



Anàlisis i implementació de solució domòtica domèstica low-cost

Daniel Casas Iglesias
Grau Informàtica
Direcció de Projectes

Professor col·laborador: Xavier Martínez Munné
Professor responsable de l'assignatura: Atanasi Daradoumis Haralabus

01/2019



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Anàlisi i implementació de solució domòtica domèstica low-cost</i>
Nom de l'autor:	<i>Daniel Casas Iglesias</i>
Nom del consultor/a:	<i>Xavier Martínez Munné</i>
Nom del PRA:	<i>Atanasi Daradoumis Haralabus</i>
Data de lliurament (mm/aaaa):	<i>01/2019</i>
Titulació o programa:	<i>Grau Informàtica</i>
Àrea del Treball Final:	<i>Direcció de Projectes</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>domòtica, low-cost, integració</i>
Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i>	
<p>La finalitat d'aquest projecte es analitzar i implementar un sistema de domòtica capaç de poder ser controlat des de una consola central web, missatges de veu o text amb l'objectiu de demostrar com poden interactuar entre ells dispositius de diferents fabricants i tecnologies.</p> <p>També es defineixen un conjunt d'escenaris relativament complexos que s'executen automàticament en funció de diferents desencadenadors.</p>	

Abstract (in English, 250 words or less):

The aim of this project is to analyze and implement a system to control several domotic objects from a centralized web console, voice or text messages to demonstrate how they interact together regardless the technology or brand.

There will be also a set of scenarios defined relatively complex that will be executed automatically depending on several triggers

Índex

1. Introducció	1
1.1. Context i justificació del Treball	1
1.2. Objectius del Treball	2
1.3. Enfocament i mètode seguit	3
1.4. Planificació del treball	5
1.5. Breu sumari de productes obtinguts	6
1.6. Breu descripció dels altres capítols de la memòria	6
2. Introducció als protocols i tecnologies	7
2.1. X10	7
2.2. Zwave	7
2.3. Zigbee	8
2.4. Bluetooth	8
2.5. Wifi	9
3. Anàlisi de components	10
3.1. Mesuradors de temperatura	11
3.2. Commutadors elèctrics	11
3.3. Activadors de persianes	12
3.4. Detectors d'estats de portes	12
3.5. Sensors de moviment	13
3.6. Sensors d'inundació	13
3.7. Activadors de caldera	13
3.8. Activadors d'aire condicionat	14
3.9. Conclusions	15
4. Anàlisi de la consola central	16
4.1. OpenHAB	16
4.2. Domoticz	17
4.3. HomeAssistant	18
4.4. Conclusions	18
5. Anàlisi del motor de processament	20
5.1. OpenHAB	20
5.2. IFTTT	20
5.3. NodeRED	21
5.4. Conclusió	22
6. Instal·lació	23
6.1. Ecosistema Xiaomi (Temperatura, moviment, inundació i portes)	23
6.2. Sonoff (Control de llum)	24
6.3. Netatmo (Termòstat)	26
6.4. Broadlink RM Mini 3 (Aire condicionat)	27
6.5. Configuració de xarxa	29
7. Prerequisits	30
7.1. Ecosistema Xiaomi	30
7.2. Sonoff	30
7.3. Netatmo	31
7.4. Broadlink RM Mini 3	32

8.	Configuració consola central.....	33
8.1.	Terminologia d'OpenHAB	33
8.2.	Convenció de noms	34
8.3.	Instal·lació	35
8.4.	Configuració de bindings, things i items	37
8.5.	Sumari d'Items.....	41
9.	Configuració de panell web.....	42
9.1.	Tipus d'elements.....	42
9.2.	Panell operatiu.....	43
9.3.	Panell de control de bateries	44
10.	Configuració del motor de processament.....	46
10.1.	Funcionament general de Node-RED	46
10.2.	Instal·lació.....	47
10.3.	Configuració dels connectors.....	48
11.	Definició d'escenaris complexos	51
12.	Control per missatgeria	54
12.1.	Elements passius.....	54
12.2.	Elements actius.....	54
13.	Conclusions.....	56
14.	Glossari.....	58
15.	Bibliografia	59

Llista de figures

Il·lustració 1 - Planificació del treball	5
Il·lustració 2 - Taula comparativa de sensors de temperatura	11
Il·lustració 3 - Taula comparativa commutadors elèctrics	11
Il·lustració 4 - Taula comparativa detectors d'estat de porta	12
Il·lustració 5 - Taula comparativa sensors de moviment	13
Il·lustració 6 - Taula comparativa sensors d'inundació	13
Il·lustració 7 - Taula comparativa dispositius controladors de temperatura	14
Il·lustració 8 - Taula comparativa d'activadors d'aire condicionat	14
Il·lustració 9 - Llistat definitiu de productes del sistema	15
Il·lustració 10 - Mida del gateway en comparació amb una moneda	23
Il·lustració 11 - Esquema connectivitat Xiaomi	24
Il·lustració 12 - Aplicació MiHome configurada	24
Il·lustració 13 - Esquema inicial de connectivitat Sonoff	25
Il·lustració 14 - Esquema final de connectivitat Sonoff	26
Il·lustració 15 - Esquema connectivitat Netatmo	27
Il·lustració 16 - Esquema de connectivitat Broadlink RM Mini 3	27
Il·lustració 17 - Exemple de comandament d'aire condicionat	28
Il·lustració 18 - Assignació d'IPs fixes als dispositius de domòtica	29
Il·lustració 19 - Configuració Xiaomi per accés en LAN	30
Il·lustració 20 - Esquema funcionament del protocol MQTT	31
Il·lustració 21 - Esquema funcionament de la API de Netatmo	31
Il·lustració 22 - Comanda per extreure els codis del dispositiu d'infrarojos	32
Il·lustració 23 - Esquema dels components d'OpenHAB	34
Il·lustració 24 - Plana inicial d'OpenHAB	36
Il·lustració 25 - OpenHAB amb tots els bindings instal·lats	37
Il·lustració 26 - Fitxer de comandes d'infrarojos	37
Il·lustració 27 - Configuració Binding Broadlink	38
Il·lustració 28 - Configuració Binding Xiaomi	38
Il·lustració 29 - Sensors de Xiaomi dins OpenHAB	39
Il·lustració 30 - Configuració Binding de Netatmo	40
Il·lustració 31 - Dispositius Netatmo després de configurar la API	40
Il·lustració 32 - Elements Zwave	41
Il·lustració 33 - Sumari d'elements de tipus Item del sistema	41
Il·lustració 34 - Panell de control de la interfície PaperUI	42
Il·lustració 35 - Configuració HabPanel - Objecte Dummy	43
Il·lustració 36 - Panell operatiu	44
Il·lustració 37 - Configuració d'un element de tipus bateria	45
Il·lustració 38 - Panell de control de bateries	45
Il·lustració 39 - Exemple fluxe a Node-RED	46
Il·lustració 40 - Aspecte de la interfície NodeRED	47
Il·lustració 41 - Cerca de connectors a NodeRED	48
Il·lustració 42 - Configuració OpenHAB a Node-RED	48
Il·lustració 43 - Llistat de Things disponibles a Node-RED	49
Il·lustració 44 - Generació bot a Telegram	50
Il·lustració 45 - Gestió de caldera amb Node-RED	51
Il·lustració 46 - Gestió d'aire condicionat amb Node-RED	52
Il·lustració 47 - Gestió de llums amb Node-RED	52
Il·lustració 48 - Control amb Telegram a Node-RED	55

1. Introducció

1.1. Context i justificació del Treball

“Oye Siri, abre la persiana de la habitación principal”. Sembla ciència-ficció però no. En el dia d'avui, qualsevol persona pot demanar a un assistent personal que interactuï amb elements de la casa, de la mateixa manera que demana que es programi una alarma per despertar-se a les 7 del matí, o que llegeixi les notícies més rellevants del dia. Ara, l'usuari pot demanar al seu dispositiu mòbil d'executar tasques rutinàries del món real, amb l'objectiu d'aconseguir un grau de control més elevat dins el seu dia a dia.

La domòtica ha evolucionat molt durant els darrers anys i es situa com a una tecnologia accessible i que de ben segur revolucionarà la vida de molta gent. Enrere queden les èpoques on calia fer grans inversions adquirint centraletes propietàries, que només podien ser configurades per personal altament qualificat. Enrere queda la necessitat de cablejar de nou tota la casa per només instal·lar-hi un parell de sensors, que indicaven si la porta principal era oberta o si es detectava moviment en una instància particular.

Avui en dia, existeixen un gran nombre de fabricants que competeixen per produir productes de domòtica econòmics i de gran qualitat, i el més important, que aquests puguin ser configurats per usuaris inexperts. Aquests productes van des de sensors capaços de mesurar amb gran exactitud paràmetres com ara temperatura, humitat o lluminositat fins a activadors de calderes o equips d'aire condicionat. Només cal que l'usuari connecti l'aparell a la xarxa wifi, obrir l'aplicació específica al mòbil i configurar les propietats de l'habitable al seu gust.

Ara, amb una petita inversió es poden tenir sensors i actuadors per tota la casa i gestionar tots els elements des del telèfon mòbil. Hi ha productes que permeten configurar l'aire condicionat, d'altres que regulen la temperatura del termòstat, d'altres que informen si les plantes de la terrassa necessiten aigua...

Un moment... potser s'ha passat d'un extrem a l'altra. Potser ara existeix un control absolut de molts elements de casa però aquests no interactuen entre ells, creant un conjunt d'ecosistemes molt potents però tancats i que no poden respondre davant d'escenaris complexos tècnicament parlant però habituals. I si l'usuari vol que cada cop que arribi a casa s'encengui el llum del rebedor i alci la persiana però només si és de nit i és un dia no festiu?

1.2. Objectius del Treball

A grans trets, l'objectiu d'aquest treball és mostrar algunes de les possibilitats que la domòtica pot oferir sense la necessitat de grans inversions. També pretén investigar la viabilitat d'integrar productes de diferents fabricants i tecnologies sota un mateix controlador que permeti la interacció i la creació d'escenaris relativament complexos.

A continuació es llisten els objectius detallats del projecte.

- Configurar una consola web on tenir visibilitat centralitzada sobre els següents elements de la casa:
 - Temperatura de 3 habitacions.
 - Estat del llum del menjador i llum exterior (apagat/encès).
 - Estat d'una persiana (abaixada/pujada).
 - Estat de la porta principal (tancada/oberta).
 - Estat de la nevera (tancada/oberta).

- Aconseguir que un usuari pugui realitzar les següents accions des de la mateixa consola web:
 - Encendre i apagar els llums esmentats anteriorment.
 - Pujar i abaixar una persiana.
 - Encendre la calefacció i configurar una temperatura concreta.
 - Encendre l'aire condicionat a una temperatura concreta.

- Demostrar que les accions descrites anteriorment es poden executar utilitzant la veu i també missatges de text.

- Executar els següents escenaris automàticament:
 - Encendre la calefacció en cas que la temperatura sigui inferior a 21 graus.
 - Encendre l'aire condicionat en cas que la temperatura sigui superior a 27 graus.
 - Apagar i encendre un llum situat al pati durant 5 minuts en cas que es detecti moviment al voltant.
 - Enviar una notificació al mòbil en cas d'inundació.

El cost total de tots els components involucrats en el projecte no pot superar els 300 euros.

1.3. Enfocament i mètode seguit

Primerament es defineixen el conjunt de tasques que s'hauran de realitzar durant l'execució del projecte, tenint en compte l'esforç que caldrà dedicar per a cadascuna d'elles. En aquesta planificació es tenen en compte les diferents fites i també vacances.

Seguidament, s'analitzen els diferents components hardware del sistema: sensors que poden reportar paràmetres i elements que poden prendre accions com encendre llums, aire condicionat... La finalitat d'aquesta anàlisi és trobar aquells que permetin assolir els objectius del treball de la manera més ràpida i econòmica possible.

De manera general, s'optarà sempre per opcions econòmiques, ja que existeix una limitació de costos per demostrar que es pot construir un sistema de domòtica amb una inversió molt baixa.

Sempre que es pugui, s'optarà per solucions ja disponibles al mercat donat que no es vol construir cap sensor des de zero per motius de seguretat i també per solucions open Source reconegudes per la comunitat.

Un cop decidits els components, s'ha de fer el mateix exercici d'anàlisi amb la consola central software encarregada de rebre la informació dels sensors tenint en compte que ha de ser compatible amb els sensors. S'analitzaran productes Open Source existents, crear un programari que permeti controlar tots els sensors és una tasca massa complexa, poden donar-se molts problemes de codi que posarien en risc la viabilitat del projecte. A més, aquest treball pretén ser un projecte d'integració de sistemes, no de programació.

Finalment, s'analitzaran diferents motors de processament que permetin executar els escenaris definits dins els objectius. Pel mateix motiu esmentat anteriorment, s'optarà per solucions Open Source.

Un cop decidits els components software i hardware, s'adquirirà tot el material necessari. Donat que hi ha elements que tenen un cost relativament elevat, es pot optar per adquirir productes en el mercat de segona mà, aquests components no tenen un desgast durant el temps d'ús més enllà de la bateria incorporada en molts d'ells que és fàcilment reemplaçable.

Després es configuraran cadascun dels elements seguint les instruccions del fabricant. Depenent de la tecnologia, només caldrà connectar el dispositiu per USB a l'ordinador central o a la xarxa wifi però en certs casos, pot ser necessària una configuració extra com ara canvis en el firmware dels sensors per poder interactuar millor amb ells o registres en la web del fabricant per poder accedir a les APIs.

En paral·lel a la configuració del hardware, s'ha d'instal·lar la consola central i el motor de processament en un ordinador i assegurar que aquesta pot tenir visibilitat bàsica sobre els sensors.

Un cop tot està configurat, caldrà modificar la consola central per poder agrupar en ella tots els elements del sistema de domòtica i crear una única interfície des d'on tenir visibilitat sobre els valors dels sensors.

Es molt important tenir en compte l'escalabilitat del sistema de manera que es puguin afegir nous components de manera senzilla. S'haurà de definir una nomenclatura per identificar tots els paràmetres i també identificar els processos d'entrada i sortida per a poder parametritzar-ho correctament després.

A mesura que els diferents sensors estiguin integrats dins la consola central, es parametritzarà el motor de processament amb els escenaris definits en els objectius definits en el punt 1.2 i es configurarà el motor per tal que sigui capaç de respondre davant missatges de veu i text.

En la darrera fase de l'execució del projecte com a tal, es definiran i executaran els jocs de proves que el sistema haurà de superar d'acord als objectius i s'aplicarà també control d'errors i alertes en cadascun dels elements com ara bateria baixa o desconnexió d'algun element.

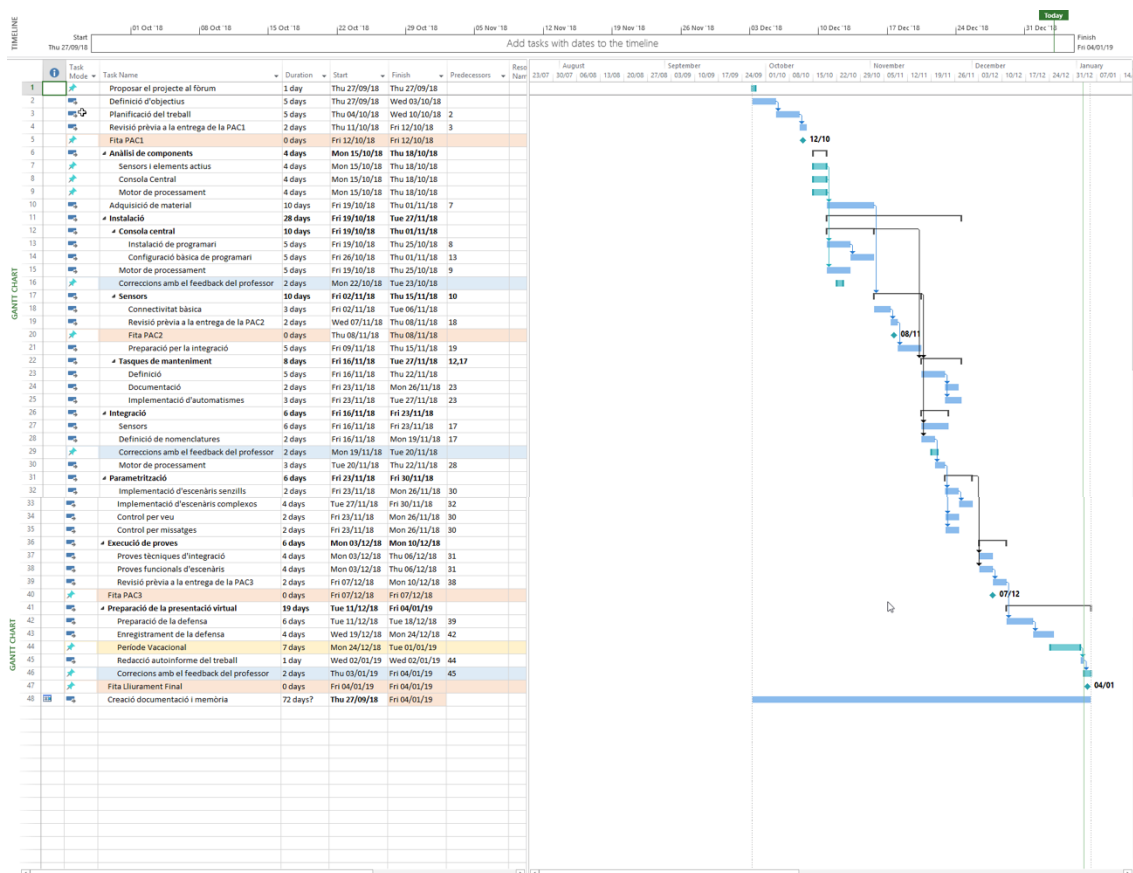
Per últim, es prepararà la presentació virtual en tots els seus aspectes: preparació de la defensa, enregistrament i redacció d'autoinforme del treball.

1.4. Planificació del treball

Existeixen tres fites, una per cada entrega de les dues PAC referents al pla de treball i una tercera que marca l'entrega de la presentació virtual. Les fites vénen marcades en el diagrama de Gantt en color taronja.

Tot i que la memòria es realitzarà durant tot el treball, es reserven uns quants dies abans d'aquestes fites per tal de revisar la memòria. També es reserva temps després de la publicació dels comentaris per a refer allò que el professor tutor consideri oportú.

Per últim, esmentar que en la planificació es preveu una setmana de vacances per Nadal que apareix indicada en color groc. Tota la informació queda reflectida en la II·lustració 1.



II·lustració 1 - Planificació del treball

1.5. Breu sumari de productes obtinguts

D'una banda, s'obindrà en una interfície web que permeti llegir els valors dels sensors i prendre accions sobre certs elements de la casa com ara persianes o equips d'aire condicionat.

D'altra banda, disposarà d'un sistema de notifikacions que permetrà a l'usuari ser capaç de llegir els valors dels sensors i controlar els elements mitjançant missatges de veu i text.

Per últim, gràcies a la integració d'un motor de processament, es podran declarar escenaris complexos des d'una interfície web que s'executaran en funció de diversos desencadenants com ara valors específics dels sensors (assolir certa temperatura, detectar moviment...).

1.6. Breu descripció dels altres capítols de la memòria

Anàlisi de components

S'analitzaran un conjunt de components de diferents fabricants amb els seus pros/cons i es decidiran quins formaran part del sistema.

Anàlisi de la consola central

S'analitzaran les principals consoles centrals de domòtica opensource seguint diversos criteris per tal de determinar la millor solució.

Anàlisi del motor de processament

S'analitzaran els principals motors de processament en funció dels requeriments.

Instal·lació

Es detallarà la instal·lació de tots els components involucrats en el sistema amb les seves particularitats i es configuraran perquè funcionin correctament.

Integració

Es mostrarà els passos a seguir per tal d'integrar tots els components dins la consola central i poder definir escenaris.

Parametrització

Es mostrarà com es configura el motor de processament per a gestionar els escenaris establerts com a objectius del projecte.

2. Introducció als protocols i tecnologies

Abans d'entrar en detall dels components, cal familiaritzar-se amb els protocols i tecnologies més habituals que es poden trobar en el món de la domòtica. De manera general, es parla de protocols creats i dedicats a la domòtica i d'altres protocols estàndards que poden també gestionar elements domòtics.

2.1. X10

X10 és un protocol creat el 1978 i utilitza la xarxa elèctrica convencional per a transmetre informació. Està pensat per gestionar únicament dispositius en mode Apagar/Encendre, i també regular intensitat en cas d'utilitzar-se amb llums que permetin aquesta funcionalitat. Aquest tipus de protocol requereix un mòdul controlador central que serà l'encarregat d'enviar les ordres als diferents dispositius.

Durant els darrers anys i a causa de l'aparició de noves tecnologies, ha quedat relegat a instal·lacions industrials amb usos molt específics on es requereix una gran estabilitat.

2.2. Zwave

Aquest protocol propietari purament sense fils va ser definit el 2001 per l'empresa Zensys, opera en diferents bandes segons la regió del món on es comercialitza (13 regions diferents). Aquest últim fet és molt important perquè si s'adquireix un dispositiu als Estats Units, pot no funcionar a Europa (la freqüència pot estar sent utilitzada per altres aparells). Tot i això, val a dir que molts dispositius permeten configurar la freqüència a nivell software, es poden configurar per treballar en una regió diferent a la que inicialment anaven destinats.

Zwave ha estat des dels seus inicis un protocol molt utilitzat i fins i tot es va crear la ZWave Alliance, un consorci d'empreses que vetllen per la interoperabilitat entre fabricants i s'encarreguen de millorar el protocol en tots els seus aspectes. El 2013, aquesta associació va decidir treure una nova versió, Z-Wave Plus que millora la gestió de bateria dels dispositius en un 50%, incrementa el rang d'abast fins a 30 metres i augmenta l'amplada de banda en un 250%. Cal recalcar que aquesta revisió manté la retro-compatibilitat amb dispositius de la generació anterior.

A grans trets, la virtut més gran d'aquest protocol és el fet que cada dispositiu actua també com a repetidor, creant un sistema en malla entre tots els accessoris. Això ajuda a millorar la cobertura sense necessitat de dispositius dedicats a repetir el senyal. Existeix un màxim de 232 nodes per xarxa i aquests no han d'estar connectats directament a la central, la informació viatja d'un node a l'altra fins a trobar el seu destinatari. Aquest protocol en concret té però una limitació, la comunicació no funcionarà si s'han de fer més de 4 salts entre dispositius per arribar al destí.

A diferència de X10, Zwave no es limita a enviar ordres d'encès i apagat. Actualment es troben al mercat un gran nombre de dispositius dedicats a gestionar infinitat de recursos, com ara mesura de temperatura, humitat, moviment, detecció de portes, gestió de persianes... El punt fort de Zwave és la seva expansió. Actualment es poden trobar al mercat més de 2400 productes. Els dispositius no són econòmics però molts d'ells incorporen més d'un sensor o actuator pel mateix preu, fet que pot abaratir el cost del sistema.

Tota la informació referent a la sincronització amb el sistema central queda emmagatzemada en la memòria, no es perd quan s'ha de reemplaçar el sistema d'alimentació. El seu consum és molt baix i una pila de botó pot proveir l'energia suficient per alimentar el dispositiu durant dos anys.

Per últim, en l'àmbit de la seguretat cal recalcar que la comunicació entre dispositius utilitza un xifrat AES de 128 bits.

2.3. Zigbee

Zigbee ha set sempre el gran competidor de Zwave. Està recolzat per la Zigbee alliance, amb membres tan coneguts com Huawei, Somfy o Texas Instruments. A diferència de Zwave, Zigbee és un protocol lliure. Com a tal, qualsevol usuari pot descarregar les especificacions i modificar el contingut segons els seus requeriments. La idea és bona, però genera en conseqüència la possibilitat que els dispositius no siguin operables entre si. Com en el cas de Philips amb el seu producte Philips Hue, cada fabricant pot optar per evolucionar el codi i fer que només sigui compatible amb els seus propis dispositius.

Igual que en el cas de Zwave, els dispositius es connecten en malla. A diferència del primer, no hi ha limitació en el nombre de salts fins a arribar a destí. Tampoc existeixen limitacions en el nombre de dispositius, ja que es pot arribar a un màxim teòric de 65.000. El xifrat de la comunicació és també AES de 128 bits i permet un intercanvi d'informació a velocitats 6 vegades superiors que Zwave.

Aquest protocol treballa en la mateixa freqüència a tot el món, 2.4Ghz i aquest és també el seu gran problema. A més de l'evident saturació de la banda utilitzada per dispositius wifi, tot i poder transmetre més quantitat de dades que Zwave, el seu abast és molt inferior, es recomana no superar mai els 10 metres entre dispositius.

2.4. Bluetooth

Des dels seus inicis l'any 1994, aquest protocol es va convertir en l'ideal per a gestionar xarxes personals (Personal Area Network), gràcies al seu baix consum i la seva flexibilitat. Al llarg del temps s'han realitzat diverses revisions del protocol i en la seva última versió (5), és capaç de gestionar fins a 2Mb per segon d'informació.

El seu baix consum el fa ideal per a elements de domòtica, però com que no pot treballar en malla com Zigbee o Zwave i també degut al seu limitat rang de connectivitat (10 metres), fa difícil la seva implantació.

2.5. Wifi

Per últim, no pot faltar la definició del sistema sense fils per excel·lència analitzat ara com a protocol de domòtica, la xarxa wifi. No cal entrar gaire en detall, és àmpliament utilitzat en molts perifèrics electrònics. La seva versatilitat el fa també apte per a gestionar accessoris de domòtica, tot i que el consum és elevat i en la majoria d'ocasions cal alimentació externa dedicada.

A més, la saturació de les freqüències (2.4Ghz i 5.0Ghz) dificulta la creació de sistemes robustos i es pateixen desconexions temporals.

Tot i que en els darrers mesos s'ha començat a parlar de les xarxes wifi en malla i ja existeixen productes com ara Google Wifi o els encaminadors del fabricant Asus, aquesta tecnologia encara no està completament implantada.

2.6. Conclusions

El protocol X10 queda descartat per aquest treball, no admet gaire tipus de dispositius i molt menys dispositius d'ús domèstic. El rang de connectivitat limitat del protocol Bluetooth fa que aquest tampoc sigui el més adient per a controlar elements situats a tota la casa, només al costat d'un ordinador que pugui actuar com a receptor.

El funcionament en malla de Zwave i Zigbee fa que aquests protocols siguin els més idonis, a més dispositius, millor la cobertura. El cost de zwave és més elevat a conseqüència del llicenciamnt vinculat a un protocol tancat, però porta més temps al mercat i existeixen més dispositius.

Per últim, l'opció de dispositius wifi també és vàlida, la cobertura de senyal a tot l'habitatge es força bona. No es recomana abusar d'aquesta tecnologia, els dispositius requereixen alimentació externa però per contra, el cost és contingut.

3. Anàlisi de components

Un cop introduïdes les diferents tecnologies i donats els objectius esmentats en la secció 1.2, es farà una cerca de tots els components que formaran part del sistema de domòtica que encaixin dins el pressupost màxim de 300 euros. S'ha d'intentar sempre que siguin dispositius alimentats amb bateries per no haver de fer grans modificacions en la instal·lació elèctrica.

Com a requeriment addicional, tots els elements que formin part de l'anàlisi han de permetre la connectivitat amb dispositius externs, de manera que es pugui interactuar amb ells sense dependre únicament d'una única aplicació propietària. Segons aquest factor, es classificaran en:

- Completa: integració sense cap mena de modificació.
- Relativa: calen petites configuracions per aconseguir una integració òptima.
- Complexa: cal modificar el sensor físicament o seguir instruccions addicionals complexes.

De manera general, es realitzarà una explicació del tipus de dispositiu que es cerca amb el seu principi de funcionament. Es mostrarà una taula amb els detalls més rellevants on apareixerà ressaltat el producte escollit i en la part inferior, es realitzarà una justificació de l'elecció.

En alguns casos els dispositius necessiten un *gateway* encarregat de gestionar el seu funcionament i centralitzar la recepció de la informació. Aquests *gateways* poden incorporar també sensors i funcionalitats extres i es detallen a continuació juntament amb el cost.

- Xiaomi MiJia Smart Home Gateway (29 €)
Tots els dispositius de la marca Aquara funcionen amb tecnologia Zigbee però no permet l'ús amb altres *gateways* externs. Aquest *gateway* envia les dades dels sensors als servidors centrals de Xiaomi a Internet on es processen, però també exposen un punt a la xarxa local des d'on es poden recollir les dades internament.

El dispositiu com a tal disposa de sensors de lluminositat, altaveu i també un llum RGB fet que el fa molt versàtil.

- Zwave USB (30 €)
Si es vol incorporar elements Zwave, es necessita un dispositiu capaç de recollir les dades i convertir-les per poder ser processades des d'un ordinador. Aquest perifèric amb terminació USB té la mida d'un *pendrive* i la seva configuració sol ser molt senzilla. Només cal prémer el botó de sincronització tant en el *gateway* com en el dispositiu final, la comunicació s'establirà i quedarà xifrada. No conté cap sensor addicional.

3.1. Mesuradors de temperatura

Es cerquen dispositius capaços de mesurar la temperatura ambient i reportar les variacions a una font de dades externes en temps real. Molts d'aquests dispositius incorporen altres sensors que s'hauran de valorar en funció dels requeriments i permetran un estalvi respecte al nombre global de perifèrics.

Nom	Protocol	Alimentació	Gateway	Interconnectivitat	Cost	Altres Sensors
Broadlink A1	TCP	USB	No	Relativa	35€	Qualitat de l'aire, Sensor de llum, sensor de gas, soroll, humitat
Aeotec Multisensor 6	Zwave	Piles	Si	Completa	49€	Moviment, lluminositat, humitat, vibració, llum ultra violeta
Aqara Temperature Sensor	Zigbee	Piles	Si	Completa	8€	Humitat
Xiaomi Temperature Sensor	Bluetooth	Piles	No	Completa	15€	Humitat
Fibaro	Bluetooth	Piles	Si	Completa	20€	N/A

Il·lustració 2 - Taula comparativa de sensors de temperatura

Donats tots els productes, s'optarà per triar el dispositiu del fabricant Aqara. Tot i necessitar un *gateway* central, s'obté un producte de molta qualitat a un preu molt baix, encaixant en els requeriments del projecte i creant una xarxa en malla amb altres sensors.

3.2. Commutadors elèctrics

Molts dels controladors que es poden trobar al mercat es presenten en format endoll de corrent. En aquest cas, només caldrà connectar el dispositiu que es vulgui controlar al controlador i aquest directament a l'endoll domèstic.

Tot i la seva aparent facilitat d'ús, es prendrà la decisió de no incloure aquest tipus de dispositius en l'anàlisi ja que es volen controlar llums domèstics que treballen directament amb la xarxa elèctrica, sense endoll propi. L'alimentació del dispositiu evidentment serà sempre proporcionada pel corrent elèctric.

Nom	Protocol	Gateway	Interconnectivitat	Cost	Comentaris
Sonoff Basic	TCP	No	Complexa	5€	Cal canviar el firmware
Shelly 1	TCP	No	Relativa	10€	Cal canviar el firmware
Fibaro Single Switch 2	Zwave	Si	Completa	48€	
Xiaomi Switch	Zigbee	Si	Completa	21€	Integrat en l'interruptor

Il·lustració 3 - Taula comparativa commutadors elèctrics

Presentades les diferents opcions, s'escollirà l'opció de Sonoff Basic. Tot i que presenta una interconnectivitat complexa, hi ha molta documentació a Internet i és l'opció més econòmica. Dit això, s'ha valorat l'opció de Xiaomi Switch però implica canviar tot l'interruptor, desentonant amb la resta d'interruptors disponibles a l'habitatge.

3.3. Activadors de persianes

En el cas dels activadors de persianes, cal esmentar que es parteix de la base que aquestes tenen ja el motor incorporat. El seu funcionament es bàsicament el mateix que faria l'interruptor, tancar el circuit elèctric.

Actualment no hi ha gaire opció al mercat domèstic disponible. Només el fabricant Fibaro ofereix el seu model FGR-222 compatible amb la tecnologia Zwave i té un cost aproximat de 46 euros.

El mòdul en si té unes dimensions molt reduïdes i està pensat per ser encastat darrere l'interruptor que controla la persiana. Com que l'alimentació la proporciona la mateixa xarxa elèctrica, no caldrà reemplaçar cap pila i l'interruptor normal també es podrà utilitzar.

Per últim, esmentar que aquesta elecció implica també l'adquisició del mòdul *gateway* de Zwave que es comenta en l'apartat 2. És un cost afegit que s'haurà de tenir en compte.

3.4. Detectores d'estats de portes

Els detectors d'estat de portes o finestres de qualsevol mena són dispositius senzills. Estan formats per dues peces imantades que es col·loquen una a cada cantó de la porta i reporten si estan unides o no. Existeixen molts fabricants però la majoria no permeten interconnectivitat entre els dispositius, queden descartats. Es detallen a continuació aquells que si ho permeten.

Nom	Protocol	Gateway	Interconnectivitat	Cost	Comentaris
Fibaro FGDW-002-3	Zwave	Si	Completa	42€	
Aeon Labs ZW089	Zwave	Si	Completa	38€	Cal fer un forat i introduir-ho dins la porta
Aqara Door sensor	Zigbee	Si	Completa	11€	

Il·lustració 4 - Taula comparativa detectors d'estat de porta

En aquest cas, es triarà l'opció d'Aqara pel seu baix cost, la seva facilitat d'instal·lació i el reduït volum. Incorpora un adhesiu que permet enganxar el dispositiu en qualsevol mena de material com ara fusta o ferro i que assegura que no es mourà.

3.5. Sensors de moviment

Aquests sensors incorporen un conjunt de leds infrarojos que detecten el moviment. Tenen un rang de detecció d'uns 5 metres i 150 graus. Emmagatzemen també l'últim instant en el qual s'ha detectat moviment.

Nom	Protocol	Gateway	Interconnectivitat	Cost	Altres sensors
Fibaro FGMS-001	Zwave	Si	Completa	58€	
Aeotec Multisensor 6	Zwave	Si	Completa	49€	Moviment, lluminositat, humitat, vibració, llum ultra violeta
Aqara Motion Sensor	Zigbee	Si	Completa	9€	

Il·lustració 5 - Taula comparativa sensors de moviment

Igual que en el punt anterior, s'optarà per la solució de baix cost i reduït volum d'Aqara. El sensor Aeotec es una bona opció però, tot i proporcionar molts mes sensors, té un cost massa elevat.

3.6. Sensors d'inundació

El principi d'un sensor d'inundació es basa en dos punts metàl·lics tanquen un circuit elèctric en cas que hi hagi aigua entre ells.

Nom	Protocol	Gateway	Interconnectivitat	Cost	Detalls
Fibaro Flood Sensor	Wifi	No	Completa	49€	Incorpora sensor de temperatura, alarma sonora i lluminosa
Aqara Flood Sensor	Zigbee	Si	Completa	8€	

Il·lustració 6 - Taula comparativa sensors d'inundació

La decisió ha set senzilla, un cop mes, Aqara proporciona la mateixa funcionalitat a un preu més baix.

3.7. Activadors de caldera

La majoria de calderes domèstiques permeten ser activades des d'un punt remot. Bàsicament proporcionen dos terminals que en entrar en contacte entre ells, tanquen el circuit elèctric i la caldera comença a funcionar. Aquests terminals solen ser gestionats des de dins de l'habitatge on es col·loca un termòstat.

Inicialment, es van instal·lar activadors analògics que incorporaven una roda on l'usuari marcava la temperatura desitjada. Gràcies a un termòmetre incorporat, si la temperatura on estava l'activador era inferior a la desitjada, simplement posava en contacte els dos terminals del cable.

Amb el pas dels anys, els fabricants van incorporar diverses millores com ara dispositius que permetien la creació de programes a mida. L'objectiu era que l'usuari pogués configurar la temperatura de l'habitació en funció de les hores. Actualment, aquests tipus de controladors tenen un cost molt reduït i es poden trobar al mercat per un preu aproximat de 20 Euros.

Donat que aquest tipus de dispositius permeten optimitzar el consum de la llar, el retorn de la inversió és molt baix, molts fabricants han decidit comercialitzar productes avançats dins el món de la domòtica. Tot i tenir un cost més elevat que un termòstat tradicional, surt a compte.

El 100% dels dispositius es gestionen mitjançant internet. No existeix la possibilitat de contactar amb cap d'ells directament dins la xarxa local. Aquest fet fa que es depengui de la disponibilitat d'APIs i també d'Internet per interactuar amb els dispositius de l'habitatge internament. També s'ha de tenir en compte que les dades estaran allotjades a Internet.

Nom	Protocol	Gateway	Interconnectivitat	Cost	Comentaris
Netatmo Termostat	TCP	No	Relativa	160€	
Nest	TCP	No	Relativa	219€	Incorpora sensor de proximitat
Honeywell T6	TCP	No	Relativa	151€	
Tado V3	TCP	No	Relativa	249€	
Momit	TCP	No	Relativa	93€	

Il·lustració 7 - Taula comparativa dispositius controladors de temperatura

En aquest cas, s'optarà pel sensor de Netatmo. El seu cost és moderat i incorpora una funcionalitat que cap altre fabricant ofereix, poder moure el termòstat de posició.

El sistema està format per un termòstat encarregat de captar la temperatura actual, amb un parell de botons per a configurar la temperatura desitjada manualment. Aquest termòstat funciona amb piles i es connecta per ones de ràdio amb un *Relay*, que és el que realment activa la caldera i reporta els valors a Internet.

El fet de separar el termòstat de l'activador i poder canviar-ho d'ubicació fa que es pugui moure a qualsevol habitació. El sistema prendrà aquella cambra com a referència en la determinació de temperatura ambient. Això, és especialment útil en llars on els terminals de la caldera estan col·locats en una zona molt freda o calenta, en comparació amb la resta d'habitacions.

3.8. Activadors d'aire condicionat

L'aire condicionat es controla en la majoria d'instal·lacions mitjançant un comandament de radiofreqüència. Cada vegada que l'usuari prem una tecla, el comandament envia la instrucció al dispositiu, aquest la interpreta segons com ha estat configurat.

Els dispositius domòtics emulen aquest funcionament. Tenen una gran base de dades que conté totes les instruccions dels principals fabricants. L'usuari, selecciona l'adient i el mòdul s'encarrega de descarregar els codis. En cas que no existeixin, pot també llegir el codi des del comandament original utilitzant un mode d'aprenentatge.

Nom	Protocol	Gateway	Interconnectivitat	Cost	Comentaris
Sensibo	TCP	No	Completa	119€	
Tado air conditioner	TCP	No	Completa	180€	
Broadlink RM Mini	TCP	No	Relativa	20€	

Il·lustració 8 - Taula comparativa d'activadors d'aire condicionat

En aquest cas, l'elecció serà senzilla. Tot i disposar de dispositius dedicats a la gestió de l'aire condicionat, el seu cost és massa elevat per a justificar la seva adquisició. Per només 20 Euros, Broadlink RM Mini és capaç de memoritzar les instruccions de radiofreqüència de qualsevol comandament i reproduir-les independentment del tipus de dispositiu.

Amb el mateix aparell es pot en un futur, gestionar qualsevol dispositiu que treballi amb radiofreqüència, com ara equips de música o televisors.

3.9. Conclusions

Es mostren a continuació el llistat de productes escollits, això com el nombre d'unitats necessàries, el seu preu unitari i també el cost del producte.

Producte	Tecnologia	Unitats	Preu Unitari	2a mà
Xiaomi Gateway	Wifi	1	29€	
Zwave USB	Zwave	1	30€	20€
Aqara Temperature Sensor	Zigbee	3	8€	
Sonoff Basic	Wifi	2	5€	
Fibaro FGR-222	Zwave	1	46€	30€
Aqara Door Sensor	Zigbee	2	11€	
Aqara Motion Sensor	Zigbee	1	9€	
Aqara Flood Sensor	Zigbee	1	8€	
Netatmo Thermostat	Wifi	1	170€	100€
BroadLink RM Mini	Wifi	1	20€	

II·l·lustració 9 - Llistat definitiu de productes del sistema

Tal com es pot apreciar, el preu total del material és d'aproximadament 368 Euros, però es poden reduir costos adquirint aquells components més cars en el mercat de segona mà. En el cas del projecte, s'aconsegueix una rebaixa substancial, marcant un preu final de 272 Euros incloent els *gateways*.

A més, com que molts dels dispositius incorporen més d'un sensor, s'adquiriran les funcionalitats extres detallades a continuació, que no s'inclouen en els requeriments inicials i permetran ampliar l'abast del sistema en un futur:

- Sensors d'humitat proporcionat per Aqara Temperature Sensor
- Sensor de lluminositat proporcionat per Xiaomi Gateway
- Llum RGB proporcionat per Xiaomi Gateway
- Alarma proporcionat per Xiaomi Gateway

4. Anàlisi de la consola central

Un cop analitzats els diferents components, es procedirà a decidir quin serà el software que centralitzarà la gestió dels diferents elements de domòtica. De manera generalitzada, les característiques que s'analitzaran i es tindran en compte a l'hora d'escollir la consola seran:

- Quantitat de dispositius compatibles o “plugins”
- Facilitat de configuració
- Facilitat d'ús
- Gestió d'usuaris
- Accessibilitat des d'Internet
- Accessibilitat des de dispositius mòbils
- Possibilitat d'integració amb altres sistemes

4.1. OpenHAB

Open Home Automation Bus (OpenHAB) és un ecosistema OpenSource basat en Java desenvolupat el 2010 per Kai Kreuzer, amb la intenció d'agrupar sota un mateix software diferents components de domòtica domèstica. Ofereix compatibilitat amb dispositius de més de 300 fabricants i es molt ràpida la integració de nous components.

OpenHAB basa el seu funcionament en una arquitectura modular que permet incorporar nous elements en calent. Tots els components es connecten a OpenHAB Event Bus, encarregat de gestionar totes les comunicacions, sia entre dispositius o també internes com ara accions fetes per l'usuari en la UI, ordres d'apagar o encendre... .

La seva configuració és senzilla i permet adaptar-se a les necessitats de diferents tipus d'usuaris. Es pot optar simplement per instal·lar el producte i configurar integracions, fent ús d'una interfície web d'usuari, o parametritzar tot el sistema amb l'ajuda de fitxers de text amb l'objectiu d'acotar més l'abast de la plataforma.

Aquesta flexibilitat per contra comporta certa incertesa per part dels usuaris avançats que no acaben de trobar la millor manera de gestionar-ho tot plegat. Existeixen tres interfícies d'usuari diferents, però moltes opcions de configuració només estan disponibles en només una d'elles. OpenHAB és conscient del problema i està treballant a unificar-ho tot, però encara no hi ha un compromís real d'arreglar-ho per una data determinada.

Tant la web de configuració com la interfície d'usuari estan construïdes amb el framework de Google Material Design. Aquest fet comporta que es pugui interactuar amb els components tant des d'un ordinador com des del mòbil. Tot i això, existeixen aplicacions natives tant d'Android com de IOS que proporcionen una millor experiència d'usuari i funcionalitats específiques com ara notificacions.

OpenHAB deixa molt clar que no consideren la gestió de permisos una prioritat. La seva filosofia és que existeixen components com ara Apache o Nginx que ofereixen aquest servei molt millor del que ells podrien fer mai. Dit això, no existeix cap mena d'usuari i contrasenya per accedir a la consola en local ni perfils, si es té accés a la web dins la xarxa local, es pot modificar el que hom vulgui.

Com a contrapartida, OpenHAB proporciona una plataforma gratuïta pròpia anomenada MyOpenHab. Permet accedir al sistema des d'Internet sense necessitat d'obrir comunicacions al nostre encaminador cap a la instància local. Aquest sistema, sí que permet la creació de perfils i especificar diferents tipus d'usuaris.

MyOpenHab és una part clau de l'ecosistema OpenHAB, ja que permet la integració amb components externs com ara IFTTT, Alexa o Google Home. Esmentar finalment que es poden realitzar filtres, per especificar quins components es volen fer accessibles des d'Internet, afegint una capa extra de protecció.

4.2. Domoticz

Domoticz va ser creat el 2012 per un usuari anomenat Guizmocuz després de comprovar que la seva plataforma de domòtica no acomplia les seves expectatives. El seu nucli i interfície web estan desenvolupats en C++ i és compatible amb Linux i Windows. Tot i l'optimització de recursos que comporta l'ús de C++, aquest llenguatge no està gaire estès actualment, implica que la comunitat de desenvolupadors serà més reduïda. Durant els darrers anys, Domoticz ha començat a treballar en un conjunt de components basats en Python que faciliten la integració de nous perifèrics a més alt nivell.

Comparat amb OpenHAB, no té tants plugins i la integració es limita a comandes relativament senzilles sense possibilitat d'interacció real amb sistemes complexos, com ara els serveis de veu d'Alexa o Google. Per contra, afegir i treure components són operacions molt senzilles i no cal reiniciar res en cap moment.

La seva interfície web resulta peculiar. Tot i poder aplicar "skins" per a personalitzar-la a gust del consumidor, no està tan cuidada com la d'OpenHAB. De cara a l'usuari final, no dóna gaire opció pel que fa a situar elements en posicions particulars o afegir icones pròpies.

Ofereix una gestió d'usuaris robusta, on es poden definir diversos tipus de perfils. Tota la informació d'usuaris i configuracions es guarden en una base de dades pròpia SQLite. Es pot modificar amb qualsevol programari compatible amb aquesta tecnologia. Aquesta possibilitat comporta per contra molta fragilitat, un error pot fer que el sistema deixi de funcionar.

Igual que OpenHAB, també disposa d'un servei Cloud que permet interactuar amb l'ecosistema des d'internet. Val a dir però que no es garanteix l'alta disponibilitat. Està en fase beta des de fa un parell d'anys i tampoc disposa

d'APIs per integració amb tercers. Com a punt positiu, recalcar que es pot aplicar doble factor d'autenticació, opció no disponible en cap dels altres productes analitzats.

4.3. HomeAssistant

Aquesta plataforma desenvolupada en Python 3 és relativament nova i la seva primera versió va ser publicada el 2015. La comunitat d'usuaris encara és reduïda, però creix exponencialment fet que de ben segur comportarà moltes millores en un temps relativament curt.

La seva estratègia de distribució es basa a simplificar al màxim les tasques d'instal·lació. Existeixen versions específicament optimitzades per sistemes encastats tipus Raspberry, que faciliten enormement la inclusió de connectors (Hass.io). També es poden descarregar els binaris i instal·lar tot manualment. En aquest últim cas però, molts dels connectors no estan disponibles o requereixen configuració addicional.

Es tracta d'un sistema molt orientat a xarxes locals, l'única manera d'accedir des d'internet és mitjançant l'obertura de ports a l'encaminador. Ofereixen un servei per 5 dòlars mensuals, que permet integrar la plataforma amb Alexa i Google Home però no permet cap altra tipus d'integració.

Tota la configuració es basa en fitxers YAML, fet que ofereix una gran flexibilitat i permet restaurar el sistema en cas d'error a baix nivell, només copiant un conjunt de fitxers en una carpeta específica. Per contra, queda reservat el seu ús a usuaris avançats.

En l'àmbit de la gestió d'usuaris, l'eina és molt completa, permet gestionar diferents perfils d'usuaris i fins i tot crear-ne de nous amb permisos específics sobre dispositius concrets. Es poden també establir grups amb visibilitat limitada a un conjunt de sensors.

4.4. Conclusions

Després d'analitzar les principals solucions de domòtica OpenSource, s'optarà per escollir OpenHAB. La seva comunitat ha construït un producte que cobreix pràcticament totes les necessitats que qualsevol usuari pugui tenir. Existeixen connectors per gairebé qualsevol dispositiu i en el pitjor dels casos, permeten també cridar comandes externes directament des de scripts en qualsevol llenguatge que el servidor sigui capaç d'interpretar.

Durant els darrers anys la domòtica està evolucionant molt i l'arribada a Espanya d'Alexa i Google Home pot canviar el mercat en qualsevol moment. La plataforma MyOpenHAB ofereix la flexibilitat necessària avui però també en un futur, sense renunciar mai a la privacitat. Evidentment les dades són processades per OpenHAB en els seus servidors però no té res a veure amb el que pot passar si acaben en mans de Google o Amazon.

Finalment, la disponibilitat d'aplicacions dedicades per iOS i Android ha jugat un paper important en la decisió. Les interfícies web han evolucionat molt però encara no ofereixen la mateixa experiència d'usuari, tampoc permeten rebre notificacions com si es pot fer amb aplicacions natives.

5. Anàlisi del motor de processament

En aquesta darrera fase, es cercarà la millor opció en l'àmbit del motor de processament. Aquest serà l'encarregat de modelar i gestionar tots els escenaris definits en els objectius del projecte. El software ha de ser evidentment compatible amb OpenHAB i ha de permetre la lectura de valors així com l'activació de perifèrics.

Igual que s'ha fet amb les consoles centrals, s'estipulen una sèrie de requeriments per ajudar en la presa de decisions.

- Facilitat de configuració
- Facilitat d'ús
- Gestió d'usuaris
- Compatibilitat amb assistents virtuals
- Compatibilitat amb eines externes
- Compatibilitat amb sistemes de missatgeria

5.1. OpenHAB

El mateix OpenHAB proposa un sistema de processament d'esdeveniments anomenat Rules ja incorporat dins la plataforma i que es gestiona mitjançant formularis molt intuïtius. El funcionament és molt senzill: donat un o més esdeveniments, executarà accions si es donen certes condicions.

Un cop configurada una regla, es pot consultar l'estat actual i es mostra també la seva darrera execució. Totes les execucions prèvies queden enregistrades als fitxers de log, però no es mostren en la interfície d'usuari (no es té accés directe a l'històric).

El sistema de regles d'OpenHAB permet evidentment la interacció bàsica amb tots els components del sistema, però no es poden generar accions on s'inclogui com a paràmetre d'entrada un valor d'un sensor. Aquest fet no permet per exemple, enviar una notificació amb la temperatura de la cambra.

Per últim, mencionar que el sistema de regles està en fase beta i avui en dia, no poden ser executades per elements externs o assistents virtuals. És un sistema molt tancat pensat per a gestionar escenaris senzills.

5.2. IFTTT

El concepte de If This Then That es força difícil de definir. A grans trets, és una plataforma Cloud que permet automatitzar accions segons fets (Si passa això, llavors allò). Agrupa serveis de centenars de proveïdors en una sola eina de manera que puguin interactuar entre ells definint escenaris senzills.

IFTTT queda vinculat a un compte d'usuari que permet definir els seus propis escenaris i atorga permisos complets d'administrador, no hi ha possibilitat de definir perfils de només lectura o només rebre notificacions. L'única manera

possible seria vinculant un compte de myOpenHAB de només lectura a l'aplicació i crear un compte IFTTT també dedicat, no és viable.

IFTTT disposa d'una aplicació per Android i iOS que permet definir escenaris de la mateixa manera que es pot fer en un ordinador. A més, es poden definir entrades o sortides referents al dispositiu mòbil com ara localització, trucades o notificacions. La seva interfície és molt intuïtiva i neta, no hi ha possibilitat d'error.

El producte com a tal no està centrat en la domòtica, però gràcies a la compatibilitat amb OpenHAB i utilitzant el portal MyOpenHAB, ofereix molta versatilitat a l'hora d'integrar l'eina. Es pot configurar per exemple encendre el llum si es rep un correu amb la paraula "llum ON", o rebre una trucada per veu IP en cas que la temperatura d'una habitació sigui inferior a 15 graus.

Està completament integrat amb els assistents Alexa i Google Home però no disponible en castellà, només anglès. En tractar-se d'un servei Cloud, no permet controlar elements del sistema HomeKit d'Apple, ja que aquests només són accessibles en xarxa local o utilitzant un Apple TV.

De manera general IFTTT presenta dos grans inconvenients. D'una banda, els elements desencadenants (IF) són els estrictament estipulats dins el servei, no se'n poden afegir de personalitzats. Si algun dispositiu no es troba actualment, no quedarà altra opció que esperar que el fabricant desenvolupi el servei juntament amb IFTTT.

D'altra banda, els escenaris que es poden plantejar es limiten a un element d'entrada i un de sortida: Si rep un correu, encén el televisor, si es puja un document a Dropbox, apaga el llum... No hi ha possibilitat de conjuntar més elements ni definir regles complexes.

5.3. NodeRED

NodeRED és una plataforma desenvolupada inicialment per IBM dedicada únicament a gestionar dispositius que formen part de la Internet Of Things. Està desenvolupada en Python i està formada per un motor intern que executa les accions, i una consola web on l'usuari defineix fluxos d'interacció.

Igual que en el cas d'OpenHAB, existeix una gran comunitat al voltant que desenvolupa tota mena de connectors per interactuar amb qualsevol element de qualsevol fabricant. De fet, podria fins i tot reemplaçar OpenHAB, si tingués la possibilitat d'oferir una visió global de l'estat del sistema i una interacció mitjançant un panell web.

La idea principal es basa a definir elements que s'utilitzaran com a desencadenadors d'esdeveniments, i enllaçar-los amb altres components encarregats de processar condicions amb l'objectiu final de generar les sortides necessàries.

Per exemple, si es vol encendre un llum durant un període de 5 min en cas de detectar activitat amb el sensor de moviment, es defineix el sensor com a entrada. La sortida d'aquest es lliga a un altre component per a determinar si s'ha executat una acció en els darrers 5 minuts i si no, encendrà el llum activant el temporitzador.

Com es comentava anteriorment, gràcies als connectors és compatible amb infinitat de sistemes entre ells, control per veu i serveis de missatgeria.

5.4. Conclusió

Com que ni OpenHAB ni IFTTT no proporcionen funcionalitats referents a la definició d'escenaris complexos, s'escull NodeRED. La seva configuració segurament serà tediosa però el resultat pot ser completament personalitzat.. Existeix també una gran comunitat de desenvolupadors que n'assegura el seu manteniment i evolució.

6. Instal·lació

En aquest capítol es detallaran totes les accions dutes a terme per tal d'instal·lar cadascun dels dispositius dins la xarxa domèstica. Cal mencionar que de moment no interactuaran entre ells, només amb la seva aplicació propietària.

6.1. Ecosistema Xiaomi (Temperatura, moviment, inundació i portes)

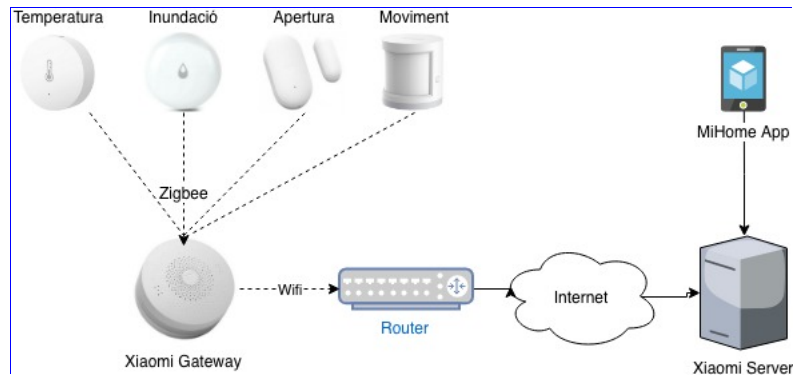
El gateway principal ve amb un endoll de tipus I pel que requereix un adaptador per a connectar-ho a un endoll europeu de tipus F. Un cop endollat, el dispositiu s'inicialitza, comença a fer pampallugues per indicar que no està encara emparellat a cap compte ni xarxa wifi. Tal com es pot comprovar en la il·lustració 10, la mida és molt reduïda.



Il·lustració 10 - Mida del gateway en comparació amb una moneda

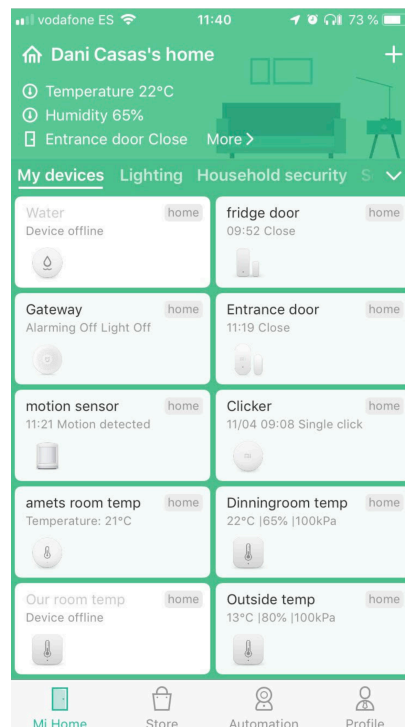
Ara, amb un mòbil cal descarregar-se l'aplicació Mi Home i prémer el símbol + que es troba a la part superior dreta, cercar el dispositiu Mi Control Hub i seguir les indicacions per a configurar el dispositiu. Un cop configurat, aquest apareixerà a la pantalla principal i es podran incorporar la resta de sensors.

Tècnicament, la primera vegada que el gateway s'encén, crea una xarxa wifi completament oberta, l'aplicació Mi Home s'hi connecta. Aquesta configura el dispositiu per a connectar-se al wifi de 2.4Ghz destí i reiniciar el controlador. Un cop s'ha connectat, la xarxa oberta deixa d'existir. D'ara endavant, i tal com es mostra en la il·lustració 11, el gateway enviarà les dades al servidor de Xiaomi. MiHome es connectarà a aquest per a recollir els resultats de les mesures dels sensors i mostrar la informació a l'usuari.



Il·lustració 11 - Esquema connectivitat Xiaomi

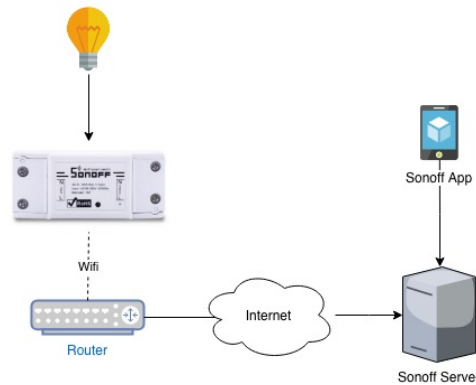
Després de configurar el gateway, es poden afegir la resta de sensors, prement el símbol + i el botó físic de sincronització disponible en cada dispositiu concret (temperatura, moviment i inundació). Un cop s'han executat tots els passos de l'assistent, la plana principal de l'aplicació mi Home mostra tots els sensors com es pot veure en la il·lustració 12.



Il·lustració 12 - Aplicació MiHome configurada

6.2. Sonoff (Control de llum)

Els dispositius Sonoff vénen configurats per a poder ser utilitzats únicament amb una aplicació propietària o els assistents virtuals Alexa i Google Home. El funcionament és molt semblant a Xiaomi, el dispositiu crea una xarxa wifi pròpia oberta, des de l'aplicació es configura la xarxa real de destí i la xarxa oberta ja no estarà mai més disponible. El dispositiu es connecta a Internet, envia tota la informació als servidors de Sonoff i es pot accedir a tot mitjançant l'aplicació dedicada, tal com es mostra en l'esquema de la il·lustració 13.



Il·lustració 13 - Esquema inicial de connectivitat Sonoff

Si es vol obrir el sistema a altres integracions, gràcies al fet que implementa un microxip molt utilitzat en domòtica anomenat ESP8266, cal reemplaçar el firmware original. Donat que el sensor anirà encastat en els conductes elèctrics, l'accés a ell després de la instal·lació serà complicat. Es detallen a continuació els passos a seguir per canviar el firmware.

Actualment existeixen tres models de firmware alternatius: Tasmota, ESPurna i ESPEasy. Tots tres són molt similars però en aquest projecte s'utilitzarà Tasmota degut a la seva facilitat en la configuració inicial.

Després d'obrir físicament el dispositiu, s'han de localitzar els 4 pins que s'utilitzaran durant el procés: 3V3, GND, RX i TX. Els dos primers estan dedicats a proporcionar alimentació al circuit, els dos últims a transmetre el firmware. Amb l'ajuda d'un soldador, es connectarà un cable a cadascun d'ells.

Un cop es tenen els terminals dels 4 cables separats, es connectaran a un convertidor TTL a USB que a la vegada es connectarà a un ordinador. Al mercat existeixen molts convertidors, en el cas d'aquest projecte, s'utilitzarà un Arduino.

Caldrà descarregar el firmware des de github (<https://github.com/arendst/Sonoff-Tasmota>) i configurar un IDE. Per senzillesa, es recomana utilitzar Atom amb el connector PlatformIO. Aquest proporciona tota la configuració necessària per interactuar amb aquest tipus de components sense haver d'instal·lar cap software adicional.

Existeixen dos fitxers, de configuració que cal modificar. Es detallen a continuació els paràmetres així com els seus valors.

platformio.ini

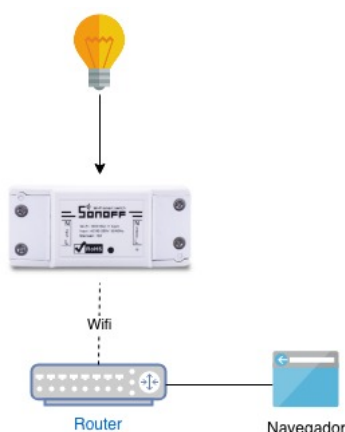
- **upload_speed:** estipula la velocitat a la qual es carregaran les dades.
- **upload_port:** adreça on està connectat el convertidor TTL a USB.

user_config.sh

- **STA_SSID1:** Nom de la xarxa wifi on es connectarà el dispositiu.
- **STA_PASS1:** contrasenya de la xarxa wifi esmentada a STA_SSID1.

Un cop els paràmetres són correctes, es pot procedir a actualitzar el firmware tot posant Sonoff en mode flash. Es prem el botó del dispositiu Sonoff i sense deixar-lo anar s'endolla el convertidor TTL-USB a l'ordinador, fins que faci pampallugues. Ara s'accedeix a Atom, pestanya PlatformIO i opció upload. El procés triga aproximadament 1 minut.

Finalment, ja es pot desconnectar tot i instal·lar el dispositiu en el seu destí definitiu. Aquest pas és el més senzill i només cal localitzar els dos cables (positiu i negatiu) que proporcionen corrent al llum i instal·lar sonoff al mig. Ara, tal com es mostra en la il·lustració 14, ja no caldrà accedir els servidors de Sonoff per a comprovar l'estat del dispositiu, tot funciona dins la xarxa local.



Il·lustració 14 - Esquema final de connectivitat Sonoff

6.3. Netatmo (Termòstat)

El sistema Netatmo està format per dos elements, un Relay encarregat de controlar la caldera i un termòstat que permet establir la temperatura desitjada i que incorpora també un sensor de temperatura. El termòstat es connecta al Relay per ones de ràdio i aquest últim a Internet utilitzant la xarxa wifi.

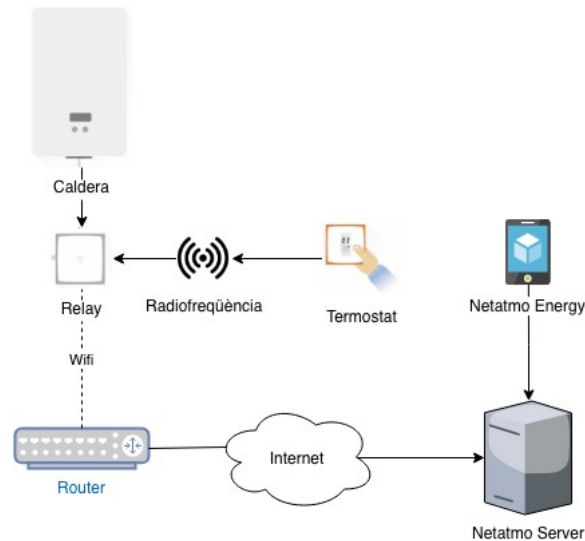
La caldera disposa de dos cables que en unir-se, tanquen un circuit i posen en marxa la calefacció. En l'habitatge on s'instal·la el producte, aquests cables anteriorment estaven connectats a un termòstat manual situat al menjador. S'ha de treure el termòstat manual i connectar els terminals al Relay de Netatmo.

A més, el Relay requereix alimentació pròpia. L'alimentació es pot proporcionar mitjançant un adaptador d'endoll o connectant els terminals positiu i negatiu directament al sistema elèctric. En el cas del projecte, s'opta per a connectar-ho al sistema elèctric per a deixar l'endoll lliure.

Similar al cas de l'ecosistema Xiaomi, quan el Relay comença a funcionar, crea una xarxa wifi oberta temporal en la banda 2.4Ghz a la que l'aplicació Netatmo Energy es connecta. L'aplicació demana a quina xarxa es vol connectar el dispositiu i realitza tots els passos necessaris.

Un cop es reinicia el dispositiu, l'aplicació Netatmo Energy es connectarà als servidors de Netatmo per a interactuar amb el sistema. A diferència de Xiaomi,

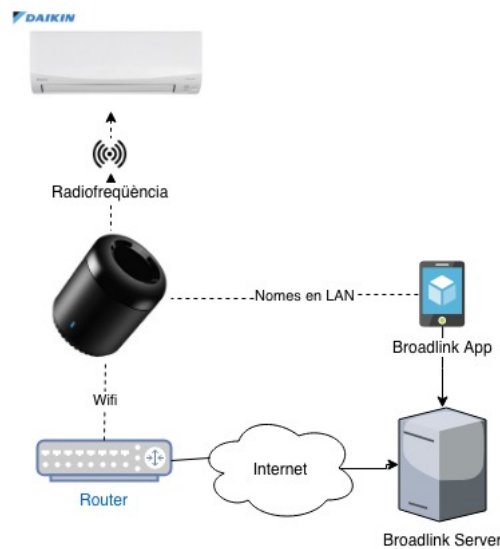
Netatmo no proporciona la possibilitat d'interactuar amb el sistema directament per xarxa local, tal com es veurà en el punt 2.6, caldrà crear un compte de desenvolupador a la web de Netatmo i vincular el dispositiu per tal d'interactuar des d'OpenHAB. L'esquema de connexió final es mostra en la il·lustració 15.



Il·lustració 15 - Esquema connectivitat Netatmo

6.4. Broadlink RM Mini 3 (Aire condicionat)

Igual que els altres dispositius basats en el protocol TCP, Broadlink crea una xarxa wifi temporal en la freqüència 2.4Ghz que utilitza l'aplicació Broadlink e-Control per a configurar el dispositiu. Tal com es mostra en la il·lustració 16, les dades s'envien al servidor de Broadlink i l'aplicació interactua amb elles, no es realitza cap operació dins la xarxa local.

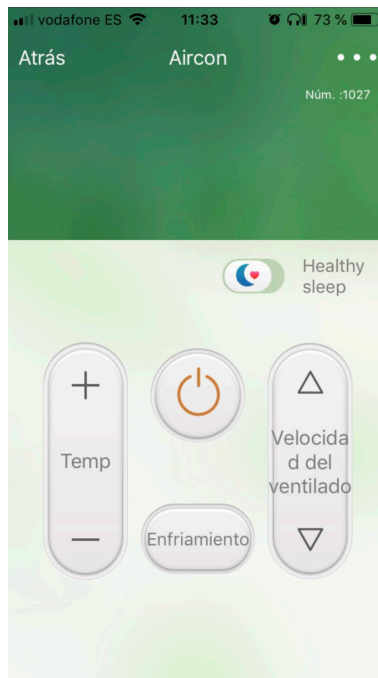


Il·lustració 16 - Esquema de connectivitat Broadlink RM Mini 3

Per a configurar els diferents dispositius infrarojos, s'han d'afegir mitjançant l'aplicació tot cercant el tipus (àudio, televisió, aire condicionat...) i el model. En cas que el model no es trobi, sempre es pot optar per capturar els codis de

cadascun dels botons d'un comandament remot. Després caldrà memoritzar-los posant Broadlink RM Mini 3 en mode aprenentatge i prement la tecla en el comandament.

En el cas del projecte, es treballa amb un aire condicionat Daikin model FTXS35G2V1B que sí que existeix a la base de dades donant com a resultat un panell molt senzill (però tremendament funcional) com el que es mostra en la il·lustració 17.












Il·lustració 17 - Exemple de comandament d'aire condicionat

6.5. Configuració de xarxa

Com que existeixen molts dispositius que requereixen connectivitat Wifi, cal fixar la seva IP de manera que aquesta no canviï quan l'encaminador es reiniciï i forci a tornar a configurar OpenHAB. S'estipula l'adreça 192.168.31.120 per a l'ordinador encarregat de gestionar el sistema i les immediatament següents IPs per a tots els dispositius.

La millor opció per a millorar la seguretat de la xarxa és crear un subrang dedicat a la domòtica, de manera que els dispositius no puguin ser accessibles per altres dispositius directament utilitzant la seva IP. Es descarta l'opció, ja que es tracta d'un entorn domèstic sense gaire risc d'atacs. En la il·lustració 18 es pot veure l'assignació corresponent d'adreces, les adreces MAC apareixen rallades per motius de seguretat.

Dirección IP asignada manualmente según la lista DHCP (Límite máximo : 64)		
Nombre del cliente (Dirección MAC)	Dirección IP	Agregar o eliminar
<input type="text"/>	<input type="text"/>	+
 raspberrypi [MAC address redacted]	192.168.31.120	-
 DiskStation [MAC address redacted]	192.168.31.100	-
 lumi-gateway-v3_miao44695362 [MAC address redacted]	192.168.31.121	-
 Broadlink_RMMINI-c7-b3-84 [MAC address redacted]	192.168.31.122	-
 Netatmo [MAC address redacted]	192.168.31.123	-
 yeelink-light-color1_miao805923 [MAC address redacted]	192.168.31.124	-
 ipcam_00626E46A45E_1 [MAC address redacted]	192.168.31.200	-
 dom-swi-snf-out-01 [MAC address redacted]	192.168.31.126	-
 dom-swi-snf-din-01 [MAC address redacted]	192.168.31.125	-
 BRWACD1B8341FCC [MAC address redacted]	192.168.31.110	-

Il·lustració 18 - Assignació d'IPs fixes als dispositius de domòtica

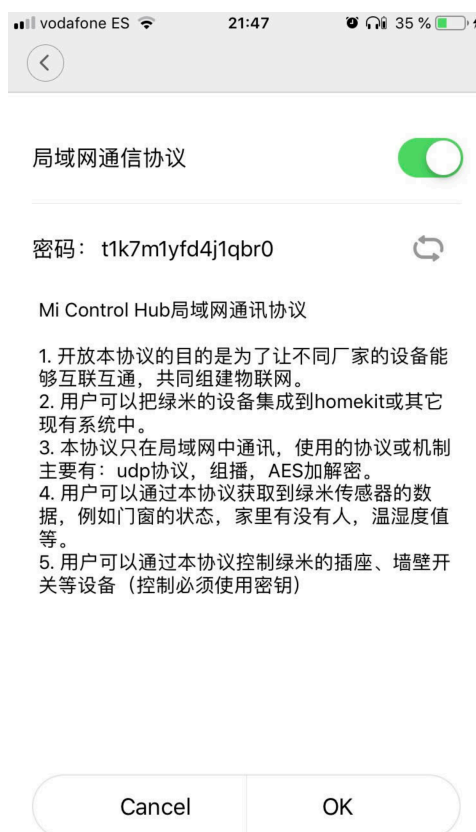
7. Prerequisites

En el punt actual del projecte, tots els dispositius estan configurats i es poden utilitzar mitjançant la seva corresponent aplicació propietària. Dit això, per tal de poder gestionar tot el sistema des d'OpenHAB, cal realitzar una sèrie de configuracions extres.

Aquest capítol s'endinsa en totes les operacions que s'han de dur a terme per a preparar tots els dispositius.

7.1. Ecosistema Xiaomi

Per tal de permetre OpenHAB interactuar directament amb els sensors sense connectar-se al servidor de Xiaomi, cal activar l'accés LAN utilitzant l'aplicació MiHome. Es selecciona el dispositiu gateway, secció About i es fan 5 pulsacions a qualsevol part de la pantalla. En aquest moment s'accedeix a un menú extra en xinès mostrat en la il·lustració 19. La tercera opció permet l'activació d'accés LAN, proporciona una clau única d'autenticació que és la que demana OpenHAB.

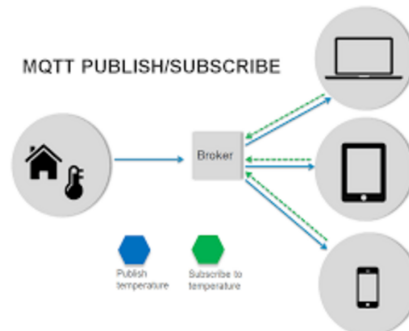


Il·lustració 19 - Configuració Xiaomi per accés en LAN

7.2. Sonoff

Tasmota utilitza el protocol MQTT per enviar informació d'estat i rebre ordres. MQTT és un sistema broker en el qual els dispositius poden publicar informació (publishers) i d'altres la poden consumir (subscritors) de manera asíncrona. La

unitat d'informació s'anomena missatge. S'utilitza en molts projectes de domòtica i en informàtica distribuïda, implementa mecanismes per tal d'assegurar l'entrega de missatges als consumidor. És impossible que la informació es perdi. En la següent figura es mostra el funcionament aplicat a un projecte de domòtica.



Il·lustració 20 - Esquema funcionament del protocol MQTT

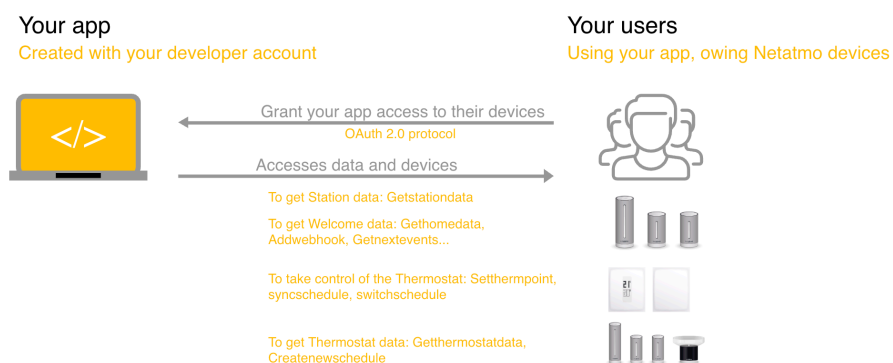
Font: <https://ricveal.com/blog/primeros-pasos-mqtt/>

S'utilitzarà el broker MQTT Mosquitto, ja que està disponible com a paquet en la majoria de sistemes operatius i en el cas d'ús del projecte, no requereix configuració. La instal·lació es realitza mitjançant la comanda `apt-get install mosquitto`. Tots els paràmetres per defecte són suficients i es pot inicialitzar el servei amb `systemctl mosquitto start`.

Ara, s'haurà d'accedir a la consola web de cadascun dels dispositius Sonoff, secció configuració, MQTT. Cal especificar el hostname on està instal·lat MQTT, amb l'usuari i contrasenya i declarar un tòpic. El nom del tòpic és SONO_LIGHT_0X on X és un incremental.

7.3. Netatmo

Per tal d'interactuar amb el termòstat netatmo des d'OpenHAB, l'usuari s'ha de registrar al web <https://dev.netatmo.com> amb el mateix usuari que es va donar d'alta per utilitzar l'aplicació Energy i crear una aplicació. Aquesta aplicació proporciona un únic punt d'interacció amb el sistema. Els usuaris s'autenticaran utilitzant el protocol OAuth 2.0 i després podran extreure les dades que vulguin, o canviar paràmetres com ara la temperatura desitjada. L'esquema de funcionament queda recollit en la il·lustració 21.



Il·lustració 21 - Esquema funcionament de la API de Netatmo

Font: <https://pro.netatmo.com/eu/thermostat/installation>

8. Configuració consola central

8.1. Terminologia d'OpenHAB

Abans d'instal·lar OpenHAB, cal familiaritzar-se amb els diferents conceptes de l'ecosistema. L'element principal d'OpenHAB són els **Add-Ons**. Es podrien definir com un conjunt d'instruccions que s'incorporen a OpenHAB amb l'objectiu que aquest entengui com gestionar un dispositiu extern. Tècnicament, es tracta d'un conjunt de llibreries normalment contingudes en un fitxer Java .jar.

Els Add-Ons es classifiquen segons la seva naturalesa en:

- **Actions:** Aquells que realitzaran accions a sistemes completament aliens a OpenHAB com ara missatges de Telegram o enviament de correu.
- **Bindings:** Encarregats de permetre OpenHAB gestionar dispositius com ara sensors o activadors.
- **Misc:** Altres tipus de components.

D'aquests tres, els Bindings són els més importants, ja que són els que s'utilitzen per a construir un sistema de domòtica captant informació dels diferents dispositius externs.

Un cop s'ha incorporat un Add-On de tipus **Binding**, OpenHAB és capaç de cercar components de la tecnologia corresponent. Cada component es representa a OpenHAB com un o més objectes de tipus **Thing**. Un Thing pot portar configuració associada com ara claus d'accés, valors de temps màxim d'espera, ports... Aquest element representa un dispositiu capaç de proveir d'un o diversos valors.

Cadascun d'aquests elements que pot proporcionar valors rep el nom de **Channel**. Els channels poden representar mesures de valors com ara temperatura, moviment... però també altre tipus d'informació com estat de la bateria. Pot donar-se la situació que no interressi activar tots els channels si aquests es consideren que no aporten informació rellevant per a l'usuari.

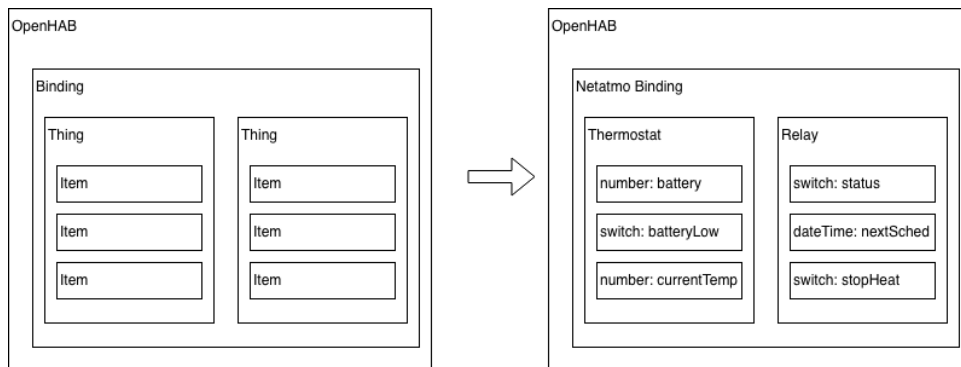
Quan s'activa un channel, openHAB crea un **Item**. Un item representa el punt d'entrada o sortida d'un sensor o actuador. Un item és l'element que permet saber si una porta està oberta o tancada i pot també encendre i apagar un llum.

Cadascun dels Items pot ser de diferent tipus en funció de les dades que gestiona:

- **Color:** utilitzat per exemple en llums de tipus RGB
- **Contacte:** utilitzat per a portes o finestres
- **Data i temps:** última vegada que una porta s'ha obert
- **Regulador:** regular intensitat en llums que ho permetin
- **Imatge:** per a gestionar càmeres web
- **Localització:** coordenades GPS
- **Nombre:** qualsevol sensor que utilitzi valors numèrics

- Reproductor: per a controlar aparells de música
- Persiana: per a tendals i persianes
- Text: utilitzat com a element estàtic per a mostrar un simple text
- Activador: útil per a qualsevol aparell amb funció encendre/apagar

Es detalla en la il·lustració 23 un exemple on apareixen els tipus d'objectes que utilitza OpenHAB amb la seva correspondència en un exemple real de dispositiu.



Il·lustració 23 - Esquema dels components d'OpenHAB

Per últim, quan es necessita interactuar amb els sensors, en certes ocasions es vol aplicar una sèrie de transformacions al valor introduït per l'usuari abans de realment enviar o mostrar el valor. En el cas del projecte es donarà aquesta situació en el moment en què l'usuari vol controlar l'aire condicionat.

Tal com es comenta en el punt 2.6.1, cada pulsació del comandament a distància es transforma en un codi, aquest és el que el dispositiu enviarà mitjançant infrarojos. Configurar una temperatura de 20 graus equival a un codi, 21 a un altre i apagar el dispositiu un altre.

Aquest tipus de transformacions es realitzen a OpenHAB utilitzant **Maps**. Aquest objecte no és més que un fitxer de text que s'ha de copiar a la carpeta `/opt/openhab2/conf/transform` i conté una línia per a cada transformació.

```
21 = 837262AFBCDE
22 = 67181ABDHF99
...
```

8.2. Convenció de noms

Cada objecte de tipus Thing i Item té associats un nom i un identificador. El nom només té finalitat informativa i és el que es mostrarà en les diferents interfícies d'OpenHAB, no cal que sigui únic. L'identificador s'utilitzarà internament per a identificar l'element i és necessari que sigui únic en tot el sistema.

Per tal d'organitzar tots els elements, és molt recomanable estipular una nomenclatura de noms. En un sistema format per dos elements potser no té

gaire importància però a mesura que s'afegeixen nous dispositius, aquesta nomenclatura pot ser de gran utilitat per tenir-ho tot sota control.

La comunitat OpenHAB no estableix cap mena de requeriment ni guia per a numerar els dispositius, depèn completament de l'usuari en funció de les seves necessitats. En el cas del projecte, en l'àmbit de requeriments, es poden definir els següents elements com a informació indispensable que cal incorporar com a part de l'identificador únic.

- Things
 - o Fabricant
 - o Nom
 - o Unitat(en cas de tenir dos dispositius iguals)
- Items
 - o Thing associat
 - o Ubicació
 - o Objecte que representa (temperatura, humitat...)

Com que la plataforma OpenHAB està disponible únicament en anglès, es recomana utilitzar aquest idioma en la mesura del possible per identificar els objectes.

Es presenta el resultat de la nomenclatura decidida així com alguns exemples de sensors i dispositius.

- Things: XXXX_XXXX_XX
- Items: XXXX_XXXX_XX_YYYY_YYYY_YY

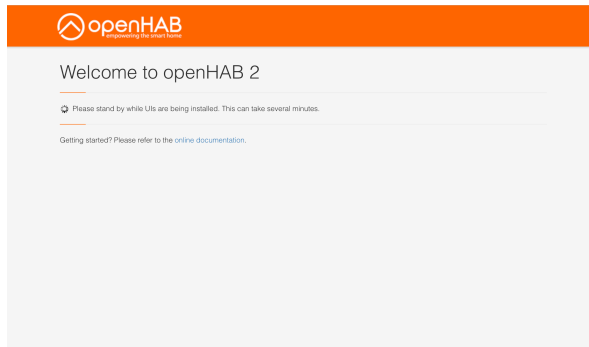
Exemples

- Thing: XIAO_GATE_01
- Thing: XIAO_TEMP_01
- Thing: NETA_RELAY_01
- Item: XIAO_TEMP_01_DINN_TEMP_01
- Item: NETA_RELAY_01_DINN_ONOF_01

8.3. Instal·lació

Per instal·lar OpenHAB, s'utilitzarà el paquet disponible al repositori Raspbian. El procés d'instal·lació genera tots els fitxers a la carpeta `/opt/openhab2` i crea un servei que es pot gestionar mitjançant `systemctl`.

Un cop s'arrenca l'aplicació, es pot accedir a la interfície web utilitzant la IP del servidor i el port que ve configurat per defecte (8080) mostrant com a resultat la pàgina mostrada en la il·lustració 24.



Il·lustració 24 - Plana inicial d'OpenHAB

La primera vegada que s'inicia OpenHAB sol trigar entre 5 i 10 minuts a carregar tots els mòduls.

OpenHAB posa a disposició de l'usuari dues interfícies inicialment, PaperUI i HABPanel. La primera s'utilitza per a gestionar la plataforma i controlar el sistema de manera bàsica. HABPanel està focalitzat a construir interfícies web d'interacció amb els components i es detallarà el seu funcionament en els següents capítols.

Com que les accions es realitzaran des de NodeRED, només caldrà instal·lar i configurar els següents Bindings:

- MQTT Binding (1.x)
- Netatmo Binding
- Xiaomi Mi Smart Home Binding
- Z-Wave Binding

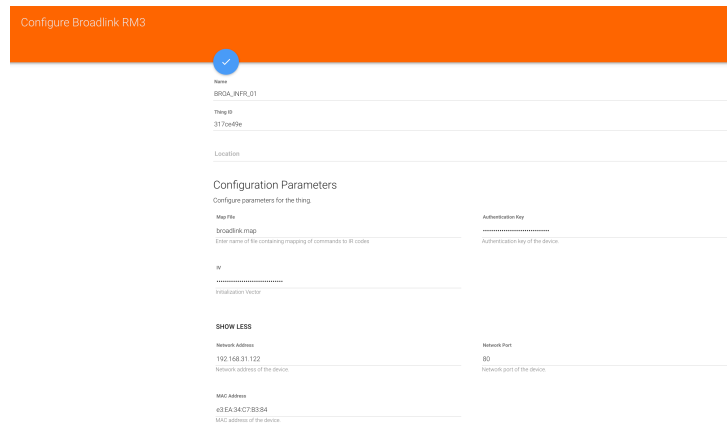
Actualment OpenHAB no conté cap Add-On per a gestionar el dispositiu Broadlink. Tot i això, un usuari de la comunitat n'ha realitzat un completament funcional. Caldrà descarregar-ho d'Internet i copiar el fitxer (.jar) a la carpeta `/opt/openhab2/addons`.

<https://dl.bintray.com/themillhousegroup/generic/org.openhab.binding.broadlink-2.4.0-BETA-7.jar>

A més, per permetre l'accés des de la plataforma Cloud MyOpenHAB, caldrà instal·lar el següent component disponible en la secció Misc.

- OpenHAB Cloud Connector

Per últim s'haurà d'especificar l'adreça IP del dispositiu, el port (80) i l'adreça MAC en les diferents caselles que es mostren en la il·lustració 27.

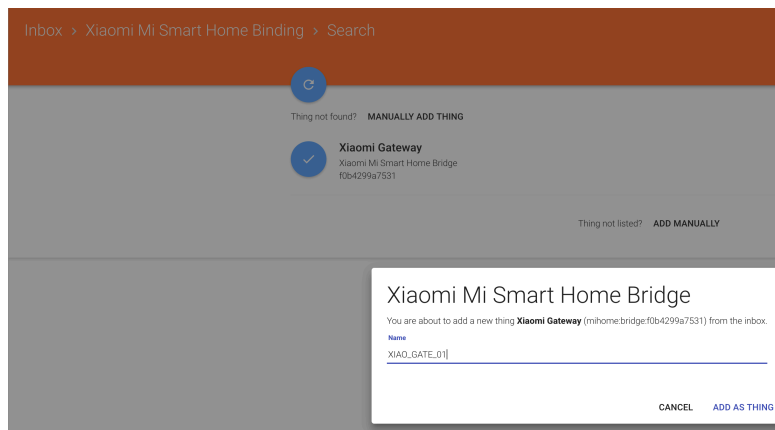


Il·lustració 27 - Configuració Binding Broadlink

Un cop l'objecte de tipus Thing ja apareix al llistat, es pot seleccionar per declarar un únic Item anomenat Command que és el que s'utilitzarà per a enviar les comandes requerides.

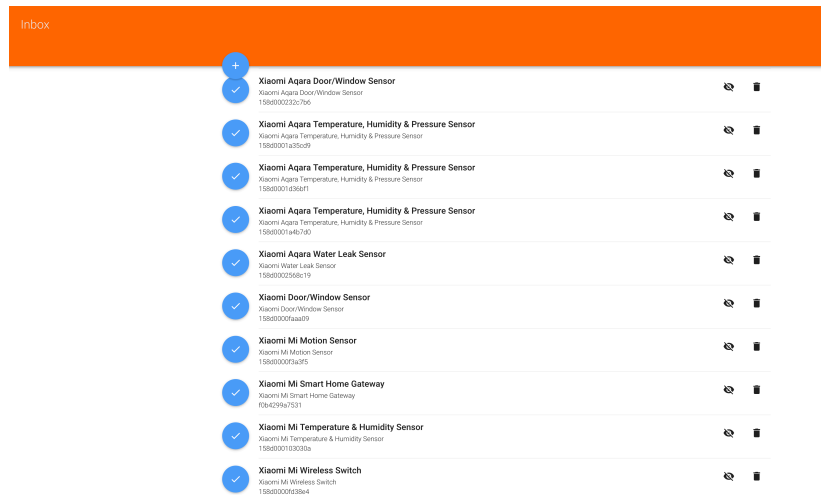
8.4.2. Ecosistema Xiaomi

En el cas de Xiaomi, primerament s'haurà de crear l'objecte de tipus Thing corresponent al gateway que controla tots els altres dispositius. Si s'ha activat correctament l'accés per LAN mencionat a la secció de prerequisits, apareixerà automàticament tal com es mostra en la il·lustració 28.



Il·lustració 28 - Configuració Binding Xiaomi

En declarar el gateway, apareixen ja tots els sensors al llistat de Things disponibles. S'ha de seleccionar cada element, declarar l'objecte de tipus Thing seguint la convenció de noms i més tard crear l'Item corresponent amb tots els Channels disponibles.



II·lustració 29 - Sensors de Xiaomi dins OpenHAB

8.4.3. MQTT (Sonoff)

Durant els darrers anys, OpenHAB ha millorat molt la configuració dels connectors i gairebé totes les accions es poden realitzar des de la interfície d'usuari. Això però no era tan senzill en la primera versió (1.0) on calia configurar els bindings utilitzant fitxers de text. MQTT Binding encara no ha set migrat al nou sistema de gestió, pel que s'haurà de parametritzar la configuració utilitzant el fitxer `mqtt.cfg` disponible a la carpeta `/opt/openhab2/conf/services`.

Existeixen múltiples paràmetres però en el cas d'aquest projecte, donat que no existeixen usuaris ni contrasenyes a la instància de MQTT, només caldrà configurar l'URL del broker.

```
broker.url=tcp://localhost:1883
```

En el cas d'aquest component específic, no es declaren objectes Thing i els Items s'han de declarar amb el format OpenHAB1, utilitzant fitxers de text. Aquests fitxers s'han de copiar a la carpeta `/opt/openhab2/conf/items` amb el nom `sonoff.items` i el següent text.

```
Switch          SONO_LIGH_01_OUTS_Status      "Light"          {mqtt=">[broker:cmdndom-swi-snf-din-01/POWER:command*:default],<[broker:statdom-swi-snf-din-01/POWER:state:default]" }
Switch          SONO_LIGH_02_DINN_Status      "Light"          <light> (LR,gLight)      {
mqtt=">[broker:cmdndom-swi-snf-out-01/POWER:command*:default],<[broker:statdom-swi-snf-out-01/POWER:state:default]" }
```

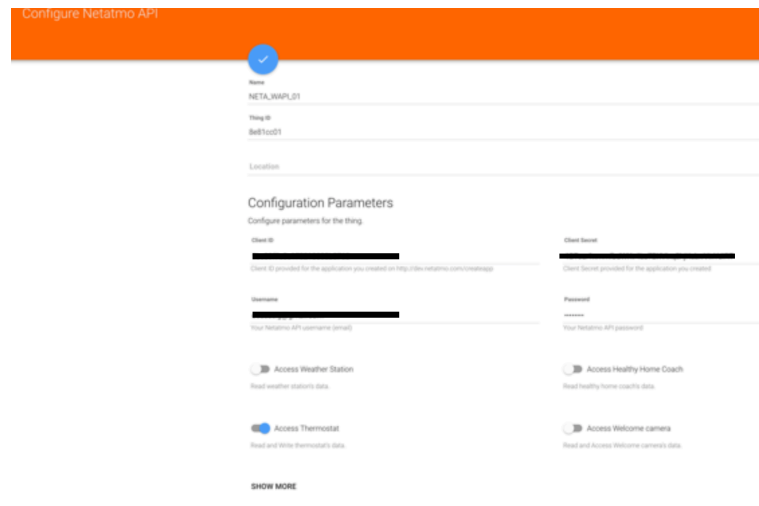
Bàsicament estipulen els mateixos paràmetres disponibles en la UI de forma tabulada.

- Tipus de Component
- Identificador únic de l'Item
- Nom amigable de l'Item
- Controlador

En el cas del controlador, s'especifica el tòpic de MQTT al que OpenHAB es subscriu.

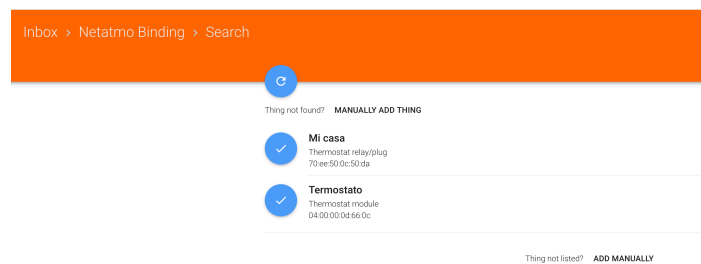
8.4.4. Netatmo

En el cas del termòstat Netatmo, s'ha de configurar primerament un objecte de tipus Thing corresponent a l'API amb els paràmetres ClientID, ClientSecret, Username i password. S'han generat durant la fase de prerequisits i es pot comprovar la configuració en la figura 30.



Il·lustració 30 - Configuració Binding de Netatmo

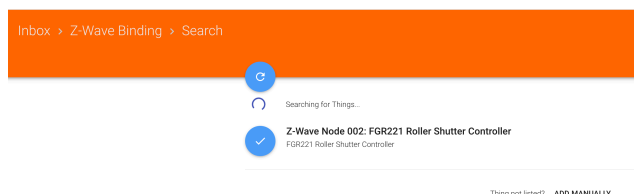
Un cop l'API està configurada, OpenHAB detecta automàticament dos dispositius, el Relay i el termòstat (il·lustració 31). Ara es pot capturar tota la informació dels diversos canals en format Item.



Il·lustració 31 - Dispositius Netatmo després de configurar la API

8.4.5. Zwave

En tractar-se d'un dispositiu que es connecta per USB i com que els controladors ja estan incorporats a la Raspberry, automàticament apareix llistat. Un cop s'ha incorporat el gateway com a objecte Thing i després de sincronitzar el dispositiu prement el botó pertinent, apareix l'element referent a la gestió de persianes com a un altre element de tipus Thing. L'item resultant es pot veure en la il·lustració 32.



Il·lustració 32 - Elements Zwave

8.5. Sumari d'Items

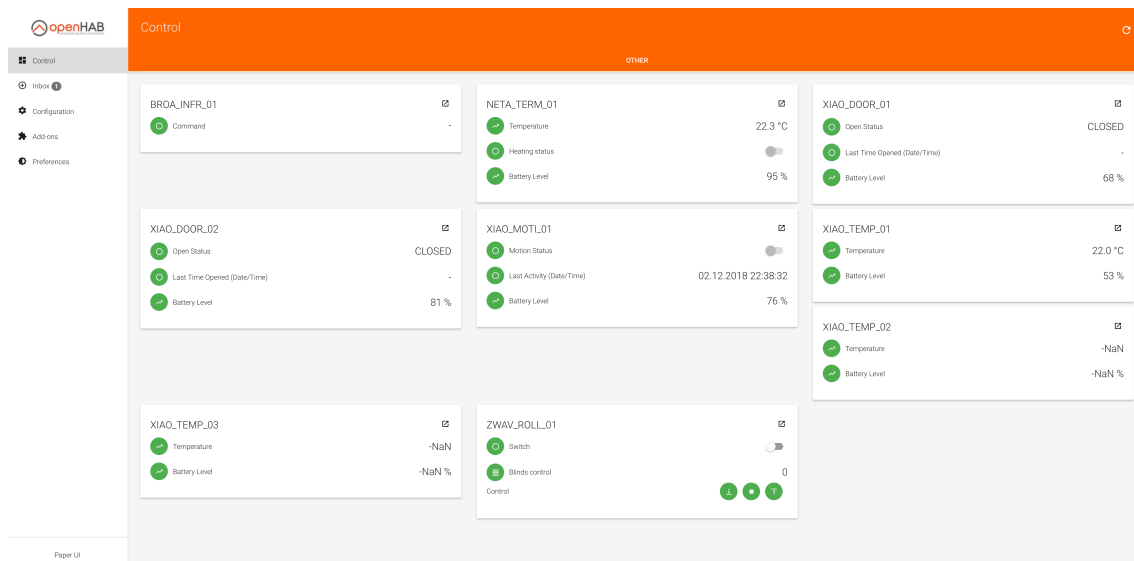
De manera global, es resumeixen en la següent taula tots els items utilitzats dins el sistema així com la seva relació amb els objectes de tipus Thing i el seu binding corresponent. Cal dir que existeixen molts més com ara avisos de bateria baixa o colors de llums però no s'utilitzaran en la present implementació, en conseqüència no apareixen llistats.

Item	Binding	Tipus	Descripció
BROA_INFR_01_DINN_Command	Broadlink	String	Envia la comanda especificada als aparells mitjançant infrarojos
NETA_TERM_01_DINN_Temperature	Netatmo	Number	Indica la temperatura actual
NETA_TERM_01_DINN_HeatingStatus	Netatmo	Switch	Indica si la caldera està en funcionament
NETA_TERM_01_DINN_BatteryLevel	Netatmo	Number	Percentatge de bateria del termòstat
NETA_TERM_01_DINN_SetPoint	Netatmo	Number	Temperatura desitjada
XIAO_DOOR_01_HALL_OpenStatus	Xiaomi	Contact	Indica si la porta està oberta o tancada
XIAO_DOOR_01_HALL_BatteryLevel	Xiaomi	Number	Percentatge de bateria del sensor
XIAO_DOOR_02_FRID_OpenStatus	Xiaomi	Contact	Porta oberta o no
XIAO_DOOR_02_FRID_LastTimeOpenedDateTime	Xiaomi	DateTime	Conté la data i hora en què la porta es va obrir per última vegada
XIAO_DOOR_02_FRID_BatteryLevel	Xiaomi	Number	Percentatge de bateria del sensor
XIAO_MOTI_01_OUTS_MotionStatus	Xiaomi	Switch	Moviment detectat en aquest moment
XIAO_MOTI_01_OUTS_LastActivityDateTime	Xiaomi	DateTime	Conté la data i hora en què es va detectar moviment per última vegada
XIAO_MOTI_01_OUTS_BatteryLevel	Xiaomi	Number	Percentatge de bateria del sensor
XIAO_TEMP_XX_XXXX_Temperature	Xiaomi	Number	Indica la temperatura actual
XIAO_TEMP_XX_XXXX_BatteryLevel	Xiaomi	Number	Percentatge de bateria del sensor
ZWAV_ROLL_01_DINN_Switch	Zwave	Switch	Persiana oberta o tancada
ZWAV_ROLL_01_DINN_BlindsControl	Zwave	Rollershutter	Control específic de la persiana
SONO_LIGH_01_OUTS_Status	MQTT	Switch	Control de llum
SONO_LIGH_01_DINN_Status	MQTT	Switch	Control de llum

Il·lustració 33 - Sumari d'elements de tipus Item del sistema

9. Configuració de panell web

Un dels objectius del treball és proporcionar un panell web des d'on l'usuari pugui gestionar tots els components del sistema de domòtica de manera senzilla. Tot i que PaperUI està pensat per a configurar Bindings i Things, també permet una interacció bàsica. Aquesta està pensada per a comprovar el funcionament més que per a ser utilitzat per l'usuari final, ja que no es pot personalitzar gairebé cap aspecte. Tot i això, tal com es pot comprovar en la il·lustració 34, es força funcional.



Il·lustració 34 - Panell de control de la interfície PaperUI

El mòdul HABPanel ja inclòs en la instal·lació bàsica té una gran flexibilitat per a mostrar informació de diverses maneres, i també crear panells mitjançant blocs que s'adapten a qualsevol mena d'usuari.

En aquest capítol, es detallarà la construcció de 2 tipus de panells. Un panell operatiu per interactuar amb els dispositius i un panell de control de bateria per a visualitzar l'estat de les bateries de tot el sistema.

9.1. Tipus d'elements

Abans d'entrar en detall, es detallen tota la terminologia utilitzada per aquest capítol amb l'objectiu de familiaritzar l'usuari.

HABPanel permet organitzar la informació mitjançant panells. Cadascun d'aquests panells és accessible des del menú de la part esquerra i està definit per un nom. A cada panell es pot aplicar un aspecte definit, utilitzant el que OpenHAB anomena Theme i a més es pot personalitzar amb fulles d'estil CSS. En el cas del projecte, aquestes dues funcionalitats no s'utilitzaran, ja que les opcions bàsiques cobreixen els requeriments.

Dins de cada panell, es poden declarar una o múltiples capsas que serviran per a mostrar valors dels sensors, text estàtic o permetre interacció amb el sistema.

Cada tipus de capsula s'anomena Widget i la seva funcionalitat es pot ampliar carregant nous components fets per la comunitat de desenvolupadors d'OpenHAB.

La configuració de tots els panells es pot enregistrar dins la base de dades pròpia d'OpenHAB, o mitjançant galetes de sessió al navegador. Les galetes són el mecanisme per defecte i en cas que s'accedeixi a la consola des d'un ordinador per primera vegada, s'haurà de carregar el panell mitjançant la pestanya Settings i l'opció Current storage location.

9.2. Panell operatiu

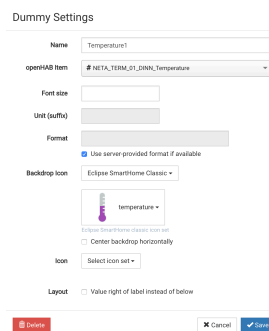
El panell operatiu és la consola central que l'usuari utilitzarà per a interactuar amb el sistema de domòtica. Està formada per dos blocs, un dedicat als sensors i l'altra als components que permeten interacció.

Primerament es creen dos blocs de tipus Label per agrupar sensors i activadors visualment. Aquest tipus de blocs serveixen només per a mostrar text i poden col·locar-se en qualsevol espai. Per a mantenir consistència amb la interfície, es defineix una mida de lletra de 15 i color taronja.

Sota l'etiqueta de sensors s'afegeixen els següents elements:

- Temperature1: mostra la temperatura de la cambra 1.
- Temperature2: mostra la temperatura de la cambra 2.
- Temperature3: mostra la temperatura de la cambra 3.
- Main Door: indica l'estat de la porta principal (OPEN/CLOSED).
- Fridge Door: indica l'estat de la porta de la nevera (OPEN/CLOSED).
- MotionDetected: indica l'últim instant que es va detectar moviment.
- WaterLeak: indica l'últim instant en què es va detectar aigua.

Per a tots aquests elements, s'utilitza un Widget de tipus Dummy tal com es pot veure en la següent figura. És l'element més senzill, ja que permet incloure un text i mostrar per pantalla la sortida del valor d'un sensor. Per facilitar la identificació, es defineix una imatge de fons que reflecteixi la finalitat del sensor tal com es pot veure en la següent figura.



II-lustració 35 - Configuració HabPanel - Objecte Dummy

Per part dels activadors, el funcionament és molt similar però s'ha d'escollir un altre tipus de Widget per a permetre la correcta interacció.

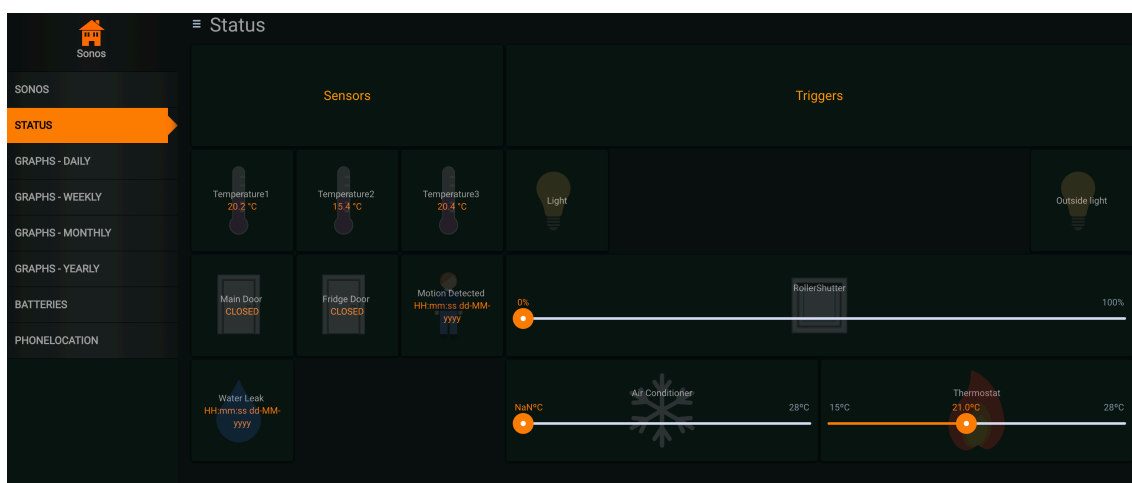
D'una banda, per a gestionar els llums, es declaren dos objectes de tipus Switch. La particularitat d'aquests elements és que el seu text canvia de color en funció de si l'element està encès o apagat. A més, igual que en el cas dels sensors, es pot definir una imatge de fons que també modificarà la seva transparència segons l'estat del dispositiu.

D'altra banda, es declara un widget de tipus Slider amb el que apareixerà una barra que permet definir el nivell d'apertura de la persiana. Els sliders requereixen un valor mínim, un màxim i l'esglaó. En aquest cas, com que es vol controlar tot el recorregut, es declara de 0 a 100 (completament tancada, completament oberta) amb un esglaó unitari.

El bloc corresponent a la gestió de l'aire condicionat també es pot gestionar amb un objecte de tipus slider. En aquest cas, estarà lligat als valors de temperatura que s'hagin capturat amb l'script mencionat en la secció anterior. Està estipulat en 16-28 amb un esglaó unitari, ja que l'equip disposa de bomba de calor i es pot utilitzar també com a calefacció.

Per últim el control del termòstat es configura exactament igual amb les temperatures marcades entre 19 i 28 graus i una escala unitària.

En la següent figura es mostra el panell resultant tenint en compte tots els components.



II-Il·lustració 36 - Panell operatiu

9.3. Panell de control de bateries

Com que molts dels dispositius funcionen mitjançant bateries, és molt convenient proporcionar una interfície on es pugui veure l'estat de totes elles independentment del seu fabricant. Això permet a l'usuari anticipar recanvis i també monitorar l'estat del sistema de manera global. Si un dispositiu no funciona, l'estat de la seva bateria serà 0.

Tots els elements són del mateix tipus. L'objecte Knob pren el valor d'un sensor com a referència i presenta la informació en format pastís en funció dels valors mínims i màxims establerts. En el cas d'aquest projecte, tal com es mostra en la següent il·lustració, la configuració és la mateixa per a tots i es pren com a referència l'escala del 0 al 100 amb un esgrao d'una unitat.

Knob Settings

General Bar & Track Scale Skin Ranges Misc.

Name Thermostat

openHAB Item # NETA_TERM_01_DINN_BatteryLevel

Read only Disable interaction

Min 0 Max 100

Step 1 Unit

Central display

Show value

Show name (below value)

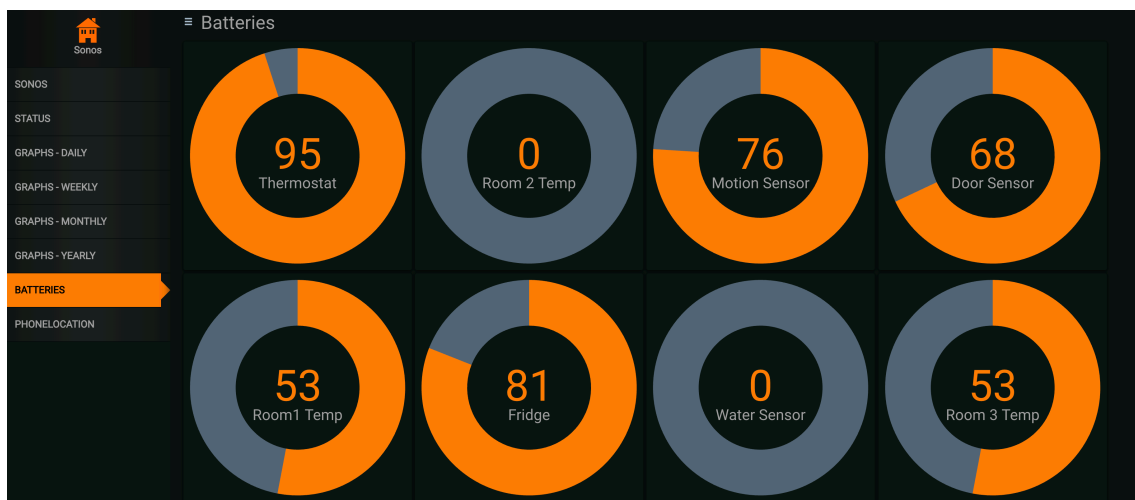
Format

Use server-provided format if available

Delete Cancel Save

II·lustració 37 - Configuració d'un element de tipus bateria

Després d'aplicar el mateix principi a tots els dispositius, en la següent il·lustració es pot veure el resultat del panell amb els diferents valors del sistema.



II·lustració 38 - Panell de control de bateries

10. Configuració del motor de processament

En aquesta darrera fase de configuració, es detallaran tots els components involucrats en la configuració i parametrització del motor de processament. A diferència de la fase anterior, on existien components de diversos fabricants i calien molts passos addicionals per a configurar-ho tot apropiadament, l'únic punt d'interacció amb el sistema serà OpenHAB.

10.1. Funcionament general de Node-RED

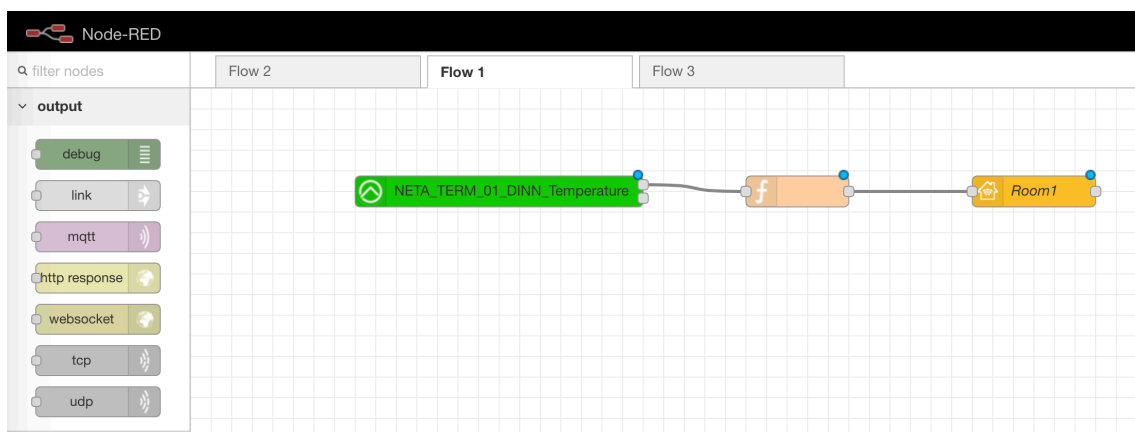
Node-RED és un sistema que permet definir escenaris utilitzant fluxos de treball. Així, partint d'un punt d'entrada (Input), la informació viatja entre blocs on s'apliquen una o més transformacions (function) per a determinar una sortida concreta (output).

La informació s'emmagatzema durant el flux en un objecte de tipus JSON anomenat msg que conté per defecte dos elements:

- `_msgid`: identificador únic del missatge
- `payload`: contingut del missatge

Durant les transformacions, es pot modificar el payload del missatge així com afegir o treure elements en funció de les necessitats de l'usuari.

S'utilitzarà com a exemple per explicar el funcionament general el flux mostrat en la següent il·lustració. Aquest pren com a paràmetre d'entrada el valor d'un dels sensors d'OpenHAB, aplica una transformació i produeix com a sortida un element de tipus HomeKit.



Il·lustració 39 - Exemple fluxe a Node-RED

En aquest cas, el bloc d'entrada representa la temperatura del menjador. Fent doble click, s'accedeix a la configuració que permet seleccionar el sensor concret. La sortida del bloc es connecta a l'entrada d'un bloc de tipus funció que realitza una transformació amb l'objectiu d'adequar el format del missatge a l'esperat per l'últim mòdul de Homekit. Finalment, aquest últim mòdul publica l'element per a poder ser utilitzat per Siri.

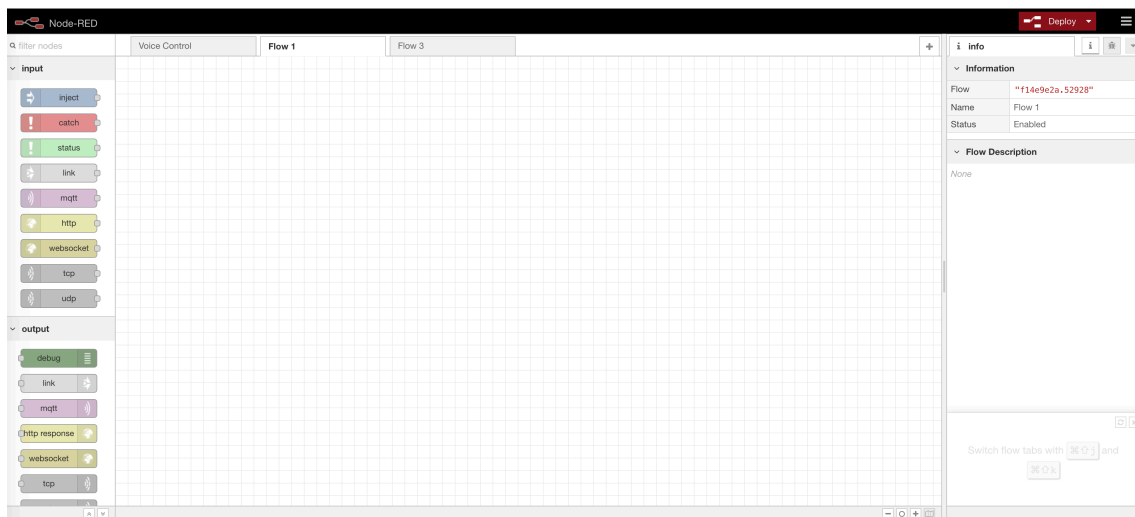
Les funcions són molt senzilles d'entendre i es poden desenvolupar en llenguatge Javascript. A continuació, es pot veure una que modifica el payload generant un element de nom CurrentTemperature amb el payload proporcionat pel bloc anterior.

```
msg.payload = {"CurrentTemperature":parseFloat(msg.payload)}  
return msg;
```

10.2. Instal·lació

NodeRED és la plataforma escollida, ja que permet tenir un control absolut sobre les interaccions i definir els fluxos d'interacció de manera visual. La instal·lació és molt senzilla, un únic paquet de NodeJS configurarà tota l'aplicació i iniciarà la interfície web tal com es pot comprovar a la il·lustració 39.

```
sudo npm install -g --unsafe-perm node-red
```

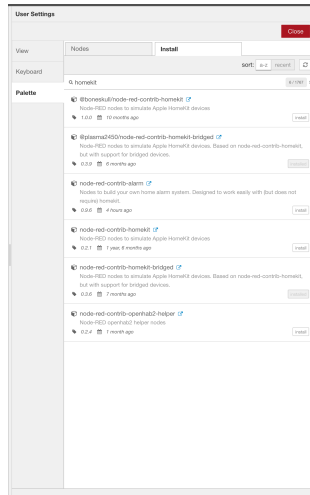


Il·lustració 40 - Aspecte de la interfície NodeRED

Igual que en el cas d'OpenHAB, NodeRED disposa d'un conjunt de components predefinits que es poden ampliar mitjançant paquets que la comunitat desenvolupa. En el cas del projecte i tenint en compte els objectius, caldrà carregar els següents connectors:

- node-red-contrib-openhab2: Interacció OpenHAB-NodeRed
- node-red-contrib-telegrambot: Interacció NodeRed-Telegram
- node-red-contrib-homekit-bridged: Interacció NodeRed-HomeKit

Tots els connectors es poden carregar des de la interfície d'usuari utilitzant l'opció manage palette i realitzant la cerca oportuna. Tal com es mostra en la següent figura, existeixen múltiples connectors per a gestionar una acció concreta, caldrà seleccionar la millor opció tot comprovant les funcionalitats a la web de l'autor.



Il·lustració 41 - Cerca de connectors a NodeRED

Els fluxos a NodeRED es realitzen mitjançant blocs que s'encadenen de manera que donada una entrada, es realitzin les operacions necessàries per aconseguir la sortida desitjada. Tot i que existeixen paquets per a realitzar gairebé qualsevol operació, moltes vegades cal fer una operació molt concreta no disponible en cap mena de bloc. El següent connector permet definir transformacions utilitzant simplement codi JavaScript.

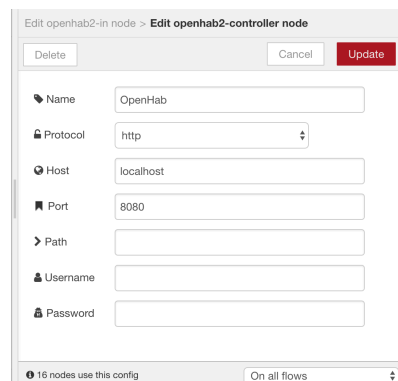
- node-red-contrib-simplejs: Definició de funcions utilitzant Javascript

10.3. Configuració dels connectors

Alguns dels connectors requereixen una parametrització específica com ara adreces IP, usuaris o contrasenyes. L'objectiu d'aquesta secció és detallar la configuració dels mòduls involucrats en la gestió del sistema.

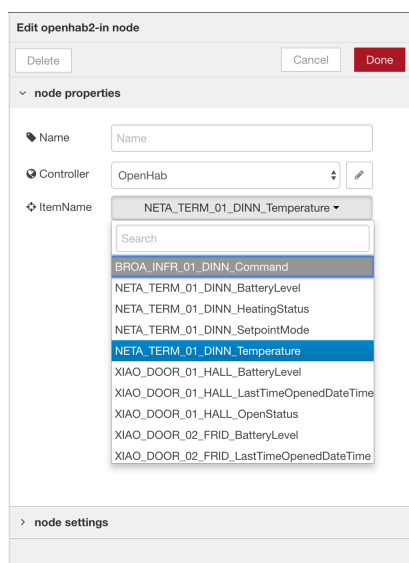
10.3.1. OpenHAB

De manera global, s'ha de donar d'alta el controlador amb les dades de la instància específica d'OpenHAB. Aquesta operació es pot fer afegint un bloc de tipus OpenHAB, fent doble click i escollir l'opció "Add new openhab2-controller". Com que Node-RED i OpenHAB estan a la mateixa xarxa, no cal protegir el sistema amb usuari i contrasenya, només caldrà especificar el host i el port per defecte tal com es mostra en la següent il·lustració.



Il·lustració 42 - Configuració OpenHAB a Node-RED

Un cop realitzada aquesta operació, a la casella inferior apareixeran automàticament llistats tots els Items declarats a OpenHAB perquè l'usuari esculli aquell que vulgui utilitzar.



Il·lustració 43 - Llistat de Things disponibles a Node-RED

10.3.2. Homekit

Apple proposa dos tipus d'elements dins HomeKit. D'una banda, existeixen aquells elements encarregats de realitzar una funció específica com ara mesurar temperatura o encendre un llum. Cada element disposa d'un codi que s'ha d'introduir a l'aplicació mòbil per a donar d'alta el dispositiu i que aparegui a la consola.

D'altra banda, existeixen elements que permeten englobar més d'un dispositiu. HomeKit demana un únic codi i es presenta el llistat de tots els dispositius adjunts disponibles perquè l'usuari incorpori al sistema aquells que més li convinguin. Aquest mode es coneix com a Bridge.

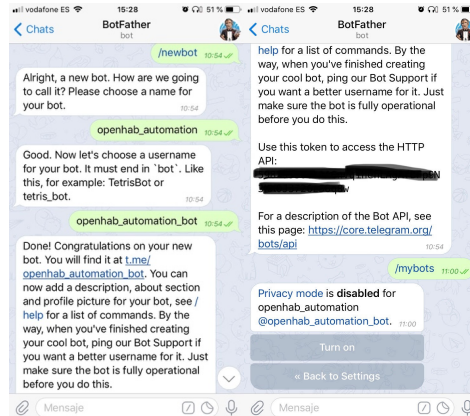
Donat el gran nombre de dispositius disponibles, s'utilitzarà el connector de Homekit en mode bridge per a facilitar la seva inclusió. Com en el cas d'OpenHAB, caldrà configurar l'element central anomenat en aquest cas Bridge i que permet definir el codi que s'haurà d'introduir al mòbil així com paràmetres opcionals extres com ara fabricant o nom.

10.3.3. Telegram

Telegram és la plataforma escollida per a interactuar amb el sistema mitjançant el servei de missatgeria, ja que permet crear bots. Un bot és un usuari estàndard de Telegram pensat per a ser controlat mitjançant un sistema extern, permetent així respondre a les peticions de l'usuari en funció de paràmetres prèviament establerts.

La primera acció que s'haurà de realitzar és crear aquest bot. Utilitzant l'aplicació de mòbil, s'estableix un nou xat amb l'usuari @BotFather. Aquest

usuari especial és el mecanisme que proporciona Telegram per a crear nous bots. En teclejar la comanda “/newbot”, el sistema respondrà demanant el nom que es vol donar al bot així com l’usuari. Com a resultat, tal com es mostra en les següents il·lustracions, es generarà el bot i es proporcionarà el codi que s’haurà d’utilitzar a les crides HTTP per a gestionar la interacció.



Il·lustració 44 - Generació bot a Telegram

Per últim, es crearà una sala de xat dedicada on s’inclourà el bot i que permetrà a més d’un usuari interactuar amb ell i rebre informació referent al sistema. Per tal de configurar després la sala de xat concreta a Node-RED, caldrà extreure el chatId. Aquest paràmetre és un nombre enter que es pot obtenir després de crear una sala de xat, afegir el bot i accedir a la següent URL.

```
https://api.telegram.org/bot<YourBOTToken>/getUpdates
```

Entre altres paràmetres, la resposta contindrà el valor que s’haurà de configurar indicat en l’exemple en color taronja.

```
{"update_id":8393,"message":{"message_id":3,"from":{"id":7474,"first_name":"AAA"}, "chat":{"id":XXXXXX, "title":""}, "date":25497, "new_chat_participant":{"id":71, "first_name":"NAME", "username":"YOUR_BOT_NAME"}}
```

Un cop el bot s’ha creat, només caldrà declarar-ho a Node-RED afegint un element de tipus Telegram i configurant els paràmetres BotName, Token i ChatID.

11. Definició d'escenaris complexos

Ara que s'ha presentat Node-RED i s'ha realitzat la configuració inicial dels diferents blocs que s'utilitzaran, es detallarà la definició dels següents escenaris complexos definits en els objectius del projecte.

- Encendre la calefacció en cas que la temperatura sigui inferior a 21 graus.
- Encendre l'aire condicionat en cas que la temperatura sigui superior a 27 graus.
- Apagar i encendre un llum situat al pati durant 5 minuts en cas que es detecti moviment al voltant..
- Enviar una notificació al mòbil en cas d'inundació.

11.1. Encendre la calefacció

El punt de partida d'aquest automatisme serà el sensor de temperatura del menjador, tot i que si es desitja es podrien afegir més d'un sensor. Això permetria per exemple encendre la calefacció en cas que qualsevol cambra estigues a una temperatura inferior a 21 graus.

El valor de sortida del bloc OpenHAB és una cadena de tipus text on s'inclou la temperatura decimal en format flotant i els caràcters °C. La primera transformació que s'haurà de realitzar és aplicar la següent funció per a aïllar el valor enter.

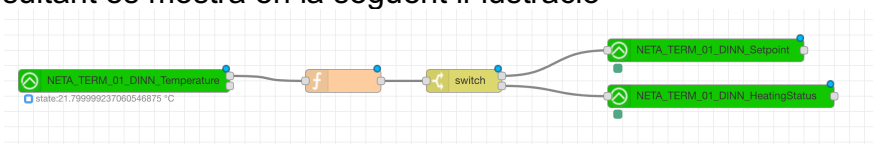
```
msg.payload = parseInt(msg.payload)
return msg;
```

El següent pas serà decidir si la temperatura del sensor ha d'encendre o apagar la caldera. Aquesta acció es realitza mitjançant un bloc de tipus switch que contindrà dos possibles sortides, la primera s'activarà en cas que el valor enter sigui menor o igual a 21, la segona en cas que sigui major.

En el primer cas, es modificarà la temperatura desitjada cridant l'item NETA_TERM_01_DINN_Setpoint en mode edició (ItemCommand) i estipulant un valor (payload) de 22.

En el segon cas, s'enviarà la comanda OFF al ítem NETA_TERM_01_DINN_HeatingStatus que farà que immediatament s'apagui la caldera independentment de l'estat actual.

El flux resultant es mostra en la següent il·lustració



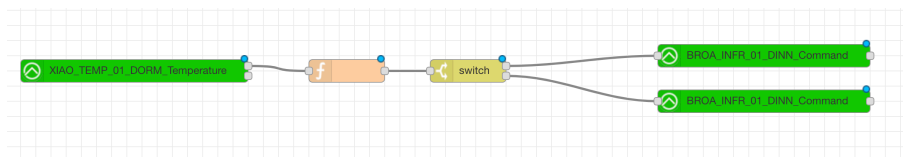
Il·lustració 45 - Gestió de caldera amb Node-RED

11.2. Encendre l'aire condicionat

De manera semblant a la caldera, caldrà definir com a entrada el valor d'un dels sensors de temperatura. Aquest es transformarà per adequar-ho a un enter i en aquest cas el bloc switch quedarà estipulat en dues opcions.

La primera s'activarà quan la temperatura sigui igual o superior a 27 graus i enviarà a l'Item BROA_INFRA_DINN_Command la comanda 21. Un cop es processi per OpenHAB i gràcies al mapa de comandes de l'infraroig, es transformarà en la instrucció necessària per activar l'aire condicionat.

La segona s'activarà quan la temperatura sigui inferior a 27 graus i enviarà la comanda OFF a l'Item BROA_INFRA_DINN_Command resultant en apagar l'aire condicionat.

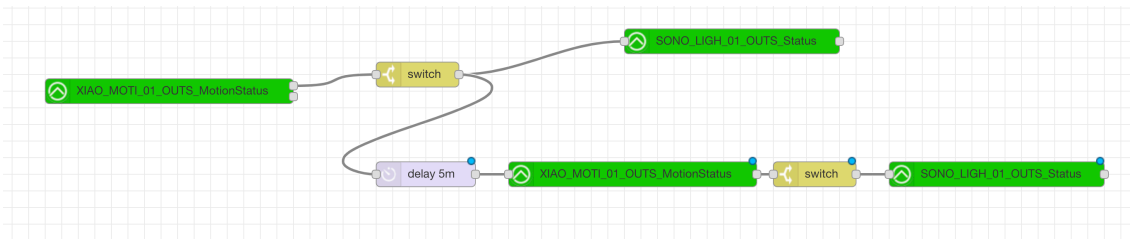


Il·lustració 46 - Gestió d'aire condicionat amb Node-RED

11.3. Apagar i encendre un llum segons el sensor de moviment

Molt semblants als anteriors casos, es declara com a punt d'entrada el valor del sensor de moviment. En aquest cas no cal fer transformacions, els valors que pot prendre són ON o OFF.

En cas que sigui ON, s'envia la comanda ON al llum exterior i a més, s'activa un bloc de tipus delay. Aquest fa que s'activi el bloc connectat a ell passat un temps determinat, en aquest cas, 5 minuts. Just després es farà una última comprovació, en cas que no hi hagi moviment, s'enviarà una comanda OFF al llum exterior.



Il·lustració 47 - Gestió de llums amb Node-RED

11.4. Enviar notificació al mòbil en cas d'inundació

Existeixen diverses maneres de resoldre aquest cas d'ús. Es pot utilitzar Telegram per informar l'usuari en cas que es detecti una fuga d'aigua, però es corre el risc que les notificacions estiguin desactivades, ja que l'aplicació es pot utilitzar per a altres tasques a més del control de la domòtica per text.

L'opció més convenient és emprar l'aplicació per a mòbils d'OpenHAB per a rebre notificacions utilitzant la plataforma MyOpenHAB. Donat que només es configurarà aquesta notificació, és un bon mecanisme perquè l'usuari sàpiga que l'avís és important.

Per contra, la gestió de regles a OpenHAB no és tan directa com a Node-RED. Les regles es defineixen en fitxers de text que s'han de copiar a `/opt/openhab2/conf/rules` amb l'extensió `.rules`.

La regla corresponent a la notificació en cas de fuga d'aigua és molt senzilla, si es detecta un canvi en l'estat del sensor i aquest retorna el valor ON (fuga detectada), s'enviarà una notificació amb la frase "Water Leak!!!!".

Existeixen diverses funcions de notificacions:

- `sendNotification("Email de l'usuari","Missatge")`
En aquest cas, s'enviarà la notificació únicament a l'usuari d'OpenHAB que s'hagi vinculat amb l'adreça de correu estipulada. Es recorda que poden existir diversos usuaris utilitzant MyOpenHAB cadascú amb la seva pròpia adreça de correu.
- `sendBroadcastNotification("Missatge")`
S'enviarà la notificació a tots els usuaris d'OpenHAB.
- `sendLogNotification("Missatge")`
S'enviarà la notificació a MyOpenHAB però no als dispositius mòbils.

En el cas del projecte, es decideix enviar la notificació a tots els usuaris. El codi de la regla es pot veure a continuació.

```
rule "push"
when
  Item XIAO_WATE_01_KITC_LeakDetected received update
then
  If (XIAO_WATE_01_KITC_LeakDetected == 'ON') {
    sendBroadcastNotification("Water Leak!!!!")}
end
```


12. Control per missatgeria

En aquest últim capítol, es detallaran totes les accions realitzades amb l'objectiu de poder controlar tot el sistema de domòtica amb Telegram utilitzant el bot creat en l'apartat 10.3.3.

Primerament, es defineixen tota una sèrie de comandes que són les que faran que els fluxos de Node-RED s'activin. Aquests són:

- /temperature1
- /temperature2
- /temperature3
- /light1
- /light2
- /door
- /fridge
- /rollershutter
- /heater
- /ac

12.1. Elements passius

En el cas dels sensors de temperatura o estat de les portes, el bot només haurà de respondre al xat informant de la temperatura captada per cadascun dels sensors o informar si la porta es troba oberta o tancada. Cada bloc de Telegram de tipus comanda s'enllaça amb el seu sensor corresponent. De manera general, quan OpenHAB rep la petició, retorna el valor en un element de nom State dins el payload.

La següent part del flux és enviar el valor llegit pel sensor a Telegram. Primerament, s'ha de crear un text amb el valor del sensor rebut d'OpenHAB amb el format esmentat anteriorment utilitzant una funció de Node-RED.

```
msg.payload.content = "Temperature is: "+msg.payload.state
return msg;
```

El bloc de resposta de Telegram espera sempre com a paràmetres d'entrada l'identificador de xat, el tipus de contingut i el cos del contingut. Es defineix una funció utilitzada per tots els escenaris que ho gestiona automàticament.

```
msg.payload.chatId = -XXXXXXXX
msg.payload.type = "message"
return msg
```

12.2. Elements actius

En el cas de llums, persianes, aire condicionat i calefacció el sistema ha de permetre interacció. D'una banda, s'ha de poder saber l'estat en què es troben els elements, d'altra, modificar el seu valor. Aquesta doble funcionalitat s'aconsegueix amb un bloc de tipus switch que detecta si l'usuari ha establert algun paràmetre addicional a la comanda. En cas que no (/rollershutter), es

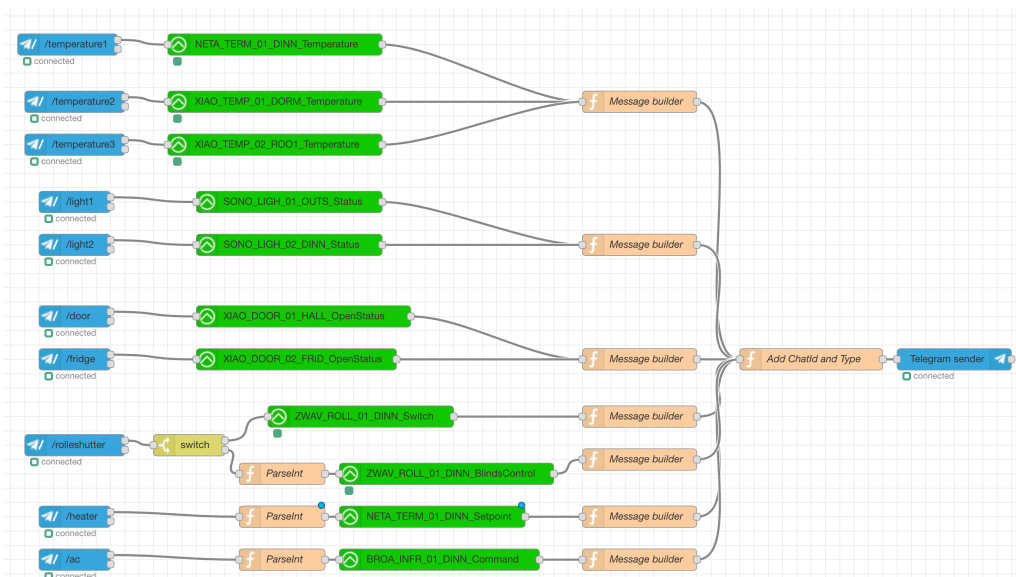
seguirà exactament el mateix patró descrit en els primers sensors. En cas que s'hagi definit, es modificarà el payload de la petició amb el valor desitjat i s'enviarà a l'objecte OpenHAB .

```
msg.payload = parseInt(msg.payload.content)
return msg
```

Per últim, caldrà també informar a l'usuari que la petició s'ha realitzat correctament. Com que el payload de la petició s'ha modificat per a contenir únicament el valor que OpenHAB espera, cal reconstruir-ho perquè la funció global de gestió de missatges de Telegram ho entengui.

```
msg.payload = {"content": "Rollershutter opened at: "+msg.payload+"%"}
return msg
```

Un cop implementades totes les funcionalitats, els fluxos de treballs resultants es mostren en la següent il·lustració.



Il·lustració 48 - Control amb Telegram a Node-RED

13. Conclusions

En l'àmbit del projecte, les conclusions generals que es poden extreure és que encara no existeix el sistema perfecte per a gestionar un sistema de domòtica de manera senzilla. OpenHAB és un producte molt potent i juntament amb Node-RED permeten cobrir moltes necessitats però la seva configuració és complexa per un usuari sense coneixements avançats.

D'una banda s'han de cercar els connectors adients dins OpenHAB. Molts d'ells son relativament nous i existeixen encara molts errors. Tot i que la comunitat genera molt contingut, no té la mateixa qualitat que un producte comercial i això es nota si es comprova l'estabilitat. Un dia funciona tot i l'endemà, després d'una actualització, la meitat d'objectes deixen de funcionar i s'han de configurar de nou.

D'altra banda, NodeRED és molt més estable però la implementació d'escenaris relativament senzills comporten la creació de fluxos complexos si es volen tenir en compte totes les casuístiques possibles. Escenaris com ara encendre l'aire condicionat davant una certa temperatura esdevenen complexos si es vol tenir en compte no enviar l'ordre cada vegada que la temperatura varia uns miligraus.

Es vol fer especial menció a la integració de Node-RED amb missatges de Telegram. Si algun lector té l'oportunitat de generar algun flux comprovarà les immenses possibilitats que el sistema de bots poden oferir només amb uns quants blocs. Node-RED es va crear per a gestionar elements de la Internet of Things però es pot estendre'n l'ús a altres mons.

Dit això, l'objectiu personal que l'autor del projecte s'havia marcat s'ha completat. S'ha demostrat com és realment possible adquirir productes de diversos fabricants i interconnectar-los entre si independentment del seu origen.

Pel que fa a l'execució del projecte, gairebé totes les fases s'han completat segons la planificació estipulada tot i que la idea de realitzar la memòria durant aquests tres mesos en paral·lel no ha set bona. Executar un projecte és relativament senzill si es tenen les idees clares però documentar-ho tot és difícil. Definitivament haguera set millor reservar un temps de la planificació exclusivament per a realitzar la memòria.

Cadascuna de les tres fases del projecte (anàlisis, consola central i motor de processament) eren molt complexes i sense cap mena de dubte hi havia prou contingut per realitzar un TFG complet amb cadascuna d'elles. Com que la longitud de la memòria del treball és limitada i el temps també, s'ha intentat resumir tot el màxim possible amb l'objectiu que el lector ho entengui tot en termes generals.

S'havien definit 4 objectius dels quals 3 s'han finalitzat al 100% però 1 s'ha realitzat en un 50%. Les proves de concepte inicials referents al control per veu es van executar bé però en el moment d'implementar-ho han sorgit problemes

tècnics que no han permès la seva finalització. El connector Node-RED – HomeKit sembla que funciona però el mòbil moltes vegades no pot interactuar amb el servei publicat.

Com a línies de treball futur, es recomana esperar uns quants mesos per veure com evoluciona el mercat. Sembla que cada vegada més fabricants opten per integrar-se amb Amazon Alexa i ara que ja està disponible a Espanya pot ser una gran revolució. En aquest projecte no s'ha tingut en compte, ja que molts fabricants no havien adaptat encara els connectors per funcionar a Espanya en el moment de l'anàlisi de la consola central.

Com a consell final pels futurs usuaris de domòtica, es recomana no lligar-se a cap fabricant ni protocol. La competència és el que fa que les empreses innovin i presentin productes cada vegada millors. Els usuaris han de forçar els fabricants a què els seus productes operin entre ells igual que ho fan les pàgines web mitjançant APIs. Qui sap si algun dia es podrà dir a Alexa que enviï un missatge de Telegram a un bot que activi Node-RED que li digui a Siri que posi una cançó de Spotify a l'equip de música de casa mentre Google Home encén el forn...

14. Glossari

- **Zwave**: Protocol propietari dedicat a la gestió d'elements de domòtica amb funcionament en malla i multifreqüència en funció de la regió.
- **Zigbee**: Protocol obert dedicat a la gestió d'elements de domòtica amb funcionament en malla. Treballa en la freqüència de 2.4Ghz.
- **Bluetooth**: protocol de transferència de dades a curta distància i baix consum.
- **PAN**: Personal Area Network. Xarxes dedicades a gestionar dispositius d'ús personal.
- **Gateway**: Element capaç d'actuar com a passarel·la entre dispositius basats en diferents tecnologies.
- **USB**: Universal Serial Bus.
- **API**: Application Programming Interface. Conjunt de mecanismes que un component informàtic proporciona per interactuar amb ell des d'una font externa.
- **Relay**: Dispositiu capaç d'obrir i tancar un circuit elèctric.
- **RGB**: Red Green Blue. Nomenclatura per indicar que un dispositiu pot emetre llum utilitzant la combinació dels colors bàsics.
- **Java**: llenguatge de programació.
- **Python**: llenguatge de programació.
- **Skin**: conjunt de mecanismes que permeten canviar l'aparença d'un component.
- **YAML**: YAML Ain't Markup Language. És un conjunt de regles per a representar dades que seran processades per un ordinador de manera que siguin fàcilment llegibles per éssers humans.
- **TTL**: Transistor-Transistor Logic. Circuit electrònic que funciona amb un conjunt d'especificacions concretes. Entre d'altres, considera un voltatge de 0V a 0,8V com a 0 i de 2V a 5V com a 1.
- **IDE**: Integrated Development Environment. Aplicació que conté tots els components necessaris per a permetre el desenvolupament de noves aplicacions en un o més llenguatges.
- **MQTT**: Message Queue Telemetry Transport. Protocol de gestió de missatges entre sistemes distribuïts.
- **IP**: Internet Protocol. Identificar únic d'un ordinador dins una xarxa.
- **LAN**: Local Area Network. Xarxa informàtica localitzada en una mateixa ubicació.
- **Raspberry**: Microordinador de baix cost molt utilitzat en projectes domèstics.
- **NodeJS**: Llenguatge de programació basat en JavaScript.
- **JavaScript**: Llenguatge de programació.
- **Bot**: Usuari virtual gestionat per una màquina.

15. Bibliografia

URL: <https://es.wikipedia.org/wiki/X10>

Enllaç visitat el 15/10/2018

URL: <https://en.wikipedia.org/wiki/Z-Wave>

Enllaç visitat el 15/10/2018

URL: <https://justclickappliances.com/z-wave-vs-z-wave-plus/>

Enllaç visitat el 15/10/2018

URL: <https://thesmartcave.com/z-wave-vs-zigbee-home-automation/>

Enllaç visitat el 15/10/2018

URL: <https://es.wikipedia.org/wiki/Bluetooth>

Enllaç visitat el 15/10/2018

URL: https://www.protectamerica.com/home-security-blog/in-review/z-wave-vs-bluetooth-pros-cons_20788

Enllaç visitat el 17/10/2018

URL: <https://www.agara.com/en/products.html>

Enllaç visitat el 19/10/2018

URL: https://www.domoticz.com/wiki/About_Domoticz

Enllaç visitat el 22/10/2018

URL: <https://www.home-assistant.io>

Enllaç visitat el 22/10/2018

URL: <https://www.openhab.org/docs/#architecture-overview>

Enllaç visitat el 22/10/2018

URL: <https://ricveal.com/blog/primeros-pasos-mqtt/>

Enllaç visitat el 25/10/2018

URL: <https://pro.netatmo.com/eu/thermostat/installation>

Enllaç visitat el 28/10/2018

URL: <https://xiaomi-mi.com/sockets-and-sensors/>

Enllaç visitat el 05/11/2018

URL: <https://github.com/davorf/BlackBeanControl>

Enllaç visitat el 15/11/2018

URL: <https://stackoverflow.com/questions/32423837/telegram-bot-how-to-get-a-group-chat-id>

Enllaç visitat el 25/11/2018

URL: <https://www.howtogeek.com/394567/zigbee-vs.-z-wave-choosing-between-two-big-smarthome-standards/>

Enllaç visitat el 27/11/2018