

ÍNDEX

CAPÍTOL 1. Introducció	15
1.1. La domòtica	15
1.1.1. Què és la domòtica?	15
1.1.2. Arquitectura	16
1.1.3. Solucions tecnològiques	17
1.2. Objectius del projecte	19
1.3. Breu descripció del paquet d'aplicacions desenvolupades	19
CAPÍTOL 2. Tecnologia utilitzada	21
2.1. Tecnologia JAVA	21
2.1.1. Introducció	21
2.1.2. J2ME	22
2.1.3. Què és un Servlet?	24
CAPÍTOL 3. Solucions actuals	27
CAPÍTOL 4. La nostra solució	29
4.1. Introducció	29
4.1.1. La Xarxa domòtica	29
4.1.2. Servidor	30
4.1.3. Telèfon mòbil	30
4.1.4. Xarxa domòtica + Servidor + Telèfon mòbil	30
4.2. Especificació de la nostra llar	31
4.2.1. Introducció	31
4.2.2. Paquet <i>llistaobjectes</i>	32
4.2.3. Classe <i>llista</i>	32

4.2.4.	Paquet <i>Casa</i>	33
4.2.5.	Interfície <i>meuPersistent</i>	33
4.2.6.	Classe <i>Aparell</i>	34
4.2.7.	Classe <i>Alarma</i>	36
4.2.8.	Classe <i>Calefacció</i>	37
4.2.9.	Classe <i>Climatització</i>	38
4.2.10.	Classe <i>DetFum</i>	38
4.2.11.	Classe <i>Finestra</i>	39
4.2.12.	Classe <i>Fogons</i>	41
4.2.13.	Classe <i>Forn</i>	42
4.2.14.	Classe <i>Llum</i>	43
4.2.15.	Classe <i>Microones</i>	44
4.2.16.	Classe <i>Nevera</i>	46
4.2.17.	Classe <i>PC</i>	47
4.2.18.	Classe <i>Persiana</i>	48
4.2.19.	Classe <i>Porta</i>	49
4.2.20.	Classe <i>Rentaplats</i>	50
4.2.21.	Classe <i>TV</i>	51
4.2.22.	Classe <i>TermoElectric</i>	53
4.2.23.	Classe <i>Video</i>	54
4.2.24.	Classe <i>Hab</i>	55
4.2.25.	Classe <i>Planta</i>	56
4.2.26.	Classe <i>Casa</i>	57
4.2.27.	Classe <i>Cases</i>	58
4.3.	El MIDlet	59
4.3.1.	Introducció	59
4.3.2.	Especificació del paquet <i>llistaobjectes</i>	63
4.3.3.	Especificació del paquet <i>Casa</i>	63
4.3.4.	Especificació del paquet <i>myRMS</i>	64
4.3.5.	Classe <i>myrms</i>	64
4.3.6.	Especificació del paquet <i>UserPass</i>	65
4.3.7.	Classe <i>UP</i>	65

4.3.8. Especificació del paquet <i>icones</i>	65
4.3.9. Especificació del paquet <i>Control</i>	66
4.3.10. Classe <i>ControlMidlet</i>	66
4.3.11. Classe <i>PSeISer</i>	68
4.3.12. Classe <i>PAfeModNouSer</i>	70
4.3.13. Classe <i>PModSer</i>	72
4.3.14. Classe <i>FConnLogin</i>	74
4.3.15. Classe <i>FUsAlta</i>	76
4.3.16. Classe <i>LldsCa</i>	78
4.3.17. Classe <i>LMenu</i>	80
4.3.18. Classe <i>LZona</i>	82
4.3.19. Classe <i>LTAp</i>	83
4.3.20. Classe <i>LAv</i>	85
4.3.21. Classe <i>FormAparell</i>	86
4.3.22. Classe <i>FAlarma</i>	88
4.3.23. Classe <i>DetFum</i>	89
4.3.24. Classe <i>FFinestra</i>	91
4.3.25. Classe <i>FFogons</i>	92
4.3.26. Classe <i>FForn</i>	93
4.3.27. Classe <i>FLlum</i>	96
4.3.28. Classe <i>FMicroones</i>	97
4.3.29. Classe <i>FNevera</i>	99
4.3.30. Classe <i>FPC</i>	101
4.3.31. Classe <i>FPersiana</i>	102
4.3.32. Classe <i>FPorta</i>	104
4.3.33. Classe <i>FRentaPlats</i>	105
4.3.34. Classe <i>FTE</i>	106
4.3.35. Classe <i>FTV</i>	108
4.3.36. Classe <i>FTemp</i>	109
4.3.37. Classe <i>FVideo</i>	112
4.4. El Servlet	114
4.4.1. Introducció	114

4.4.2.	Paquet <i>SerialNumber</i>	115
4.4.3.	Classe <i>SN</i>	115
4.4.4.	Paquet <i>UserPassSerial</i>	116
4.4.5.	Classe <i>UPS</i>	116
4.4.6.	Classe <i>Users</i>	118
4.4.7.	Paquet <i>servletcasavirtual</i>	119
4.4.8.	Servlet <i>SeCa</i>	119
4.5.	Aplicació <i>Casa Virtual</i>	121
4.5.1.	Introducció	121
4.5.2.	Especificació del paquet <i>llistaobjectes</i>	124
4.5.3.	Especificació del paquet <i>Casa</i>	124
4.5.4.	Especificació del paquet <i>CrearSimularCasa</i>	124
4.5.5.	Classe <i>CSCasaMain</i>	125
4.5.6.	Classe <i>PantPrinCasa</i>	125
4.5.7.	Classe <i>CrearCasa</i>	126
4.5.8.	Classe <i>CarrCasa</i>	133
4.5.9.	Classe <i>SimCasa</i>	134
CAPÍTOL 5. Procés de desenvolupament del paquet d'aplicacions		141
5.1.	Introducció	141
5.2.	Desenvolupament i proves	141
CAPÍTOL 6. Captures de pantalla		147
6.1.	Midlet	147
6.2.	Aplicació <i>CasaVirtual</i>	156
CAPÍTOL 7. Recursos tecnològics utilitzats		161
7.1.	Introducció	161
7.2.	Recursos hardware	161

7.3. Recursos software	162
CAPÍTOL 8. Planificació del projecte	165
8.1. Introducció	165
8.2. Planificació inicial	165
8.3. Planificació final	166
CAPÍTOL 9. Anàlisi econòmic	169
9.1. Introducció	169
9.2. Anàlisi econòmic partint de la primera planificació del projecte	169
9.3. Anàlisi econòmic partint de la segona planificació del projecte	170
9.4. Anàlisi econòmic partint de la fusió de la primera i segona planificació	171
CAPÍTOL 10. Conclusions	173
10.1.Objectius assolits	173
10.2.Possibles millores	175
10.3.Agraïments	177
CAPÍTOL 11. Bibliografia	179
11.1.Libres consultats	179
11.2.URLs consultades	179
CAPÍTOL 12. Annexe	181
12.1.Extensió de totes les classes del nostre paquet d'aplicacions	181
12.1.1. MIDlet	181
12.1.2. Servlet	181
12.1.3. Aplicació <i>CasaVirtual</i>	182

ÍNDEX DE FIGURES

1.1 Xarxa domòtica: Tipus d'arquitectura	16
1.2 Xarxa domòtica: Tipus de nivell tecnològic	17
2.1 Típic cicle de vida d'un servlet.	25
4.1 Classe <i>llista</i>	33
4.2 Classe <i>meuPersistent</i>	34
4.3 <i>Esquema dels mètodes persist i resurrect de la interfície meuPersistent</i>	34
4.4 Classe <i>Aparell</i>	35
4.5 Classe <i>Alarma</i>	36
4.6 Classe <i>Calefacció</i>	37
4.7 Classe <i>Clima</i>	38
4.8 Classe <i>DetFum</i>	39
4.9 Classe <i>Finestra</i>	40
4.10 Classe <i>Fogons</i>	41
4.11 Classe <i>Forn</i>	42
4.12 Classe <i>Llum</i>	44
4.13 Classe <i>Microones</i>	45
4.14 Classe <i>Nevera</i>	46
4.15 Classe <i>PC</i>	47
4.16 Classe <i>Persiana</i>	48
4.17 Classe <i>Porta</i>	49
4.18 Classe <i>Rentaplats</i>	51
4.19 Classe <i>TV</i>	52
4.20 Classe <i>TermoElectric</i>	53
4.21 Classe <i>Video</i>	54
4.22 Classe <i>Hab</i>	56
4.23 Classe <i>Planta</i>	57
4.24 Classe <i>Casa</i>	58
4.25 Classe <i>Cases</i>	59
4.26 Classe <i>myrms</i>	65
4.27 Classe <i>UP</i>	66
4.28 Classe <i>ControlMidlet</i>	67
4.29 Classe <i>CPSeISer</i>	68

4.30	Classe <i>PAfeModNouSer</i>	71
4.31	Classe <i>CPModSer</i>	73
4.32	Classe <i>FConnLogin</i>	74
4.33	Classe <i>FUsAlta</i>	77
4.34	Classe <i>LIdsCa</i>	79
4.35	Classe <i>LMenu</i>	81
4.36	Classe <i>LZona</i>	82
4.37	Classe <i>LTAp</i>	84
4.38	Classe <i>LAv</i>	85
4.39	Classe <i>FormAparell</i>	86
4.40	Classe <i>FAlarma</i>	88
4.41	Classe <i>FDetFum</i>	90
4.42	Classe <i>FFinestra</i>	91
4.43	Classe <i>FFogons</i>	93
4.44	Classe <i>FForn</i>	94
4.45	Classe <i>FLlum</i>	96
4.46	Classe <i>FMicroones</i>	98
4.47	Classe <i>FNevera</i>	100
4.48	Classe <i>FPC</i>	101
4.49	Classe <i>FPersiana</i>	103
4.50	Classe <i>FPorta</i>	104
4.51	Classe <i>FRentaPlats</i>	106
4.52	Classe <i>FTE</i>	107
4.53	Classe <i>FTV</i>	108
4.54	Classe <i>FTemp</i>	110
4.55	Classe <i>FVideo</i>	112
4.56	Classe <i>SN</i>	116
4.57	Classe <i>UPS</i>	117
4.58	Classe <i>Users</i>	118
4.59	Servlet <i>SeCa</i>	120
4.60	Classe <i>CSCasaMain</i>	125
4.61	Classe <i>PantPrinCasa</i>	126
4.62	Classe <i>CrearCasa</i>	127
4.63	Classe <i>CarrCasa</i>	133
4.64	Classe <i>SimCasa</i>	139
6.1	Midlet: Afegir un servidor.	147

6.2	Midlet: Consultar les dades desades del servidor afegit.	147
6.3	Midlet: A l'esquerra: seleccionar el servidor amb el que es vol contactar. A la dreta: pantalla per registrar un usuari al servidor.	148
6.4	Midlet: A l'esquerra: pantalla per iniciar sessió al servidor. A la dreta: Llista de cases enviades pel servidor, una vegada em iniciat la sessió.	148
6.5	Midlet: A l'esquerra: pantalla de selecció del criteri de cerca dels dispositius. A la dreta: Primera fase de la cerca per zona.	149
6.6	Midlet: A l'esquerra: Segona fase de la cerca per zona. A la dreta: Tercera fase de la cerca per zona.	149
6.7	Midlet: A l'esquerra: Pantalla de la primera fase de cerca per tipus d'aparell. A la dreta: Segona fase de cerca per tipus d'aparell.	150
6.8	Midlet: Pantalla on apareixen tots els aparells (criteri de cerca: Tots els aparells).	150
6.9	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Alarma</i> . A la dreta: Pantalla de control dels dispositius tipus <i>Detector de fum</i>	151
6.10	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Calefacció</i> . A la dreta: Pantalla de control dels dispositius tipus <i>Climatitzador</i>	151
6.11	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Finestra</i> . A la dreta: Pantalla de control dels dispositius tipus <i>Persiana</i>	152
6.12	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Fogons</i> . A la dreta: Pantalla de control dels dispositius tipus <i>Forn</i>	152
6.13	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Llum</i> . A la dreta: Pantalla de control dels dispositius tipus <i>PC</i>	153
6.14	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Microones</i> . A la dreta: Pantalla de control dels dispositius tipus <i>Nevera</i>	153
6.15	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Rentaplats</i> . A la dreta: Pantalla de control dels dispositius tipus <i>TermoElèctric</i> (escalfador d'aigua elèctric).	154
6.16	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>TV</i> . A la dreta: Pantalla de control dels dispositius tipus <i>Vídeo</i>	154
6.17	Midlet: A l'esquerra: Pantalla de control dels dispositius tipus <i>Porta</i> . A la dreta: Primer pas per programar el funcionament d'alguns dels dispositius programables com la <i>calefacció, climatitzador, forn, microones i vídeo</i> . Tots segueixen el mateix patró	155
6.18	Midlet: A l'esquerra: Segon pas per programar el funcionament dels dispositius programables: Introduir hora i data d'inici. A la dreta: Interfície d'usuari utilitzada per entrar la hora d'inici.	155

6.19	Midlet: A l'esquerra: Interfície d'usuari utilitzada per entrar la data (dia/mes/any) d'inici. A la dreta: Tercer pas per programar el funcionament dels dispositius programables: Introduir hora i data de finalització. Les interfícies d'usuari per entrar la data i hora de finalització són les mateixes que les d'inici.	156
6.20	Aplicació <i>CasaVirtual</i> : Pantalla Principal de l'aplicació <i>CasaVirtual</i>	156
6.21	Aplicació <i>CasaVirtual</i> : L'usuari ha executat l'opció <i>Crear...</i> del menú <i>Fitxer</i> . Primer dels cinc passos per a crear una casa domòtica.	157
6.22	Aplicació <i>CasaVirtual</i> : Segon dels cinc passos per a crear una casa domòtica. .	157
6.23	Aplicació <i>CasaVirtual</i> : Tercer dels cinc passos per a crear una casa domòtica. .	157
6.24	Aplicació <i>CasaVirtual</i> : Quart dels cinc passos per a crear una casa domòtica. .	158
6.25	Aplicació <i>CasaVirtual</i> : Cinquè i últim dels cinc passos per a crear una casa domòtica.	159
6.26	Aplicació <i>CasaVirtual</i> : L'usuari ha executat l'opció <i>Carregar Casa Virtual...</i> del menú <i>Casa</i> . Pantalla on seleccionarem la casa que vulguem carregar per a la simulació.	159
6.27	Aplicació <i>CasaVirtual</i> : Simulador de cases domòtiques.	160

ÍNDEX DE TAULES

3.1 Solucions actuals de control, sobre dispositius domòtics, des de l'exterior de l'edificació	27
4.1 Format de les peticions enviades del servlet al simulador a través de <i>sockets</i> . Aquestes peticions primer arriben al servlet via HTTP provinents del midlet.	123
4.2 Tipus de peticions enviades pel servlet al simulador a través de <i>sockets</i> . Aquestes peticions primer arriben al servlet via HTTP provinents del midlet.	123
8.1 Planificació inicial del projecte i temps real destinat	166
8.2 Segona planificació del projecte i temps real destinat	167
9.1 Personal qualificat i temps dedicat per a realitzar els diferents tipus de tasques desenvolupades al llarg del projecte. Partim de la planificació inicial	170
9.2 Sous €/h dels professionals	170
9.3 Cost en €de cada tasca partint de la planificació inicial	171
9.4 Personal qualificat i temps dedicat per a realitzar els diferents tipus de tasques desenvolupades al llarg del projecte. Partim de la segona planificació	171
9.5 Cost en €de cada tasca partint de la segona planificació	172
9.6 Hores totals destinades a cada tasca i la suma de totes.	172
9.7 Personal qualificat i Cost en €de cada tasca partint de la planificació final.	172
12.1Midlet: Tamany del paquet Casa	183
12.2Midlet: Tamany del paquet Control	184
12.3Midlet: Tamany del paquet UserPass	185
12.4Midlet: Tamany del paquet llistaobjectes	185
12.5Midlet: Tamany del paquet myRMS	185
12.6Servlet: Tamany del paquet SerialNumber	185
12.7Servlet: Tamany del paquet UserPassSerial	185
12.8Servlet: Tamany del paquet servletcasavirtual	185
12.9Aplicació <i>CasaVirtual</i> : Tamany del paquet Casa	186
12.10Aplicació <i>CasaVirtual</i> : Tamany del paquet CrearSimularCasa	186

CAPÍTOL 1. INTRODUCCIÓ

1.1. La domòtica

1.1.1. Què és la domòtica?

La domòtica és la tecnologia informàtica que converteix una edificació en "intel·ligent". Més concretament, aquest concepte es refereix a l'automatització i control de sistemes domèstics com la climatització, subministrament d'electricitat, de gas, il·luminació, electrodomèstics, portes, finestres, persianes... Prové de les paraules domus(casa en llatí) i robòtica(*robota*, esclau en txec).

La seva finalitat és la de millorar la qualitat de vida dels habitants de l'edificació on està instal·lada augmentant la seva seguretat, optimitzant la gestió del consum i reduint el treball domèstic.

En un principi està pensada per a edificis nous, però la veritat és que es pot domotitzar construccions ja existents i no únicament llars. També indústries, edificis públics o privats, hotels, col·legis, comunitats de veïns(regar correctament els jardins, il·luminació adequada de cadascuna de les plantes d'un edifici de vivendes...), etc... que rebran la denominació d'edificis in mòtics.

Els sistemes domòtics estan formats tres tipus de dispositius connectats entre si formant una xarxa d'informació: sensors, actuadors i controladors. Tot i que un únic dispositiu pot agrupar els tres, no s'acostuma a utilitzar perquè per separat sol ser més econòmic.

Els sensors s'encarreguen de recollir informació de l'entorn de l'edificació on es troben instal·lats. Aquesta, es enviada al controlador que la processa d'acord amb el seu programa de control i com a resultat generarà un seguit d'ordres dirigides a un o diferents actuadors. Per tant, podríem definir el controlador com el cervell d'un sistema domòtic centralitzat. La funció dels actuadors és la d'executar les ordres provinents del controlador, com per exemple, la d'engegar o apagar un llum.

Un exemple de comunicació entre els tres tipus de dispositius seria la següent: Suposem que és de nit i ens dirigim cap a la nostra habitació on hi tenim instal·lat un sensor de moviment. Una vegada a dins, el sensor ens detecta i emet un event al controlador. Aquest el processa d'acord amb el seu programa de control juntament amb variables com

la de que tenim la persiana de l'habitació abaixada perquè ja és de nit i per tant no ens entra llum exterior. Una vegada ha fet els seus càlculs, genera una ordre com a resposta a l'event rebut que l'envia l'actuador que controla el llum que penja del sostre de l'habitació. L'actuador la rep i l'executa: El llum s'engega.

1.1.2. Arquitectura

Es poden fer diferents classificacions de sistemes domòtics depenen des de quin punt de vista els analitzem. Una possible classificació podria ser depenent d'on resideix la seva part intel·ligent: centralitzat, distribuït o mixt (veure figura 1.1).

El sistema centralitzat està format per un únic controlador (el "cervell") centralitzat que rep tots els events generats pels diferents sensors, els processa i tot seguit envia les ordres corresponents als actuadors.

En el sistema distribuït desapareix la figura d'un únic controlador centralitzat i cada sensor o actuador en té el seu propi.

Si un sistema està format per diferents controladors i cadascun d'aquests es capaç de rebre i processar la informació de més d'un sensor i tot seguit enviar-la a altres dispositius, estarem parlant dels sistemes mixtes.

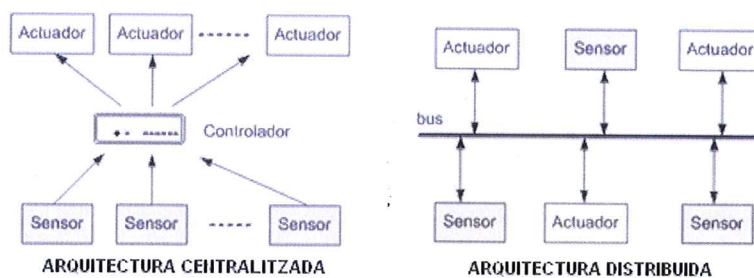


Figura 1.1: Xarxa domòtica: Tipus d'arquitectura

Una classificació diferent seria fixant-nos en el seu nivell tecnològic: sense fils, cablejats i mixtes(veure figura 1.2).

Els sense fils es caracteritzen per tenir una central de control connectada a la xarxa elèctrica(amb bateries de suport) i sensors i actuadors sense fils que es comuniquen amb la central de control per ràdio i estan alimentats per piles o bateries.

Els sistemes cablejats tenen tots els sensors i actuadors cablejats amb el controlador central. Aquest s'alimenta directament de la xarxa elèctrica. Normalment acostuma a

incloure unes bateries per poder seguir alimentant els sensors, actuadors i a ell mateix, en el cas de caiguda de la xarxa elèctrica, durant el període de temps necessari per a que tot torni a la normalitat i així poder seguir funcionant.

Els sistemes mixtes s'obtenen combinant l'ús de cablejat i de tecnologia sense fils.

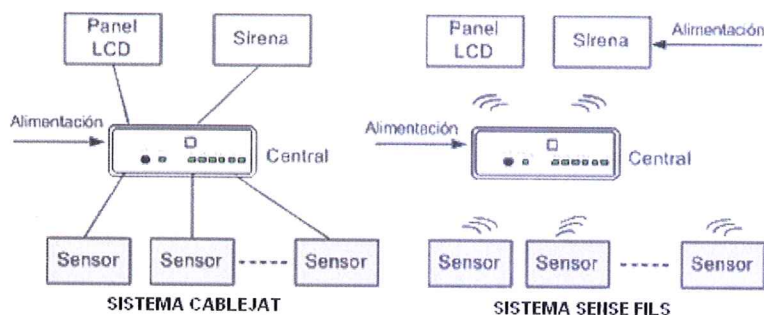


Figura 1.2: Xarxa domòtica: Tipus de nivell tecnològic

1.1.3. Solucions tecnològiques

Existeixen un gran nombre de solucions tecnològiques per a les xarxes domòtiques. Actualment, els estàndards més importants són el KNX, X10 i LonWorks.

1.1.3.1. X10

La tecnologia X10 va ser desenvolupada a finals dels anys setanta i es va començar a comercialitzar en aquests anys també. És basada en la utilització de la xarxa elèctrica ja existent a la casa com a mitjà de comunicació entre els diferents components de la xarxa domòtica. Els senyals emesos són molt simples i permeten la transmissió dels events ON, OFF i DIM que captaran els diferents elements connectats a la xarxa elèctrica. L'últim dels events anomenats, permet regular la intensitat d'il·luminació dels llums.

Actualment és la tecnologia més estesa ja que és barata i la seva instal·lació i configuració és senzilla. Per a la instal·lació s'hauria d'afegir receptors X10 per cada element que es vulgui controlar, canviar els polsadors instal·lats prèviament per polsadors X10 i col·locar un nombre de controladors X10 necessaris per a poder controlar tots els elements desitjats.

La configuració dels diferents dispositius es fa de forma manual. Cadascun d'ells s'identificarà per una lletra de la "A" a la "P", que rep el nom de codi de casa, més un número

del 1 al 16, anomenat codi d'unitat. Amb aquest sistema d'identificació es poden arribar a un màxim de 256 identificacions diferents. Això no implica que es puguin connectar únicament 256 dispositius. Per exemple, podríem utilitzar la mateixa direcció per a dos làmpades que tenim en una mateixa habitació. D'aquesta manera podem controlar un dispositiu més, però executaran simultàniament les ordres destinades a l'adreça que comparteixen. Per tant, perdríem el control individual de cadascun d'ells.

X10 és especialment indicada per a cases antigues ja que evita tenir de fer reformes, encara que lògicament també es pot utilitzar en noves.

La seva baixa velocitat de transmissió (només 50bits/segon) i a la seva poca funcionalitat (engegar, parar i "dimmer") són els seus principals desavantatges respecte KNX i LonWorks.

1.1.3.2. KNX

De la convergència dels estàndards EIB, BatiBus i EHS, en va néixer KNX(Konnex).

Pot utilitzar diferents medis físics de comunicació: parell trenat, xarxa elèctrica, cablejat ethernet i radiofreqüència.

És un sistema molt més robust i fiable que el X10.

KNX proporciona tres tipus de configuració: Mode-S ("Mode Sistema", els diferents dispositius de la xarxa son instal·lats i configurats per professionals amb l'ajuda d'una aplicació software), Mode-E ("Mode Easy", els dispositius venen programats de fàbrica, encara que petits detalls hauran de ser configurats durant la instal·lació) i el Mode-A ("Mode Automàtic", ni l'instal·lador ni el client s'hauran de preocupar de la configuració dels diferents elements ja que és basen en la filosofia *Plug&Play*¹).

Actualment encara s'acostuma a anomenar sistema EIB i s'utilitzaran els dos noms, EIB i KNX, durant un cert període de temps.

¹Tecnologia que permet connectar un dispositiu informàtic amb un ordinador sense configurar jumpers ni proporcionar paràmetres als seus controladors. L'ordinador té de funcionar amb un sistema operatiu amb suport a aquell dispositiu per tal que tot això sigui possible.

1.1.3.3. LonWorks

Pertany a la companyia Echelon Corp. És una tecnologia molt robusta i fiable. Pot utilitzar la xarxa elèctrica, cable coaxial, fibra òptica, parell trenat i l'aire com a mitjà de transmissió. Pot arribar als 2,5Mb/s de velocitat de transmissió i controlar fins a 32.000 dispositius.

El seu elevat cost econòmic ha causat que la seva implementació només hagi tingut acceptació en grans edificis i hotels.

1.2. Objectius del projecte

Aquest projecte parteix amb l'objectiu de crear un conjunt d'aplicacions informàtiques per tal de poder controlar remotament els diferents dispositius d'una xarxa domòtica, via Internet, mitjançant el dispositiu mòbil més utilitzat actualment: el telèfon mòbil.

Per tal de poder assolir aquest objectiu em pensat en dissenyar i desenvolupar un paquet de tres aplicacions. La primera ens simularà una casa domòtica des d'on visualitzarem l'estat dels dispositius que hi tinguem instal·lats. La segona aplicació(servlet) s'executarà en un ordinador personal que tindrem instal·lat a la casa a controlar. Aquest ordinador al que anomenarem servidor, tindrà accés a Internet i, per altre banda, també es podrà comunicar amb tots els dispositius. La tercera i última aplicació(MIDlet) estarà destinada per a funcionar en un telèfon mòbil. Amb ella podrem veure i modificar l'estat dels aparells de la casa.

Les comunicacions entre la casa i el telèfon mòbil es realitzaran a través d'Internet. Des del mòbil llançarem les ordres que les rebrà el nostre servidor i aquest les enviarà al simulador de la casa domòtica.

1.3. Breu descripció del paquet d'aplicacions desenvolupades

El paquet d'aplicacions desenvolupades per a assolir els objectius del projecte consta de 3 aplicacions: L'aplicació creadora i simuladora de la casa domòtica, el servlet i el MIDlet.

1. Aplicació creadora i simuladora d'una llar: Estarà instal·lada al servidor juntament amb el servlet. Ens permet crear el nombre que vulguem de llars domòtiques que després podrem monitoritzar, individualment, per conèixer l'estat dels seus aparells. Per a la seva confecció disposarem d'un total de 17 tipus de dispositiu, que més tard podrem controlar des de l'aplicació del telèfon mòbil. Els tipus de dispositius disponibles són alarma, detector de fum, calefacció, climatització, fogons, forn, microones, rentaplats, nevera, escalfador d'aigua elèctric, PC, llum, persiana, finestra, porta, TV i vídeo.

La casa estarà dividida en plantes i aquestes en diferents zones que podríem anomenar habitacions, balcons... Cadascuna d'aquestes zones pot contenir cap o varis dispositius dels 17 tipus que acabem d'anomenar.

Una vegada tinguem la llar creada, ja podrem monitoritzar l'estat dels diferents dispositius que contingui per mitjà del simulador.

2. Un servlet: Instal·lat al servidor, serà el pont que farà possible la comunicació entre el MIDlet i la casa virtual. El servlet estarà preparat per a rebre qualsevol petició que l'hi arribi a través d'Internet des del MIDlet i prolongar-la fins al simulador de la casa domòtica i viceversa. El servlet també té la funció de donar d'alta i autenticar els clients que vulguin establir comunicació amb la llar.

En un principi es va pensar que també realitzés la funció de simulador, però al final per tal d'apropar-nos una mica més al cas real, vam decidir que aquesta funció la portés a terme l'aplicació.

3. Un MIDlet: Aplicació instal·lada al telèfon mòbil que ens permetrà veure i modificar l'estat dels diferents dispositius de la casa que s'estigui monitoritzant des del simulador de la llar virtual. El MIDlet establirà comunicació amb el servlet que farà arribar les seves peticions al simulador. Aquest, una vegada hagi tractat la petició en qüestió, enviarà una contestació, la qual, el servlet serà l'encarregat d'enviar-la al MIDlet.

A partir d'un número de sèrie proporcionat pel proveïdor del paquet d'aplicacions, l'usuari que intenta connectar-se amb la llar virtual, es podrà donar d'alta al servidor (una de les tasques del servlet) i una vegada fet això, amb el seu nom d'usuari i contrasenya escollida, podrà accedir a la casa.

CAPÍTOL 2. TECNOLOGIA UTILITZADA

2.1. Tecnologia JAVA

2.1.1. Introducció

Totes les aplicacions desenvolupades per assolir els objectius d'aquest projecte han estat desenvolupades amb la tecnologia JAVA. Aquesta no només és un llenguatge de programació, sinó que és una plataforma amb una gran biblioteca, gran quantitat de codi reutilitzable i un entorn d'execució d'alta qualitat que ofereix serveis com seguretat, adaptabilitat entre sistemes i robustesa.

La seguretat en l'execució de les aplicacions ens la proporciona el fet de que es porta a terme dins de l'entorn establert per la màquina virtual Java. D'aquesta manera, si l'aplicació és penja, només provocarà que també és pengi la màquina virtual i no pas el dispositiu tal i com passa amb les aplicacions binaries.

Java és robust respecte al consum de recursos, alliberació d'objectes, recollidor de deixalles...

L'adaptabilitat entre sistemes o portabilitat és una de les seves característiques més importants. Ens fa possible crear una aplicació i executar-la en diferents dispositius. Com que el codi JAVA s'executa dins l'entorn de la màquina virtual, en un principi, l'aplicació s'hauria d'executar sense problemes.

Java va aparèixer el 1991, a mans d'un grup d'enginyers de Sun. En un principi va ser dissenyat per a la creació de petites aplicacions destinades a equips de consum, però de mica en mica ha anat migrant cap al desenvolupament de grans aplicacions per a servidors.

Sun va publicar la primera versió de Java(Java 1.0) el 1996. Més tard, va aparèixer la Java 1.1 i el 1998 va publicar-se Java 1.2 que ràpidament va ser canviada de nom a "Java 2 Standard Edition Software Development Kit Version 1.2". A part de la "Standard Edition", també es van presentar dues edicions més: la "Enterprise Edition" i la "Micro Edition". Cadascuna d'elles té el seu propi conjunt d'APIs i d'eines de desenvolupament.

1. J2EE(Java 2 Enterprise Edition): Aquesta edició està enfocada al desenvolupament

d'aplicacions JAVA per a servidors. De les tres branques, és la més orientada al desenvolupament de components i distribució d'aplicacions.

2. J2SE(Java 2 Standard Edition): És un subconjunt de J2EE i va dirigida a la programació d'aplicacions JAVA per a ordinadors personals(PCs).
3. J2ME(Java 2 Micro Edition): J2ME és un subconjunt de J2SE. El seu ús recau en el desenvolupament d'aplicacions per a dispositius amb pocs recursos, amb limitacions de la pantalla gràfica, de capacitat de processament i de memòria. Els telèfons mòbils, PDAs, dispositius de navegació dels cotxes... formen part d'aquest grup de dispositius. Amb l'aparició de J2ME, la plataforma JAVA torna als seus orígens.

Més tard han aparegut les versions 1.3, 1.4 i 1.5 de la "Standard Edition". La versió més actual va presentar-se a principis de desembre del 2006 sota el nom de Java SE 6.

2.1.2. J2ME

"Java 2 Micro Edition", al ser un subgrup de J2SE, hi manté una compatibilitat cap amunt. Per tant, una aplicació escrita en J2ME funcionarà amb la plataforma J2SE i fins i tot amb la J2EE(assumint que les classes utilitzades estan disponibles en cada edició).

Aquesta plataforma és un conjunt d'especificacions que defineixen varies versions minimitzades de la plataforma Java 2. Amb la seva aparició, es podria dir que Java torna als seus orígens, ja que com em dit abans, en un principi, va ser dissenyada per a crear petites aplicacions destinades a equips de consum.

Si mirem els avantatges respecte a altres tecnologies, les aplicacions que utilitzen J2ME poden arribar a ser molt més eficients i ràpides. Això ho provoca la necessitat de no tenir d'estar sempre connectat ni consultar la informació que ja s'hagi consultat anteriorment ja que permet que sigui emmagatzemada a la memòria local. Un altre avantatge prové del fet de que les aplicacions són petites i es poden descarregar utilitzant qualsevol tipus de protocol com per exemple el WAP¹ o el TCP/IP. Això facilita la difusió y distribució de les aplicacions. Aquestes també s'avantatgen respecte a les realitzades en entorns basats per exemple en WAP, tant en capacitat gràfica com en funcionalitat proporcionats pel propi llenguatge.

¹ *Wireless Application Protocol*: Protocol que pretén oferir als usuaris de dispositius limitats de capacitat de processament, resolució de pantalla i memòria, les mateixes funcionalitats que s'ofereixen per Internet a través d'ordinadors personals, però amb els avantatges i limitacions pròpies d'aquests dispositius.

La tecnologia WAP no ha tingut la repercussió prevista principalment per dues raons que J2ME soluciona. Per una banda, la necessitat de connexió permanent a la xarxa, que perjudica massa l'economia dels usuaris finals i, per l'altre, al ús excessiu de la web. Això ha provocat que J2ME s'estigui afermant en el terreny de les aplicacions destinades a dispositius mòbils. Tot i això, J2ME no està pensat per substituir a les aplicacions basades en WAP sinó més com una tecnologia complementària que proporciona la capacitat de navegació dels llocs WAP i, a més a més, una gran funcionalitat a l'entorn d'execució de les aplicacions.

La tecnologia J2ME va dirigida a una gran diversitat de dispositius, és molt jove i en constant desenvolupament. Això esdevé un problema considerable si ens referim a la responsabilitat de que les aplicacions creades siguin portables². Conté especificacions encara inacabades i molts fabricants s'han creat les seves pròpies implementacions que només aprofiten els seus propis productes. Existeix un estàndard format per les versions finals de les especificacions que actualment s'està definint. S'està a l'espera que els diferents fabricants s'hi acabin adherint.

2.1.2.1. *Detalls tècnics*

Java 2 Micro Edition es divideix en configuracions, perfils i paquets opcionals. No es poden crear aplicacions en aquesta plataforma sense conèixer la configuració i el perfil dels dispositius a les que van destinades.

Les configuracions estan formades per tres elements:

- Una màquina virtual Java per executar el bytecode de l'aplicació.
- Un conjunt base de classes Java que poden ser utilitzades en certa classe de dispositius.
- Codi nadiu per fer de pont entre Java i el sistema operatiu utilitzat en el dispositiu.

Actualment, J2ME té disponibles dos configuracions: CLDC i CDC.

- CLDC(Connected Limited Device Configuration): Dissenyada per a dispositius petits amb connexions de xarxa intermitents, com telèfons mòbils i PDAs. La seva

²La capacitat de crear una aplicació i executar-la sobre diferents sistemes operatius

màquina virtual s'anomena KVM(Kilo Virtual Machine). És una implementació completa de una màquina virtual Java optimitzada per ser utilitzada en dispositius amb molts pocs recursos.

- CDC(Connected Device Configuration): Enfocada per aparells amb més memòria, capacitat de processament i connexions de xarxa robustes. La seva màquina virtual s'anomena CVM(Compact Virtual Machine)

Un perfil descriu el tipus de dispositiu suportat, com per exemple, el MIDP(Perfil per a Dispositius de Informació Mòbil). Treballa sobre una configuració, però afegeix classes específiques per fer un entorn complet de construcció d'aplicacions. Per exemple, les classes que permeten la implementació de interfícies d'usuari estan definides al perfil. El perfil més utilitzat és el MIDP, que està basat en la configuració CLDC per executar aplicacions en telèfons mòbils i cercapersones, dispositius amb pantalles petites, connexió HTTP sense fils i memòria limitada. Inclou classes de la interfície d'usuari, de cicle de vida de les aplicacions i algunes d'emmagatzemament persistent.

Els paquets opcionals proporcionen funcionalitats que poden estar o no relacionades amb alguna configuració o perfil.

2.1.3. Què és un Servlet?

Es podria definir un servlet com a un programa JAVA que s'executa en un servidor web o contenidor de servlets i que rep i respon a les peticions d'un client. El contenidor de servlets que s'ha utilitzat en aquest projecte és l'*Apache Tomcat*.

Té un cicle de vida definit. La interfície que s'utilitza per implementar-los, defineix diferents mètodes per a cada una de les seves etapes i és el seu entorn qui s'encarrega d'invocar-los (veure figura 2.1):

- Quan el servidor rep la primera petició d'un client, carrega el servlet invocant el mètode *init*. Una vegada fet això, ja no es tornarà a cridar aquest mètode.
- Quan la càrrega ha finalitzat, totes les peticions seran processades pel mètode *service*.
- Quan el servidor vulgui descarregar el servlet, executarà el mètode *destroy*.

La API utilitzada per a la seva programació conté dos paquets:

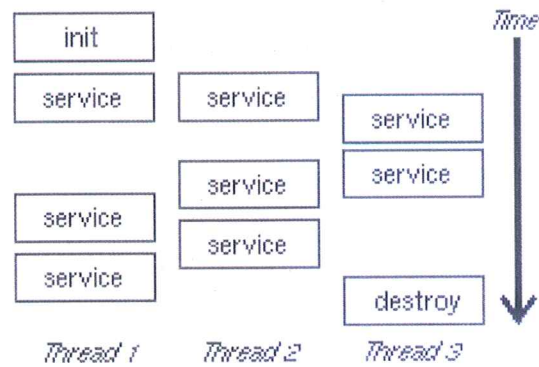


Figura 2.1: Típic cicle de vida d'un servlet.

- *javax.servlet*: Defineix la interfície *servlet* i la seva implementació. En aquesta interfície es defineixen tots els mètodes de cadascuna de les etapes del seu cicle de vida. El paquet també defineix la forma en que es tractaran les peticions i generaran les respostes mitjançant les interfícies *ServletResponse* i *ServletRequest*.
- *javax.servlet.http*: És una especialització del paquet *javax.servlet*. S'utilitza pel protocol HTTP. Defineix la classe *HttpServlet* i inclou les interfícies *HttpServletRequest* i *HttpServletResponse*, que són extensions de la *ServletRequest* i *ServletResponse*.

Els servlets formen part de l'edició J2EE de la plataforma JAVA.

CAPÍTOL 3. SOLUCIONS ACTUALS

Les solucions de control sobre els dispositius domòtics que ofereixen les empreses dedicades a aquest camp, es divideixen en dos tipus. Una seria el control des de l'interior de la llar i l'altre des de l'exterior. Els sistemes de control domòtic més avançats inclouen els dos tipus.

Per portar a terme el control desde l'interior, els dispositius més utilitzats són des de pulsadors empotrats a la paret i comandaments a distància, fins a, pantalles tàctils o PCs.

Els més emprats per al control desde l'exterior de l'edificació són el PC, PocketPC, telèfon, telèfon mòbil i PDA. Segons les ofertes actuals, per aquest tipus de control, és solen utilitzar una o varies vies de comunicació: WEB, WAP, SMS o RTC¹(veure taula 3.1).

Vía de Comunicació	Dispositiu
Connexió convencional a Internet	PC, Pocket PC
WAP	PDA, telèfon mòbil
SMS	Telèfon mòbil
RTC	Telèfon convencional/mòbil

Taula 3.1: Solucions actuals de control, sobre dispositius domòtics, des de l'exterior de l'edificació

Un exemple de sistema de control per a una xarxa domòtica podria ser:

- Control desde l'interior a través de pulsadors de diferents elements, pantalla tàctil o PC.
- Control remot mitjançant tecnologia WEB: PC/Pocket PC.

¹Reconeixement de veu/codis introduïts des del teclat del telèfon

