i amb el que es pot fer tot el que voldríem fer amb un ordinador normal. Un inconvenient és la potència de processament, memòria ram i espai d'emmagatzematge, però en el cas d'aquest projecte, aquests 3 punts no són crítics, ja que el volum de dades a tractar és poc. Tampoc tindrem un gran tràfic de dades ni consultes donat que poques persones utilitzaran i consultaran les dades.

La resta de programari s'ha escollit perquè és programari lliure o gratuït i per ja tenir experiència amb el software.

7.2 Python:

El codi per llegir les dades és un codi simple i de poques línies. Python és un llenguatge de programació d'alt nivell que sovint s'utilitza per fer scripts i codis per realitzar tasques concretes.

A més a més personalment ja l'havia utilitzat en altres projectes i disposem de moltes llibreries i informació sobre la programació amb Python.



Per la part de desenvolupament en Python s'ha utilitzat PyCharm, ja que és un entorn de desenvolupament integrat amb una versió gratuïta, fet principalment per al llenguatge de programació Python.

Per desenvolupar la resta de codi s'han utilitzat editors de text com ara Kate o Atom que són eines lliures.

7.3 MySQL

Per desar les dades s'havia d'utilitzar un sistema gestor de bases de dades. MySQL és un software lliure i amb el que s'ha treballat durant la carrera i les seves funcionalitats i rendiment són més que suficients per a les necessitats del projecte.

7.4 Highcharts

Per la part dels gràfics, no sabia com fer-ho així que es van provar diverses opcions fins a trobar la més adequada pel projecte.

Primer es va provar amb gnuPlot però els gràfics eren estàtics, ja que es generava una imatge i després s'havia de mostrar a la web mitjançant PHP.

Aquesta opció es va descartar, ja que era complicat realitzar tot el procés i els resultats no eren els desitjats.

Després de buscar llibreries i software per fer gràfics en una web, vaig optar per fer-ho amb d3.js.

d3.js són unes llibreries escrites en JavaScript que permeten fer tot tipus de gràfics dinàmics i interactius. Aquest software és molt potent i ofereix una infinitat de possibilitats, però requereix un gran aprenentatge.

Finalment després de seguir buscant es va optar per Highcharts perquè és una bona eina que compleix amb les nostres necessitats, és fàcil d'utilitzar i compta amb molts exemples a la seva web.

Un cop après el seu funcionament, i mitjançant PHP per extreure les dades de la base de dades, resulta fàcil i entenedor poder crear gràfics.

7.5 Protocol de trames:

Al principi les trames no tenien un format estàndard, les variables tenien noms diferents en alguns Wapsmotes i dins d'una trama la separació entre variables era diferent entre unes i altres.

Des del començament del projecte es va proposar aplicar un protocol per tal de generalitzar i facilitar el processament de trames, però es va descartar aquesta opció. Un cop un Wapsmote està instal·lat i configurat, el procés per modificar el software és lent i tediós.

Inicialment es va fer el programa per detectar 4 variables i no era possible afegir-ne més, cosa que no era gaire escalable. Finalment sí que es va optar per establir un protocol, ja que això facilitava el procés parseig de la trama i oferia la possibilitat d'afegir i treure variables sense haver de modificar el programa.

8. Anàlisi i disseny del sistema

Per les característiques d'aquest projecte, no ha hagut grans requeriments quant a anàlisi i disseny. Cada part descrita a continuació no té més de 50-100 línies de codi i s'ha fet segons les necessitats dels requeriments que havien de complir.

Dit això, explicarem quines són les parts del sistema i els seus punts a destacar.

8.1 Recollir dades

El primer pas és captar les trames que envien els Waspmotes.

Cada Waspmote envia una trama cada cert temps. Per capturar les trames utilitzem un dispositiu Xbee amb USB.

S'ha de tenir en compte que les dades ens arriben en una única línia de caràcters que conté totes les variables que haurem de tractar. En el punt 9.2 expliquem quina és l'estructura de les trames rebudes. Per tant, és necessari separar-les i per això utilitzem un Diccionari que és una estructura de dades organitzada per clau-valor. La clau serà el nom de la variable i el valor la dada.

8.2 Emmagatzemar Dades

Cada trama serà desada a la base de dades com una nova fila i cada variable és una columna de la taula de la base de dades.

Per tant per cada trama rebuda, primer separem cada valor de la trama i després inserim una nova fila a la base de dades amb totes les dades rebudes d'aquesta trama.

Aquesta fila inserida en la base de dades correspon a la següent trama:

-# -mac: 4090C5A04090C5A0 -temp:0 -bat:86% -#

id	mac	temp	bat	x	y	z	date	dateOK
1065	4090C5A04090C5A0	0	86%	0	0	0	2014-12-01 16:38:08	2014-12-01 17:00:00

Com que en aquest cas no s'han enviat les dades 'x', 'y' i 'z' els seus valors prenen el valor per defecte de la base de dades.

8.3 Visualitzar dades

Un dels requeriments és poder consultar algunes dades de forma més visual. Per aquesta tasca s'ha utilitzat Highcharts, unes llibreries escrites en javascript.

Highcharts és fàcil d'utilitzar i configurar i disposa de molts exemples a la seva pàgina web.

Simplement haurem de crear un 'div' en el que inserirem els gràfics de la nostra web.

```
$(function () {  
  
    var ranges = [  
        [1246406400000, 14.3, 27.7],  
        [1246492800000, 14.5, 27.8],  
        ...  
        [1248998400000, 10.8, 16.1]  
    ],  
    averages = [  
        [1246406400000, 21.5],  
        [1246492800000, 22.1],  
        ...  
        [1248998400000, 13.6]  
    ];  
  
    $('#container').highcharts({  
  
        title: {  
            text: 'July temperatures'  
        },  
  
        xAxis: {  
            type: 'datetime'  
        },  
  
        yAxis: {  
            title: {  
                text: null  
            }  
        },  
  
    },
```

```
        tooltip: {  
            crosshairs: true,  
            shared: true,  
            valueSuffix: '°C'  
        },  
  
        legend: {  
        },  
  
        series: [{  
            name: 'Temperature',  
            data: averages,  
            zIndex: 1,  
            marker: {  
                fillColor: 'white',  
                lineWidth: 2,  
                lineColor:  
                    Highcharts.getOptions().colors[0]  
            }  
        }, {  
            name: 'Range',  
            data: ranges,  
            type: 'arearange',  
            lineWidth: 0,  
            linkedTo: ':previous',  
            color:  
                Highcharts.getOptions().colors[0],  
            fillOpacity: 0.3,  
            zIndex: 0  
        }  
    ]
```

```
<div id="container" ></div>
```

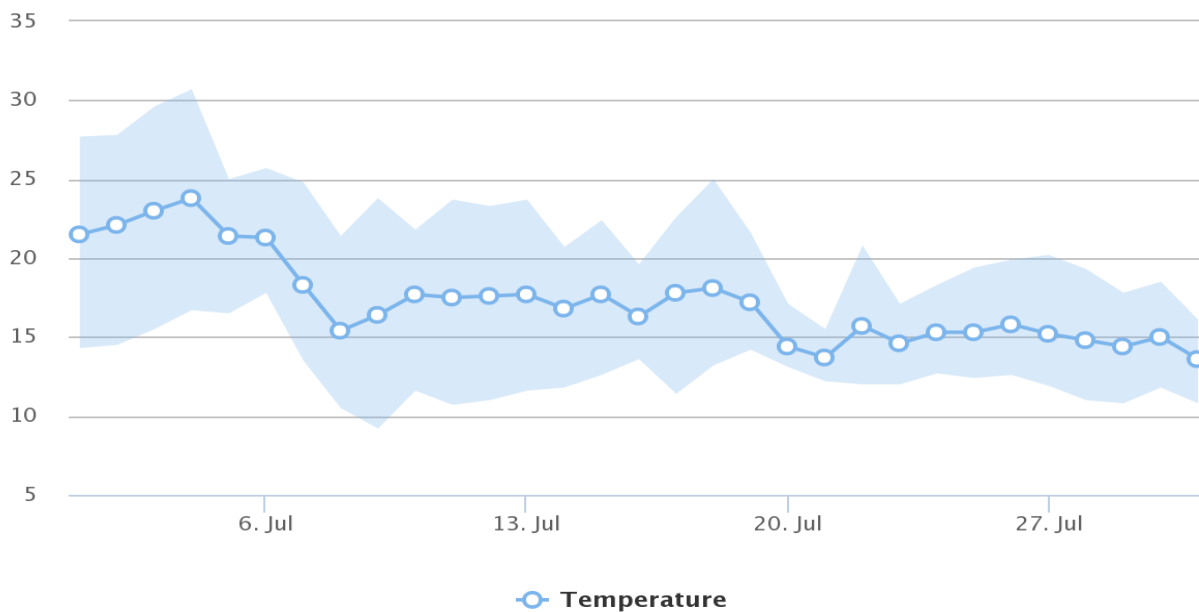
En aquest cas les dades estan introduïdes en el codi (són dos llistes, ranges i averages), però nosaltres les extraurem de la base de dades mitjançant PHP.

Highcharts es configura amb etiquetes com 'title' pel títol, 'xAxis' per les propietats de l'eix X...

Finalment en l'etiqueta 'series' introduïrem les nostres dades.

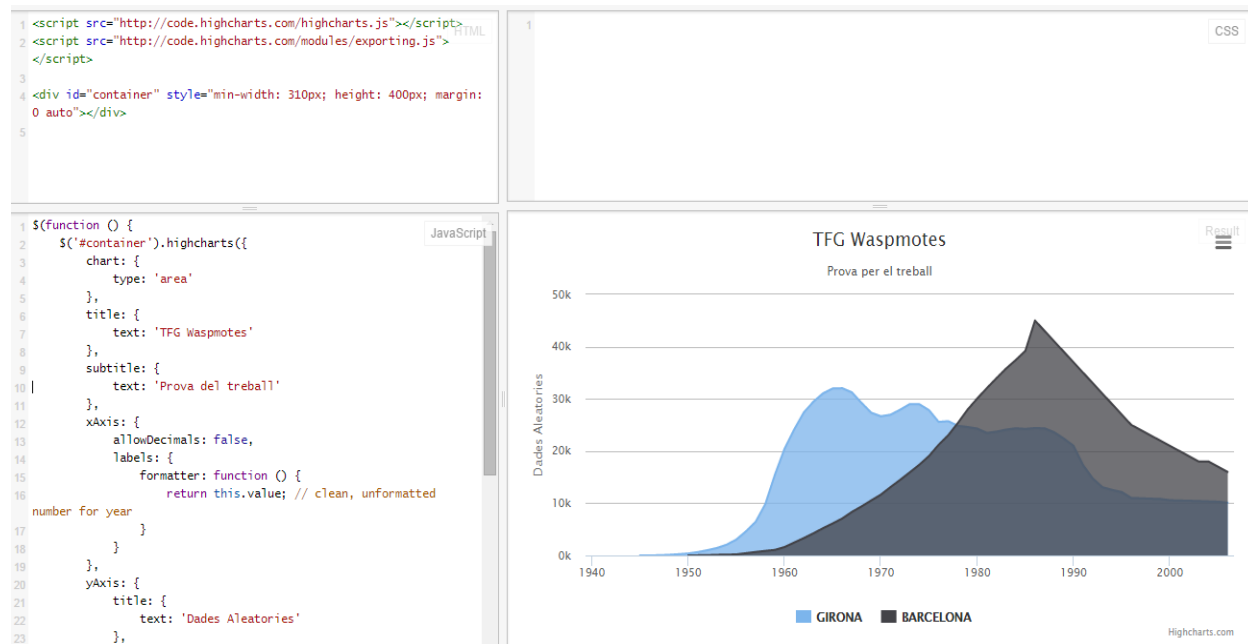
El resultat d'aquestes dades seria el següent:

July temperatures



Highcharts.com

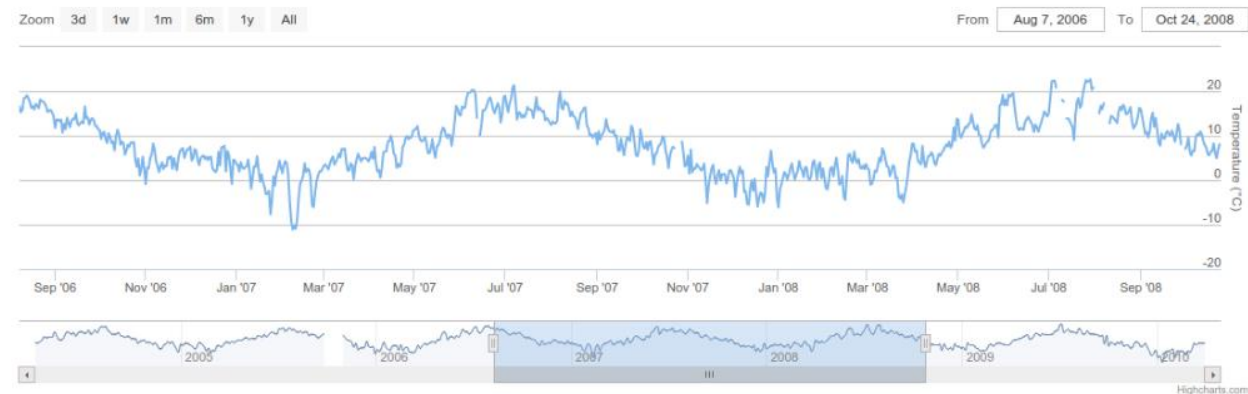
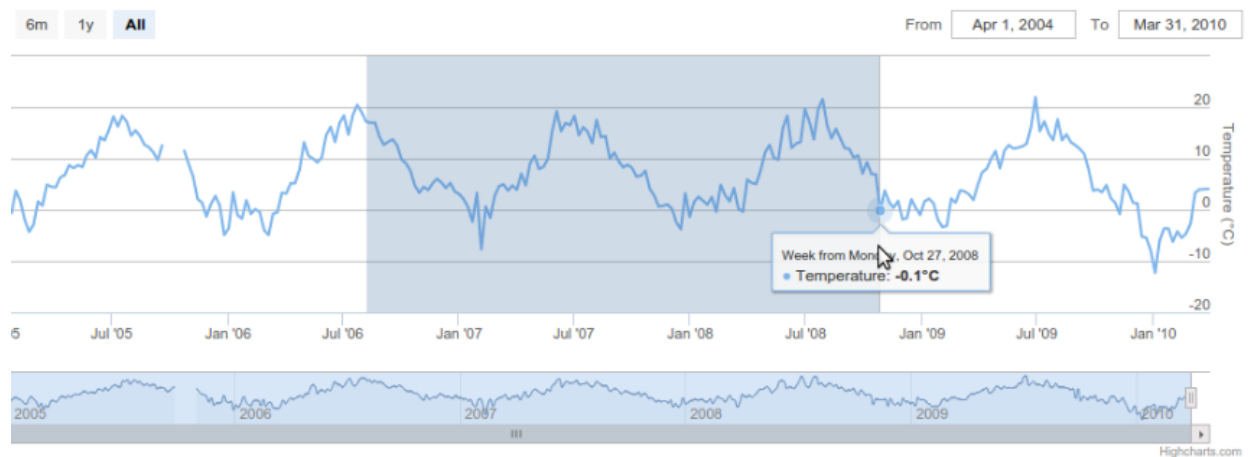
En la pàgina web de Highcharts en l'apartat demo <http://www.highcharts.com/demo> disposem d'una gran varietat d'exemples. A més a més, es poden fer proves i modificacions amb els exemples directament des de la web amb l'eina de jsfiddle que ens permet executar codi JavaScript.



Per extreure les dades de la base de dades simplement s'ha creat una classe amb PHP que es connecta a la base de dades i consulta les dades de cada dispositiu.

Per facilitar la visualització de les dades, en la web creada es mostra una llista de dispositius disponibles. Es consulta la base de dades i es mostra una llista de tots els dispositius disponibles identificats per la mac. En clicar en un dispositiu, se'n mostrarà dos gràfics, un per Temperatura i temps i l'altre de Bateria i Temps.

Highcharts ens permet tenir gràfics dinàmics i interactius, per exemple podem filtrar per dates, cosa que ens permet poder ampliar el detall dels gràfics i veure les dades per dates concretes. En aquest exemple podem veure com es pot filtrar per data a dalt a la dreta, o directament seleccionant la zona que volem ampliar en el gràfic.



8.4 Tractament de Dades

Les dades recollides pels Waspmotes serviran per poder analitzar consums energètics, possibles previsions i altres estudis.

Per facilitar aquestes tasques, es van afegir alguns requisits a les funcionalitats inicials del projecte.

Un punt important era que les dades fossin el més regular possible en el temps.

Cada trama de tots els dispositius s'envia un cop per hora. Com hem explicat anteriorment, les hores d'arribada de les trames s'arrodoneixen a l'hora més propera per tal d'intentar tenir una dada per hora per cada dispositiu.

En cas que s'esgotés la bateria d'algun Waspmote o hi hagués una caiguda de llum i la Raspberry s'apagués, es perdrien dades. És per això que a l'hora d'inserir cada dada, es comprova que l'anterior sigui de l'hora precedent i si no ho és, s'afegeixen trames amb tots els valors a 0 o Null per cada hora que falti.

El fet d'haver d'arrodonir les hores ens facilita aquesta tasca d'afegir trames perdudes, ja que és més fàcil comprovar quina és l'última dada i la seva hora d'inserció.

Per altra banda, per estudiar les dades, un membre de l'equip les exporta en format Excel. Per això es va afegir una funcionalitat a la web que mitjançant PHP permet exportar les dades directament en un arxiu Excel.

	A	B	C	D	E	F	G	H
1	id	mac	temp	bat	x	y	z	date
2	423	0013A200408C9D05	29	49%	16	-19	1049	2014-11-24 00:40:08
3	424	0013A200408C9D05	29	50%	16	-19	1049	2014-11-24 00:40:08
4	425	0013A200408C9D05	29	50%	16	-19	1048	2014-11-24 00:40:08
5	426	0013A200408C9D05	29	50%	16	-19	0	2014-11-24 00:40:08
6	427	0013A200408C9D05	29	49%	16	-19	0	2014-11-24 00:40:08
7	428	0013A200408C9D05	29	49%	16	-19	0	2014-11-24 00:40:08
8	429	0013A200408C9D05	29	50%	16	-18	0	2014-11-24 00:40:08
9	430	0013A200408C9D05	29	49%	29	-19	0	2014-11-24 00:40:08
10	431	0013A200408C9D05	29	50%	29	-19	0	2014-11-24 00:40:08
11	432	0013A200408C9D05	29	50%	30	-19	0	2014-11-24 00:40:08
12	433	0013A200408C9D05	29	50%	30	-20	0	2014-11-24 00:40:08

9. Implementació i proves

En aquest apartat es detallen els problemes i les solucions apareguts en el desenvolupament del projecte.

9.1 Aprenentatge i configuració dels Wasmotes

Un dels primers passos del projecte va ser familiaritzar-se amb els sensors sense fil i conèixer els diferents dispositius utilitzats en el projecte.

Als Wasmotes s'hi ha de carregar un codi per poder enviar les dades i configurar-los perquè puguin comunicar-se de forma segura amb la Raspberry i també s'hi poden afegir altres configuracions.

9.2 Parsejar les trames

Al principi també hi havien problemes amb el parseig de les dades.

No hi havia una estructura en la trama de dades que enviaven els Wasmotes. Això complicava bastant el tractament d'aquesta trama perquè qualsevol canvi en la trama comportava també un canvi en el codi del programa.

És per això que es va decidir establir un protocol per enviar les dades:

Es van establir uns caràcters d'inici i final (-# ... -#) i cada variable estava separada per un espai en blanc. Primer hi havia un '-' , després el nom de la variable, després ':' i seguidament la dada en si.

Exemple de trama: -# -Mac:00112233 -Temperatura:23 -Bateria:55 -#

D'aquesta manera es podien afegir i treure variables de la trama sense afectar el programa.

9.3 Gràfics de dades

Com s'ha explicat en l'apartat 7, a l'hora de fer els gràfics amb les dades, va haver-hi alguns problemes. Mai havia treballat amb gràfics i per tant no sabia quines eines eren les adequades per aquest projecte.

Els requeriments no eren gaire complexos, només s'havien de mostrar certes dades en funció del temps.

Després de buscar i provar diferents eines es va escollir Highcharts, ja que és una eina que cobria les nostres necessitats i ofereix una bona experiència a l'usuari.

Highcharts posa molts exemples a disposició dels desenvolupadors per tal de facilitar la feina i a més a més l'estructura del codi que s'utilitza és força entenedora.

Per extreure les dades de la base de dades s'han d'utilitzar altres eines com en aquest cas PHP, cosa que facilita bastant el procés de tractament de les dades.

9.4 Tractament posterior de les dades.

Una de les tasques que es va afegir, és la de detectar les trames que falten i afegir-les amb valors Null o a zero.

Pels estudis que es volen fer amb les dades, és convenient que l'hora de cada trama s'arrodoneixi a l'hora més propera. Per exemple si arriben dades d'un dispositiu a les 14:28, aquesta hora s'arrodonirà a les 14:00 i una trama que arribi a les 14:31 s'arrodonirà a les 15:00.

Les dades dels Waspmotes s'envien cada hora, i per tant si alguna trama no arriba, ja sigui perquè un Waspmote no té bateria o per algun error d'enviament, fins que no se solucioni el problema, faltaran dades.

Així que es va decidir afegir dades amb valors Nulls per cada hora que no hi hagin dades.

Com que es va prendre aquesta decisió un cop ja començat el projecte, vam tenir problemes per afegir aquesta nova funcionalitat.

Inicialment s'utilitzaven triggers de MySQL per poder rectificar les hores d'arribada de les dades.

Es va optar per fer-ho amb triggers perquè era una forma d'automatitzar el procés i assegurar-se que sempre s'actualitzaven les hores.

Per poder afegir dades en lloc de les trames perdudes ens vam trobar amb problemes per mantenir la consistència i ordre de les dades.

Al principi es va optar per fer-ho amb triggers però no va ser possible. Després es va intentar fer una part amb triggers i l'altre directament des del codi que llegeix les dades però també van sorgir problemes.

Finalment es van descartar els triggers i es va fer tot des de la part del codi.

9.5 Reemplaçar Dispositiu

Un punt important també era poder reemplaçar el dispositiu en cas d'avaria o qualsevol incident.

La Raspberry no disposa de disc dur, les dades i sistema operatiu son emmagatzemats en una targeta SD. Això també va ser un punt a favor d'utilitzar la Raspi, ja que en cas que es fes malbé, només cal reemplaçar-la, posar la mateixa targeta i tot segueix funcionant.

També facilita reemplaçar la targeta SD, ja que simplement cal fer-ne una còpia (una imatge de la SD) i copiar aquesta imatge en una nova targeta.

10. Implantació i resultats

10.1 Legislació i normativa vigent

El projecte desenvolupat no presenta cap problema en aspectes legislatius.

No s'ha tingut en compte la llei orgànica de protecció de dades de caràcter personal (LOPD), ja que el sistema en cap moment tracte cap tipus de dades relatives a l'usuari.

Des del punt de vista de seguretat, només s'ha afegit una autenticació per accedir a la web simplement per evitar que algú amb accés a la xarxa pugui consultar les dades sense permís.

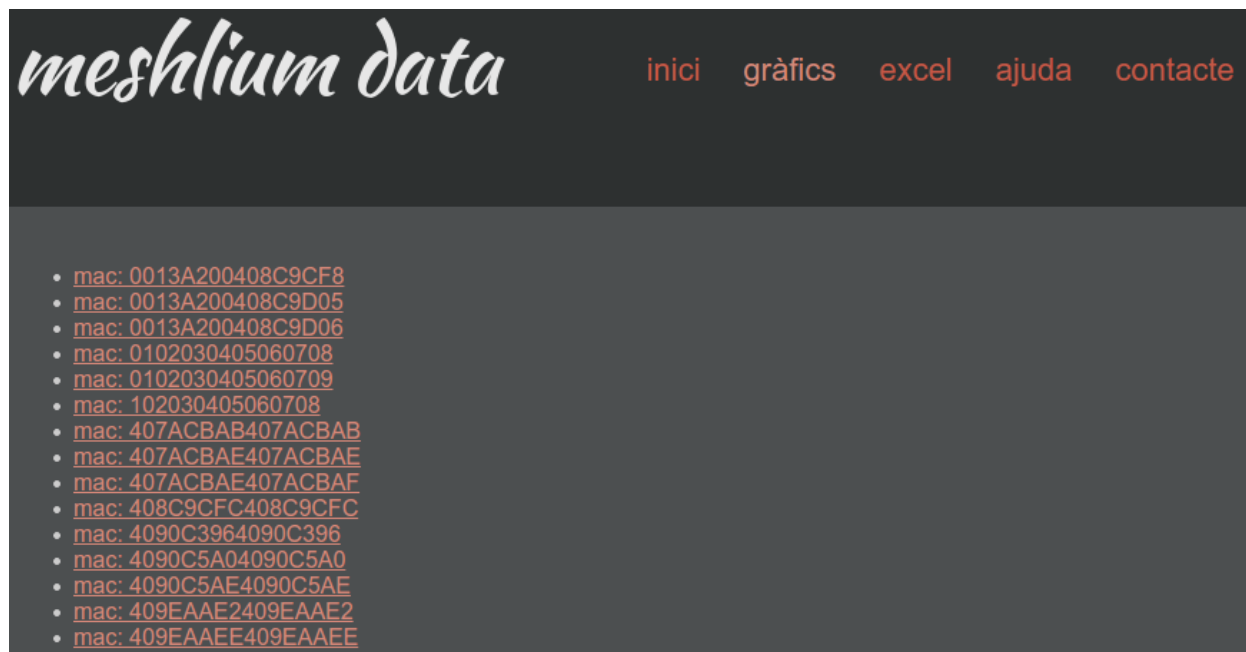
Aquest projecte no s'utilitzarà per treure'n benefici ni per cap finalitat lucrativa i per tant no hi ha conflictes amb el software utilitzat.

10.2 Resultats

Per les característiques del projecte és difícil demostrar els resultats.

En l'apartat 15 s'adjuntarà un manual detallat de tots els passos per instal·lar totes les parts del projecte. S'explica tot el software utilitzat, les llibreries necessàries i com instal·lar-ho tot.

A continuació mostrarem algunes captures de pantalla de la pàgina web i la base de dades perquè són els únics punts dels quals podem mostrar els resultats obtinguts.



En aquesta imatge podem veure com es llisten tots els dispositius disponibles a la base de dades.

En clicar en un d'ells, ens mostren els gràfics disponibles.



En clicar en l'apartat 'excel' ens descarrega un arxiu amb totes les dades disponibles.

	A	B	C	D	E	F	G	H
1	id	mac	temp	bat	x	y	z	date
2	423	0013A200408C9D05	29	49%	16	-19	1049	2014-11-24 00:40:08
3	424	0013A200408C9D05	29	50%	16	-19	1049	2014-11-24 00:40:08
4	425	0013A200408C9D05	29	50%	16	-19	1048	2014-11-24 00:40:08
5	426	0013A200408C9D05	29	50%	16	-19	0	2014-11-24 00:40:08
6	427	0013A200408C9D05	29	49%	16	-19	0	2014-11-24 00:40:08
7	428	0013A200408C9D05	29	49%	16	-19	0	2014-11-24 00:40:08
8	429	0013A200408C9D05	29	50%	16	-18	0	2014-11-24 00:40:08
9	430	0013A200408C9D05	29	49%	29	-19	0	2014-11-24 00:40:08
10	431	0013A200408C9D05	29	50%	29	-19	0	2014-11-24 00:40:08
11	432	0013A200408C9D05	29	50%	30	-19	0	2014-11-24 00:40:08
12	433	0013A200408C9D05	29	50%	30	-20	0	2014-11-24 00:40:08

Des de Phpmyadmin podem visualitzar les dades més còmodament, i podem veure com corresponen a les dades extretes de la web amb l'opció d'exportar en format Excel:

+ Opciones

← T →

					Id	mac	temp	bat	x	y	z	date
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	423	0013A200408C9D05	29	49%	16	-19	1049	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	424	0013A200408C9D05	29	50%	16	-19	1049	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	425	0013A200408C9D05	29	50%	16	-19	1048	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	426	0013A200408C9D05	29	50%	16	-19	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	427	0013A200408C9D05	29	49%	16	-19	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	428	0013A200408C9D05	29	49%	16	-19	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	429	0013A200408C9D05	29	50%	16	-18	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	430	0013A200408C9D05	29	49%	29	-19	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	431	0013A200408C9D05	29	50%	29	-19	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	432	0013A200408C9D05	29	50%	30	-19	0	2014-11-24 00:40:08
<input type="checkbox"/>	Editar	Editar en línia	Copiar	Borrar	433	0013A200408C9D05	29	50%	30	-20	0	2014-11-24 00:40:08

10.3 Arxius del projecte

En aquest punt descriurem els arxius del projecte. El directori principal Xbee conté 3 directoris:

-src

Aquí trobarem el codi necessari per llegir i desar les trames enviades pels Waspnotes.

Conté 3 arxius:

- xbee.py és el codi del programa.
- start.sh és un script que utilitzarem per iniciar el programa automàticament.
- variables.txt arxiu per configurar les variables que volem desar a la base de dades.

-web

En aquest directori trobarem tots els arxius necessaris per al funcionament de la web.

- En el subdirectori Highcharts hi trobarem les llibreries dels gràfics.
- El subdirectori images conté algunes imatges per la web.
- Index.php i grafics.php són les dues pàgines de la web, index.php és la primera pàgina que es mostra i grafics.php és la pàgina que ens llista tots els dispositius disponibles per veure'n els gràfics.
- DrawChart.php i charts.php s'encarreguen de mostrar els gràfics del dispositiu seleccionat.
- L'arxiu excel.php ens permetrà descarregar el contingut de la base de dades en format Excel.
- Finalment, tenim style.css i title.woff que configuren l'aspecte de la web i el tipus de lletra del títol de la web respectivament.

-waspmote ide

Aquest directori conté l'eina per configurar els Waspnotes.

11. Conclusions

11.1 Objectius

L'objectiu principal era el de substituir el Meshlium per un altre dispositiu de baix consum i de mida el més reduïda possible.

S'han complert tots els objectius proposats inicialment pel projecte. Com que disposàvem de temps suficient, es van ampliar els requeriments inicials.

Inicialment només es volia recollir i emmagatzemar les dades, però més endavant es va decidir també que seria convenient poder visualitzar i extreure les dades i facilitar-ne el posterior tractament.

Tots aquests objectius s'han complert.

11.2 Terminis Dels Objectius

Els terminis proposats durant el projecte s'han respectat tot i que en alguns casos s'han allargat o ajornat a causa dels canvis esmentats anteriorment.

11.3 Opinions Personals

Al llarg de la carrera aprenem a treballar en grup per fer projectes i sempre se'ns ha recalcat la importància del treball en equip i de complir els terminis proposats.

És cert, i s'ha pogut comprovar que és difícil posar-se d'acord i coincidir amb el client o persona que encarrega una feina o un projecte.

L'anàlisi dels requeriments d'una aplicació o projecte requereix un esforç per les dues parts, tant la part del client com la del desenvolupador del projecte. A vegades sembla que tothom s'ha entès, però després s'ha de refer o desfer part de la feina feta perquè les dues parts tenien percepcions diferents del mateix concepte i el resultat no és el desitjat.

Amb aquest projecte he après i reforçat tant, conceptes de programació i disseny com de coordinació i treball en equip.

Per altra banda el treball també m'ha permès conèixer més el món dels sensors i dels dispositius i xarxes sense fils. He descobert i entès la gamma de productes dels Waspnotes i mòduls Xbee i les possibilitats que ens ofereixen.

12. Treball futur

En aquest apartat es comentaran les possibles ampliacions, millores o treballs futurs que es podrien realitzar.

12.1 Avisos

Es podria implementar una aplicació que en cas de detecció de dades inusuals enviés un correu o ens ho notifiques d'alguna forma. Podria ser per internet per correu electrònic o per SMS amb la Raspi afegint-hi un mòdul especial.

També ens podria notificar en cas de caigudes de llum o si algun dispositiu té la bateria massa baixa, per tal de minimitzar la pèrdua de dades.

12.2 Identificació de Wasmotes

Els Wasmotes només s'identifiquen amb la seva MAC. Això és un inconvenient, ja que per veure la MAC s'ha de desmuntar l'aparell. En cara que s'etiquetessin per facilitar la identificació, un cop col·locats s'hauria de saber quin dispositiu està en cada punt de l'edifici. Es podria fer una aplicació per poder visualitzar i reconèixer els dispositius dins de l'edifici. A la mateixa pàgina web que utilitzem, mitjançant un mapa de l'edifici que es pogués col·locar cada Wasmote amb la seva MAC.

Com hem comentat en la introducció, aquest sistema es pot instal·lar en una ciutat o complex urbà i llavors es podrien visualitzar els dispositius en un mapa amb els seus identificadors.

12.3 Intervals d'enviament

Un dels punts del projecte era l'arrodoniment d'hores d'arribada de les trames a l'hora més propera. Es va especificar que aquest interval no canviaria i per tant no s'ha dissenyat perquè sigui ajustable sense haver de modificar el codi del programa. Tot i que en teoria aquest interval sempre hauria de ser el mateix, és convenient que aquest paràmetre sigui fàcilment modificable, ja que en un futur podria ser que s'hagués de canviar.

13. Bibliografia

www.digitalocean.com/community/tutorials/how-to-use-the-htaccess-file

www.wikipedia.com

www.highcharts.com

www.stackoverflow.com

www.python.org

www.raspberrypi.org

www.ubuntuforums.org

www.wiki.apache.org/httpd/PasswordBasicAuth

www.libelium.com/forum/

www.php.net/manual/es/

14. Manual d'usuari i/o instal·lació

14.1 Instal·lar sistema operatiu

El primer pas serà accedir a la web de Raspberry per descarregar l'última versió del Sistema operatiu.

Accedim a www.raspberrypi.org i busquem l'apartat de descàrregues.

<http://www.raspberrypi.org/downloads/>

Seguim les instruccions per instal·lar Raspbian, que serà el sistema operatiu que utilitzarem.

La forma més senzilla de fer l'instal·lació és amb NOOBS tal com ens ho expliquen a la web.

Seguidament adjuntem el manual de la web traduït al català:

DESCARREGAR

Utilitzant un ordinador amb un lector de targetes SD, visiteu la pàgina de descàrregues.

Feu clic al botó Baixa ZIP a "noobs (fora de línia i d'instal·lació de xarxa) ", i seleccioneu una carpeta per guardar-lo.

Traieu els arxius del fitxer comprimit.

Formatar la targeta SD

El millor és formatar la targeta SD abans de copiar els arxius NOOBS en la mateixa. Per a això: Visiteu el lloc web de l'Associació SD i descarregar formatador SD 4.0 (<https://www.sdcard.org/downloads>) per a Windows o Mac.

Seguiu les instruccions per instal·lar el programari.

Inseriu la targeta SD en el lector de l'ordinador o la targeta SD de l'ordinador portàtil i prengui nota de la lletra d'unitat assignada a la mateixa, per exemple, G: /

En SD formatador, seleccioneu la lletra d'unitat per a la targeta SD i formatar.

Copiar els arxius de NOOBS

Quan la targeta SD s'ha formatat, arrossegament tots els arxius al directori noobs els deixem anar en la unitat de targeta SD.

Els arxius necessaris seran transferits a la targeta SD.

Quan aquest procés hagi acabat, treure amb seguretat la targeta SD i inserir-la en el Raspberry Pi.

La primera arrencada

Connecteu els cables de teclat, ratolí i monitor.

Ara connecteu el cable d'alimentació USB al vostra Pi.

El seu Raspberry Pi arrencarà, i apareixerà una finestra amb una llista dels diferents sistemes operatius que es poden instal·lar. Us recomanem que utilitzeu Raspbian - marqui la casella al costat de Raspbian i feu clic a Instal·la.

Raspbian s'executarà a través del seu procés d'instal·lació. Nota que això pot portar un temps.

Quan el procés d'instal·lació s'ha completat, el menú de configuració del Raspberry Pi (Raspi-config) es carregarà. Aquí es podrà d'ajustar l'hora i la data per a la seva regió i crear els usuaris. Pot sortir d'aquest menú utilitzant Tab del teclat per moure a Finalitza.

Inici de sessió i accés a la interfície GRÀFICA D'USUARI

El nom d'usuari per defecte per Raspbian és **pi** amb contrasenya **raspberry**. Tingueu en compte que no veuran res escrit quan escriviu la contrasenya. Aquesta és una característica de seguretat en Linux.

Per carregar l'interfície gràfica d'usuari heu d'executar la comanda **startx**.

14.2. Actualitzar i preparar la Raspberry

Amb aquesta comanda podem canviar la contrasenya per defecte: **sudo passwd pi**

El següent pas serà connectar la Raspi a internet. Simplement connectem un cable ethernet al nostre router i comprovem que tenim connexió. Podem provar de fer un ping a google amb: **ping 8.8.8.8**

Executem les següents comandes per actualitzar la Raspberry.

```
sudo apt-get update  
sudo apt-get upgrade
```

També hem de configurar correctament l'hora, ja que serà la que utilitzarem per a les dades.

```
sudo dpkg-reconfigure tzdata Per configurar la data i hora.
```

Per defecte l'usuari administrador(root) no està habilitat, per habilitar-lo utilitzem aquesta comanda:

```
sudo passwd root (per habilitar l'usuari root) i introduïm la contrasenya.
```

14.3. Instal·lar Software

Ara ja tenim una Raspberry pi amb Linux(Raspbian) instal·lat.

Per defecte, python ja ve instal·lat. Podem comprovar la versió que te amb `python --version` per veure la versió que tenim. En aquest cas la versió 2.7.3.

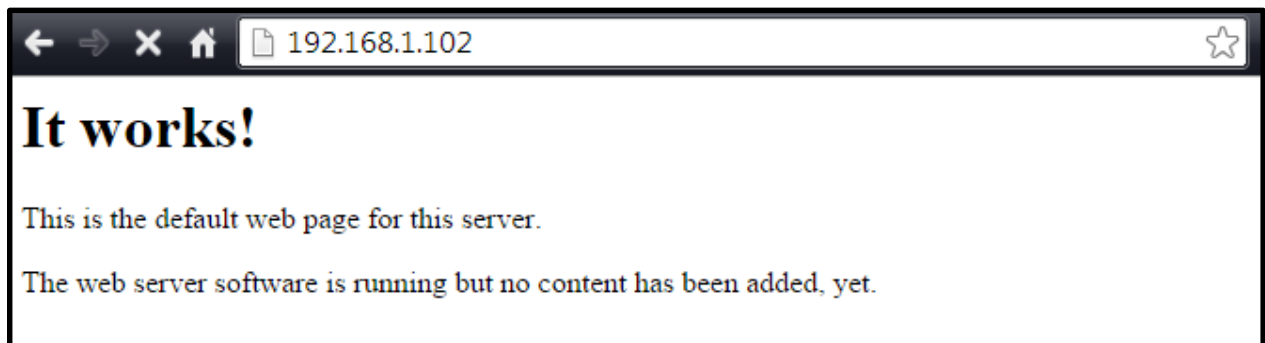
Instal·lem la llibreria per treballar amb la base de dades de MySQL : `sudo apt-get install python-mysqldb`

A continuació Instal·lem un servidor LAMP(Linux Apache MySQL PHP):

Amb la comanda `sudo apt-get install apache2 php5 libapache2-mod-php5` instal·lem Apache i PHP.

Executem `sudo service apache2 restart` per comprovar que s'ha instal·lat correctament, haurem de saber la ip de la Raspberry. Ho podem veure amb `ifconfig` o amb la comanda `ifconfig eth0 | grep inet | awk '{ print $2 }'`

Des de qualsevol ordinador (dins de la mateixa xarxa que la raspi) posem aquesta ip al navegador i hauríem de veure això:



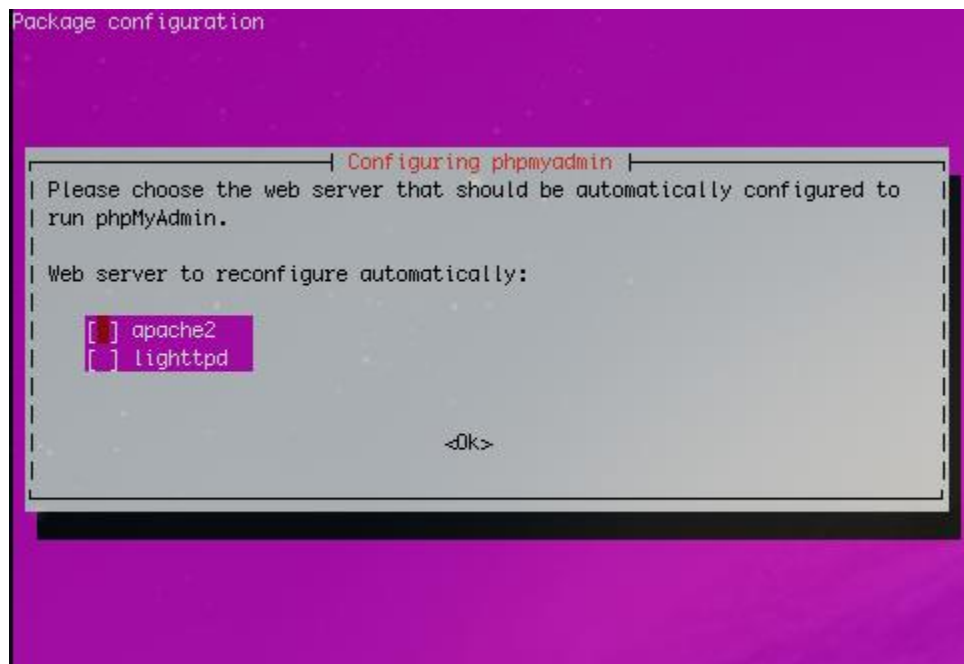
També necessitem MySQL:

`sudo apt-get install mysql-server mysql-client php5-mysql` i seguim les instruccions de l'instal·lador.

Per facilitar la gestió de la base de dades utilitzarem l'eina PhpMyadmin

`sudo apt-get install phpmyadmin`

Segurament ens preguntin quin servidor web utilitzarem, llavors seleccionem apache2 i continuem.



Ara necessitem configurar l'Apache per què reconegui el phpmyadmin:

sudo nano /etc/apache2/apache2.conf per editar el fitxer de configuració i afegim **Include /etc/phpmyadmin/apache.conf** al final del fitxer. Premem **CTRL + X** per sortir de l'editor i **Y** per desar els canvis.

Finalment reiniciem Apache amb: **sudo service apache2 restart**

Per comprovar que funciona correctament, des d'un navegador accedim a **ipRaspberry/phpmyadmin** en aquest cas 192.168.1.102/phpmyadmin/



← → ↻ 🏠 192.168.1.102/phpmyadmin/ ☆

phpMyAdmin

Bienvenido a phpMyAdmin

Idioma - *Language*

Español - Spanish ▼

Iniciar sesión ⓘ

Usuario:

Contraseña:

Continuar

Podrem accedir amb l'usuari i contrasenya que hem configurat a l'hora d'instal·lar phpmyadmin.

En aquest cas:

Usuari: root

Contrasenya: raspberry

Finalment, per facilitar la transferència d'arxius a la Raspi, instal·larem **vsftpd** per poder transferir arxius per FTP.

Primer hem de canviar els permisos del directori web `sudo chown -R pi /var/www`

Instalem **vsftpd** amb la comanda següent:

```
sudo apt-get install vsftpd
```

i editem el seu arxiu de configuració:

```
sudo nano /etc/vsftpd.conf
```

Busquem i editem les següents línies:

reemplaçem ***anonymous_enable=YES*** per ***anonymous_enable=NO***,

Descomentem ***local_enable=YES***

Descomentem ***write_enable=YES***

al final de l'arxiu afegim ***force_dot_files=YES***

*per descomentar, esborrem el símbol(#) davant de la línia.

Premem CTRL + X per sortir de l'editor i Y per desar els canvis.

14.4. Configurar entorn

Ara haurem de crear un usuari de MySQL i una base de dades.

Ho farem mitjançant Phpmyadmin encara que també es podria fer des del terminal.

Accedim a phpmyadmin amb l'usuari i contrasenya.



The screenshot shows the phpMyAdmin interface for a MySQL server on localhost. The 'Privilegios' tab is selected, displaying the 'Vista global de usuarios' (Global user view). The interface includes a navigation menu on the left with options like 'information_schema', 'mysql', 'performance_schema', and 'phpmyadmin'. The main content area shows a table of users with columns for 'Usuario', 'Servidor', 'Contraseña', 'Privilegios globales', 'Conceder', and 'Acción'. Below the table, there is a link to 'Agregar un nuevo usuario' (Add a new user).

Usuario	Servidor	Contraseña	Privilegios globales	Conceder	Acción
<input type="checkbox"/> debian-sys-maint	localhost	Si	ALL PRIVILEGES	Si	Editar los privilegios Exportar
<input type="checkbox"/> phpmyadmin	localhost	Si	USAGE	No	Editar los privilegios Exportar
<input type="checkbox"/> root	127.0.0.1	Si	ALL PRIVILEGES	Si	Editar los privilegios Exportar
<input type="checkbox"/> root	:::1	Si	ALL PRIVILEGES	Si	Editar los privilegios Exportar
<input type="checkbox"/> root	localhost	Si	ALL PRIVILEGES	Si	Editar los privilegios Exportar
<input type="checkbox"/> root	raspberrypi	Si	ALL PRIVILEGES	Si	Editar los privilegios Exportar

[↑ Marcar todos / Desmarcar todos](#)

[Agregar un nuevo usuario](#)

Des de la pestanya **Privilegios** afegirem un nou usuari i també crearem una nova base de dades:

Agregar un nuevo usuario

Agregar un nuevo usuario

Información de la cuenta

Nombre de usuario: Use el campo de te ▾ xbee

Servidor: Local ▾ localhost 1

Contraseña: Use el campo de te ▾

Debe volver a escribir:

Base de datos para el usuario

Ninguna

Crear base de datos con el mismo nombre y otorgar todos los privilegios

Otorgar todos los privilegios al nombre que contiene comodín (username_%)

En aquest cas creem un usuari **xbee** amb contrasenya **xbee1234** i seleccionem l'opció de crear una base de dades per l'usuari amb el mateix nom i privilegis i premem el botó de **Crear Usuari**.

Ara crearem l'estructura de la base de dades **xbee**.

Primer sortim de phpmyadmin i tornem a entrar amb el nou usuari **xbee** i contrasenya **xbee1234**.

Creem la taula **dades**:

Primer seleccionem la base de dades **xbee** a l'esquerra.



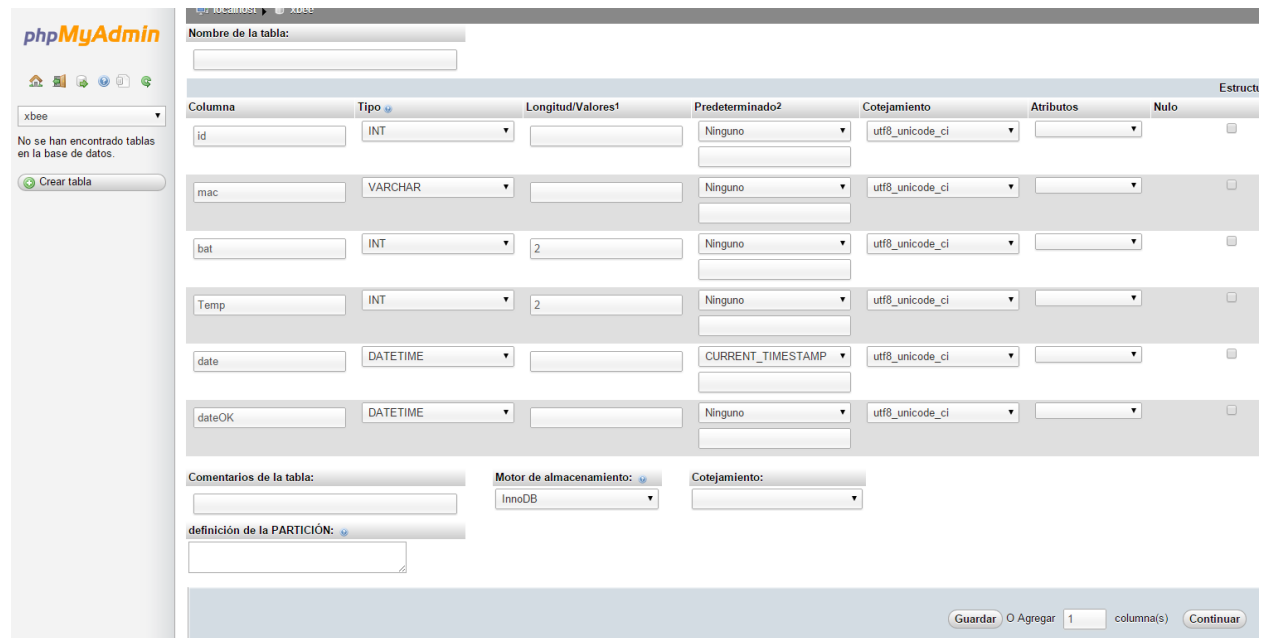
L'opció més senzilla és importar l'arxiu '**dades.sql**' que proporcionem.

Això ho farem des de la pestanya **Importar** i després seleccionem l'arxiu.

L'altre opció és fer-ho des de phpmyadmin.

Seleccionem la base de dades **xbee** a l'esquerra i creem una taula nova.

Creem una taula nova dades amb les columnes necessàries.



Finalment tindrem la taula **dades** creada:

The screenshot shows the phpMyAdmin interface for a MySQL database. The left sidebar shows the navigation tree with 'xbee' selected and 'dades' as the current table. The main area displays the table structure for 'dades' with the following columns:

#	Columna	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
<input type="checkbox"/>	1	id			No	Ninguna	AUTO_INCREMENT	Cambiar Eliminar Más ▼
<input type="checkbox"/>	2	mac	utf8_unicode_ci		No	Ninguna		Cambiar Eliminar Más ▼
<input type="checkbox"/>	3	bat			No	Ninguna		Cambiar Eliminar Más ▼
<input type="checkbox"/>	4	Temp			No	Ninguna		Cambiar Eliminar Más ▼
<input type="checkbox"/>	5	date			No	CURRENT_TIMESTAMP		Cambiar Eliminar Más ▼
<input type="checkbox"/>	6	dateOK			No	0000-00-00 00:00:00		Cambiar Eliminar Más ▼

Below the table structure, there are options to 'Marcar todos / Desmarcar todos' and actions like 'Examinar', 'Cambiar', 'Eliminar', 'Primaria', and 'Único'. At the bottom, there is an 'Agregar' section with a dropdown menu set to 'id' and a 'Continuar' button.

14.5. Seguretat

Afegirem un control d'usuaris per accedir a la web i les dades.

Crearem un usuari **xbee** que tindrà accés a la web.

```
sudo mkdir /etc/htpasswd/  
sudo htpasswd -c /etc/htpasswd/passwords xbee
```

introduïm la contrasenya i confirmem.

Ara hem d'habilitar l'autenticació pels usuaris creats:

```
nano /etc/apache2/sites-enabled/000-default
```

busquem:

```
<Directory /var/www/>  
  Options Indexes FollowSymLinks MultiViews  
  AllowOverride None  
  Order allow,deny  
  allow from all  
</Directory>
```

i canviem **AllowOverride None** per **AllowOverride All**.

A continuació accedim al següent directori `cd /var/www/` i creem un arxiu de configuració `nano .htaccess` i hi copiem això:

```
AuthUserFile /etc/htpasswd/passwords  
AuthGroupFile /dev/null  
AuthName "Please Enter Password"  
AuthType Basic  
Require valid-user
```

Finalment reiniciem Apache: `sudo service apache2 restart`.

Ara si accedim a la web, ens hauria de demanar un usuari i contrasenya.

Per afegir usuaris simplement tornarem a executar:

```
sudo htpasswd /etc/htpasswd/passwords xbee
```

(en comptes de **xbee** al final, posarem el nom del nou usuari)

14.6. Carregar el projecte

Copiem el directory xbee del projecte al directori `/var/www/` de la Raspi.

Amb un client FTP com per exemple FileZilla carregarem els arxius.

Dins del directori **xbee/src/** hi ha l'arxiu 'start.sh' per iniciar el programa. Perquè s'iniciï automàticament el programa en encendre o quan es reiniciï la Raspberry o en cas de caiguda de llum, haurem d'afegir una línia en un arxiu de la Raspberri.

Executem la comanda `sudo nano /etc/rc.local` per obrir el fitxer rc.local. Aquest fitxer s'executa sempre a l'inici de la Raspberry.

Haurem d'afegir la següent línia `/var/www/xbee/src/start.sh` al final del fitxer just abans de `exit 0`

En aquest cas, l'arxiu rc.local queda així:

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
    printf "My IP address is %s\n" "$_IP"
fi

/var/www/xbee/src/start.sh

exit 0
```

El programa crearà l'arxiu 'log.txt' on es desa informació del que fa el programa.

Per evitar que l'arxiu contingui masses dades, l'esborrarem cada cert temps. Per realitzar aquesta tasca utilitzarem el CRON.

Executem `crontab -e` i al final de l'arxiu afegim la següent línia:

```
0 0 * * 0 echo Reset log: $(date) > /var/www/xbee/src/log.txt
```

Els primers 5 caràcters indiquen quan s'ha d'executar la comanda.

Minut	Hora	Dia del Mes	Mes	Dia de la Setmana	Comanda
(0-59)	(0-23)	(1-31)	(1-12 o Jan-Dec)	(0-6 o Sun-Sat)	

Per tant amb '0 0 * * 0' indiquem que s'executi a les 00:00 del diumenge.

Minut	Hora	Dia del Mes	Mes	Dia de la Setmana	Comanda
0	0	*(tots els dies)	*(tots els mesos)	0(diumenge)	comanda a executar.