



Universitat
Autònoma
de Barcelona



escola
d'enginyeria

4793: Millora en la creació i configuració automatitzada de dispositius a la xarxa lliure guifi.net

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica

realitzat per

ALBERT SARLÉ PUIG

i dirigit per

DIEGO JAVIER MOSTACCIO MANCINI

Bellaterra, 14 de Juny de 2012

El sotasignat, Diego Javier Mostaccio Mancini
Professor/a de l'Escola d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat
realitzat sota la seva direcció per en Albert Sarlé Puig

I per tal que consti firma la present.

Signat:

Bellaterra, 14 de Juny de 2012

“mai no és massa tard per ser qui voldries haver estat”

George Eliot

Agraïments

A en Diego Mostaccio per acceptar dirigir aquest projecte i ajudar-me amb la seva experiència i recomanacions.

A en Miquel Martos per la gran feinada de programació que ha fet tots aquests anys a la web de Guifi.net i per la seva ajuda des dels primers moments d'aquest projecte. A en Lluís Dalmau i en Ramon Roca de la Fundació Privada per a la Xarxa Oberta, Lliure i Neutral Guifi.net per el suport des de la signatura del conveni fins a la redacció de la memòria. A tot Guifi.net, per ser tant i tant gran, fer-me conèixer moltes persones interessants i aprendre moltes coses noves (algunes no tecnològiques també).

A tots els amics que vaig fer durant la primera època a Informàtica, a Ciències i a Ganàpies, sense els que no seria qui sóc i molt especialment als que insistint, finalment em vareu convèncer de reprendre els estudis.

Als meus pares, a l'Emma i en Joan per estar al meu costat i a en Ton per posar-hi sentit a tot plegat.

Continguts

Agraïments	iv
Índex de Figures	vii
1 Introducció	1
1.1 Motivació	1
1.2 Proposta de projecte i objectius	2
1.3 Planificació inicial	3
1.4 Plataforma de desenvolupament i recursos	3
1.5 Anàlisi de viabilitat i riscos.	4
1.6 Estructura de la memòria	4
2 Introducció a Guifi.net	6
2.1 Introducció a la xarxa lliure Guifi.net	6
2.1.1 Filosofia	6
2.1.2 Funcionament de la comunitat	8
2.1.3 Eines	11
2.1.4 Programari	12
2.1.5 Evolució i reptes de futur	13
2.2 Anàlisi de l'estat de l'art actual	13
2.2.1 Com connectar: la recepta dels tres passos	13
2.2.2 Anàlisi tècnica	15
3 Anàlisi	17
3.1 Requeriments funcionals	17
3.2 Requeriments no funcionals	18
3.3 Diagrames i casos d'ús	19
3.3.1 Identificació dels actors	19
3.3.2 Noves entitats	19
3.3.3 Casos d'ús	20
3.3.4 Diagrama de fluxe general.	22
4 Disseny	23
4.1 Diagrames	23
4.1.1 Disseny de la base de dades	26
4.2 Avaluació dels sistemes de plantilles estàndard	27
4.3 Pla d'implementació	28
4.3.1 Gestió web	28
4.3.2 Nodes en mode client	29
4.3.3 Supernodes	29

4.3.4	Rutines de Validació	30
5	Implementació	31
5.1	Gestió web	31
5.1.1	Càrrega inicial de dades	33
5.1.2	Fabricants	34
5.1.3	Models	35
5.1.4	Firmwares	36
5.1.5	Paràmetres	37
5.1.6	Configuracions UnSolClic	37
5.1.7	Assignació de configuracions	39
5.2	Orígens i parametritzacions del node	39
5.3	Integració del sistema de plantilles	42
5.4	Nodes en mode client	45
5.4.1	Model Ubiquiti Nanostation2 - Firmware AirOs3.0	45
5.4.2	Model Linksys wrt54GL - Firmware dd-guifi	46
5.5	Nodes en mode Access Point i Supernodes	47
5.5.1	Model Mikrotik RouterBoard 433 - Firmware RouterOS4.7	48
5.5.2	Model Linksys wrt54GL - Firmware dd-guifi	50
5.6	Procés d'instal·lació	51
5.7	Dificultats sorgides	52
6	Resultats i proves	54
6.1	Distribució de maquinari a guifi.net	54
6.2	Comprovació automatitzada	55
6.3	Resultats de les proves	57
6.3.1	Anàlisi de desviacions i solucions aportades	58
7	Conclusions	60
A	Glossari	63

Índex de Figures

1.1	Diagrama de Gannt de la planificació del projecte	3
2.1	Mapa del desplegament de Guifi.net a Catalunya	9
2.2	Passos per afegir un node	14
2.3	Passos per afegir un aparell definint Fabricant, Model i Firmware	15
2.4	Passos per definir un enllaç en un node client	16
3.1	Diagrama de flux general	22
4.1	Diagrama Entitat Relació Extesa general	24
4.2	Diagrama EER Noves entitats	25
4.3	Diagrama de taules que originen l'UnSolClic	26
4.4	Diagrama de Taules de la BD	27
4.5	Aparells utilitzats en configuracions de nodes client	29
4.6	Mikrotik routerboard RB433	30
5.1	Pàgina inicial de gestió per al PFC	32
5.2	Llistat de Fabricants	34
5.3	Formulari de Fabricant	34
5.4	Llistat de Models	35
5.5	Formulari de Models	35
5.6	Llistat de Firmwares	36
5.7	Formulari de Firmwares	36
5.8	Llistat de Paràmetres	37
5.9	Formulari de Paràmetres	37
5.10	Llistat de configuració UnSolClic	38
5.11	Formulari de configuració UnSolClic	38
5.12	Fragment d'UnSolClic real d'un node client amb Ubiquiti Nanostation 2	45
5.13	Fragment de la plantilla d'UnSolClic per a Ubiquiti Nanostation 2	46
5.14	Fragment de la plantilla d'UnSolClic per a Linksys wrt54GL amb DD-guifi	47
5.15	Capçalera de la plantilla d'UnSolClic per a RouterBoard 433 amb RouterOS4.7	48
5.16	Definició dels servidors DNS de la plantilla d'UnSolClic per a RouterBoard 433 amb RouterOS4.7	48
5.17	Iteració sobre ràdios i condició de mode de la plantilla d'UnSolClic per a RouterBoard 433 amb RouterOS4.7	49
5.18	Crida a funcions del mòdul Guifi.net UnSolClic per a Mikrotik RouterBoard 433 amb RouterOS4.7	49
5.19	Definició de macro UnSolClic per a RouterBoard 433 amb RouterOS4.7	50
5.20	Llista de macros UnSolClic per a Linksys wrt54GL - DD-guifi	50
5.21	Consulta de la interfície WAN Linksys wrt54GL - DD-guifi	51
5.22	Codificació de l'actualització	51

5.23	Pujada del codi al repositori GIT de gitorious	52
5.24	Aplicació de l'actualització del PFC	52
5.25	Resultat de l'actualització amb el codi i dades del projecte	52
6.1	Totalització entitats	54
6.2	Total d'aparells per estat	55
6.3	Distribució d'aparells segons fabricant- model i firmware	55
6.4	Eina de comprovació de fitxers UnSolclics	56
6.5	Anàlisi de resultats amb Meld	58

Capítol 1

Introducció

En aquest capítol, després d'una breu descripció de la xarxa lliure **Guifi.net** que emmarca aquest PFC, trobarem la motivació que l'ha impulsat, així com la proposta que pretén millorar-ne la gestió i contribuir al seu desenvolupament. També inclou l'avantprojecte amb la planificació inicial, els recursos necessaris i l'estudi de viabilitat i riscos.

1.1 Motivació

L'any 2005 vaig començar a formar part de Guifi.net¹, impulsant la xarxa lliure en diferents vessants: la social (creant xarxa on no n'hi havia, ajudant a gent nova a connectar-se o organitzant xerrades, fent comunitat local), la d'instal·lador voluntari (muntant antenes a les teulades de forma voluntària i configurant enllaços entre ciutats) i la de blocaire (generant continguts als sites locals i escampant el projecte Guifi.net a les xarxes socials). Essent un projecte basat en programari lliure i jo un futur Enginyer en Informàtica, em faltava entrar a col·laborar activament en el desenvolupament de la pròpia plataforma, per això crec que aquesta és una bona oportunitat per aportar-hi el meu granet de sorra en forma de PFC.

En la vessant personal, em crida contribuir en un projecte de programari lliure on conèixer i practicar eines de programació en grup (GIT) i, per què no dir-ho, acabar la carrera i fer feliç la meva parella, els meus pares, els sogres i els amics a qui deuré un sopar.

¹<http://guifi.net>

1.2 Proposta de projecte i objectius

Guifi.net és una xarxa lliure de telecomunicacions construïda a iniciativa dels propis participants, ja siguin particulars, empreses, institucions o administracions que mitjançant un acord entre iguals s'interconnecten per compartir tota mena de serveis i recursos. En la base del seu ideari hi trobem la llibertat de l'usuari per connectar, utilitzar, comprendre i estendre la xarxa sense restriccions. L'agnosticisme tecnològic, que pregona que es pugui utilitzar qualsevol protocol o aparell mentre segueixi la filosofia global, en el cas del maquinari requereix d'eines que facilitin a l'usuari la seva configuració i integració a la xarxa. L'anomenat ²UnSolclic, és un configurador genèric de dispositius que ja ha demostrat amb escreix el seu paper clau en l'expansió de la xarxa³, però al qual li manca flexibilitat per incorporar els nous aparells.

Aquest projecte vol crear un sistema de gestió web que permeti augmentar la llista de dispositius disponibles sense haver de retocar el codi de l'aplicació, eliminant així el coll d'ampolla que suposa el procés actual d'incorporació de nou maquinari. L'increment dels dispositius suportats ha de contribuir a fer créixer la xarxa i augmentar la independència tecnològica respecte els fabricants.

Objectius

- **Nova Gestió.** Crearem un conjunt de noves entitats que ens permetran modelar els fabricants, els seus models, els firmwares que suporten, les característiques que incorporen i els paràmetres que admeten. Aquesta gestió avançada de dispositius permetrà que els usuaris amb un perfil més tècnic puguin definir i crear relacions entre aquestes entitats definint noves configuracions UnSolClic a les que podrem especificar una plantilla base que editarem des del propi gestor.
- **Sistema de plantilles.**

En la majoria de les webs actuals trobem que, per facilitar el manteniment i l'optimització de codi, es busca la manera de separar la presentació del contingut, el patró de disseny del Model-Vista-Controlador⁴ és el més utilitzat en la majoria de frameworks actuals tot i que també n'hi ha d'altres com per exemple el de Drupal⁵ Per implementar la capa de presentació és molt útil recórrer a un motor de plantilles estàndard que ens faciliti la feina. En el nostre sistema utilitzarem Twig⁶ per la seva potència i simplicitat.

²Veure entrada al glossari

³A data de 21/4/2012 figuraven 16234 punts connectats

⁴Patró MVC a wikipedia : <http://es.wikipedia.org/wiki/ModeloVistaControlador>

⁵Popular gestor de continguts de programari lliure : <http://drupal.org>

⁶Motor de plantilles de programari lliure : <http://www.twig-project.org>

1.3 Planificació inicial

El projecte s'ha planificat per dur-lo a terme en 6 mesos, considerant el 14/11/2011 com a data inicial, i es compon de les següents etapes on s'han treballat cadascun dels aspectes mencionats

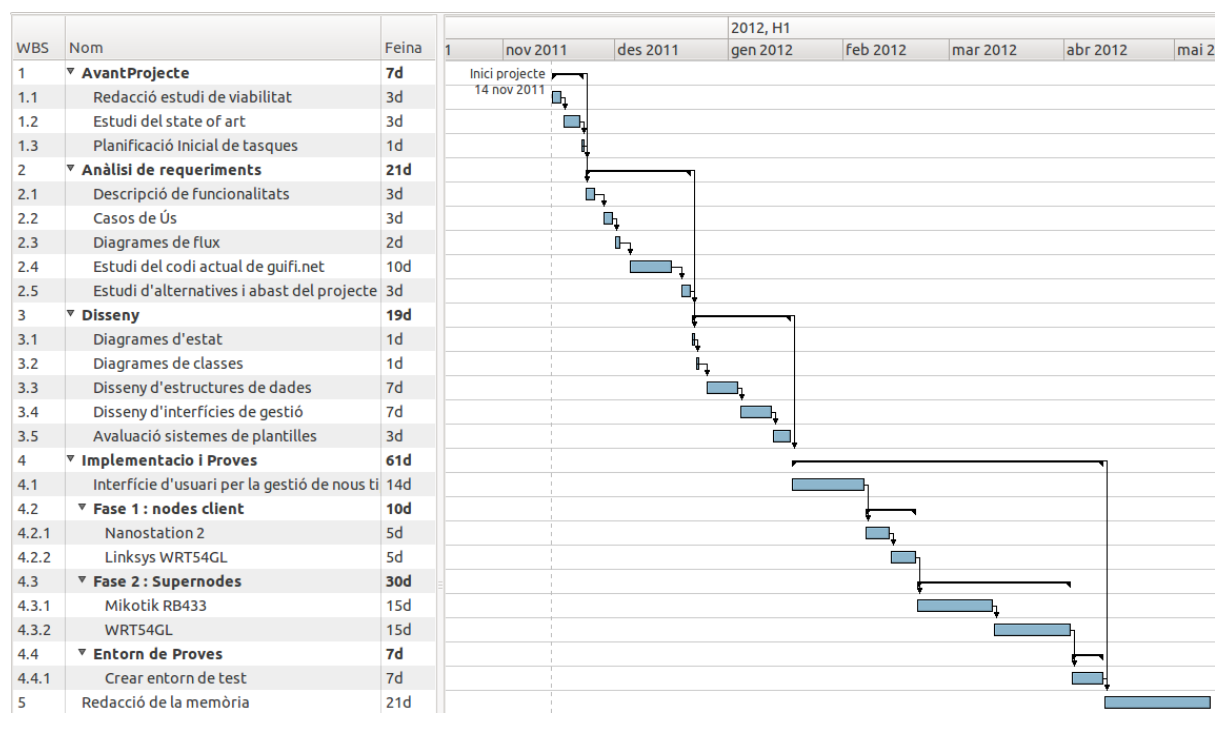


FIGURA 1.1: Diagrama de Gantt de la planificació del projecte

1.4 Plataforma de desenvolupament i recursos

El sistema de gestió proposat haurà d'integrar-se amb la plataforma actual de Guifi.net i per tant, en compartirà els requisits, determinant també la plataforma de desenvolupament.

Tecnologies/llenguatges de programació: PHP v 5.03, MySQL 5.1.30, Twig 1.6.

Entorn de desenvolupament: Ubuntu 11.10, Eclipse 3.5, MySQLWorkWench, SmartGit.

Servidor web : LAMP Stack, Drupal CMS, CVS GIT en gitorious.org, Modul drupal Guifi.net, Web de proves test.guifi.net, Web de producció de Guifi.net.

Els recursos utilitzats en el desenvolupament d'aquest projecte són els següents:

- Ordinador amb entorn local de desenvolupament de Guifi.net : Costos associats : 500 €

1.5 Anàlisi de viabilitat i riscos.

Riscos segons el tipus

Tecnologia: Ni el wifi ni els voltatges dels aparells presenten cap risc.

Relatiu a persones: Al ser un projecte obert, podria passar que no seduís als possibles contribuïdors .

Organitzacionals: Guifi.net és una estructura molt des-jerarquitzada.

Eines: S'utilitza programari lliure, amb ampli suport i documentació de la comunitat.

Requeriments: El projecte és propi i no té requeriments externs que puguin canviar.

Estimació: Potser el més elevat al haver-hi molts aparells disponibles als que assumim una complexitat similar en la codificació de les plantilles.

Motius de viabilitat

- Hi ha una demanda per part de la comunitat, directament del desenvolupador principal de l'aplicació de Guifi.net i indirectament de tots els usuaris que dia a dia utilitzen l'aplicació tant per fer-la créixer com per vetllar pel seu correcte funcionament.
- El desenvolupament es planifica per complir uns requisits mínims que permetin la seva integració a la plataforma de producció.
- El projecte permetrà futures iteracions en el desenvolupament que estenguin el concepte més enllà del proposat en aquest PFC.
- Es documentarà el procés d'implementació i es redactarà un manual d'usuari avançat perquè la comunitat pugui aportar noves configuracions en el futur.

1.6 Estructura de la memòria

En el primer capítol trobarem una introducció al PFC .

En el segon capítol presentaré el funcionament de la xarxa lliure Guifi.net, tant en la vessant social com en la tècnica, incidint en les eines que aporta Guifi.net i avaluant l'estat de l'art del procés actual d'incorporació de nou maquinari i les seves limitacions.

En el tercer capítol estudiarem els requeriments funcionals i no funcionals que volem que compleixi la nostra solució i acotarem els casos d'ús que ha de cobrir.

En el quart capítol dissenyarem l'estructura de les noves entitats de gestió que introduirà aquest projecte i les detallarem en diagrames que ens indicaran el disseny de la base de dades. També marcarem quin serà el pla d'implementació a realitzar.

En el cinquè capítol desgranem els passos de la implementació: crear la interfície i la lògica de la gestió de les noves entitats, inicialitzar i parametritzar-les amb les dades que reflexin els comportaments requerits, integrar el sistema de plantilles Twig i comentar el mecanisme de funcionament, i veure dos exemples d'adaptació de plantilles per a nodes client i dos més per a supernodes.

En el sisè capítol presentarem un mecanisme automatitzat de proves que ajudarà a validar les noves plantilles que es vagin confeccionant i en comentarem els resultats, aplicant-lo sobre els exemples de plantilles configurades en el capítol d'implementació.

En el setè capítol trobarem les conclusions del projecte i les futures línies d'actuació que podria obrir la integració d'aquest projecte dins de la plataforma de Guifi.net

Finalment, als annexes hi trobarem un glossari de termes utilitzats en aquesta memòria.

Capítol 2

Introducció a Guifi.net

En aquest capítol trobarem una introducció a la xarxa lliure Guifi.net, imprescindible per contextualitzar les necessitats que motiven el projecte i enfocar els punts clau on pretén incidir. Per conèixer altres vessants de la xarxa que no tractarem aquí, podem consultar les pàgines de documentació referenciades.

2.1 Introducció a la xarxa lliure Guifi.net

Guifi.net és una xarxa lliure de telecomunicacions construïda a iniciativa dels propis participants, ja siguin particulars, empreses, institucions o administracions que, mitjançant un acord entre iguals, s'interconnecten per compartir tota mena de serveis i recursos.

2.1.1 Filosofia

A principis de l'any 2004, a la població de Gurb¹, la necessitat de connexió a internet i la deixadesa de les operadores en donar servei de banda ampla a zones rurals, va fer que un parell d'usuaris decidissin interconnectar-se entre ells mitjançant la tecnologia més assequible de que disposaven, l'estàndard 802.11b conegut com a wireless. Des del primer moment es va entendre aquesta connexió com una auto-prestació de servei, constituint un dels pilars de la filosofia de Guifi.net.

Aquest fet però no era pas cap novetat ja que l'any 2000 havia sorgit Seattle Wireless² com una de les primeres comunitats que tenien a l'ideari crear una xarxa lliure, independent de les operadores, que permetés que tothom s'hi connectés de forma fàcil i econòmica. La filosofia d'aquesta i altres comunitats compartia conceptes de l'ètica hacker, del Do It Yourself³ i el Lo-fi per trobar solucions tecnològiques de baix cost i fàcil desplegament.

¹la G inicial de Gurb és la que dona el nom a Guifi.net

²<http://seattlewireless.net>

³En català, Fes-ho tu mateix com a filosofia i mètode d'aprenentatge

Comuns de la Xarxa Oberta, Lliure i Neutral ("XOLN") i vinculació amb el programari lliure

Els anys següents van aparèixer al nostre país diferents comunitats similars aïllades, entre les quals es troba Guifi.net, que es va definir ben aviat com una xarxa lliure, alternativa, oberta i neutral. En aquest context la seva aportació diferencial va ser definir les bases de les xarxes lliures mitjançant la redacció d'una llicència d'ús basada en les quatre llibertats del programari lliure, que va anomenar el Comuns de la Xarxa Oberta, Lliure i Neutral ("XOLN")⁴, i que textualment diu⁵:

1. Ets lliure d'utilitzar la xarxa per qualsevol propòsit mentre que no perjudicis el funcionament de la pròpia xarxa, la llibertat dels altres usuaris, i respectis les condicions dels continguts i serveis que lliurement hi circulen.
2. Ets lliure de saber com és la xarxa, els seus components, com funciona i difondre'n el seu esperit i funcionament.
3. Ets lliure d'incorporar serveis i continguts a la xarxa amb les condicions que vulguis.
4. Incorporant-te a la xarxa, ajudes a estendre aquestes llibertats en les mateixes condicions.

L'adhesió voluntària dels participants a aquesta llicència visualitza el compromís en mantenir aquest acord entre iguals, construint en el domini públic una infraestructura de propietat distribuïda i aportant un criteri per distingir-la de les xarxes tancades (o privatives).

Una altra aportació clau va ser confeccionar tot un seguit d'eines per tal de desplegar-les més fàcilment i adoptar els estàndards oberts disponibles com per exemple CNML⁶.

Agnosticisme tecnològic

Rellegint els punts del XOLN veiem que no fa referència a cap tecnologia concreta, i això és l'anomenat agnosticisme tecnològic, entès de manera que sempre s'utilitzarà la millor tecnologia disponible que compleixi les condicions. Per això si Guifi.net va començar com una xarxa wireless, va ser perquè el State Of Art del moment la feia la tecnologia més barata disponible, juntament amb el fet que utilitzava freqüències que no requerien llicència. En aquest sentit la xarxa ha anat adoptant els diferents protocols wireless que han anat sorgint, i quan ha estat possible, ha fet la transició a enllaços de fibra òptica.

Guifi.net no imposa requisits pel què fa a la vessant del programari, però evidentment la seva naturalesa fa que el programari lliure i els estàndards oberts siguin els més utilitzats i potenciats. Per la vessant del maquinari, evidentment tampoc imposa cap restricció tot i que,

⁴Inicialment va rebre el nom de Comuns Sense Fils

⁵<http://guifi.net/ComunsXOLN#ComunsXOLN>

⁶<http://www.cnml.info>, Community Network Markup Language Project

com veurem, sí que proporciona eines per a configurar certs models per als quals s'ha desenvolupat el suport.

2.1.2 Funcionament de la comunitat

Sota la denominació Guifi.net hi trobem diferents conceptes depenent del punt de vista. Per una banda, és una xarxa informàtica que actualment esta constituïda per més de 16.000 punts. Per una altra, és una comunitat d'usuaris que constitueix una veritable xarxa social vertebrada en diferents grups locals. També és una fundació sense ànim de lucre que actua com a operador de telecomunicacions i vetlla per la pròpia xarxa. Finalment, també és una web que aglutina i incorpora un munt d'eines que contribueixen al seu desenvolupament

La web <http://guifi.net> és el pal de paller de la comunitat, ja que és on trobem la documentació i els fòrums on acudim quan volem formar part de Guifi.net. Allà també hi trobem els enllaços a les llistes de correu de cada subzona i les webs de cadascuna de les agrupacions locals que empenyen el seu creixement.

Topologia de la xarxa lliure

Sense una estratègia definida de desplegament, la xarxa lliure ha anat creixent a mida que s'hi anaven incorporant persones i grups interessats en formar-ne part, molt sovint amb nous nuclis aïllats que amb els temps han anat creixent fins a connectar amb el gruix de la xarxa troncal. Simultàniament, la xarxa s'esta començant a estendre per altres parts del país de forma testimonial i també per altres països.

Tot i que a Guifi.net hi conviuen diferents topologies de xarxa, històricament podem classificar els aparells que la formen en els següents tipus :

- *Node client*: Punt final de xarxa que ofereix connectivitat en una ubicació. (constitueixen la majoria de punts de la xarxa)
- *Super Node*: Punt on conflueixen els enllaços punt a punt entre dues zones més grans. Ofereixen connectivitat als nodes client.

A la Figura 2.1 es pot veure el desplegament de Guifi.net a Catalunya, on hi trobem uns 13.500 nodes dels 16.300 totals a 5/5/2012. Les línies verdes representen enllaços punt a punt entre supernodes de diferents zones, constituint l'anomenada xarxa troncal, que vertebrada el gruix de Guifi.net, i les línies grogues representen connexions de nodes client. Tots sumats arriben als 30.000 Km d'enllaços.

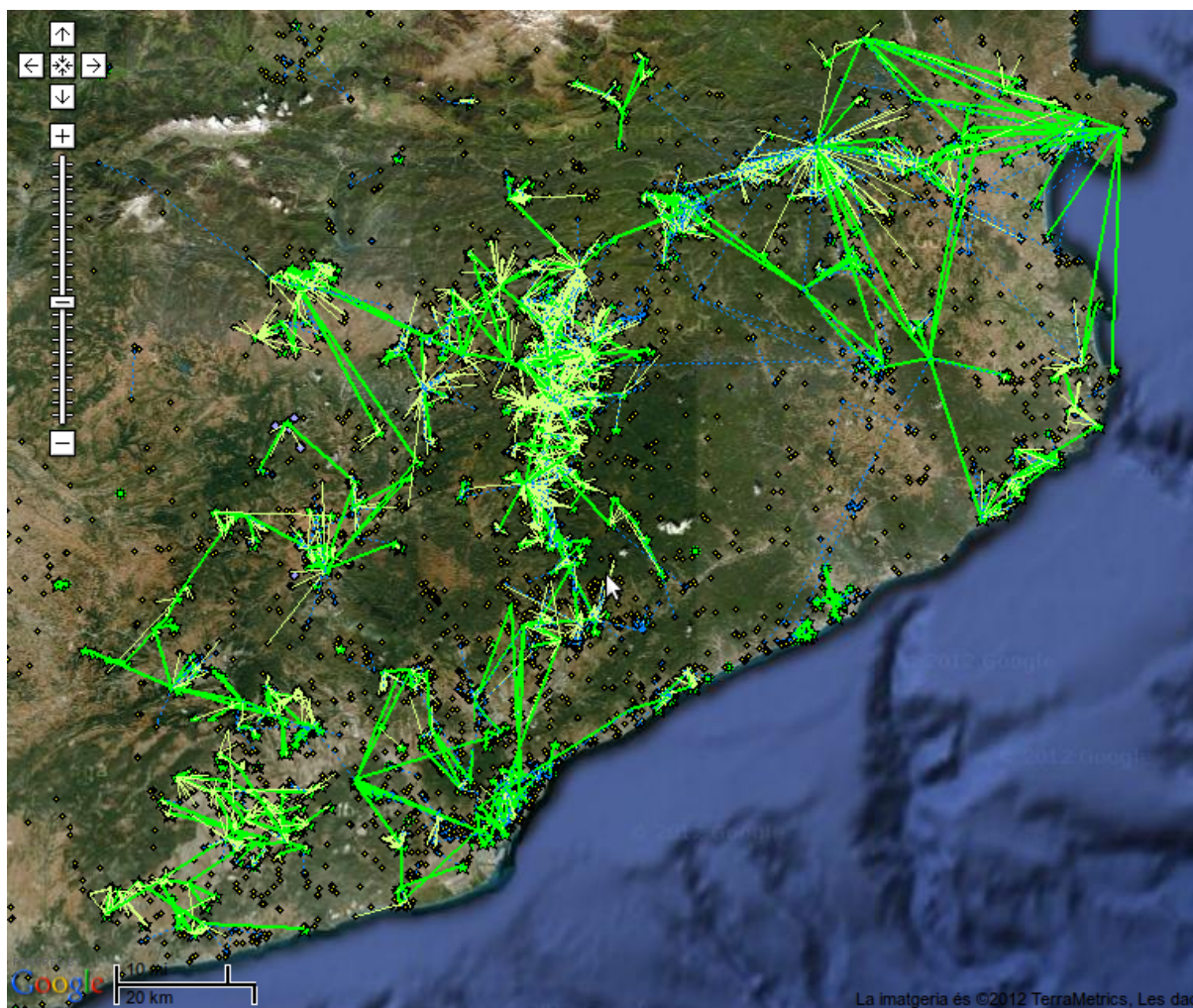


FIGURA 2.1: Mapa del desplegament de Guifi.net a Catalunya

La distribució de nodes es pot consultar en cadascuna de les pàgines resum de zona, per exemple a Catalunya⁷ o a la península Ibèrica⁸.

Infraestructura necessària i serveis disponibles

Per descriure de forma simplificada la infraestructura necessària per la connexió wifi a la xarxa Guifi.net, considerarem el cas d'un node client. Abans però cal tenir en compte que per establir enllaços wireless de bona qualitat cal tenir sempre visió directa⁹ amb el punt a connectar, és per aquest motiu que és habitual fer una instal·lació a la teulada. Així doncs que necessitem per connectar? :

- Router inalàmbric amb antena integrada, i preparat per la intempèrie, (per exemple Ubiquiti Nanostation2)

⁷<http://guifi.net/ca/Catalunya>

⁸<http://guifi.net/ca/node/17711>

⁹En anglès Line Of Sight(LOS)

- Cable de xarxa per transportar les dades fins a casa i alimentació mitjançant PoE(Power Over Ethernet)
- Suport físic per al router (un màstic o una fixació).

Amb la instal·lació física completada i la configuració indicada per la web traspassada al nostre aparell, ja estem connectats a Guifi.net i per tant podem accedir a tots els recursos que aporten els propis usuaris. A la mateixa web trobarem un directori amb tots els continguts disponibles¹⁰ agrupats per tipus : Servidors Web, Servidors FTP, Centraletes Asterisk, Càmeres en Xarxa, Servidors P2P interns, Streaming de radio..., i el més utilitzat, l'Accés compartit a internet mitjançant Servidors Proxy.

Aquest accés a internet, dins de la xarxa es considera un recurs més que els usuaris comparteixen en les condicions que determina qui el posa a disposició de la comunitat, ja sigui un particular, una empresa o un ens local. En molts casos es comparteix la connexió entre igual i en altres s'utilitzen recursos comunitaris disponibles(telecentres, biblioteques, etc) pel que a la pràctica molts usuaris gaudeixen d'un accés a internet sense cost.

Fundació Guifi.net com a Operador de comunicacions

Arran del premi Nacional de Comunicacions aconseguit el 2007 es va constituir la Fundació Privada per a la Xarxa Oberta, Lliure i Neutral Guifi.net que vetlla per mantenir-ne l'esperit obert i constitueix una eina de suport al projecte en totes les seves vessants: (voluntariat, entitat jurídica, projectes europeus, convenis amb universitats, entitats i empreses, projectes de joventut, ong, formació, ...)

Nou ecosistema per a empreses

Cal remarcar la importància que la necessitat d'accés a la banda ampla, juntament amb el fet que Guifi.net ha desplegat a molts llocs l'anomenada 'última milla' de connexió i el fet de donar cobertura al model de xarxa, la fundació es va constituir com a operador de telecomunicacions i, per tant, té els mateixos drets que qualsevol altre, per exemple tenir una connexió al punt neutre de Catalunya (CATNIX). Així doncs, l'accés compartit a internet que comentàvem també es pot suplir amb un accés sense ànim de lucre proveït per la Fundació que cal finançar entre els propis usuaris.

Això però no exclou que altres empreses es puguin constituir com a operadores de telecomunicacions dins el model obert de Guifi.net. Aquestes utilitzen la xarxa per proporcionar als seus clients els serveis de valor afegit pels quals obtenen un benefici econòmic

¹⁰<http://guifi.net/ca/node/3671/view/services>

que es reflexa en un millor manteniment i estabilitat. El model de dades públiques facilita que les diferents empreses puguin col·laborar entre elles i amb la comunitat.

Per altra banda, el creixement de la xarxa ha alimentat un nou ecosistema local d'empreses de serveis i equips que es nodreixen de les necessitats dels usuaris i que també són part important del seu desenvolupament i manteniment.

2.1.3 Eines

Per tal que tothom pugui conèixer com funciona la xarxa i fer-la créixer, és indispensable que en conegui detalls tècnics com les configuracions dels aparells; per això a la web de Guifi.net sempre trobarem disponible tota la informació possible i un conjunt d'eines per fer-ne el seguiment

Tota la gestió tècnica de la xarxa es coordina des de la web de Guifi.net, tant la reserva d'adreces, les configuracions dels enllaços troncal, com també les configuracions que han de tenir tots els routers que integren la xarxa.

Gestió

El programari desenvolupat per a Guifi.net ens aïlla de la vessant més tècnica de manera que, quan donem d'alta un nou punt amb un nou aparell, la web de Guifi.net s'encarrega tant de la reserva d'adreces com de les configuracions dels enllaços troncal i dels protocols d'enrutament utilitzats.

Monitorització

A la web tenim disponible la connectivitat de cadascun dels aparells que formen la xarxa; aquesta funcionalitat s'obté a partir dels anomenats servidors de gràfiques, ordinadors que extreuen de la web la informació dels nodes de la seva zona i fan una monitorització. Aquesta permet diagnosticar la qualitat dels enllaços i registrar el percentatge de connectivitat. També existeix d'una eina per visualitzar sobre un mapa les diferents rutes disponibles des d'un node concret (traceroute)

Configuració : l'UnSolClic com a factor d'èxit

Una de les conseqüències de la llibertat a l'escollir el maquinari és que podem utilitzar el router que vulguem i configurar-lo segons ens indiqui la gestió centralitzada de la web. Però això requeriria per part de l'usuari uns coneixements tècnics avançats del funcionament del seu

router que frenarien a molts possibles usuaris interessats. Per evitar-ho es va desenvolupar el "UnSolClic".

L'UnSolClic és un fitxer generat per la web que facilita la connexió a la xarxa de manera que es pugui carregar d'un sol cop tota la configuració necessària. Començant per indicar quin és el model del nostre router, la versió del firmware que incorpora i a quin punt de la xarxa el volem connectar, la web ens retornarà un script confeccionat d'acord amb la sintaxi de cada router, incorporant tots els paràmetres necessaris.

Podem afirmar sense cap dubte que aquest procés ha estat un dels factors clau de l'èxit i el creixement de Guifi.net.

2.1.4 Programari

Tot el codi que fa funcionar les eines de Guifi.net està publicat sota llicència GPL¹¹ i es pot trobar als repositoris públics de Guifi.net, actualment allotjats a gitorious.org.¹²

Al voltant de la web de Guifi.net, a mida que s'anaven connectant pobles veïns, s'anava construint una comunitat que requeria cada vegada més gestió, tant de la pròpia xarxa com de la informació necessària per estendre-la. Es va instal·lar un Drupal com a base de la comunitat i es van programar una sèrie de mòduls integrats que automatitzaven la gestió de la pròpia xarxa i feien pública tota la informació com a manera de permetre que tothom s'hi apuntés.

LAMP, Drupal, GIT, trac

La web de Guifi.net utilitza un servidor LAMP¹³ i un gestor de continguts Drupal al que se li ha integrat una sèrie de mòduls que encapsulen la lògica de funcionament de Guifi.net. El mòdul que incorpora les funcionalitats principals de gestió s'anomena Drupal -guifi¹⁴ i està acompanyat d'altres mòduls com per exemple el de la visualització de mapes.

Incorporació de nous aparells

En els darrers anys l'augment de la demanda i l'evolució del maquinari en l'àmbit de les xarxes wireless, han fet aparèixer molts fabricants que competeixen en preu i prestacions per posicionar al mercat els seus productes per a empreses i particulars. Aquest ràpid augment de dispositius disponibles fa de la programació del mòdul UnSolClic que els hauria de suportar, un coll d'ampolla per a l'adopció d'aquests dispositius a la xarxa lliure.

¹¹Llicència Pública General GNU

¹²Repositoris públics de Guifi.net: <https://gitorious.org/guifi>

¹³Linux, Apache, MySQL i PHP

¹⁴Drupal-guifi : <https://gitorious.org/guifi/drupal-guifi>

2.1.5 Evolució i reptes de futur

L'evolució de Guifi.net cap a altres mitjans de transmissió com la fibra òptica portarà la necessitat de interoperar entre un nombre més elevat de dispositius amb característiques i capacitats molt diferents, per això és necessari disposar d'un sistema de gestió dinàmic que permeti ampliar la base de dispositius pels quals Guifi.net dona suport.

2.2 Anàlisi de l'estat de l'art actual

2.2.1 Com connectar: la recepta dels tres passos

La intenció inicial de Guifi.net de facilitar el procés de connexió en uns passos molt simples per a l'usuari estàndard, va fer que es bategés el procés amb el nom de "Tres Passos". Són els següents:

1. *Afegir un node*: Consisteix en marcar la nostra ubicació en un mapa per obtenir distàncies i línies de visió amb altres punts de la zona. Veure la Figura 2.2
2. *Afegir un trasto*: Considerant que ja disposem del maquinari (històricament anomenat trasto a Guifi.net) amb el que volem realitzar la connexió, en aquest pas n'escollirem el mode de funcionament (node client o supernode) i triarem entre una llista de dispositius i firmwares suportats per l'aplicació. En cas d'afegir un supernode, es reserven els rangs de xarxa per aprovisionar les futures connexions de nodes client.

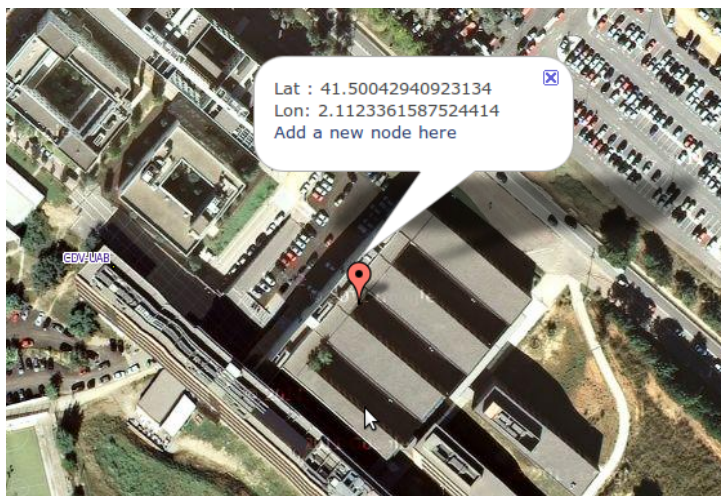
Com podem veure al gràfic, el formulari que presenta l'aplicació en el segon pas correspon a la única part visible a l'usuari en què incidirà aquest PFC, ja que filtrarà els valors vinculats que apareixen als combos de model i firmware (segons les configuracions validades per l'usuari administrador). Veure 2.3

3. *Definir un enllaç*, si l'aparell és en mode client indicarem a quin supernode connectarem i per tant de quin rang de xarxa obtindrem una adreça ip fixa, així com altres paràmetres imprescindibles per la connexió com el servidor DNS locals de la zona o el protocol d'enrutament utilitzat. Veure 2.3

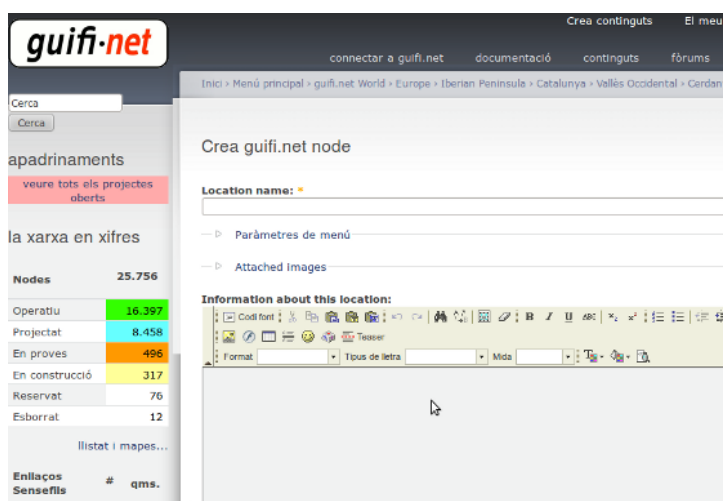
Quan hem completat aquests passos l'aplicació ha realitzat i emmagatzemat totes les configuracions necessàries per definir una connexió, aquesta es troba disponible permanentment a la web de Guifi.net actuant com un autèntic repositori o backup de configuracions.

Encara manca però traspasar aquesta configuració a l'aparell que hem seleccionat per connectar. Aquí és on intervé l'UnSolClic¹⁵, un script de configuració generat per l'aplicació i

¹⁵Exemple d'UnSolClic: <http://guifi.net/ca/guifi/device/2385/view/unsolclic>



(a) Marcar ubicació al mapa



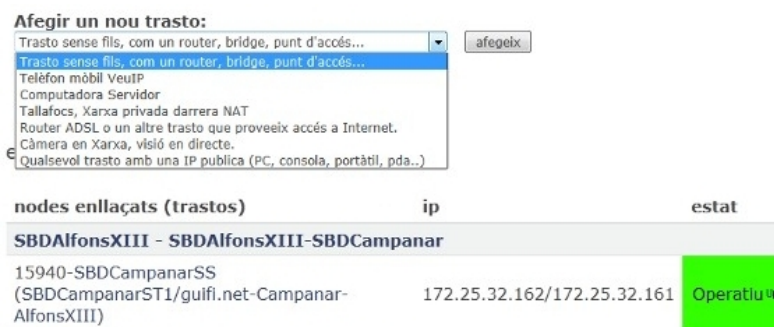
(b) Omplir dades bàsiques

FIGURA 2.2: Passos per afegir un node

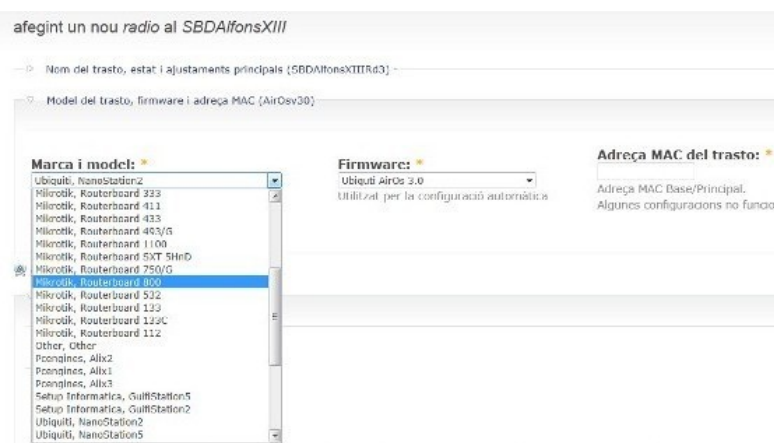
que codifica tots els paràmetres necessaris segons la sintaxi del firmware que incorpora el nostre model.

La càrrega d'aquest script al nostre aparell dependrà del mode de funcionament. Històricament es feia mitjançant una sessió telnet des de la qual s'enganxava tota la configuració, tot i que actualment és freqüent que es pugui carregar mitjançant un upload habilitat a la gestió de l'aparell (ja sigui web o amb una aplicació adhoc)

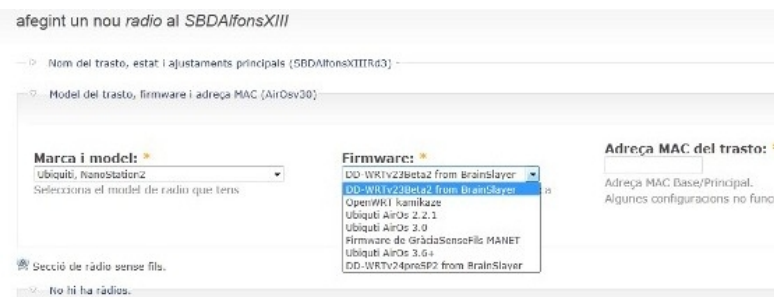
Després de carregar la configuració amb l'UnSolClic, el sistema ha deixat la màquina preparada per tenir connectivitat, tot assumint que la part de l'enllaç físic es farà en les millors condicions possibles (visió directa entre els punts, antenes i material adequats, etc. Així doncs, en zones amb molta densitat de xarxa no tindrem gaires dificultats en connectar, però en zones on la xarxa té pocs supernodes, existeixen impediments geogràfics o arquitectònics, o hi ha un espectre saturat d'interferències, caldria plantejar altres enllaços.



(a) Afegir Trasto



(b) Escollir Fabricant-Model



(c) Escollir Firmware

FIGURA 2.3: Passos per afegir un aparell definint Fabricant, Model i Firmware

2.2.2 Anàlisi tècnica

El sistema actual que implementa la lògica d'aquests passos, tot i haver estat extremament útil al llarg d'aquest anys, presenta algunes limitacions importants:

Limitacions

- Absència de gestió de dispositius: Malgrat que a la base de dades existeixen les taules de Fabricant i Model on s'han introduït a mà els valors corresponents, no existeix una interfície de gestió que permeti la crear, llegir, modificar i esborrar registres. D'altra banda, les claus primàries de Fabricant/Model i Firmware de cada dispositiu és troben

Model del trasto, firmware i adreça MAC (AirOsv30)

Marca i model: *
Ubiquiti, NanoStation2
Selecciona el model de radio que tens

Firmware: *
Ubiquiti AirOs 3.0
Utilitzat per la configuració automàtica

Secció de ràdio sense fils.

No hi ha ràdios.

Nova radio (mode)

Wireless client Afegir una radio nova

Usage:
Choose **wireless client** mode for a normal station with full access to the network. That's the right choice in general. Use the other available options only for the appropriate cases and being sure of what you are doing and what does for doing this.
You will not be able to define your link and get connected to the network until you add at least one radio.

(a) Afegir una radio a l'aparell

Radio #0 - client - BergoBRGVllDn1Rd1CPE0 - 1 interfície(s)

Ajustaments principals de la Ràdio (SSID, MAC, Canal...)

MAC: *
00:15:6D:E6:BE:58
Adreça MAC Sense fils.
Algunes configuracions no funcionen si es deixa en blanc.

Ajustaments de l'antena

Tipus (angle):
patch 30 degrees
Angle (depends on the type of antenna you will use)

Guany:
14
Millora (Db)

Graus (º):
0
Àzimut (0-360º)

Connexió a l'AP - 0 adreça(es)

(b) Definir propietats radio

Tria un Punt d'Accés de la llista per enllaçar amb *BRGVllDan1Rd1*

tot dades gràfiques interfície enllaços serveis traceroute unsolcllc **editar**

Filtres

Distància des de: 0 **fins al:** 10 **Cerca cadena:**

La llista comença a aquesta distància ...i acaba a aquesta distància Zona, node o trasto que conté aque:

select the device which do you like to link with:

- Berga , wds-BRGSambRGPsns**troncal** (0.894 Qms)
- Berga , BRGFmnyAP1 (0.945 Qms)
- Berga , BRGQrlTAP1 (1.2 Qms)
- Berga , guifi.net-BRGQueralTAP2 (1.2 Qms)
- Avià , AVMoliCastellAP (1.698 Qms)

(c) Escollir un supernode

FIGURA 2.4: Passos per definir un enllaç en un node client

serialitzats en un camp de text de la taula guifi-devices mitjançant un comportament proporcionat per Drupal, és a dir, sense el suport d'una estructura relacional.

- Anti-patró de disseny: Degut a l'evolució de l'aplicació, el codi font conté diverses parts que requereixen ser actualitzades quan es volen incorporar nous aparells.
- Dificultat en el manteniment i reutilització: La complexitat afegida derivada d'aquest anti-patró frena la incorporació de nous desenvolupadors, dificulta la comprensió de la lògica de l'aplicació i afavoreix l'aparició d'errors.

Capítol 3

Anàlisi

A l'estudi del State of Art del Capítol 2 hem vist les limitacions actuals de l'aplicació pel què fa a la gestió i el manteniment del maquinari i els impediments al desenvolupament que suposa. La proposta d'aquest projecte passa doncs per crear una estructura de dades nova i un nou gestor web que permeti flexibilitzar la gestió, així com proporcionar un sistema de plantilles que encapsulin les configuracions disponibles per als usuaris.

3.1 Requeriments funcionals

Entre els requeriments funcionals que volem per a la nostra proposta de gestió trobem els següents :

- *Oferir una gestió de fabricants*: Els usuaris avançats han de poder realitzar les operacions bàsiques de gestió CRUD¹ sobre les entitats dels fabricants de dispositius. Tot i que inicialment només guardarem les propietats del nom i la url, aquesta entitat serà imprescindible per contextualitzar les altres entitats relacionades.
- *Oferir una gestió de firmwares*: Els firmwares que existeixen actualment poden ser creats per un fabricant específicament per a una família de dispositius (com per exemple RouterOS) o es pot tractar de una versió de programari lliure que es pugui executar en models de diferents fabricants. Amb aquesta gestió, l'usuari avançat ha de poder fer CRUD i associar paràmetres genèrics dependents del firmware.
- *Oferir una gestió de dispositius*: Darrerament alguns fabricants estan incorporant molts models nous amb moltes innovacions. Amb aquesta gestió, l'usuari avançat podrà vincular dispositius a fabricants, assignar-los algunes propietats relatives a les seves característiques físiques i especificar quins són els firmwares que pot executar.

¹Acrònim en anglès per a Create/Read/Update/Delete

- *Especificació i gestió de plantilles*: L'usuari avançat ha de poder definir per a cada combinació de dispositiu-firmware una plantilla que servirà de base per a confeccionar el fitxer UnSolClic que utilitzaran tots els aparells de la xarxa que funcionin amb aquest firmware.
- *Suport als dispositius existents*: A la xarxa Guifi.net hi ha molts dispositius diferents, alguns dels quals són força vells però encara estan operatius. Per això caldria que tots ells poguessin ser gestionats amb la nova eina i tinguessin les seves pròpies plantilles.
- *Integració amb el CMS Drupal*: Tota la funcionalitat actual del programari de suport de la xarxa està vinculat en més o menys grau a un sistema de gestió de continguts Drupal. Per permetre la re-utilització de codi existent, per una coherència visual i de workflow que faciliti la feina als usuaris, cal que el gestor s'integri dins del Drupal en un menú propi per a usuaris avançats.
- *Generar estadístiques de cada entitat(opcional)*: Un dels avantatges que es desprenen de la classificació de les entitats segons la nova estructura de dades és que podem generar uns informes estadístics consultables des del propi gestor que ens poden aportar dades interessants. Podríem veure quina quota té cadascun dels fabricants i cadascun dels models; podríem saber quin percentatge de dispositius no té actualitzada l'última versió del firmware i generar avisos als propietaris; i potser també podríem realitzar estimacions dels costos totals o parcials d'una zona.

3.2 Requeriments no funcionals

Com a requeriments no relacionats amb la funcionalitat de la nova aplicació, trobem els següents:

- *Integració de les eines control de codi*. Com tots els projectes de programari lliure, l'accés al codi és essencial, i el cas de Guifi.net no n'és una excepció. Per això, tot el codi relacionat amb el projecte es troba disponible en un repositori públic gestionat amb GIT². El codi d'aquest projecte, concretament, formarà part del modul Drupal-guifi³
- *Documentació a les plataformes col·laboratives* (wiki.Guifi.net): Tota la documentació de Guifi.net està confeccionada pels propis usuaris de la comunitat i es troba disponible en una wiki⁴ que cal anar fent periòdicament perquè reflexi la realitat tecnològica de la xarxa
- *Millorar la interfície de gestió (opcional)*: Tot i que ja existeixen unes plantilles per al disseny general de la pàgina (templates) ,que es poden utilitzar per a l'aparença de les

²<https://gitorious.org/guifi>

³<https://gitorious.org/guifi/drupal-guifi>

⁴<https://wiki.Guifi.net>

noves pantalles del gestor, en la mesura del possible s'intentaran aplicar tècniques que en millorin l'aspecte o la funcionalitat.

3.3 Diagrames i casos d'ús

3.3.1 Identificació dels actors

Considerarem els següents tipus d'usuari segons el nivell de permisos de què disposen els actors del sistema:

Usuari estàndard : Qualsevol persona registrada a la web de Guifi.net.

Usuari avançat : Usuari estàndard amb permisos addicionals per utilitzar la interfície de gestió i definir noves configuracions de maquinari, sense possibilitat d'habilitar-les per a la comunitat.

Usuari administrador : Usuari amb permisos superiors per poder validar la feina de l'usuari avançat i obrir nous aparells a la comunitat en general.

3.3.2 Noves entitats

A les entitats que ja es troben disponibles en l'estructura de l'aplicació, caldrà afegir o ampliar les següents:

Fabricant (manufacturer) : Marca comercial que fabrica l'aparell.

Camps principals : fid, name, url

Model : Denominació única que un fabricant assigna a un dels seus productes.

Camps principals : mid, fid, model, url, comments

CaracterísticaModel : Característica associada a les característiques concretes de un aparell.

Camps principals : id, mid, cid

Firmware : Denominació del sistema operatiu que utilitza l'aparell.

Camps principals : id, nom, descripcio, enabled

paràmetresFirmware : Valors possibles que admet un firmware per configurar l'aparell.

Camps principals : id, fid, pid

paràmetres : Paràmetres generals dels sistema aplicables a qualsevol firmware.

Camps principals : id, nom, defaultvalue, origen, dinamic

ConfiguracióUnSolClic : Plantilla associada a la combinació de model-firmware.

Camps principals : id, mid, fid, enabled, plantilla, tipologia

paràmetresConfiguracióUnSolClic : Paràmetres específics associats a una configuració UnSolClic.

Camps principals : id, pid, uscid, valor, dinamic

3.3.3 Casos d'ús

Associats als actors identificats prèviament, desgranem cadascun dels casos d'ús sobre les diferents entitats.

- Usuari estàndard
 - Trastos/Aparells
 - * CRUD trastos/aparells
 - * Assignar/Des-assignar model
 - * Assignar/Des-assignar firmware
 - * Generar fitxer UnSolClic (i baixar-lo)
- Usuari avançat
 - Fabricant: CRUD fabricant
 - Model
 - * CRUD model
 - * Clonar model
 - * Associar/Des-associar a un fabricant
 - * Gestionar característica
 - CaracterísticaModel
 - * Associar/Des-associar característica a un model
 - * Afegir característica
 - * Modificar característica
 - * Clonar característica
 - Firmware
 - * CRUD firmware
 - * Clonar firmware
 - * Vincular/Desvincular a un model
 - ParàmetresFirmware
 - * CRUD paràmetre per firmware
 - * Definir origen
 - * Modificar origen

- * Associar/Des-associar a una configuració UnSolClic
- ConfiguracióUnSolClic
 - * Associar/Des-associar UnSolClic a un model
 - * Associar/Des-associar parametre requerit a UnSolClic
 - * Activar/Des-activar UnSolClic
 - * Afegir plantilla UnSolClic
 - * Modificar plantilla UnSolClic
 - * Verificar UnSolClic (comprovar la substitució de tots els paràmetres)
 - * Generar fitxer UnSolClic (matching plantilla-dades)
- Usuari Administrador
 - Fabricant
 - * Validar/Desvalidar fabricant
 - Model
 - * Validar/Desvalidar model
 - Firmware
 - * Validar/Desvalidar firmware
 - ConfiguracióUnSolClic
 - * Verificar UnSolClic (comprovar la substitució de tots els paràmetres)
 - * Activar/Des-activar UnSolClic

3.3.4 Diagrama de fluxe general.

El diagrama de flux general per a l'aplicació correspon amb el que mostra la Figura 3.1.

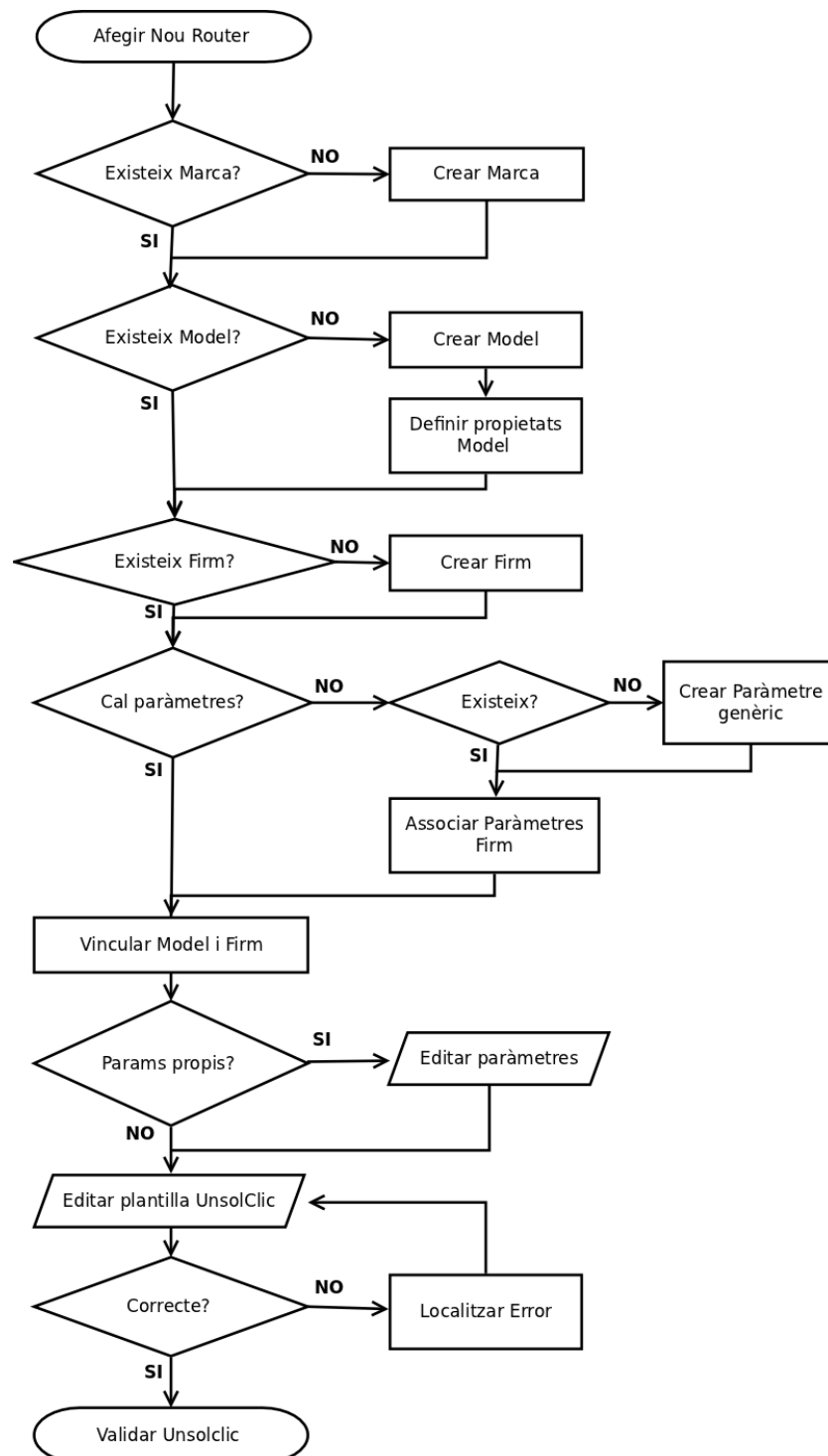


FIGURA 3.1: Diagrama de flux general

Capítol 4

Disseny

En aquesta fase concretarem els conceptes recollits durant l'anàlisi en esquemes i processos que ens han de aclarir i detallar com resoldrem els aspectes tècnics durant la implementació. Aquests esquemes determinaran les funcionalitats de la nostra aplicació.

4.1 Diagrames

En aquesta secció descriurem les relacions entre les entitats existents i les noves que hem creat per a poder implementar la funcionalitat de la nostra aplicació.

Diagrama d'entitats existents

L'actual procés de configuració del fitxer UnSolClic requereix recopilar dades de les següents entitats:

Node : Cadascuna de les ubicacions que componen la xarxa, independentment del tipus de node que implementin. A la base de dades actual correspon a la taula 'node'.

Device (aparell) : Cadascun dels aparells que existeixen a la xarxa i estan associats a un node, que pot tenir diversos devices. A la base de dades actual correspon a la taula 'devices'.

Interfaces : Cadascuna de les interfícies físiques que té definides un cert aparell i que corresponen a connectors de xarxa(normalment del tipus RJ45). A la base de dades actual correspon a la taula 'links'.

Links : Cadascun dels enllaços, ja siguin cablejats o inalàmbrics, que té definits un aparell amb altres nodes de la zona o altres aparells del node, tant poden ser per cable o per wireless. Cada registre té marcat de quin tipus d'enllaç es tracta (cable, WDS, client...)

Ipv4 : Cadascuna de les adreces IP fixes que associem a les interfícies de l'aparell. A la base de dades actual correspon a la taula 'ipv4'.

A continuació podem veure el diagrama Entitat Relació Estesa que relaciona aquestes entitats i ens mostra els seus principals atributs:

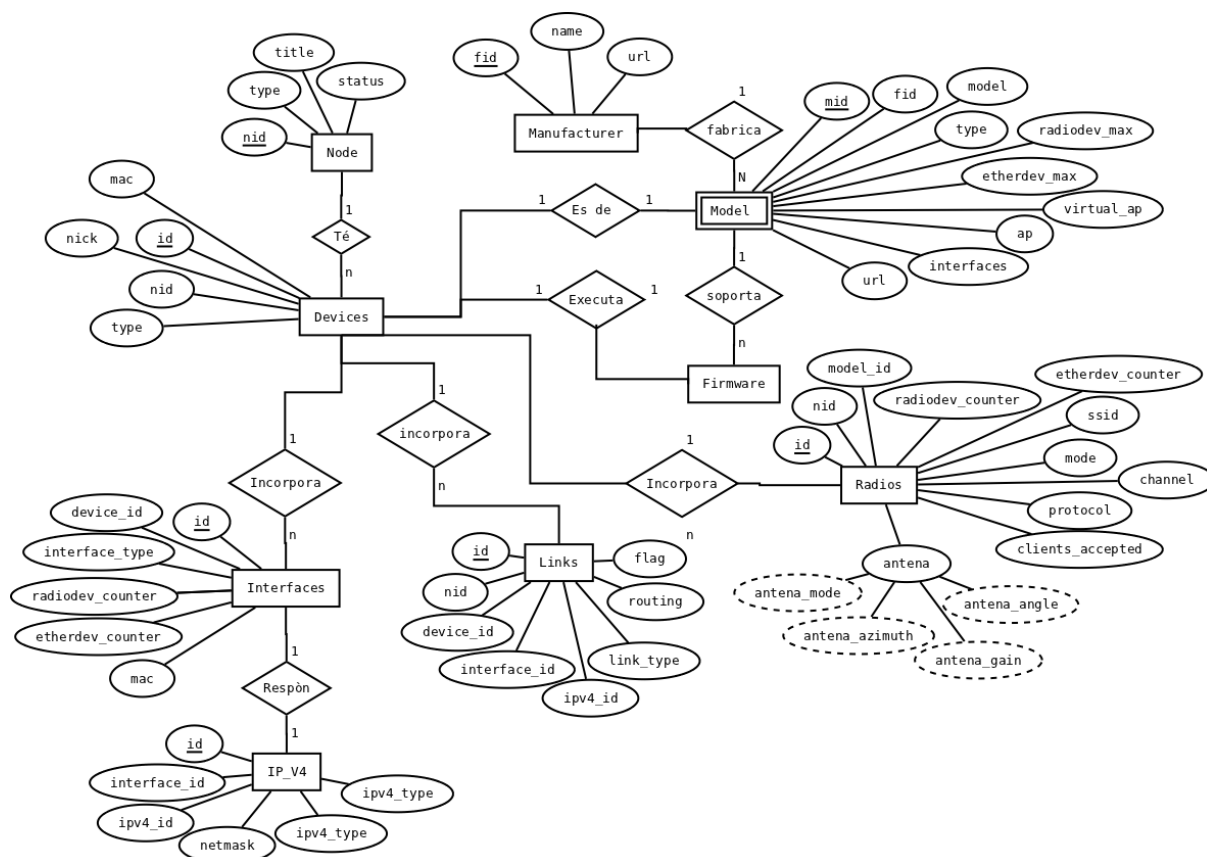


FIGURA 4.1: Diagrama Entitat Relació Estesa general

Diagrames de noves entitats

Per a les noves entitats proposades en l'anàlisi del Capítol 3, veiem al diagrama Entitat Relació Estesa (Figura 4.2) que totes elles estenen a partir de l'entitat Devices sense que es relacionar-se amb les altres. Això ens acotará l'abast dels canvis que la nostra aplicació introduirà en l'estructura actual, minimitzant el risc de conflictes.

La idea de funcionament de l'aplicació queda resumida en aquest diagrama on podem veure que l'entitat Devices (aparells) està relacionada per una banda amb un 'Model' concret i que executa una certa configuració UnSolClic. Aquestes dues relacions són les que engloba el cas d'ús de creació de node per a l'usuari estàndard del sistema, en què triem un aparell d'una llista de Models de Fabricants i a continuació escollim un Firmware associat a aquest Model, de manera que automàticament queda definida quina configuració UnSolClic s'aplicarà.

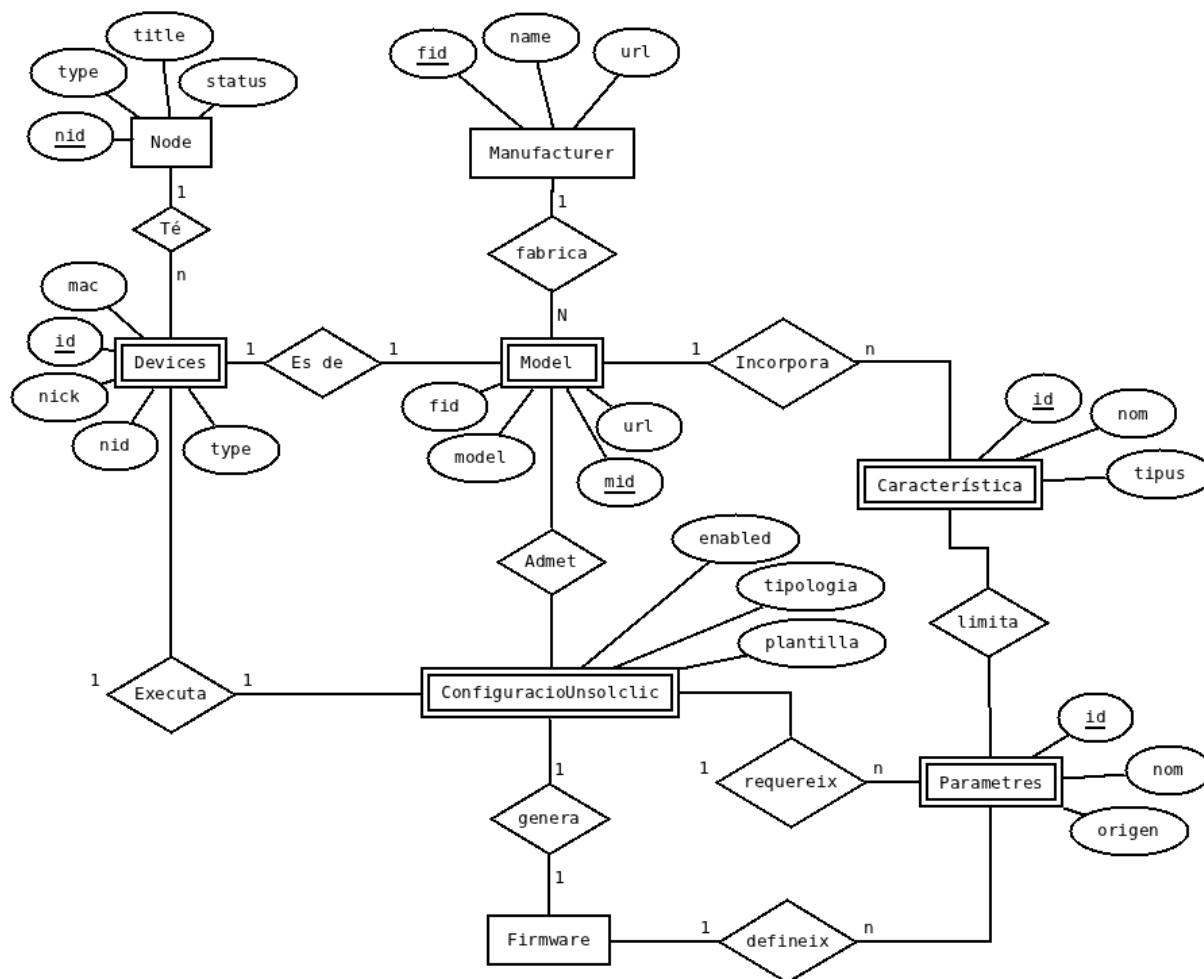


FIGURA 4.2: Diagrama EER Noves entitats

Aquesta dependència entre Model i Firmware es crea cada vegada que per a un cert model li especifiquem que suporta un firmware concret.

D'altra banda, veiem que existeixen les entitats Característica i Paràmetre relacionades amb Model la primera i amb ConfiguracioUnSolClic i Firmware la segona, i que constitueixen el punt on l'aplicació recolzarà la seva flexibilitat a l'hora de poder definir models i comportaments nous.

Les característiques d'un model representen les capacitats físiques de l'aparell i han d'actuar com a topalls en el moment d'associar cadascuna de les dades per a un cert node. Així per exemple malgrat que l'estructura de la base de dades permeti definir un nombre indeterminat d'interfícies, definir una Característica 'max_interfaces' associada a un Model podria establir un límit a comprovar en temps d'execució.

En el cas dels paràmetres dels firmwares, la idea és que es puguin definir uns paràmetres associats que en un moment donat es puguin modificar i que el nou valor es pugui traspasar automàticament a totes les configuracions que utilitzen aquest firmware. Això permetria, per exemple, que certs atributs associats a ordres concretes, codificats com a paràmetres, es

poguessin actualitzar. Un exemple podria ser el valor que prenen les definicions de la freqüència utilitzades en un enllaç, ja que en alguns firmwares s'utilitzen, per exemple, els valors '802.11' o '2.4' o inclús en diferents versions d'un mateix firmware es pot veure '5ghz' o '5ghz-a/n' per referenciar els mateixos rangs.

Finalment, també trobem que per cada ConfiguracióUnSolClic podem definir quins dels Paràmetres disponibles per al nostre Firmware són requerits i quin comportament tindran respecte a l'origen de les seves dades. Per una banda, poden ser estàtics si prenen valors de la definició general de paràmetres, o poden ser dinàmics quan el seu valor prové directament de la informació associada al device concret per al qual estem generant el fitxer UnSolClic. Exemples de paràmetres dinàmics serien el nom del node, la adreça IP principal, l'identificador SSID, etc.

4.1.1 Disseny de la base de dades

Dins l'estructura de dades actual del mòdul Drupal-guifi hi ha moltes taules, però ara només ens fixarem en les que tenen una relació directa amb el procés actual de generació del UnSolClic (Figura 4.3) i que estan representades a l'EER de la Figura 4.1.

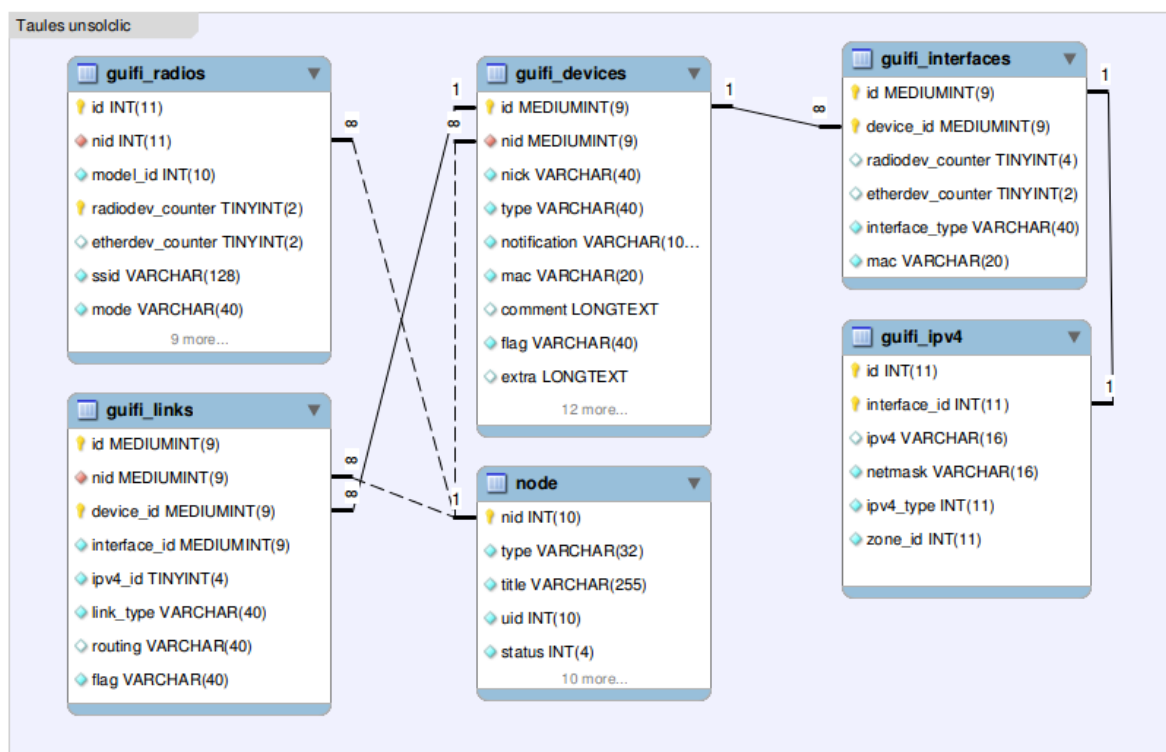


FIGURA 4.3: Diagrama de taules que originen l'UnSolClic

Pel que fa a les taules de les noves entitats proposades, podem veure a la Figura 4.4 com a partir de la taula guifi_devices (que correspon als aparells reals de la xarxa) trobem la relació amb una configuració UnSolClic

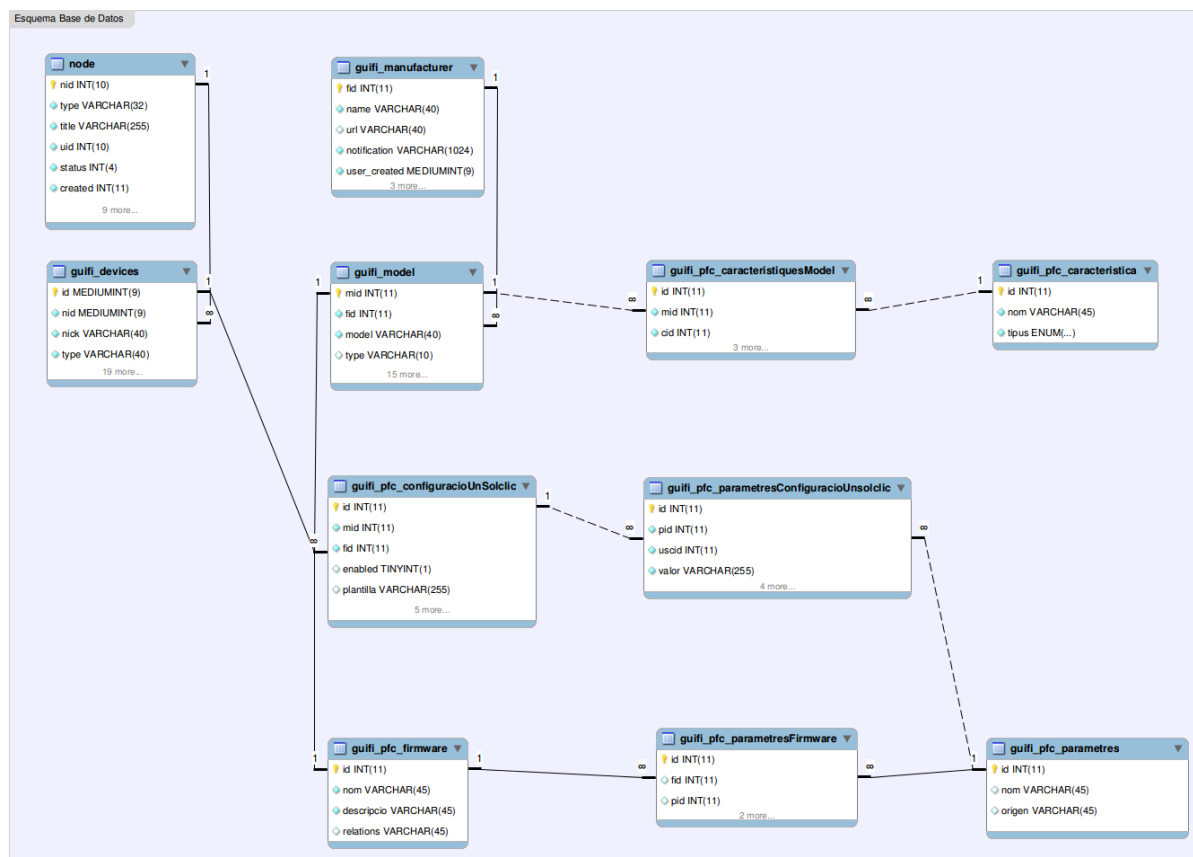


FIGURA 4.4: Diagrama de Taules de la BD

4.2 Avaluació dels sistemes de plantilles estàndard

Per una gestió efectiva de continguts és imprescindible que l'eina que utilitzem separi clarament la capa de presentació de la de l'anomenada lògica de negoci (que manipula el contingut). És per això que la majoria de frameworks actuals implementen el patró de disseny Model-Vista-Controlador, on la Vista fa referència a la visualització de les dades del Model, tasca que recau en els anomenats motors de plantilles (o templates).

En el nostre cas, volem que per cada configuració UnSolClic dependent de un Model i un Firmware, existeixi una plantilla a partir de la qual el sistema pugui crear per cada aparell cadascun dels fitxers individuals de configuració que li corresponen. Així doncs, la part del Model vindrà representada per la informació d'un Aparell concret de la xarxa, la Vista per una plantilla associada i el Controlador per el codi de l'aplicació que genera el fitxer UnSolClic.

El sistema permetrà que aquestes plantilles base es puguin editar directament des d'un formulari web, facilitant així la creació de noves plantilles i l'actualització de les existents. Caldrà doncs que l'usuari avançat conegui la sintaxi que permetrà el sistema de plantilles escollit, per això voldrem que sigui el més senzill possible sense perdre totes les capacitats que exigim a un sistema de plantilles modern. Entre les característiques que demanem per al nostre motor de plantilles, trobem les següents:

- Substitució de variables, filtres de text i crida de funcions natives
- Estructures de control de flux
- Inclusio de fitxers
- Sistema de Catxé per augmentar el rendiment
- Herència de plantilles
- Mecanismes implícits de seguretat

D'entre els motors de plantilles per a PHP existents, hem considerat els següents :

- *Smarty*: Motor amb llicència LGPL, històricament ha estat un dels motors de plantilles més utilitzats en entorns web. Tot i que la darrera versió incorpora moltes millores, ha rebut crítiques per tenir una sintaxi antiga i ferragosa. ¹
- *Twig*: Amb llicència BSD, va aparèixer l'any 2009 del nucli de desenvolupament de Symfony, un dels frameworks PHP més utilitzats, sobretot per la seva filosofia de components desacoblats, essent Twig un d'ells. Incorpora totes les característiques més modernes que puguem demanar i mecanismes per ampliar-ne les funcionalitats. ²

D'entre aquests dos motors de plantilles , el finalment escollit ha estat Twig, sobretot per la baixa corba d'aprenentatge i perquè les seves prestacions en cap cas limiten les possibilitats de configuració que demanarem al sistema; al contrari, presenten escenaris que inicialment no s'havien contemplat, com per exemple l'herència de plantilles o la possibilitat de crear noves paraules clau. En el següent capítol introduïrem els seus conceptes bàsics que necessitarem per a les nostres plantilles.

4.3 Pla d'implementació

Després de crear totes les estructures de dades proposades, podem dividir el projecte en tres parts ben diferenciades. Per una banda, cal crear l'entorn de gestió amb el qual definirem i inicialitzarem totes les entitats creades, i per l'altra emprendrem la confecció de plantilles per a uns casos d'estudi determinats que diferenciarem segons el tipus de node.

4.3.1 Gestió web

Dins l'entorn del CMS Drupal crearem un menú exclusiu per als usuaris avançats des del qual podran accedir a totes les pàgines de gestió i eines associades.

¹<http://www.smarty.net/>

²<http://twig.sensiolabs.org/>

4.3.2 Nodes en mode client

D'entre tot el conjunt de models que existeixen a Guifi.net i que implementen configuracions de node client, s'han seleccionat dos dels models més emblemàtics.

D'una banda hem escollit el Linksys WRT54G³, que com hem esmentat al capítol d'introducció, va constituir un dels pioners de les xarxes wifi, executant el firmware de DD-guifi⁴ que la comunitat Guifi.net va derivar del firmware de programari lliure DD-WRT⁵. Aquest model va ser molt important en el desplegament inicial de la xarxa Guifi.net. Actualment encara n'hi ha un centenar en funcionament.



(a) Linksys WRT54G



(b) Ubiquiti Nanostation 2

FIGURA 4.5: Aparells utilitzats en configuracions de nodes client

D'altra banda, hem escollit el pare d'una família de dispositius que, per prestacions, preu i rendiment, ha esdevingut la opció majoritària per a nodes client. Estem parlant de l'Ubiquiti Nanostation 2⁶ executant el firmware AirOS 3.0⁷. Aquest aparell va suposar una gran avantatge competitiva al ser dels primers en oferir una combinació de router de baix preu resistent a la intempèrie, amb antena integrada i alimentació per PoE⁸, que va simplificar molt la instal·lació dels nodes.

4.3.3 Supernodes

Pel que fa a l'implementació de les plantilles per a supernodes també hem escollit dos models emblemàtics al llarg de l'evolució de la xarxa.

Com a primer candidat, hem escollit el mateix aparell que en el punt anterior, el Linksys WRT54G amb firmware DD-guifi, però en aquest cas utilitzant una configuració en mode Punt

³<http://es.wikipedia.org/wiki/WRT54G>

⁴<http://Guifi.net/project/dd-guifi>

⁵<http://www.dd-wrt.com/site/index>

⁶<http://www.ubnt.com/nanostation>

⁷http://wiki.ubnt.com/AirOS_3

⁸PoE : Power Over Ethernet http://ca.wikipedia.org/wiki/Power_over_Ethernet

d'Accés. Els primers supernodes que es van construir estaven formats de diversos Linksys (fins a 5) en una caixa estanca, units per cables de xarxa, amb tota la dificultat que implicava el muntatge. En aquest mode la configuració és força més complexa i requereix de les rutines pròpies dels supernodes.

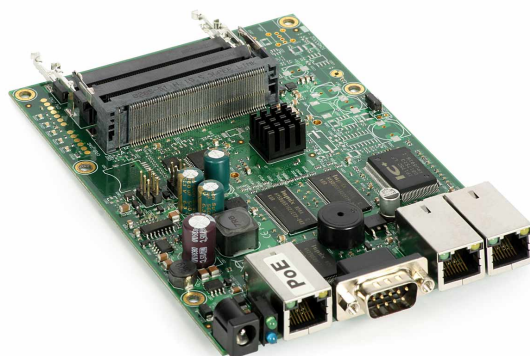


FIGURA 4.6: Mikrotik routerboard RB433

Per al segon cas d'estudi per a supernodes, hem triat el Mikrotik RouterBoard 433⁹ executant el firmware propietari RouterOS¹⁰ en la versió 4.7. Després que s'haguessin començat a utilitzar amb èxit models anteriors per evolucionar els supernodes, el RB433 es va començar a utilitzar extensivament a la xarxa degut al gran salt de rendiment que representava el seu potent processador i els 128 Mb de memòria respecte els 8Mb dels vells Linksys.

4.3.4 Rutines de Validació

Finalment, implementarem un sistema de validació automatitzat que, mitjançant la comparació dels UnSolClics generats per una plataforma de desenvolupament amb els de la plataforma de producció, ens ajudarà a confeccionar les plantilles i detectar els possibles errors de codificació.

⁹<http://routerboard.com/RB433>

¹⁰<http://acacha.org/mediawiki/index.php/RouterOS>

Capítol 5

Implementació

En aquest capítol resseguirem cadascun dels passos que ha requerit la implementació dels requisits de l'aplicació, enfocant l'atenció en els aspectes més rellevants.

5.1 Gestió web

En primer lloc, cal confeccionar una pàgina inicial des d'on poder accedir a cadascuna de les pàgines de gestió de cada entitat; per això hem creat el menú que es pot veure a la Figura 5.1:

The screenshot shows the Guifi.net web interface. At the top, there's a navigation bar with links like 'Crea contingut', 'El meu compte', and 'Menú guifi.net'. Below that, a search bar and a sidebar with 'Apadrinaments' and 'network figures & facts'. The 'network figures & facts' section includes a table of nodes and links. The main content area is titled 'Eines de desenvolupament' and lists various tools for network management.

Nodes	25.758
Operatiu	16.403
Projectat	8.457
En proves	493
En construcció	317
Reserved	76
Dropped	12

Enllaços Sensefils	#	qms.
ap/client	14.244	16.903,50
Enllaç P2P	1.454	6.037,30
Total	15.698	22.940,80

FIGURA 5.1: Pàgina inicial de gestió per al PFC

El procés per afegir cadascuna d'aquestes pàgines al mòdul Drupal-guifi consisteix en afegir una ruta al fitxer `guifi.module` de configuració. Així per exemple, per a la pàgina de gestió de models hem de definir dues rutes diferents, `'guifi/menu/devel/device'` que ens mostrarà el llistat de models i `'guifi/menu/devel/device/%/%'` que rebrà dos paràmetres, l'identificador del model i la acció a realitzar, i ens mostrarà el formulari adient.

- Afegir la ruta al fitxer de configuració del mòdul:

```
$items['guifi/menu/devel/device'] = array(
  'title' => 'Device Models',
  'description' => 'A full list of devices we use in the network',
  'page callback' => 'guifi_devel_devices',
  'access arguments' => array('guifi module developer'),
  'weight' => 8,
  'file' => 'guifi_devel.inc.php'
);
$items['guifi/menu/devel/device/%/%'] = array(
  'title' => 'Device Models',
  'page callback' => 'guifi_devel_devices',
  'page arguments' => array(4, 5),
  'access arguments' => array('guifi module developer'),
  'file' => 'guifi_devel.inc.php'
);
```

- Implementar la funció corresponent al fitxer `'guifi_devel.inc.php'` on s'han concentrar totes les funcions relacionades amb aquest projecte:

```
// Device Models output
function guifi_devel_devices($devid , $op) {
  switch($devid) {
    case 'add':
      $devid = 'New';
      return drupal_get_form('guifi_devel_devices_form',$devid);
  }
  switch($op) {
    case 'edit':
```

- Maquetar el formulari corresponent `'guifi_devel_devices_form'`

```
function guifi_devel_devices_form($form_state, $devid) {

  $sql = db_query('SELECT * FROM {guifi_model} WHERE mid = %d', $devid);
  $dev = db_fetch_object($sql);
```



```

if ($devid == 'New' ) {
    $form['new'] = array('#type' => 'hidden', '#value' => TRUE);
} else {
    $form['mid'] = array('#type' => 'hidden', '#value' => $devid);
}

$query = db_query('SELECT * FROM {guifi_manufacturer}');
while ($manufacturers = db_fetch_array($query)) {
    $manuf_array[$manufacturers["fid"]] = $manufacturers["name"];
}

$form_weight=0;
$form['fid'] = array(
    '#type' => 'select',
    '#title' => t('Manufacturer'),
    '#required' => TRUE,
    '#default_value' => $dev->fid,
    '#options' => $manuf_array,
    '#description' => t('Select Device Manufacturer.'),
    '#prefix' => '<table><tr><td>',
    '#suffix' => '</td>',
    '#weight' => $form_weight++,
);

$form['model'] = array(
    '#type' => 'textfield',
    '#title' => t('Model Name'),
    '#required' => TRUE,
    '#default_value' => $dev->model,
    '#size' => 32,
    '#maxlength' => 50,
    '#description' => t('Device model name, please, use a clear and short description.'),
    '#prefix' => '<td>',
    '#suffix' => '</td>',
    '#weight' => $form_weight++,
);

```

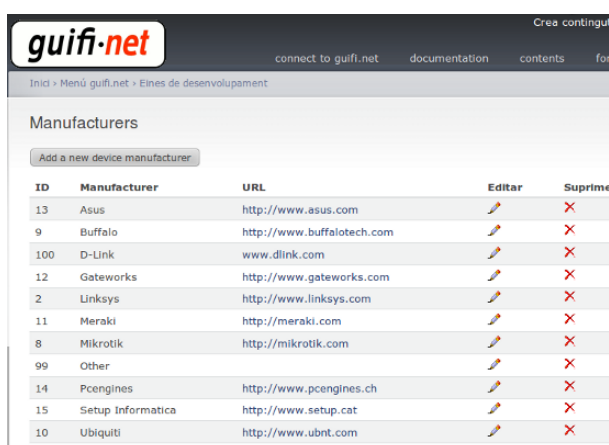
5.1.1 Càrrega inicial de dades

Com ja havíem comentat, en l'estructura inicial de la base de dades, ja existien les taules per a Fabricants i Models, que s'associaven als aparells concrets de la taula devices mitjançant un camp de text serialitzat. D'altra banda, la llista dels firmwares disponibles es troba en una taula genèrica de tipus on trobem un camp de text amb els models que suporten.

En primer lloc, cal normalitzar aquestes dades i fer els inserts a cadascuna de les noves taules que hem creat. Després d'executar els scripts de MySQL que trobem als annexes, comprovem que a la taula `guifi_manufacturer` tenim 11 registres, a la taula de `guifi_model` 45 registres i a la taula `guifi_pfc_firmware` 21 registres.

5.1.2 Fabricants

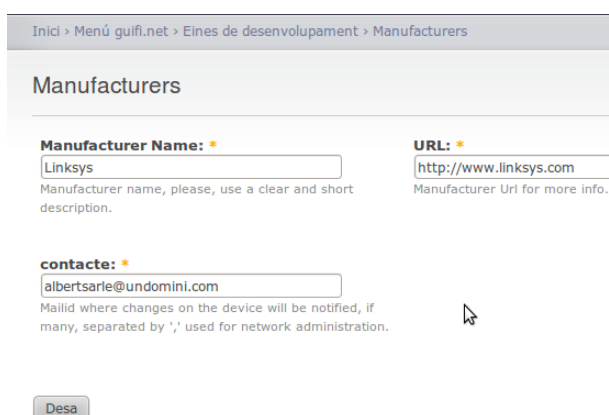
Per a la gestió de fabricants, tenim un llistat i un formulari mínims des d'on podem actualitzar el nom, desar una URL per a les novetats i introduir l'adreça de correu de l'usuari avançat que ha donat d'alta el registre.



The screenshot shows the 'Manufacturers' page on the guifi-net website. It features a table with columns for ID, Manufacturer, URL, Editar, and Suprime. Below the table is a button to 'Add a new device manufacturer'.

ID	Manufacturer	URL	Editar	Suprime
13	Asus	http://www.asus.com		
9	Buffalo	http://www.buffalotech.com		
100	D-Link	www.dlink.com		
12	Gateworks	http://www.gateworks.com		
2	Linksys	http://www.linksys.com		
11	Meraki	http://meraki.com		
8	Mikrotik	http://mikrotik.com		
99	Other			
14	Pcengines	http://www.pcengines.ch		
15	Setup Informatica	http://www.setup.cat		
10	Ubiquiti	http://www.ubnt.com		

FIGURA 5.2: Llistat de Fabricants



The screenshot shows the 'Manufacturers' form on the guifi-net website. It includes fields for 'Manufacturer Name', 'URL', and 'contacte', each with a description and a 'Desa' button at the bottom.

Manufacturer Name: *

 Manufacturer name, please, use a clear and short description.

URL: *

 Manufacturer Uri for more info.

contacte: *

 Mailid where changes on the device will be notified, if many, separated by ',' used for network administration.

FIGURA 5.3: Formulari de Fabricant

5.1.3 Models

La gestió de models ja incorpora més funcionalitats; per una banda, al llistat veiem que es mostren el número total de Firmwares actius que estan associats a aquest model, i per l'altra,

una funcionalitat de Duplicar per tal de facilitar la creació de nous models a partir d'altres ja existents.

ID	Model	Manufacturer	Model	Enabled Firms	Editar	Suprimetx	Dupl
18		Linksys	WRT54GSv4	3			
31		Mikrotik	Routerboard 433	3			
25		Ubiquiti	NanoStation2	1			
40		Asus	Asus WL-500xx	0			
15		Buffalo	WHR-HP-G54, WHR-G54S	0			
56		D-Link	DIR-600 B1/B2	0			
39		Gateworks	Avila GW2348-4	0			
16		Linksys	WRT54GL	0			
17		Linksys	WRT54GSv1-2	0			
1		Linksys	WRT54Gv1-4	0			
57		Manufacturer1	Model1	0			
30		Meraki	Meraki/Fonera	0			
54		Mikrotik	RouterBoard 1100	0			

FIGURA 5.4: Llistat de Models

Device Models

Manufacturer: Mikrotik

Model Name: Routerboard 433

Select Device Manufacturer: Device model name, please, use a clear and short description.

firmwares compatibles amb aquest model:

- RouterOSv2.9
- RouterOSv3.x
- RouterOSv4.0+
- RouterOSv4.7+
- RouterOSv5.x

Tots els firmwares:

- AirOsv221
- AirOsv3.6+
- AirOsv30
- AirOsv52
- Alchemy
- DD-guifi
- DD-WRTv23
- DD-WRTv24preSP2
- Freifunk-BATMAN
- Freifunk-OLSR
- gsfirm
- GuifiStationOS1.0
- kamikaze
- n/s
- Talisman
- whiterussian

contacte: albertsarie@undomini.com

Mailid where changes on the device will be notified, if many, separated by ',' used for network administration.

Desa

FIGURA 5.5: Formulari de Models

Pel què fa al formulari, podem actualitzar el fabricant, el nom del model, el contacte de l'usuari avançat que l'ha creat i, al centre, triar d'entre els firmwares definits als sistema els que pot executar aquest model concret.

És important observar que per cadascun dels firmwares que associem a un model, podrem definir a la pantalla corresponent una plantilla base per crear els fitxers de configuració UnSolClic. Per l'assignació de firmwares no cal fer cap comprovació, però per des-assignar un firmware que cal comprovar que aquest no formi part de cap configuració UnSolClic habilitada, ja que es deixaria de donar servei als aparells que implementessin aquesta combinació.

5.1.4 Firmwares

Al llistat de Firmwares podem veure que els primers en aparèixer són els que estan habilitats per a ser usats pels usuaris estàndard al configurar els seus aparells, i s'inclouen el nombre de configuracions UnSolClic de les que formen part.

Firmware				
Add a new firmware				
ID	Nom	Descripció	Habilitat	used on #USC
13	AirOsv30	Ubiquiti AirOs 3.0	1	1
5	DD-guifi	DD-guifi from Miquel Martos	1	1
4	DD-WRTv23	DD-WRTv23Beta2 from BrainSlayer	1	1
21	DD-WRTv24preSP2	DD-WRTv24preSP2 from BrainSlayer	1	1
14	RouterOSv4.0+	RouterOSv4.0+	1	1
16	RouterOSv4.7+	RouterOS 4.7 and newer 4.x	1	1
19	RouterOSv5.x	RouterOS 5.x or newer from Mikrotik	1	1
11	AirOsv221	Ubiquiti AirOs 2.2.1	0	0
20	AirOsv3.6+	Ubiquiti AirOs 3.6+	0	0
15	AirOsv52	Ubiquiti AirOs 5.2	0	0
2	Alchemy	Alchemy from sveasoft	0	0
9	Freifunk-BATMAN	OpenWRT-Freifunk-v1.6.16 with B.A.T.M.A.N	0	0
12	Freifunk-OLSR	OpenWRT-Freifunk-v1.6.16 with OLSR	0	0
18	gsffirm	Firmware de GràciaSenseFils MANET	0	0

FIGURA 5.6: Llistat de Firmwares

Firmware

Firmware short name: *
 Habilitat
Check if firmware is available for use

The firmware name, please, use a clear and short name. ex: "FirmwarevXX" where XX = version

Firmware Description: *

The firmware description, please, use a clear and short description. ex: "FirmwarevXX from creator"

Parametres associats:

- antenna.selection
- device_id
- device_name
- devnick
- firmware_name
- firmware_version
- locnick
- primarydns
- prova1
- radio.1.acktimeout
- radio.1.ext_antenna
- radio.1.ieee_mode
- radio.1.mcastrate
- radio.1.rate.max
- radio.1.rx_antenna
- radio.1.tx_antenna
- radio1txpower
- wan.netmask
- wireless.1.ssid

Parametres associats a la definició d'aquest firmware.

Parametres disponibles:

- firmware_description
- nombre
- radio.1.antenna_gain
- radio.1.channel
- radio.1.order
- radio.1.protocol
- radio.1.ssid
- routerOS_24_band_value
- routerOS_5_band_value
- snmp_contact
- ternarydns
- ternary_dns
- txpwr
- zone_id
- zone_ternary_dns

Parametres disponibles per a definir aquest firmware

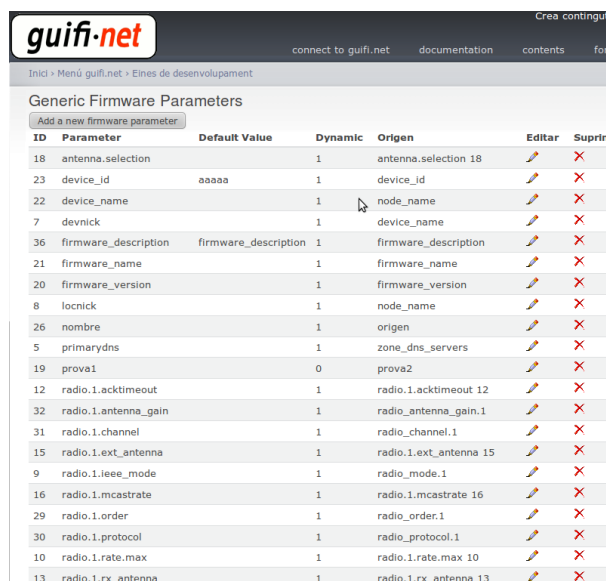
contacte: *

Mallid where changes on the device will be notified, if many, separated by ',' used for network administration.

FIGURA 5.7: Formulari de Firmwares

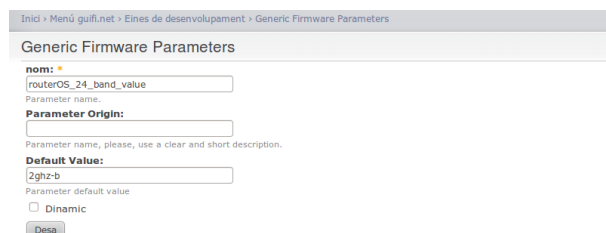
5.1.5 Paràmetres

Els paràmetres genèrics es poden inicialitzar de manera global per tal que puguin ser compartits per diferents firmwares.



ID	Parameter	Default Value	Dynamic	Origin	Editar	Suprim
18	antenna_selection		1	antenna_selection 18		
23	device_id	aaaaa	1	device_id		
22	device_name		1	node_name		
7	devnick		1	device_name		
36	firmware_description	firmware_description	1	firmware_description		
21	firmware_name		1	firmware_name		
20	firmware_version		1	firmware_version		
8	locnick		1	node_name		
26	nombre		1	origen		
5	primarydns		1	zone_dns_servers		
19	prova1		0	prova2		
12	radio.1.acktimeout		1	radio.1.acktimeout 12		
32	radio.1.antenna_gain		1	radio_antenna_gain.1		
31	radio.1.channel		1	radio_channel.1		
15	radio.1.ext_antenna		1	radio.1.ext_antenna 15		
9	radio.1.ieee_mode		1	radio_mode.1		
16	radio.1.mcastrate		1	radio.1.mcastrate 16		
29	radio.1.order		1	radio_order.1		
30	radio.1.protocol		1	radio_protocol.1		
10	radio.1.rate_max		1	radio.1.rate_max 10		
13	radio.1.rx_antenna		1	radio.1.rx_antenna 13		

FIGURA 5.8: Llistat de Paràmetres



Generic Firmware Parameters

nom:

Parameter name:

Parameter Origin:

Parameter name, please, use a clear and short description.

Default Value:

Parameter default value

Dinamic

FIGURA 5.9: Formulari de Paràmetres

5.1.6 Configuracions UnSolClic

Les Configuracions UnSolClic representen cadascuna de les combinacions de Model i Firmware, entenent que dues versions diferents es consideren firmwares diferents. Per tant, no es poden crear des del llistat sinó que cada vegada que associem un firmware a un model crearem un registre nou en aquest llistat. Com podem veure a la Figura 5.11, el llistat apareix ordenat segons si l'UnSolClic està disponible per als usuaris o no i podem veure quants paràmetres incorpora.

A la capçalera del formulari se'ns indica quina combinació de Fabricant, Model i Firmware estem tractant (els noms ens porten al formulari d'edició corresponent), i un checkbox ens indica si la plantilla ha estat validada, i per tant, permetem que els usuaris estàndards la puguin triar per als seus aparells.

Manufacturer	Model	FirmWare	enabled	#parameters	Editar	Suprimeix
Linksys	WRT54GSv4	DD-guifi	1	13		
Linksys	WRT54GSv4	DD-WRTV23	1	15		
Linksys	WRT54GSv4	DD-WRTV24preSP2	1	13		
Mikrotik	Routerboard 433	RouterOSv4.0+	1	23		
Mikrotik	Routerboard 433	RouterOSv4.7+	1	21		
Mikrotik	Routerboard 433	RouterOSv5.x	1	25		
Ubiquiti	NanoStation2	AirOsv30	1	16		
Linksys	WRT54GL	DD-guifi	0	12		
Mikrotik	Routerboard 433	RouterOSv2.9	0	0		
Mikrotik	Routerboard 433	RouterOSv3.x	0	0		
Ubiquiti	NanoStation5	AirOsv30	0	0		

FIGURA 5.10: Llistat de configuració UnSolClic

Configuracions UnSolClic

Linksys WRT54GSv4 DD-guifi Habilitat

Template File:

```
nvrnm set ntp_server="{{ zone_primary_ntp }}"
{% endif %}
#
{% if dev.radios[0].mode == 'ap' or dev.radios[0].mode == 'AP' %}
# {{ '(AP mode)' }}
nvrnm set wi_mode="ap"
nvrnm set wi0_mode="ap"
nvrnm set wi_channel="{{ dev.radios[0].channel }}"
nvrnm set wi_ssid="guifi.net-{{ guifi_to_7bits(dev.radios[0].ssid) }}"
nvrnm set wi_macmode="disable"
nvrnm set wi0_macmode="disable"
nvrnm set wi_macmode1="disable"
nvrnm set wi0_macmode1="disable"
{{ _self.guifi_unsolclic_ospf(dev, zone, ospfzone, interfaceWanLan) }}
{{ _self.guifi_unsolclic_rbrnd(dev, interfaceWanLan) }}
```

Paràmetres Associats al USC : 13

Paràmetre	Origen	Valor Fixe	Editar	Suprimeix
txpwr	-	28		
device_id	device_id	-	-	
device_name	node_name	-	-	
firmware_name	firmware_name	-	-	
firmware_version	firmware_version	-	-	
primarydns	zone_dns_servers	-	-	
secondarydns	secondarydns	-	-	
ternarydns	zone_ternary_dns	-	-	
wan.gateway	radios_1_interfaces_1_ipv4_0_links_1_interface_ipv4_ipv4	-	-	
wan.ipv4	ipv4_ip	-	-	
wan.netmask	radios_1_interfaces_1_ipv4_0_links_1_interface_ipv4_ipv4	-	-	
wireless.1.ssid	linkedto_ssid	-	-	
zone.ntp_servers	zone_ntp_servers	-	-	

Desa

FIGURA 5.11: Formulari de configuració UnSolClic

LA gran àrea de text central és on hi editarem la plantilla que crearà la seqüència d'ordres necessària per generar un fitxer UnSolClic que els usuaris es puguin descarregar. Aquí podem fer ús de totes les variables, expressions, filtres i estructures de control que ens proporciona Twig. Cal anar amb molt de compte perquè en alguns firmwares, com per exemple RouterOS, utilitzen els mateixos caràcters que delimiten els blocs de Twig per crear bucles en estructures, i en aquesta versió no s'ha implementat un validador de la sintaxi de la plantilla (tot i que Twig ho permet).

Finalment, veiem un llistat on trobem tots els paràmetres associats al firmware que s'aplicaran en aquesta configuració. La primera columna 'Paràmetre' ens indica el nom que utilitzarem per referenciar el paràmetre dins la plantilla, el camp 'Origen' ens determina si el valor d'aquest paràmetre ens vindrà del seu valor per defecte o si caldrà agafar-lo de les dades reals de l'aparell que estem configurant; en aquest cas hi trobem un nom fruit d'una convenció aplicada

als diferents camps que podem obtenir de la base de dades. Els paràmetres que obtenen un valor inicial en la seva definició permeten que es modifiquin només per una certa combinació.

5.1.7 Assignació de configuracions

Aquesta és una de les eines que aporta l'aplicació per tal que la migració de sistema pugui ser més fluida. La pàgina executa un script que, per a cada aparell de la xarxa, extreu la informació existent del camp serialitzat, en busca la configuració UnSolClic adient al Model i Firmware de cadascun, l'assigna al registre de l'aparell en una clau forània a la taula de configuracions.

5.2 Orígens i parametritzacions del node

Com hem esmentat, el sistema de plantilles bàsicament substitueix les variables presents en la plantilla pels valors que li passem en la crida de la visualització, però no tota la informació necessària per configurar el fitxer UnSolClic prové del mateix origen; després de crear la nova estructura de dades, la trobem segons l'entitat a què pertanyin.

Paràmetres dependents del firmware

Quan definim un paràmetre genèric, li donem un nom que ens permetrà referenciar-lo dins les plantilles i triar-lo de la llista de paràmetres disponibles per a un firmware. Aquest nom pot ser completament genèric o, si tenim previst que només l'utilitzarem en una determinada família de firmwares, podem prendre la convenció de posar un prefix amb el nom de la família que volguem i només vincular-lo als seus integrants. El paràmetre admet un valor per defecte que s'utilitzarà per igual en tots els firmwares.

Paràmetres dependents del UnSolClic

Al crear l'associació de Model-Firmware, tots els paràmetres associats al firmware passen a formar part de la configuració i, en el cas de no tractar-se de paràmetres marcats com a dinàmics, permetran que, per a aquesta configuració UnSolClic concreta, puguin prendre valors diferents modificant-los amb un formulari accessible des del llistat de paràmetres.

Paràmetres dependents del device (dinàmics)

Quan donem d'alta el nostre aparell li introduïm dades concretes com, per exemple, l'adreça mac, la banda de treball, el guany de l'antena i l'angle. Quan definim un enllaç amb un altre

node, queden determinades les adreces IP, la ruta per defecte, el protocol d'enrutament, els servidors DNS, etc. Tota aquesta informació és la que anomenem dinàmica en el sentit que, des del punt de vista dels fitxers de configuració, correspon a dades reals de la xarxa i, per tant, obligatòriament han de ser diferents per a cada node.

En el procés actual de configuració de l'UnSolClic ens trobem amb una funció que rep un objecte de tipus Device (Aparell) i que, de forma seqüencial, realitza crides a funcions auxiliars que van acumulant tota la informació necessària per confeccionar un fitxer de sortida. En el nostre cas, volem que l'usuari pugui tenir una pàgina en blanc sobre la que començar a incloure informació estàtica i dinàmica sense conèixer el programa o la estructura de dades interna.

Ens trobem, doncs, que cal definir un sèrie de convencions que determinin com l'usuari referenciarà la informació que vol incloure.

- **Jerarquia d'objectes**

El motor de plantilles Twig ens permet treballar amb variables i estructures de dades complexes, ja siguin arrays o objectes d'una classe, accedint als seus elements o propietats de la mateixa manera, per exemple `{{ foo.bar }}` pot significar moltes coses, però Twig s'encarrega de verificar-les totes per ordre: si foo és array i bar un element, o si foo és un objecte i bar una propietat, o si foo és un objecte i bar un mètode, o si foo és un objecte amb un getter `getBar`, en els altres casos retorna un null que no es visualitza a la sortida.

Donat que la nostra funció substitueix a l'existent per a les combinacions UnSolClic habilitades i per tant rebem el mateix objecte Device, si coneixem la estructura del objecte, podem utilitzar comportament per referenciar totes les seves propietats a dins de la plantilla.

```
object(stdClass)#4 (28) {
  ["id"]=>
  string(5) "22979"
  ["nick"]=>
  string(20) "vistabellaCalvariRd1"
  ["type"]=>
  string(5) "radio"
  ["flag"]=>
  string(7) "Working"
  ["extra"]=>
  string(66) "a:2:{s:8:"model_id";s:2:"31";s:8:"firmware";s:13:"Router0Sv4.0+";}";
  ["radios"]=>
  array(5) {
    [0]=>
    array(18) {
```



```

["id"]=>
string(5) "22979"
["radiodev_counter"]=>
string(1) "0"
["ssid"]=>
string(22) "Vistbll-CalvariAP0-353"
["mode"]=>
string(2) "ap"
["protocol"]=>
string(7) "802.11a"

```

En la nostra plantilla podríem accedir a les propietats de la primera radio iterant sobre l'array i accedint als elements:

```

{% for radio_id, radio in dev.radios %}
# Radio#: {{ radio.radiodev_counter }} {{ radio.ssid }}
/interface wireless set wlan{{ radio.radiodev_counter+1 }} \
    name="wlan{{ radio.radiodev_counter+1 }}" \
radio-name="{{ radio.ssid }}" mode={{ radio.mode }}-bridge \
    ssid="Guifi.net-{{ radio.ssid }}" \

```

Només ens caldria conèixer el nom de tots els camps i totes les estructures que trobem associades que al Device. La llista completa dels noms mitjançant els quals podem accedir a les propietats i estructures relacionades d'un Device es pot trobar a l'Apèndix C i també segueix una convenció determinada segons la relació.

- **Convenció de noms**

Una altra manera de poder accedir a la informació provinent del node seria definint una convenció per la qual, mitjançant un conjunt limitat de noms associats als diferents orígens possibles del objecte Device, poguéssim determinar qualsevol de les propietats del Device amb una única paraula, independentment de la profunditat a la que es trobin en l'array multi-dimensional. Amb aquesta idea és va crear el camp Origen, que podem definir als paràmetres.

Per determinar el format que tindrien les paraules, es van escollir els mateixos noms que les propietats de l'objecte. Els camps que contenien un array es van 'aplanar' creant una paraula resultat de concatenar el nom de tots els subelements implicats i substituint les claus amb identificadors de la base de dades per comptadors. Així, tot i que pot quedar una sintaxi molt llarga, podem referenciar amb una sola paraula una propietat molt profunda en l'objecte sense haver de crear estructures de control que vagin iterant per cada subelement. Seguint amb l'exemple anterior, podríem haver definit un paràmetre 'ssidradio1' amb origen 'dev_radio_0_radio_ssid' i incloure'l a la plantilla de la següent forma:

```
ssid="Guifi.net-{{ ssidradio1 }}"
```

5.3 Integració del sistema de plantilles

Abans d'abordar la confecció de plantilles per a cadascun dels casos tractats farem una breu introducció al funcionament de Twig.

Instal·lació

La versió de Twig amb la que hem treballat és la 1.6¹, tot i que en el moment de la redacció d'aquesta redacció ja havia sortit la versió 1.7². Els passos per executar git són els següents:

- Baixar la versió de Twig:

```
cd ./contrib
wget -q -O - https://github.com/fabpot/Twig/tarball/v1.6.0 | tar zxv
```
- Registrar la llibreria:

```
include_once('contrib/Twig_1.6/lib/Twig/Autoloader.php');
Twig_Autoloader::register();
```
- Definir el carregador i les variables d'entorn:

```
$loader = new Twig_Loader_String();
$Twig = new Twig_Environment($loader);
```
- Inicialitzar la plantilla i les variables:

```
$totalParameters['dev'] = $dev;
```
- Mostrar el resultat retornat per Twig:

```
$plantilla = $Twig->render($plantilla, $totalParameters);
```

Funcionament bàsic

Una plantilla de Twig és bàsicament un fitxer de text on hi trobem variables i expressions que prendran el valor indicat i etiquetes que en controlen la lògica. Les variables i expressions apareixeran limitades pels caràcters `{{ variable }}`, les etiquetes de control per `{% etiqueta %}` i els comentaris per `{# etiqueta #}`.

És habitual que en aquestes plantilles hi hagi codi HTML, per això afegim un exemple basat en el que trobem a la documentació oficial³:

¹<https://github.com/fabpot/Twig/tree/v1.6.0>

²<http://pear.twig-project.org>

³<http://Twig.sensiolabs.org/doc/templates.html>

```

<!DOCTYPE html>
<html>
  <head><title>{{ titolProjecte }}</title></head>
  <body>
    <h1>Taula de continguts</h1>
    <ul id="capitols">
      {% for item in capitols %}
        <li><a href="{{ item.href }}">{{ item.caption }}</a></li>
      {% endfor %}
    </ul>
  </body>
</html>

```

La crida i els paràmetres per visualitzar aquest plantilla (index.html) podria ser aquesta:

```

for ($i=1; $i<=5; $i++) {
    $capitols[] = array('href'=> '/capitol' . $i . '/index.html',
                      'caption'=> 'Capitol ' . $i)}
$template = $Twig->loadTemplate('index.html');
echo $template->render(array('titolProjecte' => 'El projecte PFC',
                          'capitols' => $capitols));

```

En la nostra implementació, enlloc de carregar la plantilla des d'un fitxer, ens vindrà del contingut del camp plantilla de la taula ConfiguracioUnSolClic que haurà confeccionat un usuari avançat. Així doncs, enlloc de tractar amb HTML hi tindrem seqüències d'ordres del firmware que estiguem tractant.

Funcionament avançat (filtres, macros, funcions, sandbox)

Com hem vist a l'exemple de plantilla anterior, Twig utilitza els operadors i les estructures de control habituals que ens permeten incorporar lògica a la plantilla. Aquesta és una de les característiques que ens permeten traspasar la complexitat del codi font de l'aplicació a les plantilles que podem modificar des del gestor web.

El propòsit de tractar i formatar text del motor de plantilles fa que també tinguem a la nostra disposició un munt de filtres⁴ que podem aplicar a les nostres variables (per exemple: replace, capitalize, nl2br, striptags, length, format...), funcions de suport, capacitat d'incloure altres fitxers, possibilitat de definir macros per optimitzar codi, heretar propietats d'una altra plantilla, etc... A més podem incorporar extensions creades per la comunitat d'usuaris de Twig i també la possibilitat de modificar l'arbre lexicogràfic per defecte i incloure les nostres pròpies paraules reservades.

⁴<http://Twig.sensiolabs.org/doc/filters/index.html>

Confecció de plantilles

El procés de confecció de plantilles el podríem dur a terme des de dues visions diferents depenent de si l'aparell ja està suportat per l'aplicació o si és un dispositiu completament nou.

- **Per a aparells existents:** En aquest cas volem confeccionar una plantilla que reproduïxi exactament el contingut de l'UnSolClic. Per aconseguir-ho, podem examinar el codi font del mòdul Drupal-guifi i analitzar pas a pas quines són totes les accions que realitza i amb quines dades; aleshores a partir d'un aparell similar o d'un exemple de configuració, confeccionem el UnSolClic aplicant les expressions i elements de Twig amb les convencions que hem definit.

En alguns casos podrem seguir la mateixa seqüència de passos i anar 'traduint' el codi PHP a la sinàxi de Twig, però en d'altres haurem de repensar l'estructura del codi i implementar-la de forma diferent, per exemple, unint diversos bucles quan s'acumulen valors generats en iteracions diferents.

La validació d'aquestes plantilles es pot fer directament comparant la sortida obtinguda amb la nova plantilla amb la configuració generada pel procés actual. Al Capítol 6 explicarem el mètode de validació i els resultats.

- **Per a aparells nous:** Per codificar plantilles de les quals no hi ha suport previ a l'aplicació, haurem de buscar a la documentació del fabricant o a internet alguna configuració semblant a la que volem aplicar, tenint en compte el mode de funcionament del dispositiu i altres variables relacionades.

Partint d'aquesta seqüència d'ordres hauríem d'identificar quines de les comandes inclouen dades que es puguin correspondre amb els diferents tipus de paràmetres definits i codificar-les segons les convencions establertes. El procés de validació d'aquestes configuracions UnSolClic requerirà disposar d'un d'aquests aparells per a carregar i verificar el seu bon funcionament de l'aparell.

Registre de funcions natives

Dins l'aplicació actual de Guifi.net hi ha definides moltes funcions que podríem definir com a utilitats, per exemple, per calcular màscares de xarxa, obtenir els servidors dns de una zona o determinar el SSID de un supernode al que volem connectar. Algunes d'aquestes funcions s'utilitzen en el procés de generació de l'UnSolClic i no es poden implementar en les plantilles.

El fet que Twig ens permet registrar aquelles funcions (natives o no) que vulguem tenir accessibles des de la plantilla, ens resol aquesta problemàtica, així per a confeccionar les plantilles dels models analitzats, hem registrat les següents funcions.

```
$Twig = new Twig_Environment($loader);
```

```
$Twig->addFunction('ip2long', new Twig_Function_Function('ip2long'));
$Twig->addFunction('long2ip', new Twig_Function_Function('long2ip'));
$Twig->addFunction('t', new Twig_Function_Function('t'));
$Twig->addFunction('guifi_to_7bits', new Twig_Function_Function('guifi_to_7bits'));
$Twig->addFunction('guifi_get_alchemy_ifs',
    new Twig_Function_Function('guifi_get_alchemy_ifs'));
$Twig->addFunction('guifi_main_ip', new Twig_Function_Function('guifi_main_ip'));
$Twig->addFunction('explode', new Twig_Function_Function('explode'));
```

5.4 Nodes en mode client

Com hem vist a la part d'anàlisi, els nodes client són aquells que es connecten a la xarxa però que representen un punt terminal, de manera que acostumen a tenir una única interfície i un únic enllaç amb un supernode, i requereixen una configuració mínima.

L'objecte de la implementació és, en essència, el contingut de la pròpia plantilla, que es pot trobar als annexes d'aquesta memòria. Per això en els següents punts comentarem el procés que ha suposat aquesta confecció.

5.4.1 Model Ubiquiti Nanostation2 - Firmware AirOs3.0

En primer lloc, analitzem a la web en producció un exemple concret de configuració UnSolClic per al que determinarem el conjunt de paràmetres dependents del Device. Considerem el cas d'un dels primers node client que consta amb aquesta combinació, SevaCasagroga⁵ a Seva(Osona), del que en mostrem una secció del seu UnSolClic⁶

```
wireless.1.ssid=Guifi.net-SevaCanBassesAP
netconf.2.ip=10.138.17.166
netconf.2.netmask=255.255.255.224
route.1.gateway=10.138.17.161
resolv.nameserver.1.ip=10.138.103.2
resolv.nameserver.2.ip=10.138.38.5
resolv.host.1.name=SevaCasagrogaRd1
snmp.location=SevaCasagroga
radio.1.ieee_mode=b
radio.1.rate.max=11M
radio.1.txpower=6
radio.1.acktimeout=45
```

FIGURA 5.12: Fragment d'UnSolClic real d'un node client amb Ubiquiti Nanostation 2

⁵<http://Guifi.net/ca/node/22013>

⁶<http://Guifi.net/files/nanostation/SevaCasagrogaRd1.cfg>

Observem que tot el que siguin adreces IP i noms concrets associats els podem considerar paràmetres del Device i els podem substituir, mitjançant la sintaxi de Twig, per la paraula que hem designat com a origen del paràmetre vinculat o com el nom únic que mitjançant la convenció establerta, es mapeja a la informació desitjada del Device.

Per al primer cas, veiem que podem incloure el SSID del supernode al que connectarem mitjançant el paràmetre `{{ wireless1ssid }}` que té per origen la informació del device `"linkedto_ssid"`.

Per altra banda, podem accedir a totes les propietats que ens arriben del device mitjançant el prefix `'dev'` seguit del nom provinent de la convenció, de manera que podem obtenir el nom del nostre router amb el codi `{{ dev.nick }}`. Tal com l'interpreta Twig, `'dev'` és l'objecte Device que ens arriba i `'nick'` és la paraula propietat amb la que ens arriba el nom del Device.

```
wireless.1.ssid=Guifi.net-{{ wireless1ssid }}
netconf.2.ip={{ ipv4_ip }}
netconf.2.netmask={{ ipv4_netmask }}
route.1.gateway={{ linkedto_gateway }}
resolv.nameserver.1.ip={{ zone_primary_dns }}
resolv.nameserver.2.ip={{ zone_secondary_dns }}
resolv.host.1.name={{ dev.nick }}
snmp.location={{ node_nick }}
radio.1.ieee_mode=b
radio.1.rate.max=11M
radio.1.txpower={{ radio1txpower }}
radio.1.acktimeout=45
```

FIGURA 5.13: Fragment de la plantilla d'UnSolClic per a Ubiquiti Nanostation 2

En aquest fragment també podem veure el cas del paràmetre `'radio1txpower'`, que s'utilitza per regular la potència d'emissió del router i que no depèn del device. En aquest cas, el paràmetre està associat al firmware i està inicialitzat amb un valor per defecte que s'aplicarà a tots els models que implementin aquest firmware. Això ens permet definir un comportament global a nivell de Firmware, que podríem canviar més endavant si fos necessari.

Per altra banda, aquesta mena de paràmetres també permeten la seva particularització per a la configuració UnSolClic que estem tractant, de manera que es pot especialitzar el valor per a aquest model concret. Aquests comportaments en la inicialització dels paràmetres respon a l'objectiu que l'aplicació sigui molt flexible.

5.4.2 Model Linksys wrt54GL - Firmware dd-guifi

La plantilla per a node client amb aquest firmware es va realitzar abans que la configuració per a supernode i no s'havia tractat cap dels elements més complexes que requereix, per això no té estructures de control i és pràcticament el mateix cas que la Nanostation 2, canviant

únicament les comandes pròpies del firmware que inclou la plantilla. A la Figura 5.14 es mostra la selecció de les comandes que inclouen paràmetres, per ordre d'aparició.

```
# Generat per a:
# {{ firmware_name }}
# {{ dev.nick }}
nvram set router_name="{{ dev.nick }}"
nvram set wan_hostname="{{ dev.nick }}"
nvram set wan_ipaddr="{{ ipv4_ip }}"
nvram set wan_netmask="{{ ipv4_netmask }}"
nvram set wan_dns="{{ zone_primary_dns }} {{ zone_secondary_dns }} {{ secondarydns }}"
nvram set sv_localdns="{{ zone_primary_dns }}"
nvram set txpwr="{{ txpwr }}"
nvram set ntp_server="{{ zone_primary_ntp }}"
nvram set wl_ssid="Guifi.net-{{ linkedto_ssid }}"
nvram set wan_gateway="{{ ipv4_ipgateway }}"
# radio: {{ dev.id }}-{{ dev.nick }}
```

FIGURA 5.14: Fragment de la plantilla d'UnSolClic per a Linksys wrt54GL amb DD-guifi

5.5 Nodes en mode Access Point i Supernodes

En general, podem considerar que els supernodes interconnecten diverses àrees de la xarxa, ja sigui amb diferents zones o amb altres supernodes i clients propers; per tant, tindran almenys dues o més interfícies inalàmbriques amb la respectiva antena. Com hem vist a l'anàlisi del Capítol 3, per cadascuna d'aquestes interfícies tenim un seguit de blocs de configuració que podem definir, per exemple: els paràmetres físics de l'antena, els paràmetres de l'enllaç, els protocols d'enrutament, els pools d'adreces, l'assignació fixa de clients, els filtres de seguretat i altres com l'accés a convidats.

En aquest punt és on es fan imprescindibles les estructures de control que ens proporciona Twig com a sistema de plantilles, juntament amb la capacitat de tractar el device com un objecte amb propietats i mètodes que podem cridar des de la plantilla. D'aquesta manera podem per exemple, recórrer totes les interfícies per inicialitzar-ne el mode de funcionament o tots els clients d'un determinat punt d'accés i inicialitzar-ne el binding d'adreces IP fixes que determina l'aplicació.

Per la implementació d'aquestes plantilles, a més d'identificar els paràmetres, ha calgut anar a buscar el codi PHP existent que generava els fitxers UnSolClic i anar 'traduint' cadascuna de les accions a la sintaxi de les plantilles Twig.

5.5.1 Model Mikrotik RouterBoard 433 - Firmware RouterOS4.7

La sintaxi de la família de RouterOS no facilita precisament la confecció de plantilles, ja que els caràcters '{' i '}' també s'utilitzen per definir els seus propis blocs de control de flux. Per tractar amb un device concret considerem el cas d'un dels primers node client que consta amb aquesta combinació, FRCasetesMikrotik1 ⁷ situat a Fontrubí (Alt Penedès), del que en mostrem una secció del seu UnSolClic⁸.

A continuació, alguns comentaris a fragments significatius de la plantilla d'UnSolClic confeccionada per aquesta combinació:

A la Figura ?? veiem la part de la plantilla que genera la capçalera del fitxer de configuració UnSolClic. A part del nom del model i el firmware, podem veure que fem ús de la funció PHP que utilitza Drupal per a localitzar les cadenes de text. Aquesta és una de les funcions que hem registrat per a poder-la tenir disponible a les plantilles i que condicionarà la sortida a l'idioma(Locale) amb què visitem la pàgina corresponent al fitxer UnSolClic.

```
# Generat per a:
# {{ firmware_name }}
:log info "UnSolClic for {{ dev.id }}-{{ dev.nick }} going to be executed."
# {{ t('Configuration for') }} {{ firmware_description}}
# {{ t('device') }} : {{ dev.id }}-{{ dev.nick }}
```

FIGURA 5.15: Capçalera de la plantilla d'UnSolClic per a RouterBoard 433 amb RouterOS4.7

A la Figura 5.15 podem veure l'ús que fem dels blocs condicionals de Twig per comprovar si els valors que rebem prenen algun valor; en aquest cas, per determinar si configurarem un servidor DNS o dos.

```
# DNS (client &#038; server cache) zone: {{ zone_id }}
{% if zone_primary_dns %}
{% if zone_secondary_dns %}
/ip dns set servers={{ zone_primary_dns }},{{ zone_secondary_dns }}
allow-remote-requests=yes
{% else %}
/ip dns set servers={{ zone_primary_dns }} allow-remote-requests=yes
{% endif %}
{% endif %}
```

FIGURA 5.16: Definició dels servidors DNS de la plantilla d'UnSolClic per a RouterBoard 433 amb RouterOS4.7

En la següent figura 5.16 veiem l'ús que fem de les estructures de control per a iterar sobre l'array de radios pertanyents al device concret i com anem accedint a les propietats de cadascun dels seus elements; en aquest cas, per inicialitzar en quin mode de funcionament es troba cadascuna d'aquestes ràdios i en quina freqüència.

⁷<http://Guifi.net/guifi/device/5056>

⁸<http://Guifi.net/guifi/device/5056/view/UnSolClic>


```

{% for radio_id, radio in dev.radios %}
# Radio#: {{ radio.radiodev_counter }} {{ radio.ssid }}
{% spaceless %}
/interface wireless set wlan{{ radio.radiodev_counter+1 }}
name="wlan{{ radio.radiodev_counter+1 }}" radio-name="{{ radio.ssid }}"
mode={{ radio.mode }}-bridge ssid="Guifi.net-{{ radio.ssid }}" \
{% if radio.mode == 'ap' %}
    {% set mode = 'ap-bridge' %}
    {% if radio.channel < 5000 %}
        {% set band = '2.4ghz-b' %}
    {% else %}
        {% set band = '5ghz' %}
    {% endif %}
{% elseif (radio.mode == 'client' or radio.mode == 'clientrouted') %}
    {% set mode = 'station' %}

```

FIGURA 5.17: Iteració sobre ràdios i condició de mode de la plantilla d'UnSolClic per a Router-Board 433 amb RouterOS4.7

Després de registrar algunes de les funcions d'utilitats disponibles al mòdul Drupal-guifi, podem accedir, per exemple, a funcions de càlculs amb adreces IP, com mostra la figura 5.18.

```

{% if interface.interface_type != 'HotSpot' and interface.interface_type != 'Wan' %}
    {% if mode == 'ap-bridge' %}
        {% set maxip = ip2long(ipv4.netstart) +1 %}
        {% if maxip +5 > ip2long(ipv4.netend) -5 %}
            {% set maxip = ip2long(ipv4.netend) %}
            {% set dhcpDisabled = 'yes' %}
        {% else %}
            {% set maxip = maxip +5 %}
            {% set dhcpDisabled = 'no' %}
        {% endif %}
    
```

FIGURA 5.18: Crida a funcions del mòdul Guifi.net UnSolClic per a Mikrotik RouterBoard 433 amb RouterOS4.7

Finalment, a la Figura 5.19 mostrem un exemple de macro que es defineix dins el mateix fitxer i la seva crida, com un mètode que ens pot ajudar a fer més modular la nostra plantilla. En aquest cas, es reproduïx el comportament del codi anterior, on aquest bloc de definició d'enrutament BGP també s'encapsulava en una funció.

Cal observar que l'enfoc de les macros podria permetre incloure un fitxer general de definició de macros de manera que es poguessin crear rutines genèriques utilitzables per a totes les plantilles. Si encara no en tinguéssim prou, podríem crear extensions i incorporar-les a Twig i, fins i tot, crear el nostre propi lèxic. Tanmateix en cap dels casos tractats en aquest projecte ha estat necessari.

```
{% macro bgp_peer(id, host_name, ipv4, disabled) %}
/ routing bgp peer
:foreach i in [find name={{ host_name }}] do={/routing bgp peer remove $i;}
add name="{{ host_name }}" instance=default remote-address={{ ipv4 }}
remote-as={{ id }} multihop=no route-reflect=no ttl=default
in-filter=ebgp-in out-filter=ebgp-out disabled={{ disabled }}{% endmacro %}

{{ _self.bgp_peer(link.device_id, link.interface.ipv4.host_name,
link.interface.ipv4.ipv4, 'yes') }}
```

FIGURA 5.19: Definició de macro UnSolClic per a RouterBoard 433 amb RouterOS4.7

5.5.2 Model Linksys wrt54GL - Firmware dd-guifi

En començar a confeccionar la plantilla per al mode Access Point, es va partir de la plantilla client però, resseguint el codi original PHP, de seguida es va veure que la implementació seria fins i tot molt més complexa que el cas del Mikrotik RB433. La causa principal és que el firmware DD-guifi és un derivat del firmware DD-WRT, que al seu temps era el principal derivat de l'Alchemy de SveaSoft, el qual també derivava del firmware original que Linksys va haver d'alliberar per complir la GPL. Les petites diferències de cadascun fan que hi hagi molts matisos.

A més, atès que que la sintaxi per configurar aplicacions avançades no és tant compacta com la que pugui tenir el RouterOS de calen blocs molt llargs d'ordres sovint relacionades que acaben complicant més el traspàs de funcionalitats a la plantilla, fins i tot amb l'encapsulació en macros (Figura 5.20)

```
{% macro guifi_UnSolClic_ospf(dev,zone, ospfzone,interfaceWlanLan) %}
{% macro guifi_UnSolClic_dhcp(dev, interfaceWlanLan) %}
{% macro guifi_UnSolClic_wds_vars(dev) %}
{% macro guifi_get_alchemy_ifs(dev) %}
{% macro guifi_UnSolClic_vlan_vars(dev, rc_startup) %}
```

FIGURA 5.20: Llista de macros UnSolClic per a Linksys wrt54GL - DD-guifi

Degut a la implementació existent, en la que es consultava a la base de dades la informació relativa als tipus de interfícies del dispositiu i se n'obtenia l'estructura, per evitar fer aquestes crides es resseguien una a una totes les interfícies del Device comprovant si corresponien amb el tipus que volíem, tal com reflexa la Figura 5.21 podem veure reflexada aquesta dinàmica.

Finalment, cal comentar que va ser necessari registrar més funcions de PHP per tal d'evitar traspassar encara més complexitat a les plantilles quan es tractava de consultes aniuades o interdependents, per exemple 'guifi_get_alchemy_ifs', 'guifi_main_ip' i fins i tot la nativa de PHP 'explode' per tractar els resultats que provenien de les primeres.

```

{# busquem interfície Wan #}
{% set interfaceWan = null %}
{% for interface_id, interface in dev.interfaces %}
  {% for ipv4_id,ipv4 in interface.ipv4 %}
    {% if link.interface.interface_type == 'Wan' %}
      {% set interfaceWan = interface %}
      {% if ipv4.ipv4 %}
        nvram set lan_ipaddr="{{ ipv4.ipv4 }}"
        nvram set lan_gateway="{{ ipv4.netstart }}"
        nvram set lan_netmask="{{ ipv4.netmask }}"
        nvram set fullswitch="1"
        nvram set wan_dns="{{ zone_primary_dns }} {{ zone_secondary_dns }} \
        {{ zone_ternary_dns }}"
      {% else %}
        nvram set wan_proto="dhcp"
      {% endif %}
    {% endif %}
  {% endfor %}
{% endfor %}

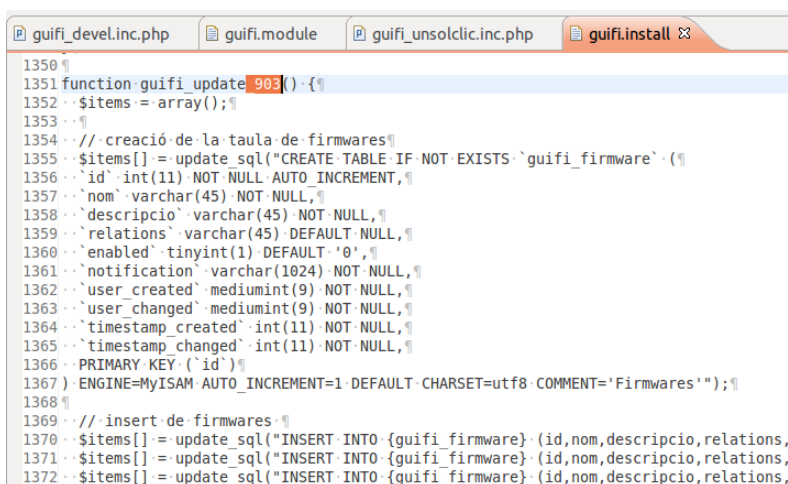
```

FIGURA 5.21: Consulta de la interfície WAN Linksys wrt54GL - DD-guifi

5.6 Procés d'instal·lació

Tots els scripts necessaris per actualitzar l'estructura de dades s'incorporen al fitxer `guifi.install` del mòdul Drupal-guifi que recopila seqüencialment tots els canvis per tal que es puguin reproduir de forma automàtica a qualsevol altra instal·lació de l'entorn de Guifi.net.

- Codificar una actualització en format esperat per Drupal:



```

1350
1351 function guifi_update($items) {
1352   $items = array();
1353   //
1354   // creació de la taula de firmwares
1355   $items[] = update_sql("CREATE TABLE IF NOT EXISTS `guifi_firmware` (
1356     `id` int(11) NOT NULL AUTO INCREMENT,
1357     `nom` varchar(45) NOT NULL,
1358     `descripcio` varchar(45) NOT NULL,
1359     `relations` varchar(45) DEFAULT NULL,
1360     `enabled` tinyint(1) DEFAULT '0',
1361     `notification` varchar(1024) NOT NULL,
1362     `user_created` mediumint(9) NOT NULL,
1363     `user_changed` mediumint(9) NOT NULL,
1364     `timestamp_created` int(11) NOT NULL,
1365     `timestamp_changed` int(11) NOT NULL,
1366     PRIMARY KEY (`id`)
1367   ) ENGINE=MyISAM AUTO INCREMENT=1 DEFAULT CHARSET=utf8 COMMENT='Firmwares');");
1368   //
1369   // insert de firmwares
1370   $items[] = update_sql("INSERT INTO {guifi_firmware} (id,nom,descripcio,relations,
1371   $items[] = update_sql("INSERT INTO {guifi_firmware} (id,nom,descripcio,relations,
1372   $items[] = update_sql("INSERT INTO {guifi_firmware} (id,nom,descripcio,relations,

```

FIGURA 5.22: Codificació de l'actualització

- Pujar el codi (commit) a la còpia personal del repositori oficial del mòdul Drupal-guifi



FIGURA 5.23: Pujada del codi al repositori GIT de gitorious

- Sol·licitar un Merge Branch al repositori oficial , Autoritzar el Merge i llançar actualització de dades mitjançant el sistema d'actualització de mòduls que incorpora Drupal

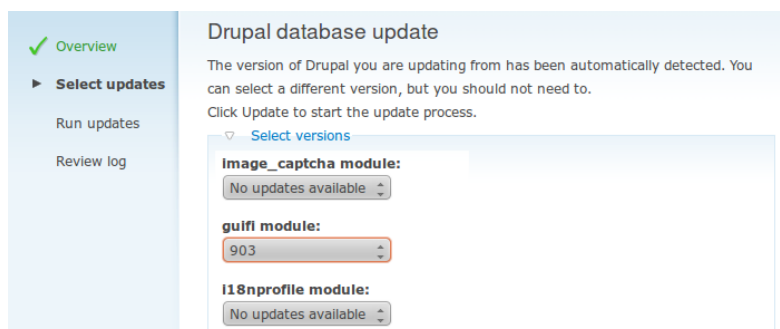


FIGURA 5.24: Aplicació de l'actualització del PFC

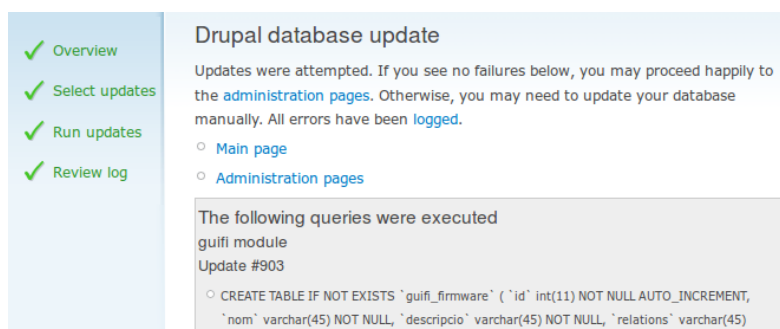


FIGURA 5.25: Resultat de l'actualització amb el codi i dades del projecte

5.7 Dificultats sorgides

Durant les diferents fases del procés d'implementació han anat sorgint diverses dificultats:

- L'adopció de les convencions i mètodes de programació existents al mòdul Drupal-guifi presenten una corba d'aprenentatge considerable ja que hi ha molts conceptes relacionats amb el funcionament de les xarxes en general i de Guifi.net en particular. Per altra banda, la integració dels formularis de gestió al mòdul es fa mitjançant el codi, dificultant d'aquesta manera la maquetació i la integració de validacions en el client amb JavaScript.
- En el cas d'aparells existents per als que cal traspassar una lògica de programació seqüencial a una plantilla amb estructures de control limitades, sovint trobem que cal repensar la manera de poder generar el mateix resultat sense fer servir variables globals o búffers de text.
- La crida repetida de funcions comuns de Guifi.net pot fer necessari que algunes d'elles retornin més informació per tal d'evitar fer aquestes crides diverses vegades des de la plantilla. Així, per exemple, les crides a funcions que manipulaven adreces IP sobre estructures complexes del Device, s'han eliminat de les plantilles gràcies a la seva inicialització prèvia en la etapa de construcció de l'objecte.
- La depuració de les plantilles durant el procés de confecció ha demostrat que la gestió web, pel fet d'editar sobre una àrea de text sense cap ajuda a l'editor, introdueix retards respecte al que podria ser una edició local de la plantilla en un fitxer. Per això, durant l'etapa de desenvolupament es va modificar el 'Loader' del Twig per poder treballar en fitxers locals i guanyar en velocitat.
- El mètode de verificació comentat en el Capítol 6, inicialment només comparava que els string fossin exactament iguals, de manera que molt sovint el resultat era que tots els fitxers generals eren incorrectes i que no aportava cap informació. Després d'utilitzar l'aplicació de comparació de fitxers i directoris 'Meld' i afegir algorismes qualitius d'avaluació, tot el mètode té més sentit.
- El replantejament de les estructures de dades després de triar el motor de plantilles Twig va aportar noves maneres d'accedir a les dades.

Capítol 6

Resultats i proves

En aquest capítol desglossarem la distribució d'aparells existents a la xarxa guifi.net segons la combinació de fabricant, model i firmware amb l'objectiu de contextualitzar les plantilles desenvolupades, de les quals examinarem el resultat generats amb el nou sistema, comentant-ne les desviacions i les sol·lucions proposades.

6.1 Distribució de maquinari a guifi.net

La normalització de les dades necessària per implementar la nova estructura ens permet obtenir més fàcilment dades estadístiques sobre la distribució de maquinari a la xarxa. Així, per exemple, podem visualitzar les següents dades obtingudes a data 5/5/2012:

Totalització d'entitats	
Fabricants	11
Models	44
Firmwares	21
Aparells	22191

FIGURA 6.1: Totalització d'entitats

Com es pot veure a la Figura 6.2 el nombre de nodes operatius és molt elevat, però sens dubte un dels indicadors més clars que guifi.net encara es troba en una fase d'expansió és la suma dels nodes 'En construcció' i 'En Proves' indica els que aviat es sumaran als operatius.

La Figura 6.3 mostra els 15 primers registres de l'agregació de Model i Firmware d'un total de 196 combinacions diferents. Observant el llistat, si considerem que el funcionament de les Ubiquiti Nanostation2, Nanostation5, Nanostation Loco2 i Nanostation Loco5 és el mateix, veiem que cobrim 5 de les 10 primeres combinacions, amb un total de 9744 dispositius (un 44% del total).

Total d'aparells per estat	
Operatius	17442
Projectats	3525
En proves	603
En construcció	429
Reservats	177
Esborrats	15

FIGURA 6.2: Total d'aparells per estat

Total	Fabricant	Model	Firmware
3730	Ubiquiti	NanoStation2	AirOsv30
2138	Ubiquiti	NanoStation5	AirOsv30
1851	Linksys	WRT54GL	DD-guifi
1621	Ubiquiti	AirMaxM5 Rocket/Nano/Loco	AirOsv52
1468	Ubiquiti	NanoStation2	AirOsv221
1254	Buffalo	WHR-HP-G54, WHR-G54S	DD-guifi
1126	Ubiquiti	NanoStation Loco5	AirOsv30
929	Ubiquiti	AirMaxM5 Bullet/PwBrg/AirGrd/NanoBr	AirOsv52
899	Ubiquiti	NanoStation Loco2	AirOsv30
457	Mikrotik	Routerboard SXT 5HnD	RouterOSv5.x
430	Linksys	WRT54GL	DD-WRTv23
429	Ubiquiti	NanoStation Loco5	AirOsv3.6+
366	Ubiquiti	AirMaxM2 Rocket/Nano/Loco	AirOsv52
298	Mikrotik	Routerboard 133C	RouterOSv2.9
258	Buffalo	WHR-HP-G54, WHR-G54S	DD-WRTv23
224	Ubiquiti	NanoStation5	AirOsv3.6+

FIGURA 6.3: Distribució d'aparells segons fabricant- model i firmware

6.2 Comprovació automatitzada

Un dels requeriments funcionals del projecte és que doni suport als aparells que actualment estan en ús a la plataforma de guifi.net. Per aquest motiu calia que amb el nou sistema, els fitxers de configuració UnSolClic fossin idèntics. Després de confeccionar les plantilles per als casos de nodes i supernodes indicats al Capítol 5, i sobre uns aparells coneguts anar comparant el resultat obtingut amb el correcte, calia contrastar també els altres aparells amb la mateixa combinació de Model-Firmware però amb diferències de configuració, diferent nombre d'enllaços, diferents protocols, etc.

Degut al gran nombre de dispositius per cada combinació calia un mètode automatitzat que pogués comprovar cadascun dels fitxers UnSolclic generats a partir de les plantilles aprovades aplicant les diferents configuracions. Es va crear una pàgina al menú principal del projecte des d'on a partir d'especificar una web amb la plataforma de Guifi.net considerada correcte (anomenada Trusted) i una web amb una plataforma de desenvolupament (Local), es poguessin buscar totes les configuracions i aparells afectats, per consultar per HTTP el seu fitxer UnSolclic a les dues les plataformes i comparar-les entre sí.

Inici > Menú guifi.net > Eines de desenvolupament

Check UnSolclics

Trusted Site Testing Site Device Id

Date Limit Algorithm Max Devices

Configuració UnSolClic : Routerboard 433-RouterOSv4.7+ (id 6)
 Checking device 1904 : 98.85 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 1911 : 98.85 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 1936 : 98.86 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 5056 : 99.12 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 25356 : 98.98 with Comparison Algorithm - Text_similar Local | Trusted
 Results for UnSolClic #Routerboard 433-RouterOSv4.7+(6) - [Total] : 5

Configuració UnSolClic : NanoStation2-AirOsv30 (id 19)
 Checking device 19950 : 100.00 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 16469 : 100.00 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 12747 : 100.00 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 12996 : 100.00 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 23487 : 100.00 with Comparison Algorithm - Text_similar Local | Trusted
 Results for UnSolClic #NanoStation2-AirOsv30(19) - [Total] : 5

Configuració UnSolClic : WRT54GSv4-DD-guifi (id 20)
 Checking device 315 : 6.27 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 1640 : 90.05 with Comparison Algorithm - Text_similar Local | Trusted
 Checking device 2001 : 99.44 with Comparison Algorithm - Text_similar Local | Trusted

FIGURA 6.4: Eina de comprovació de fitxers UnSolclics

Aquesta eina de comprovació té la limitació que les consultes via web alenteixen molt l'execució del script, encara que les dues instàncies de la web s'allotgin en la màquina local. Això, que juntament amb el gran nombre d'aparells, fa que es pugui esgotar el temps disponible d'execució per al script si no disposem d'accés per augmentar-ne el valor màxim. Tot i així, aquest enfoc no és el recomanable a nivell de rendiment per una màquina en producció; per això s'ha fet que es pugui filtrar i limitar els aparells a testejar, tot i que això implica que no es puguin cobrir tants casos en cada prova i, per tant, calguin més proves i més temps.

En el cas del desenvolupament de les plantilles aportades, la forma de visualitzar les diferències per cada fitxer de configuració consistia en generar en dos directoris, local i trusted, cadascun dels fitxers indexats per l'identificador de l'aparell segons l'origen. Un cop disposàvem de tots els fitxers, s'utilitzava una aplicació de visualització de diferències, com per exemple Meld¹ per identificar cada divergència. L'ideal seria poder integrar un visualitzador de les diferències a la pròpia eina, però això quedava fora de l'abast d'aquest projecte.

A la Figura 6.4 podem veure l'aparença del comparador automàtic. A la part superior trobem totes les opcions possibles, els dos dominis (trusted i local) sobre els que farem les proves, un filtre de data màxima per poder ignorar els aparells creats en els darrers N dies, un filtre per identificador i el nombre màxim d'aparells a comprovar per cada configuració UnSolclic.

¹<http://meldmerge.org> és un visualitzador de diferències per a fitxers i directoris

També hi trobem un combo que ens permet seleccionar l'algorisme de comparació amb les següents opcions:

- **Exact:** S'utilitza una funció que considera tot el fitxer `UnSolClic` com un string molt llarg i comprova si és idèntic
- **Similar Text:** Utilitza la funció nativa de PHP `'similar_text'`² a la que li podem especificar que ens retorni un índex de semblança en percentatge.
- **Levenshtein:** utilitza la funció nativa de PHP `'Levenshtein'`³ que ens calcula el nombre de caràcters que hem de canviar per convertir una cadena en l'altra.

Les dues darreres opcions no són adients per a realitzar comprovacions per un nombre elevat de nodes, ja que la seva complexitat per a cadenes de longitud m i n són $O(\max(n,m)^3)$ i $O(m*n)$ per a `Similar_text` i `Levenshtein`, respectivament. El fet que les cadenes que tractem siguin de longituds molt grans aconsella que aquest dos mètodes s'utilitzin amb filtres molt i molt restrictius. Tot i això, hem trobat interessant d'implementar-les perquè la comparació per exactitud no ens donava cap indicador de la qualitat de la plantilla.

6.3 Resultats de les proves

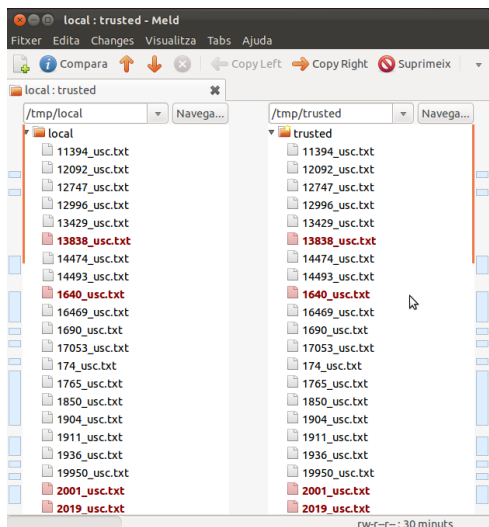
Després de generar un conjunt d'`UnSolClics` mitjançant l'eina del comparador, podem analitzar els resultats de forma visual amb `Meld`. En primer lloc, a la Figura 6.5(a), podem veure a nivell de directori quins fitxers que són iguals i quins diferents. Els fitxers ressaltats amb negreta i color són els que presenten alguna diferència (cal tenir en compte que per a les comparacions no es comptabilitzen espais, tabuladors, etc.)

Per als fitxers que no apareixen ressaltats, podem clicar per comprovar que no presenten diferències i veure una pantalla com la de la Figura 6.5(b).

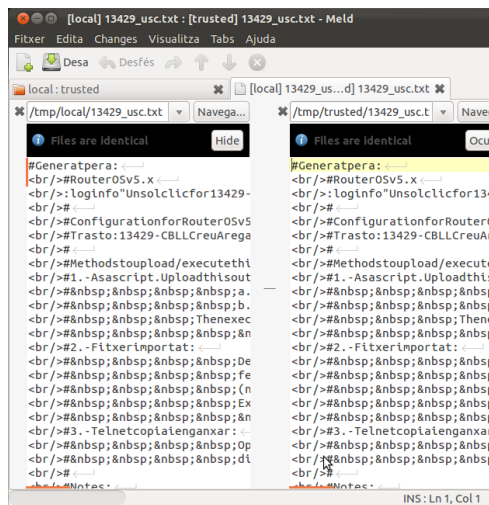
En canvi, per als que sí que presenten diferències, podrem visualitzar-les ubicades en el context de cada fitxer i anar-les seleccionant tal com mostra la Figura 6.5(c). Una de les funcionalitats principals d'aquesta aplicació, que seria seleccionar i sincronitzar la versió correcta de cadascun dels fitxers, no la podem utilitzar directament en aquest cas ja que estem visualitzant el resultat d'aplicar les plantilles que tenim a la base de dades, per tant hem de editar la plantilla original al gestor web per codificar els canvis necessaris.

²<http://www.php.net/manual/es/function.similartext.php>

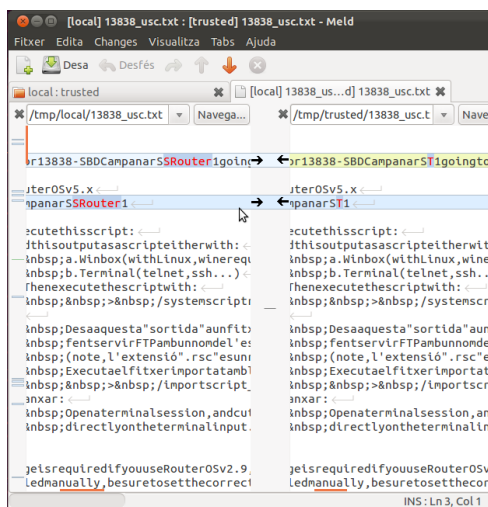
³<http://php.net/manual/es/function levenshtein.php>



(a) Visualització de diferències a nivell de directori



(b) Visualització de fitxers idèntics



(c) Visualització de fitxers amb diferències

FIGURA 6.5: Anàlisi de resultats amb Meld

6.3.1 Anàlisi de desviacions i solucions aportades

Mitjançant l'eina de comparació, hem pogut detectar i corregir moltes desviacions. A continuació detallem els tipus més freqüents i la solució proposada :

- *Utilització de còpies obsoletes de la base de dades:* Guifi.net té un creixement constant i la plataforma de desenvolupament va partir d'una imatge de la base de dades feta al principi de la fase d'implementació d'aquest projecte, mentre que per la instància que considerem trusted es va baixar una versió més nova; per aquest motiu, tots els canvis efectuats als nodes (nom, enllaços, clients) es reflexen en la comparació com a informació sobrant. La solució seria fer una còpia actual de la BD, instal·lar-hi totes les estructures necessàries i dades del projecte i repetir les proves.

- *Traducció de cadenes de comentaris*: Alguns dels comentaris que apareixen al UnSolClic estan localitzats segons l'idioma del navegador del visitant. En la confecció de plantilles podem posar els textos directament o fer una crida a la funció nativa que ens inclourà la traducció. Si no es realitza aquesta crida podem obtenir diferències depenent de l'idioma utilitzat(Locale). Caldria definir una convenció en els textos de les plantilles i unificar el comportament de les diferents webs de prova utilitzant els mateixos idiomes.
- *Codificacions de caràcters*: Twig per defecte fa un autoescape dels strings que mostra, de manera que si no codifiquem un bloc en mode autoescape = false pot ser que alguns caràcters es codifiquin i provoquin diferències. Per solucionar aquest problema podem forçar a Twig a no codificar els caràcters, ja sigui per tota la plantilla o per seccions concretes.
- *Errors en la implementació de blocs*: En alguns casos s'ha detectat que alguns comentaris separadors de blocs no apareixen correctament, fent que comptabilitzi com a diferència malgrat que no influeixen en la semàntica de l'UnSolClic. Aquests errors són deguts al incorrecte traspàs de funcionalitat del codi actual a la plantilla, fent que algun bloc de comentari estigui fora de l'àmbit d'un dels bucles que el genera. La solució seria depurar el codi i corregir la ubicació del comentari dins la plantilla.
- *Errors en la lògica de la plantilla*: En el procés de traspàs de funcionalitats actuals a la plantilla, es poden produir mal-interpretacions en reproduir la lògica i generar de forma errònia iteracions sobre bucles o inicialitzacions de propietats. En el cas de confeccionar plantilles noves també es podrien reproduir aquests errors però no tindriem una font original contrastada amb la que comparar. La solució per aquests casos passa per depurar la plantilla resseguint el codi original.

Capítol 7

Conclusions

En aquest darrer capítol avaluarem l'experiència i extreurem conclusions del procés que ha seguit aquest projecte, valorant l'assoliment dels objectius inicials i descrivint les línies d'actuació futures.

Avaluació general: Pel que fa al compliment dels requisits funcionals detallats al Capítol 3, podem afirmar que tots els relacionats amb la gestió de noves entitats i plantilles integrades en Drupal s'han acomplert, mentre el requisit de crear les plantilles per a tots els models existents a Guifi.net ha demostrat ser massa ambiciós. Com ja s'ha vist al Capítol 6, l'elevat nombre d'aparells i firmwares fa inabastable per aquest projecte cobrir totes les combinacions (196).

D'altra banda, no s'han creat pantalles d'estadístiques sobre les entitats perquè no eren estrictament necessàries per al funcionament de l'aplicació, tot i que l'esquema relacional introduït permetrà que es puguin crear fàcilment.

Pel que fa a la planificació inicial del projecte, cal dir que per una banda escollir Twig com a sistema de plantilles va treure gran part de la feina prevista de parametrització d'aparells, mentre que per l'altra, la confecció de les plantilles per a supernodes va resultar força més complicada del que semblava a priori, especialment en el cas del Linksys WRT54GL en mode AP. De fet es pot afirmar que la potència de Twig en la manipulació d'objectes va fer replantejar l'enfoc inicial del projecte basat en paràmetres configurables.

Cal afegir també que les proves realitzades han donat resultats molt satisfactoris en un conjunt elevat de aparells, amb índexs de similitud propers al 100 % per als que s'ha comprovat que presentaven diferències que no afectaven al funcionament de l'UnSolClic. Tot i així, algunes configuracions apareixien amb percentatges molt baixos, i caldrà esmerçar més temps en detectar-ne les causes a les plantilles.

De l'objectiu de crear documentació del projecte i publicar-la a la wiki de Guifi.net cal dir que encara no s'ha fet però que espero que fragments d'aquesta memòria es puguin utilitzar amb aquest propòsit.

Estat final del projecte :

Pel que fa integrar el codi dins dels repositoris públics de Guifi.net, un dels requisits no funcionals, podem incloure l'adreça web repositori personal¹ clonat del repositori master de Drupal-guifi on s'han publicat la primera versió de l'aplicació. Aquesta branca caldrà provar-la per altres membres de la comunitat a la plataforma de proves abans no es pugui integrar definitivament en la plataforma de producció i estigui disponible per a tots els usuaris.

Evolució Futura : Després que l'aplicació superi el període de proves serà quan es posarà a prova davant els usuaris avançats de Guifi.net, que seràn els qui l'utilitzaran per a crear noves configuracions. Els usuaris estàndard no notaran cap canvi en el funcionament general de l'aplicació. S'espera que es puguin familiaritzar fàcilment amb la idea general de l'aplicació i puguin adquirir ràpidament els coneixements necessaris de l'estructura lògica de Guifi.net i la sintaxi de Twig per poder confeccionar noves plantilles.

Pel que fa a la evolució futura d'aquesta aplicació, podem pensar que amb una parametrització adequada, es podrien confeccionar plantilles per a altres tipus de dispositius de xarxa que no siguin routers, sobretot considerant que els futurs desplegaments de fibra òptica poden incorporar aparells que també caldrà configurar.

Pensant en treure profit de les noves estructures de dades es podria per exemple, a més de treure estadístiques d'ús de cada aparell, afegir-hi informació sobre els costos aproximats de cada aparell i així generar una mena d'informes econòmics a diferents nivells, ja sigui per un node concret, per un tram de xarxa concret, per una zona, etc. Aquesta informació podria utilitzar-se tant per conscienciar als usuaris del cost de la infraestructura que utilitzen com per, donat el cas, mostrar a algú proveïdor o fabricant el gran volum d'aparells que formen la xarxa per tal que hi col·labori d'alguna manera.

Una altra línia possible seria crear una nova entitat que fos una relació de diversos aparells, els anomenats nodes híbrids a Guifi.net, de manera que és pogues crear plantilles associades a model, firmware i tipus de híbrid, de manera que, per exemple, un Ubiquiti Nanostation2 utilitzés la plantilla desenvolupada, però en cas de formar part d'una associació de nodes que incloguin certs models, utilitzi una altra plantilla diferent. Estaria per veure quants tipus de nodes híbrids sortirien, tot i que actualment la majoria consisteixen en unir aparells de Mikrotik per fer la feina pesada d'enrutar i aparells de d'Ubiquiti per connectar i donar cobertura.

¹https://gitorious.org/albertsarle/guifi/albertsarle-drupal-guifi_pfc

Valoració personal : En el moment d'escriure aquestes línies veig que el temps que a principi del projecte semblava molt llarg en realitat ha passat molt ràpid, i que al llarg d'aquests mesos s'ha concretat tot el que esperava del projecte. A part de posar en pràctica les tècniques necessàries per publicar codi en un projecte de programari lliure, també em veig capaç de contribuir en altres aspectes del desenvolupament de l'aplicació de Guifi.net.

Per altra banda, he de reconèixer que més que el projecte proposat per un professor, em feia il·lusió poder realitzar un projecte que sortís de mi, i que hagi estat sobre Guifi.net, una iniciativa a la que he dedicat moltes hores, em fa especialment content.

I si a això li afegim que ha de servir per acabar la carrera després de tant de temps, aleshores la satisfacció ja no té límit.

Appendix A

Glossari

Aparell / Device : Una màquina concreta dins de la xarxa guifi.net amb una configuració determinada segons el seu entorn.

BSD : Llicència de programari Berkeley Software Distribution menys restrictiva que la GPL.

CapacitatsModel : Cadascuna de les propietats físiques que incorpora un model concret i que determinen la seva funcionalitat.

CRUD : Acrònim en anglès per a Create/Read/Update/Delete és a dir, gestió bàsica de registres d'una base de dades.

Fabricant / Manufacturer : Casa comercial que desenvolupa un cert aparell.

GIT : Sistema distribuït de control de versions dissenyat per Linus Torvalds.

GPL : Llicència de programari lliure General Public License.

Interfície / Interface : Un o més adaptadors de xarxa d'un model concret.

Links : Cadascuna de les connexions que un aparell concret de la xarxa realitza amb altres aparells veïns.

LOS : De l'anglès Line Of Sight, línia de visió directa.

Model : Denominació que rep un dels productes del fabricant al que associem certes prestacions.

Networks : Cadascun dels rangs de xarxa pels quals escolta una interfície de un Aparell concret.

Node Client : Aparell que no expandeix la xarxa i que depèn d'un supernode per obtenir accés a la xarxa .

Node Hibrid : Combinació de dos o més aparells configurats per repartir-se les tasques d'un supernode.

P2P : Acrònim de l'anglès Peer-to-Peer, designa una relació entre iguals.

Proxy (Servidor) : Servidor que té connectivitat a a Guifi.net i a Internet i actua de pasarella entre les dues.

PHP : Acrònim recursiu de 'PHP Hipertext Pre-processor', llenguatge interpretat amb el que s'ha desenvolupat el projecte.

PoE : Power Over Ethernet, estàndard per alimentar dispositius utilitzant els parells de cables no utilitzats dels cables de xarxa.

Ràdio : Un o més adaptadors inalàmbrics de xarxa d'un model concret.

SSID : Acrònim de Service Set IDentifier, el nom amb el que es publica una xarxa sense fils.

Supernode : Aparell que ofereix connectivitat als nodes client del seu voltant i que connecta amb altres supernodes de zones veïnes.

Twig : Motor de plantilles amb llicència BSD associat al framework Symfony.

UnSolClic : Fitxer de configuració d'un aparell concret amb tota la informació necessària per connectar, confeccionada adhoc segons el model i firmware que li hem definit.

Signat: Albert Sarlé Puig

Bellaterra, Juny de 2012

RESUM

Guifi.net és una xarxa lliure de telecomunicacions construïda a iniciativa dels propis participants que, mitjançant un acord entre iguals, s'interconnecten per compartir serveis i recursos. Del vincle filosòfic amb el programari lliure en deriva que tota la informació sigui pública, mentre que de l'agnosticisme tecnològic, l'utilització de qualsevol aparell del mercat. Per això existeix l'unsolclic', una seqüència d'ordres de configuració genèrica de dispositius i veritable factor d'èxit.

L'objectiu del projecte és millorar el procés d'incorporació de nous dispositius a l'aplicació actual. Mitjançant una nova gestió web i un sistema de plantilles estàndard, els usuaris avançats podran crear els configuradors 'unsolclic' per als nous dispositius del mercat i mantenir els existents amb més facilitat i eficiència.

RESUMEN

Guifi.net es una red libre de telecomunicaciones construida a iniciativa de los propios participantes que, mediante un acuerdo entre iguales, se interconectan para compartir servicios y recursos. Del vínculo filosofico con el software libre deriva que toda la información sea pública, mientras que del agnosticismo tecnologico, la utilización de cualquier dispositivo del mercado. Por esto existe el 'unsolclic', una secuencia de órdenes de configuración genérica de dispositivos y verdadero factor de éxitos.

El objetivo del proyecto es mejorar el proceso de incorporación de nuevos dispositivos a la aplicación actual. Mediante una nueva gestión web y un sistema de plantillas estándar, los usuarios avanzados podran crear los configuradores 'unsolclic' para los nuevos dispositivos del mercado y mantener los existentes con más facilidad y eficiencia.

ABSTRACT

Guifi.net is a free, open and neutral, telecommunications network build on the initiative of its own community members, who share services and resources through interconnection agreement between equals. In the spirit of open source and technological agnosticism, all the information is publicly available and the network allows to use any commercial device. This purpose is reflected by 'UnSolClic', a script for generic device configuration and a real factor of success.

The aim of this project is to improve the process of adding new devices to the current application. By using a new web management interface and a standard template system, advanced users will be able to create the 'UnSolClic' configurators for the new devices on the market, and to maintain the existing ones with greater ease and efficiency.