



471 Desenvolupament d'aplicacions per a dispositius mòbils

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica

realitzat per:

Sergi Torralba i Santamaria

i dirigit per:

Jordi Ceballos

Bellaterra, 16 de Setembre del 2008

471 Desenvolupament d'aplicacions per a dispositius mòbils

La finalitat d'aquest projecte és obtenir una visió global de la programació per dispositius mòbils i a més fer una avaluació del que podria costar (tant les eines com el reciclatge de programadors) a una empresa dedicada a TIC començar a implementar aplicacions per a dispositius mòbils. Amb un plantejament molt ampli al principi i reduït només a tres opcions (BlackBerry, Windows Mobile i Android), es desenvolupa una aplicació per a aquests dispositius. Els resultats obtinguts mostren que a nivell econòmic les millors opcions són el desenvolupament per Android (gratuït) i per BlackBerry (20\$). Però en canvi a nivell d'adaptació dels llenguatges de programació la opció de Windows Mobile és la que presenta menys obstacles per passar de coneixements de programació d'aplicacions de servidor o sobre taula (en entorn .NET) a dispositius mòbils.

El objetivo de este proyecto es tener una visión global de la programación para dispositivos móviles y realizar una evaluación del coste (tanto de las herramientas como del reciclaje de programadores) que tendría desarrollar este tipo de aplicaciones para una empresa dedicada a las TIC. El planteamiento amplio inicialmente se tuvo que reducir a tres opciones (BlackBerry, Windows Mobile y Android), para las cuales se ha desarrollado una aplicación. Los resultados obtenidos nos muestran que económicamente las mejores opciones son el desarrollo para Android (gratis) y para BlackBerry (20\$). Pero en cambio a nivel de adaptación de programadores de entornos de escritorio o servidores (en entorno .NET) a dispositivos móviles, Windows Mobile es el que presenta menos dificultades.

The aim of this project is to obtain a global vision about programming oriented to mobile devices and to evaluate the different costs –the tool costs and the difficulties of adapting programmers from Desktop programs to mobile devices applications – for a company dedicated to IT. The broad initial objectives had to come down to three options (BlackBerry, Windows Mobile and Android), and we have developed an application for each. We show that the cheapest options are developing for Android (free) and BlackBerry (20\$). However the easiest one to adapt programmers who have experience programming desktop or server applications (in .NET environment) is Windows Mobile.

INDEX

1	Introducció.....	3
1.1	El projecte	3
1.2	Motivacions	5
1.3	Requeriments.....	5
1.4	Planificació	7
2	Diferents tecnologies per dispositius mòbils.....	9
2.1	MIDlets	9
2.1.1	Característiques	10
2.2	Symbian	11
2.2.1	Característiques	11
2.3	Apple iPhone.....	13
2.4	BlackBerry.....	14
2.4.1	Característiques	14
2.5	Windows Mobile.....	16
2.5.1	Característiques	18
2.5.2	Tecnologies .NET.....	18
2.6	<i>Android</i>	21
2.6.1	Característiques	22
3	Els entorns de programació.....	27
3.1	BlackBerry JDE 4.5.0.....	27
3.2	Visual Studio 2008.....	30
3.3	<i>Android (ECLIPSE)</i>	34
4	Descripció de l'aplicació.....	39
4.1	Requeriments funcionals	39
4.2	Requisits no funcionals.....	40
4.3	Diagrama casos d'us	41

4.4	Explicació del casos d'us.....	42
4.5	Comunicacions	50
5	El Comunicador.....	53
5.1	Dispositius mòbils	53
5.2	L'aplicació de sobretaula	60
6	Adaptació a les diferents tecnologies i avaluació econòmica.....	65
6.1	Adaptació a les diferents tecnologies.....	65
6.2	Avaluació econòmica.....	66
7	Conclusions i ampliacions.....	69
7.1	Conclusions.....	69
7.2	Ampliacions.....	70
	Annex 1: Índex de taules i figures.....	73
	Bibliografia i Referències.....	75
	Bibliografia i adreces web per el desenvolupament.....	75
	Referències.....	75

1 Introducció

En els últims mesos hi ha hagut un augment considerable de dispositius mòbils i tecnologies afins, ja sigui pel novíssim *Apple iPhone* o pel sistema operatiu per mòbils *Google Android*. Per tant, creiem que és un bon moment per intentar avançar en aquest camp. L'objectiu del projecte és doncs comparar les diferents opcions de dispositius mòbils que hi ha actualment al mercat i crear una petita aplicació de mostra que ens serveixi per comparar i avaluar tant les tecnologies de cada plataforma com del seus entorns de programació. A més a més, aquesta aplicació exemple ens permetrà realitzar ofertes a futurs clients en l'entorn d'aquestes tecnologies.

1.1 El projecte

Cada cop és més habitual que tant les empreses com els usuaris particulars disposin de terminals mòbils amb connexió a Internet, els quals també són capaços de fer córrer aplicacions que no formin part del *firmware* del telèfon. Tot i que globalment Nokia és la marca més present en dispositius mòbils a nivell empresarial, hi ha altres marques que tenen una gran acceptació, com ara el conegut *BlackBerry* (dispositius creats per l'empresa RIM). També hi ha moltes empreses que posen a disposició dels seus empleats aparells que no són només telèfons sinó també són ordinadors de butxaca.

Worldwide converged smart mobile device market					
Market shares Q4 2007, Q4 2006					
Vendor	Q4 2007 shipments	% share	Q4 2006 shipments	% share	Growth Q4'07/Q4'06
Total	35,522,360	100.0%	20,667,200	100.0%	71.9%
Nokia	18,802,480	52.9%	11,114,630	53.8%	69.2%
RIM	4,046,860	11.4%	1,829,260	8.9%	121.2%
Apple	2,320,840	6.5%	-	0.0%	NA
Motorola	2,301,260	6.5%	1,463,090	7.1%	57.3%
Others	8,050,920	22.7%	6,260,220	30.3%	28.6%

Source: Canalys estimates, © canalys.com ltd. 2008
Converged smart mobile device market: smart phones and wireless handhelds

Taula 1 Nombre d'unitats de telèfons mòbils de les diferents marquesⁱ

Per aquesta raó s'ha plantejat aquest projecte a nivell empresarial en comptes de a nivell personal.

Cal tenir també en compte que les tarifes de connexió a Internet són cada vegada més assequiblesⁱⁱ. Per tant, és el moment idoni per oferir aplicacions que s'aprofitin d'aquesta situació i creiem que aquestes aplicacions seran ben vistes pels nostres clients, els quals possiblement estan desaprofitant el potencial dels terminals que ja posseeixen.

Amb aquest projecte no s'ha intentat aconseguir un programa complet i totalment funcional, sinó fer una aplicació que mostri el potencial de les diferents plataformes actuals més comunes i veure quina plataforma ens pot ser més profitosa a l'hora de la creació de programes.

Així doncs, el projecte es divideix en dos grans blocs:

- Investigació de les tecnologies, història i evolució tant de plataformes més aviat clàssiques com poden ser *Symbian* i *BlackBerry*, fins les més noves en temes de telèfons mòbils com poden ser *Windows Mobile 6* i *iPhone*. També tractarem la plataforma *Android* ja que, al estar estretament relacionada amb *Google*, creiem que en un futur proper s'haurà de tenir molt en compte. En aquest punt s'ha de remarcar dos problemes, que tota al feina feta per l'*iPhone* es va haver de desestimar ja que un cop es va fer públic el kit de desenvolupament de programari (SDK) un dels requeriments era fer servir el programari i el maquinari de MACⁱⁱⁱ del qual no disposàvem, i que la feina feta per *Symbian* també es va haver de desestimar per la inestabilitat de l'entorn de programació.
- Creació d'una aplicació a mode d'exemple per intentar mostrar el potencial dels dispositius mòbils i els seus entorns de desenvolupament. A l'exemple es planteja un programa per a una empresa que té treballadors desplaçats (com per exemple tècnics) i vol tenir un petit control de la seva posició des de l'oficina.

1.2 Motivacions

Hi ha varies motivacions en aquest projecte. Primer de tot personal. Tot i que a la feina estic sempre desenvolupant aplicacions per servidors i sistemes de sobretaula o similars, m'agrada molt la tecnologia i, en concret, els dispositius mòbils. Potser la raó d'aquesta afinitat son les llargues hores en transport públic, on acostumo a portar diversos aparells a sobre com un *PocketPC* i diferents reproductors, a més del telèfon mòbil, que a mesura que han passat els anys els diferents telèfons guanyaven funcionalitats, posant com excusa per canviar de dispositiu que algun dia em faria algun petit programa.

També hi ha motivacions més aviat comercials. Davant l'augment de dispositius mòbils que són més que un simple telèfon, els quals permeten connexió a Internet, comunicacions WIFI, etc., juntament amb uns entorns gràfics que permeten crear aplicacions més atractives, s'obre un mercat cara a obtenir nous projectes. Sobretot en aquest moment, on la situació econòmica no és la millor pel desenvolupament, es pot oferir al client la possibilitat d'aprofitar els dispositius que ja tenen pagats i que la majoria de vegades estan infrautilitzats.

Així doncs, l'objectiu d'aquest projecte consisteix en l'avaluació de diferents tecnologies i entorns de programació dels sistemes que hi ha actualment disponibles, o que hi haurà en un futur proper, per veure cap a quina o quines ens podem encarar a l'hora de definir un objectiu dins de nous projectes. D'aquesta manera, també s'aconsegueix aprendre a utilitzar les eines més importants disponibles, tant a nivell professional com comercial, i adquirir un bon nivell de disseny i programació en aquest tipus d'aplicacions.

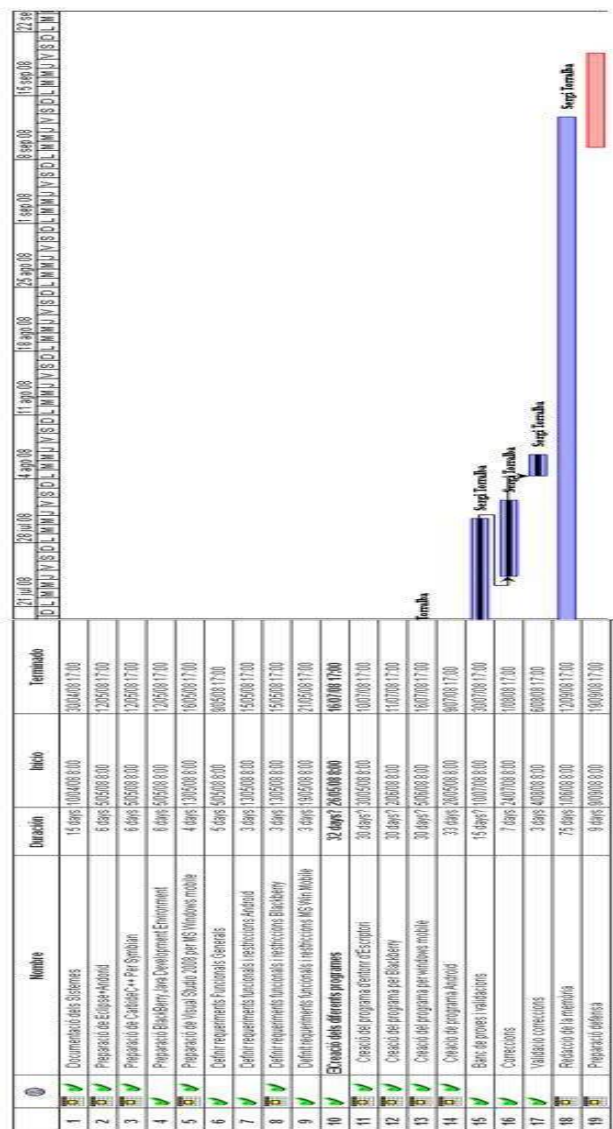
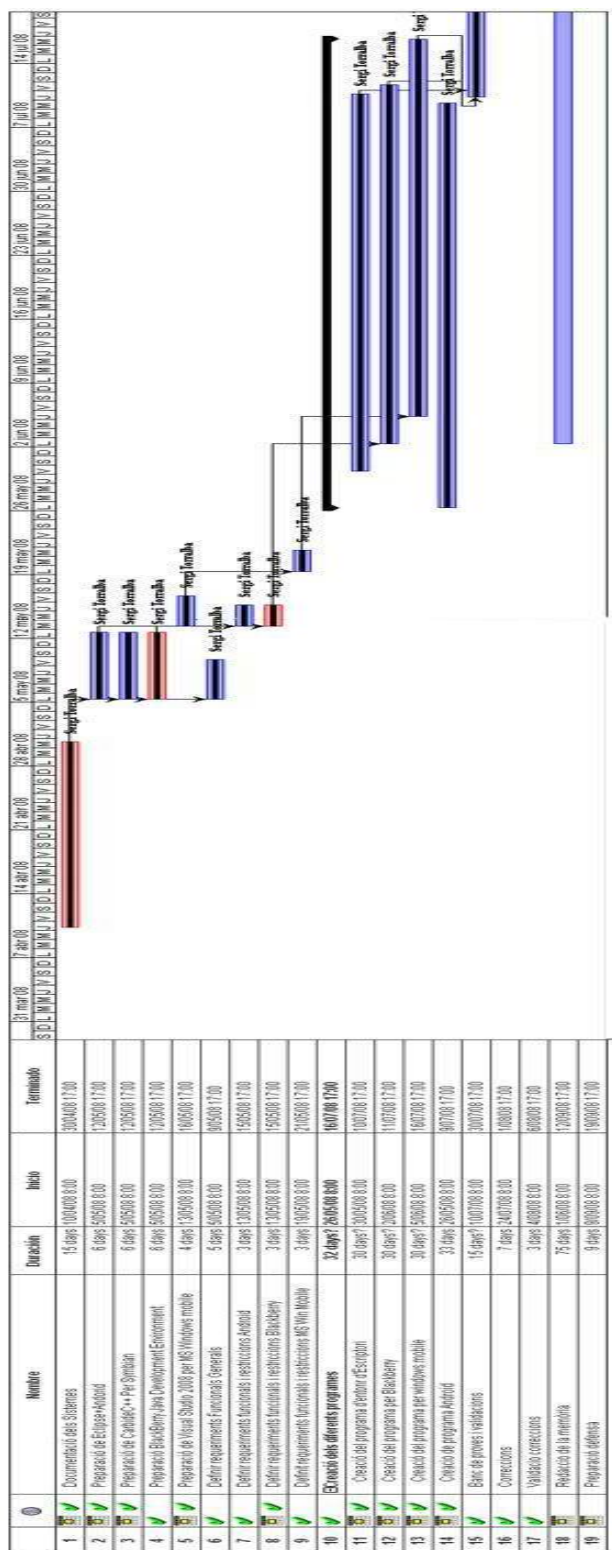
1.3 Requeriments

Pel desenvolupament de les aplicacions cal un ordinador amb *Windows* (les eines oficials de desenvolupament tant de *BlackBerry* com de *Windows Mobile* només estan disponibles per *Windows XP* o *Windows Vista*). Aprofitant les tecnologies actuals de virtualització es farà el desenvolupament en una màquina virtual amb el *Windows XP* com a sistema operatiu, facilitant les proves amb els simuladors ja que, d'aquesta manera, es poden realitzar proves entre dos ordinadors quan realment només es té una màquina física.

Les eines de desenvolupament tant per *BlackBerry* com per *Android* són gratuïtes i es poden trobar per Internet. En canvi, per desenvolupar aplicacions per dispositius mòbils de les plataformes de *Microsoft Windows Mobile* no serveix cap de les versions gratuïtes disponibles de *Visual Studio* (versions Express)^{iv}.

Com a coneixements inicials són necessaris coneixements de programació orientada a objectes, ja que els llenguatges que cal fer servir són Java i C# (aquest darrer no és obligatori, ja que la plataforma de desenvolupament .NET permet programar en diferents llenguatges sense modificar-ne el resultat, 2.5.2). També és recomanable tenir coneixements de comunicacions i temes web.

1.4 Planificació



2 Diferents tecnologies per dispositius mòbils

En aquesta secció es descriuen les eines per les diferents plataformes que s'han analitzat per poder tenir una base teòrica amb la qual poder prendre les decisions inicials i començar a desenvolupar l'aplicació principal del projecte. També es valorarà si ens interessa fer servir cadascuna d'aquestes tecnologies o si és preferible deixar-la de banda per aconseguir els nostres objectius.

2.1 MIDlets

Mobile Information Device Profile (MIDP) és una especificació publicada per l'ús de Java en dispositius mòbils i PDAs.

Els MIDlets són programes independents de l'arquitectura, tot i presentar certs requeriments Java Micro Edition, Connected Limited Device Configuration i Mobile Information Device Profile, i són el sistema més estès. Les principals raons consisteixen en la seva simplicitat arquitectònica i que depenen únicament del software. En aquest moment són el sistema utilitzat per als jocs per mòbils i petits programes. Com molts altres programes Java, els MIDlets tenen la característica que s'escriuen una vegada i es poden executar arreu. És a dir, un MIDlet programat per a un telèfon *BlackBerry* es podrà executar en un telèfon *Nokia*. Aquest avantatge és també un inconvenient. Degut al caràcter genèric d'aquest tipus de programes, no es pot arribar al màxim de les possibilitats de cada aparell o tecnologia.

A nivell de programació, els programes per *BlackBerry* són MIDlets. Tot i que presenten classes pròpies que ajuden i donen accés a elements propis de l'arquitectura, s'ha optat per no fer un desenvolupament de cap aplicació MIDlet. Malgrat tot, el programa realitzat per *BlackBerry* no funcionarà en altres dispositius, ja que s'utilitzen llibreries pròpies de *BlackBerry*.

2.1.1 Característiques

Els MIDlets han de complir els següents requeriments per poder ser executats en un telèfon mòbil:

- La classe principal ha de ser una subclasse de *javax.microedition.midlet.MIDlet*
- Han de ser empaquetats dins d'un arxiu *.jar*
- L'arxiu *.jar* ha de ser pre-verificat fent servir un pre-verificador.
- En alguns casos, l'arxiu *.jar* ha de ser signat per l'operador de telefonia mòbil al qual estigui associat el dispositiu.

2.2 Symbian

symbian OS

Symbian és un sistema operatiu per a dispositius mòbils, el qual disposa de llibreries pròpies i interfície d'usuari. Aquest sistema operatiu està produït per *Symbian Ltd.* i només funciona en processadors ARM. *Symbian* no és exclusiu d'una companyia de telefonia mòbil, sinó que pertany a diferents marques.

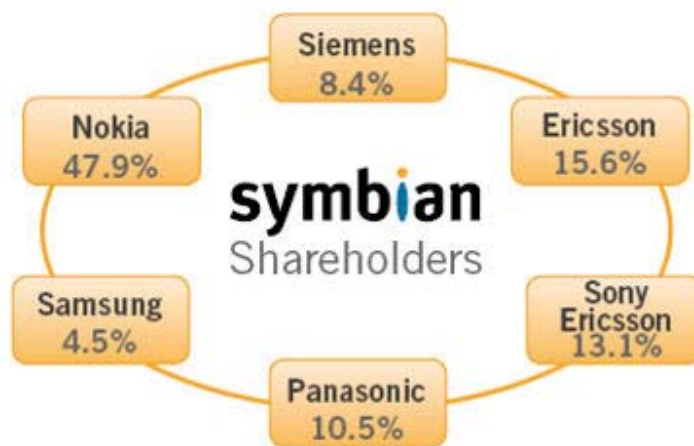


Figura 1 Percentatges de possessió de *Symbian* (Font www.symbian.com)

A nivell europeu, *Nokia* és qui més ha apostat per aquest sistema operatiu. En aquests moments, la companyia disposa d'uns quaranta models en el mercat que utilitzen el sistema *Symbian*. Per contra, *Sony Ericsson* només té nou models al mercat que disposin del sistema tot i ser un dels propietaris de *Symbian*^{vi}.

2.2.1 Característiques

Tot i que el sistema operatiu *Symbian* té com a llenguatge natiu el C++, també suporta el llenguatge *Java* (concretament *JavaME*). Tot i un acurat disseny per intentar millorar el rendiment de les aplicacions *Java*, les aplicacions programades en C++ dins d'aquest sistema són més eficients ja que el codi resultant de la compilació és codi assemblador

ARM.^{vii} *Nokia* recomana l'ús de l'eina de desenvolupament *Carbide.c++* per a la programació de programes per als seus dispositius.^{viii}

Information	Description and Link	EXP	DEV	PRO	OEM
Tutorials (Screencasts)	Getting Started with Carbide.c++	*	*	*	*
	On-Device Debugging		*	*	*
	Performance Investigator			*	*
	Getting Ready For Symbian Signed with Carbide.c++	*	*	*	*
White Papers	Carbide.c++ Introduction	*	*	*	*
	Switching to Carbide.c++	*	*	*	*
	On-Device Debugging		*	*	*
	Carbide.c++ UI Designer		*	*	*
	Carbide.c++ Performance Investigator			*	*
Other	Carbide.c++ Datasheet	*	*	*	*
	Carbide.c++ FAQ	*	*	*	*
	E-store purchase and upgrading		*	*	*
	Developer Edition Screen shots		*	*	*
	Professional Edition Screen shots			*	*
Pricing ¹	Go to the e-Store	Free	299€	1299€	3999€

1) Node-locked license. Floating licenses are double the price per license. Floating licenses are not available on the Developer Edition.

Taula 2 Especificacions dels diferents models de *Carbide.c++*

Tot i que l'eina de desenvolupament *Carbide.c++* és gratuïta en la versió *Express*, ofereix moltes menys opcions que les eines proporcionades tant per *BlackBerry* com per *Windows Mobile*. *Android* no es pot incorporar en aquesta comparació ja que encara no hi ha cap dispositiu real que funcioni amb aquest sistema operatiu. Cal remarcar que Microsoft Visual Studio 2008 és gratuït per a estudiants, però sinó té un preu força elevat (6.2).

El desenvolupament d'aplicacions per a *Nokia/Symbian* fent servir l'entorn de programació *Carbide.c++* es va desestimar per diverses raons. Primerament, l'entorn de programació és difícil de configurar. A més a més, la seva estabilitat és reduïda; errors d'execució de l'entorn de programació, poca documentació d'instal·lació, etc. Per aquestes raons, i després de discutir-ho amb el cap de projecte, aquesta opció va quedar descartada.

2.3 Apple iPhone



En el plantejament inicial del projecte estava inclòs el dispositiu *Apple iPhone*, just abans de l'aparició al mercat d'aquest telèfon mòbil. *Apple* havia comunicat que alliberaria el kit de desenvolupament de programari per aquest dispositiu (SDK). Però vam tenir una gran decepció quan la companyia americana va alliberar l'SDK. No només degut a les restriccions posades per *Apple* tant al sistema operatiu com al maquinari^{ix} (no vaig poder disposar d'un ordinador *Apple* per la realització de tot el desenvolupament) sinó també pel fet de la prohibició legal de la companyia d'instal·lar el sistema operatiu *MacOSX 10.5* en maquinari que no sigui d'*Apple*^x, s'ha descartat el desenvolupament d'aplicacions pel dispositiu *iPhone* en el projecte. Ens consola saber que, tot i la gran acollida de l'*iPhone* al mercat, aquest està més enfocat a l'ús personal i multimèdia que no pas a l'empresarial.

2.4 BlackBerry



BlackBerry és una gamma de productes fabricats per l'empresa *Research In Movement* (RIM) que va ser fundada l'any 1984 al Canadà. RIM va crear el primer *BlackBerry* l'any 1997. Aquest era un dispositiu de pantalla monocrom i el seu tret distintiu era la possibilitat de llegir el correu electrònic, fet que el va fer molt popular als Estats Units entre els directius de les grans empreses. Una altra característica que va incrementar la popularitat del dispositiu canadenc, va ser disposar d'aplicacions pròpies de PDAs com per exemple eines d'ofimàtica, calendari, notes...

En els últims anys s'ha avançat molt en el tema de missatgeria entre dispositius *BlackBerry*, gràcies a la reducció de tarifes que moltes companyies operadores de telefonia ofereixen i gràcies a la possibilitat de transferència gratuïta de dades per tots els serveis que facin servir "BlackBerry Enterprise Server" (BES). Aquest últim inclou, no només productes *BlackBerry* sinó també altres dispositius compatibles amb el protocol de "BlackBerry Email Client".

2.4.1 Característiques

Els models actuals, a diferència del primer dispositiu, tots tenen pantalles en color. Per contra, encara no hi ha cap dispositiu *BlackBerry* amb pantalla tàctil al mercat. Aquest fet influeix en el desenvolupament de les aplicacions, ja que s'hauran de basar en les opcions de menú en comptes de múltiples botons. El que singularitza als dispositius *BlackBerry* és que utilitzen un sistema operatiu propi, *BlackBerry OS*. Aquest sistema operatiu és compatible amb el sistema MIDP 2.

Tot i que molta gent utilitza els dispositius *BlackBerry* a nivell personal, aquests dispositius estan principalment destinats a empreses. El BES, comentat anteriorment, està pensat per integrar-se amb el sistema de correu electrònic de l'empresa client. El *BlackBerry Enterprise Server* és un paquet de *software middleware* que permet connectar *Microsoft*

Exchange, Lotus Domino, Novell GroupWise de la xarxa de l'empresa client amb els dispositius *BlackBerry*.

Per altra banda, *BlackBerry* posa a mans dels administradors de xarxa eines per controlar quins tipus de connexió es permeten fer i quins tipus no. Mitjançant *BlackBerry MDS* (Mobile Data Services) l'administrador pot crear serveis nous pels empleats que tinguin dispositius *BlackBerry*, i decidir quins serveis i quines aplicacions han d'estar disponibles.

Apart dels serveis BES també es poden crear aplicacions *Java* per a *BlackBerry*, tal i com s'ha comentat en la secció explicativa dels *MIDlets*. Aquestes aplicacions no deixen de ser *MIDlets* però amb algunes classes pròpies que ajuden i proporcionen accés i control a elements més específics dels dispositius *BlackBerry*. Per aquesta raó, les aplicacions creades amb aquestes eines específiques no són simples *MIDlets* i, per tant, no es poden fer servir en dispositius que no siguin *BlackBerry*. A més a més, en funció del dispositiu i de la versió del sistema operatiu que tingui instal·lada, obligarà a fer servir una versió concreta del *BlackBerry JDE*.

Personalment, el que no m'agrada dels dispositius *BlackBerry* és el lligam tant estret que manté amb l'operador. Aquests no només estan bloquejats per ser utilitzats únicament amb la xarxa del venedor, sinó que per poder actualitzar el SO (que pot proporcionar millores com per exemple l'accés a Internet per part de les aplicacions sense fer servir BES, millores de comunicacions, passar de GPRS -General Packet Radio Service- a EDGE -Enhanced Data rates for GSM Evolution-) també es depèn de l'operador. Per exemple, en el meu cas dispo d'una *BlackBerry* que un amic em va proporcionar per tal d'entendre'n millor el seu funcionament i tenia una versió antiga del sistema operatiu. M'assabento que RIM ha tret una nova versió del Sistema Operatiu per aquest dispositiu, però al visitar el web el meu operador no hi apareix. Encara que sigui un dispositiu alliberat (no està bloquejat per utilitzar només la xarxa original) cal una versió feta expressament per a l'operador original. Realment es pot actualitzar fent servir la versió d'algun altre operador, fins i tot d'algun altre país, però no de manera oficial ja que cal la modificació del software que es fa servir per poder pujar el nou sistema operatiu al dispositiu mòbil. A diferència dels ordinadors de sobre taula, la versió del sistema operatiu dels dispositius *BlackBerry* no és general, sinó que quan es publica una versió nova del sistema operatiu es publica una versió per cada un dels models compatibles.

2.5 Windows Mobile



Windows Mobile és un sistema operatiu compacte, o sigui que combina el sistema operatiu amb un conjunt d'aplicacions bàsiques. Hi ha diferents tipus de dispositius que poden fer servir *Windows Mobile*: PocketPCs (o ordinadors de butxaca), els anomenats telèfons intel·ligents, centres portàtils multimèdies i també certs ordinadors que el porten com a sistema operatiu encastat. El sistema està dissenyat perquè estèticament sigui semblant a les versions de sobretaula de *Windows*. De la mateixa manera que la marca *Palm* s'ha anat decantant per *Windows Mobile 6^{xi}* tot i tenir el seu propi sistema operatiu (el PALM OS, però PAL OS no està enfocat a telefonia mòbil), *SonyEricsson* ha incorporat com a sistema operatiu *Windows Mobile^{xii}* al seu l'últim model de gamma alta (anomenat *Experia*) tot i ser propietari de *Symbian*. (veure 2.2)

L'evolució del *Windows Mobile* es pot descriure amb els següents punts:

- **Pocket PC 2000:** Va ser alliberat l'abril de l'any 2000, es basava en el Windows CE 3.0 que és un sistema operatiu per sistemes encastats (no és un sistema operatiu retallat com el Windows XP Embbebed que es basa en el nucli NT, sinó que és un sistema que prové d'una branca pròpia). Va ser dissenyat per ordinadors de butxaca en comptes de telèfons intel·ligents. El dispositiu tenia incorporades versions reduïdes d'aplicacions conegudes, com el Pocket Office i el Windows Media Player. Visualment era similar al Windows 98 i al Windows 2000
- **Pocket PC 2002:** Alliberat l'octubre del 2001 també estava basat en el Windows CE 3.0 però en aquesta versió ja es tenien en compte el telèfons intel·ligents. També es va canviar l'entorn visual per assemblar-se al Windows XP
- **Windows Mobile 2003:** Alliberat el juny del 2003 va ser el primer dispositiu amb la denominació *Windows Mobile*. Aquest ja estava basat amb el sistema Windows CE 4.2. i va sorgir en 4 versions: "Windows Mobile 2003 for Pocket PC Premium Edition", "Windows Mobile 2003 for Pocket PC Professional Edition", "Windows

Mobile 2003 for Smartphone" i "Windows Mobile 2003 for Pocket PC Phone Edition". Aquest últim estava pensat per Pocket PCs que tenien funcionalitats de telèfon.

- *Windows Mobile 2003 SE*: Alliberat el març del 2004 va ser una actualització del *Windows Mobile 2003*.
- *Windows Mobile 5.0*: Va ser alliberat el maig del 2005 i està basat en Windows CE 5.0 i fa servir .NET Compact Framework 1.0 SP 2. Una de les grans millores va ser que permetia l'emmagatzematge persistent (permanent storage) en Pocket PCs
- *Windows Mobile 6*: Va ser alliberat el Febrer del 2007. Aquesta versió ve en tres modalitats diferents: "Windows Mobile 6 Standard" per telèfons intel·ligents sense pantalla tàctil, "Windows Mobile 6 Professional" per Pocket PCs amb funcions de telèfon, i "Windows Mobile 6 Classic" per Pocket PCs sense funcions de telèfon. Aquesta versió es basa en el Windows CE 5.0, que porta preinstal·lat en la ROM el .NET Compact Framework v2 i el Microsoft SQL Server 2005 Compact Edition. L'entorn visual és molt semblant a l'última versió de Windows per a escriptori, Windows Vista. Windows Mobile 6 Professional, tal i com es veu a la Taula 3, seria una barreja del *Windows Mobile Standard* i el *Windows Mobile Classic*.
- *Windows Mobile 6.1*: Una millora de la versió 6, només aporta millores gràfiques.

		New Product Family		
		Standard	Pro	Classic
Platform	Basic Platform and Applications – including: Outlook Mobile Internet Explorer Mobile Media Player Mobile ActiveSync	✓	✓	✓
	Touch screen UI	✗	✓	✓
	Windows Mobile Update	✓	✓	✓
	Windows Mobile Marketplace	✓	✓	✓
	IP Telephony	✓	✓	✗
Applications	Office Mobile	✓	✓	✓
	Window Live Mobile	✓	✓	✓
	Voice Command 1.6	✓	✓	✓
	Remote Desktop Mobile	✗	✓	✓

Taula 3 Comparativa de les diferents versions de *Windows 6*^{xiii}

2.5.1 Característiques

Deixant de banda la versió *Windows Mobile* Standard que està dirigida a telèfons intel·ligents, les altres dues versions estan més aviat enfocades a la pantalla tàctil i l'entorn gràfic per tal de competir amb el dispositiu *iPhone*. El nivell d'aquest últim està però una mica per sobre dels altres dos. No serà fins que surti la versió 7 de *Windows Mobile*^{xiv} que realment serà un autèntic rival per al dispositiu d'*Apple* en temes de pantalla tàctil.

Tot i així la versió actual, la 6, té millores respecte les versions anteriors i característiques molt interessants. Per una banda, ens permet llegir i enviar missatges en format HTML perquè siguin més atractius. També té *Internet Explorer Mobile* millorat amb una representació de les pàgines web més evolucionada, i altres millores diverses de cara a l'usuari. Per altra banda, cara als desenvolupadors de programes, *Windows Mobile 6* utilitza .NET Compact Framework 2.0 (a l'apartat 2.5.2 s'explica amb més deteniment la tecnologia .NET).

2.5.2 Tecnologies .NET

Visual Studio 2008 té, al igual que Visual Studio .Net 2005 i Visual Studio .NET 2003 (nom que es va atorgar a la primera versió del IDE de Microsoft que integrava .NET Framework), suport natiu pel Framework .Net, el qual és un entorn de programació que permet, entre d'altres:

- Simplificació del desenvolupament i distribució d'aplicacions per entorns compatibles.
- Model Unificat de programació que permet interactuar entre els diferents llenguatges suportats de manera única.
- Proporcionar un entorn d'execució totalment compatible; robustesa i seguretat.

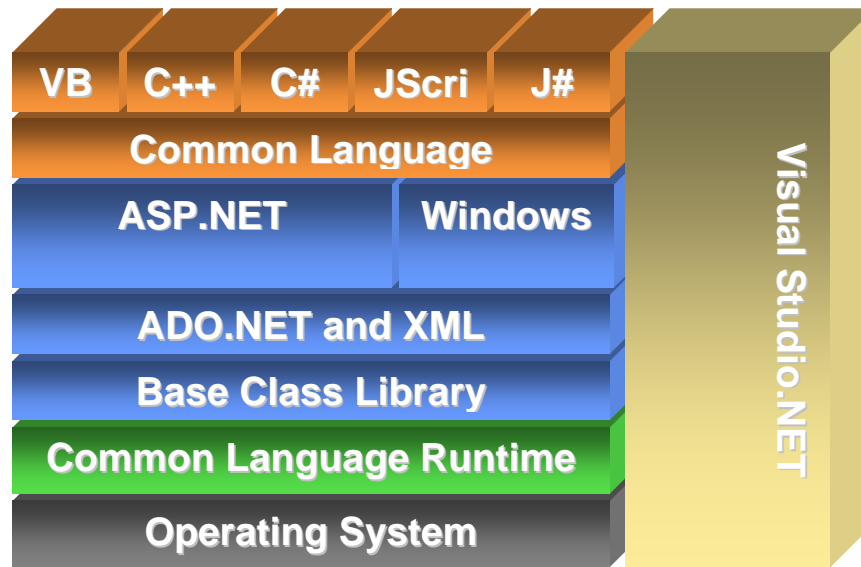


Figura 2 Esquema de l'estructura del *Visual Studio .NET*

Tot i que hi ha un gran suport per utilitzar *Visual Basic* i es troba tant integrat amb el Framework .NET com el C#, he optat per fer servir aquest últim ja que presenta una estructura del llenguatge molt més semblant al Java (que és el llenguatge que estic més acostumat a fer servir per temes de feina) i al C++.

C# és un híbrid entre Java i C++. Aquest invent de *Microsoft* presenta els avantatges de tots dos llenguatges i en el qual s'ha intentat corregir el què la companyia creu que són defectes.

En resum, C# és un llenguatge de programació que consta de les millors característiques dels llenguatges existents com Visual Basic, Java o C++ i les combina en un de sol. Apart dels motius explicats anteriorment, podem destacar-ne:

- Senzillesa: Elimina molts elements que altres llenguatges inclouen i que són innecessaris en .Net.

Per exemple: El codi escrit en C# és autocontingut. És a dir, no necessita fitxers addicionals com ara fitxers de capçalera o fitxers de fonts.

- La mida dels tipus de dades bàsiques és fix i no depèn ni del compilador ni del sistema operatiu, fet que facilita la portabilitat del codi.

- No s'inclouen els elements complexos de llenguatges com C++, com podrien ser les *macros*, l'herència múltiple, etc.
- C# suporta totes les característiques pròpies del paradigma de la programació orientada a objectes: encapsulació, herència i polimorfisme.
- Orientació a components: la pròpia sintaxi de C# inclou elements del mateix disseny de components que altres llenguatges han de simular mitjançant construccions més o menys complexes. És a dir, la sintaxi de C# permet definir fàcilment propietats, esdeveniments o atributs.
- Seguretat de tipus: C# inclou mecanismes que permeten assegurar que els accessos a tipus de dades sempre es realitzen correctament. Això evita que es produeixin errors difícils de detectar per accés a memòria que no pertany a cap objecte. Per aquest motiu es prenen les següents mesures:
 - Només s'admeten conversions entre tipus compatibles.
 - No es poden usar variables no inicialitzades.
- Es pot controlar la producció de desbordaments en operacions aritmètiques, informant d'això amb una excepció quan succeeix
- Sistema de tipus unificat: a diferència de C++, en C# tots els tipus de dades que es defineixin sempre derivaran, encara que sigui de manera implícita, d'una classe comuna anomenada `System.Object`, pel què disposaran de tots els membres definits en aquesta classe.
- Potent: en principi en C# tot el codi inclou nombroses restriccions per tal d'assegurar la seva seguretat i no permet l'ús de punters. En canvi, a diferència de Java, en C# és possible ignorar aquestes restriccions manipulant objectes a través de punters.

2.6 Android

Mesos abans de l'alliberament d'*Android* hi va haver moltes especulacions sobre què seria exactament. Es va especular que *Google* volia crear un nou dispositiu mòbil^{xv}; un telèfon per competir amb l'*iPhone* d'*Apple* (fins i tot se l'anomenava *gPhone*, de *Google Phone*).

El dia 5 de novembre Andy Rubin (Andy Rubin –Director de Plataformes Mòbils de *Google*) va anunciar: “*Android* és més important i ambiciós que un simple telèfon”^{xvi}. Així es va descobrir que *Google*, juntament amb *Open Handset Alliance* (OHA)^{xvii}, volien proporcionar un joc complet de software: el sistema operatiu, middleware i aplicacions bàsiques o clau.

Va ser el 12 de novembre del 2007 quan es va alliberar el kit de desenvolupament de software (SDK) de *Google Android*. L'alliberament va provocar un gran impacte a les pàgines relacionades amb la tecnologia o amb la programació. Durant els dos primers dies posteriors al 12 de novembre, la pàgina web del grup d'*Android* va rebre més de dos mil missatges. El que fa tan especial aquest nou sistema és la gran flexibilitat que en principi ha de proporcionar. Se suposa que es podrà canviar fins i tot les aplicacions inicials i es tindrà accés a qualsevol zona del dispositiu (però amb matisos de seguretat). Tot i que es va avisar que l'SDK no era definitiu, que ni tan sols era una versió Beta sinó que era simplement un alliberament per donar una primera impressió i que tothom pogués veure com seria *Android*, la xarxa es va començar a omplir de missatges de problemes i queixes per falta de documentació. Tot i que ja ha passat gairebé un any del dia de l'alliberament, el kit encara està en fase beta i en constant millora. Fins la data hi ha hagut cinc versions prèvies a l'actual, que és “*Android 0.9 SDK beta*”.

2.6.1 Característiques

Característiques que fan diferent i innovador a *Android* segons els seus responsables.

La principal és que volen que *Android* sigui obert. Però no només obert en temes de software lliure, sinó que inclou el no fer distincions entre aplicacions pròpies del dispositiu i aplicacions desenvolupades per altres empreses.

Dóna una alta llibertat a les aplicacions per poder ser canviades, donar accessos i poder ser personalitzades. Un cas seria la possibilitat de crear aplicacions que s'associïn a successos propis del dispositiu, com per exemple el fet de rebre un missatge curt ("*Android.provider.Telephony.SMS_RECEIVED*"). *Android* reconeix que hi ha més d'una aplicació que està escoltant aquest succés i per tant li pregunta a quina aplicació se li vol donar permís.

Un altre fet important és que amb *Android* es vol trencar amb els límits i barreres entre el dispositiu, les aplicacions i la xarxa. És factible crear una aplicació que combini dades d'Internet amb dades del propi telèfon. Per exemple, una aplicació que combini les adreces de l'agenda amb el servei de *Google Maps* podria mostrar les cases dels amics que estan properes a la nostra posició.

Tot i estar desenvolupat en C++, *Android* fa servir com a llenguatge vehicular el Java. Tot i així, els programes no s'executen en una màquina virtual normal sinó que s'ha desenvolupat una màquina virtual pròpia, anomenada *Dalvik*.

L'arquitectura d'*Android* és la següent:

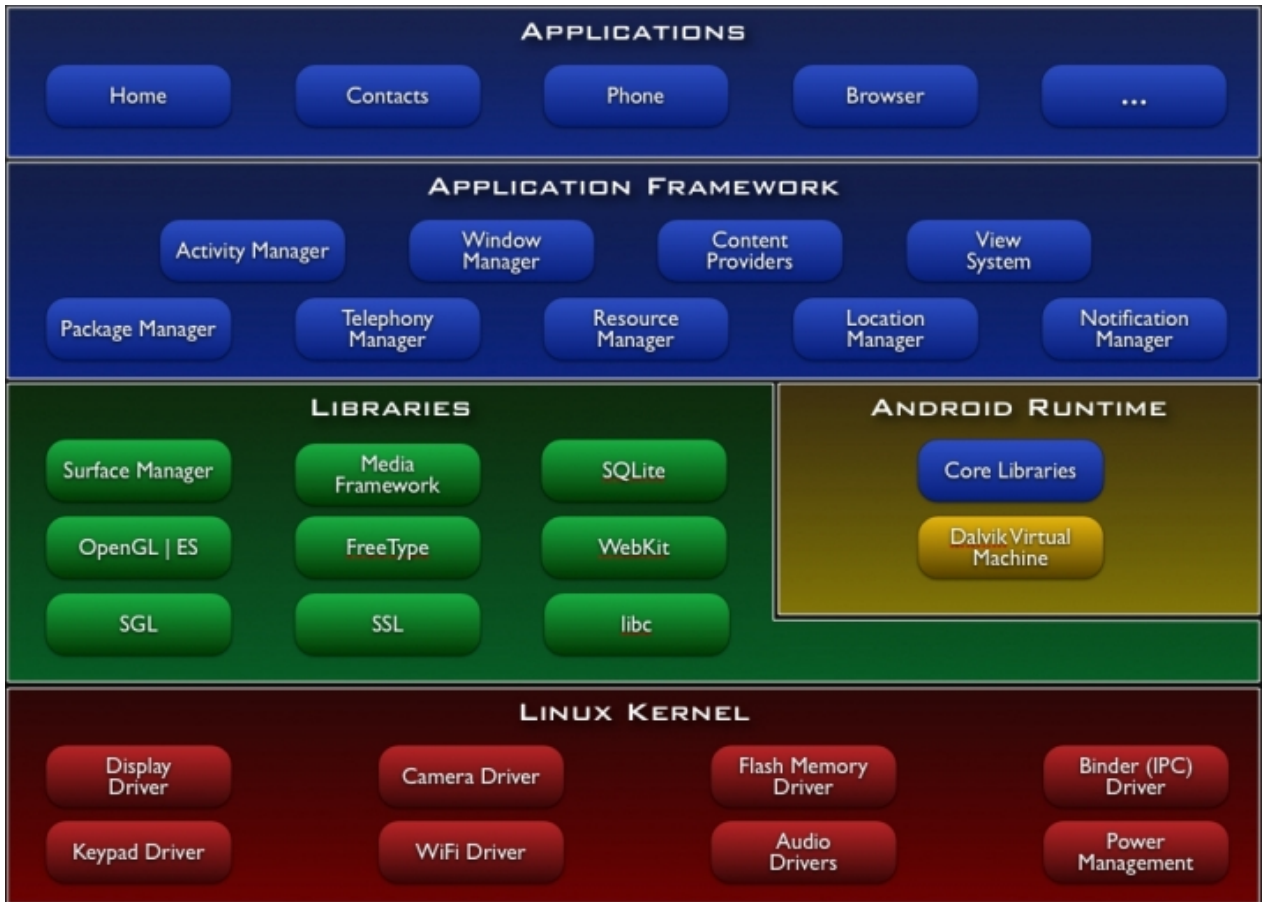


Figura 3 Esquema de l'arquitectura d'*Android*

Nivell d'aplicació

Android es proporciona amb algunes aplicacions per defecte, com per exemple un client de correu electrònic, un programa per rebre SMS, calendari, mapes i altres. Totes les aplicacions estan escrites en Java.

Entorn de treball d'aplicació

Els desenvolupadors tenen accés al mateix entorn de treball amb el què s'han desenvolupat les aplicacions del nucli de l'*Android*. L'arquitectura de les aplicacions està pensada per a la reutilització de components. Qualsevol aplicació pot publicar les seves dades i recursos, i aquests poden ser utilitzats per altres aplicacions (dins dels paràmetres de seguretat del framework).

Lliberies

Android inclou un grup de lliberies de C/C++ que són utilitzades per alguns dels components de l'arquitectura *Android*. A aquestes lliberies s'hi accedeix a través de l'entorn de treball. Algunes de les lliberies del nucli són:

- **System C library** – llibreria que deriva del sistema BSD, modificada per a aparells amb Linux encastat.
- **Lliberies Multimèdia** – basades en el PacketVideo OpenCore. Les lliberies suporten la reproducció i la gravació dels formats més coneguts d'àudio, vídeo i imatge estàtica.
- **Lliberies gràfiques** – Connecten amb el subsistema de la pantalla i s'encarreguen de dibuixar tant en dues com en tres dimensions.
- **LibWebCore** – Llibreria que tant fa servir el navegador d'*Android* com la vista web.
- **SQLite** – Un potent motor de base de dades relacional lleuger disponible per a totes les aplicacions.

Android Runtime

Android inclou un conjunt de lliberies nucli que donen gairebé la mateixa funcionalitat que les lliberies nucli del llenguatge Java.

Cada aplicació corre en el seu propi procés, com una instància de la màquina virtual *Dalvik*. *Dalvik* s'ha dissenyat perquè cada aparell sigui capaç de fer córrer varies màquines virtuals a la vegada. La màquina virtual *Dalvik* es recolza en el nucli de Linux per les tasques de control dels fils i per l'administració de la memòria a baix nivell.

Dalvik no és una màquina virtual *Java* convencional. *Dalvik* es basa en l'execució per registres, ja que els processadors mòbils estan optimitzats per l'execució a través d'aquests.

El nucli de Linux

Android es recolza en la versió 2.6 del *kernel* de Linux per les tasques seguretat, control de memòria, la pila de la xarxa, el control de processos i el control de *drivers*. El *kernel* també actua com a capa d'abstracció entre el maquinari i la resta de la pila de programari.

3 Els entorns de programació

En aquesta secció es descriuen les eines de desenvolupament d'aplicacions que s'han analitzat. Compararem els diferents entorns, la disponibilitat d'eines gràfiques de disseny, així com d'altres elements.

Els diferents entorns de programació s'instal·laran a la mateixa màquina, però per millorar-ne l'execució simultània hi haurà dos entorns instal·lats en un màquina virtual. L'ús de màquines virtuals permet tenir un segon entorn limitat actuant com una màquina independent a canvi d'una pèrdua d'eficiència. A més a més, serveix per evitar conflictes amb les instal·lacions gràcies a aquesta delimitació. Utilitzarem el programari de *VMWare player*. Tot i haver-hi altres programaris de codi obert com el *VirtualBox* de *Java*, he optat per utilitzar el *VMWare* principalment perquè és al que sóc més afí. A la màquina virtual de *Windows XP* hi haurà instal·lats el *Visual Studio 2008* i el *BlackBerry JDE 4.5.0*. Per altra banda, en la màquina real hi haurà l'*Eclipse* amb el *plugin* per al desenvolupament de programes per *Android* i l'*SDK* (en realitat estarà en una memòria USB permetent-me així treballar en qualsevol ordinador) . En el moment de fer les proves a la màquina virtual s'executaran el emuladors de *Windows Mobile 6* i de *BlackBerry*, mentre que a la màquina real s'hi executarà el programa de sobretaula i l'emulador d'*Android* (a la pràctica al no haver aconseguit que l'emulador d'*Android* es connectés a Internet, no caldrà fer les proves per *Android*) .

3.1 BlackBerry JDE 4.5.0

Research In Motion (RIM) proporciona de manera gratuïta, demanant només un registre, els entorns de programació des de la versió 4.0 fins la més nova, la 4.6, alliberada fa menys d'un mes (per aquesta raó ens hem centrat amb la versió 4.5)^{xviii}. *BlackBerry* manté les diferents versions de JDE degut a que se n'ha d'utilitzar la versió que correspongui amb la dels dispositius objectius. *BlackBerry* no es fa responsable del que pugui passar en un dispositiu al desplegar-hi una aplicació realitzada en un JDE posterior a la versió del propi dispositiu. Els diferents *BlackBerry JDEs* utilitzen com a llenguatge de programació el Java (utilitzen la versió Java2Me, pensada principalment per a dispositius mòbils). Hem

optat per la versió 4.5.0 ja que és la versió que no només disposa de connexió telefònica sinó també té connexió *wifi*. Aquest és l'aspecte de l'entorn de programació de la part de codificació:

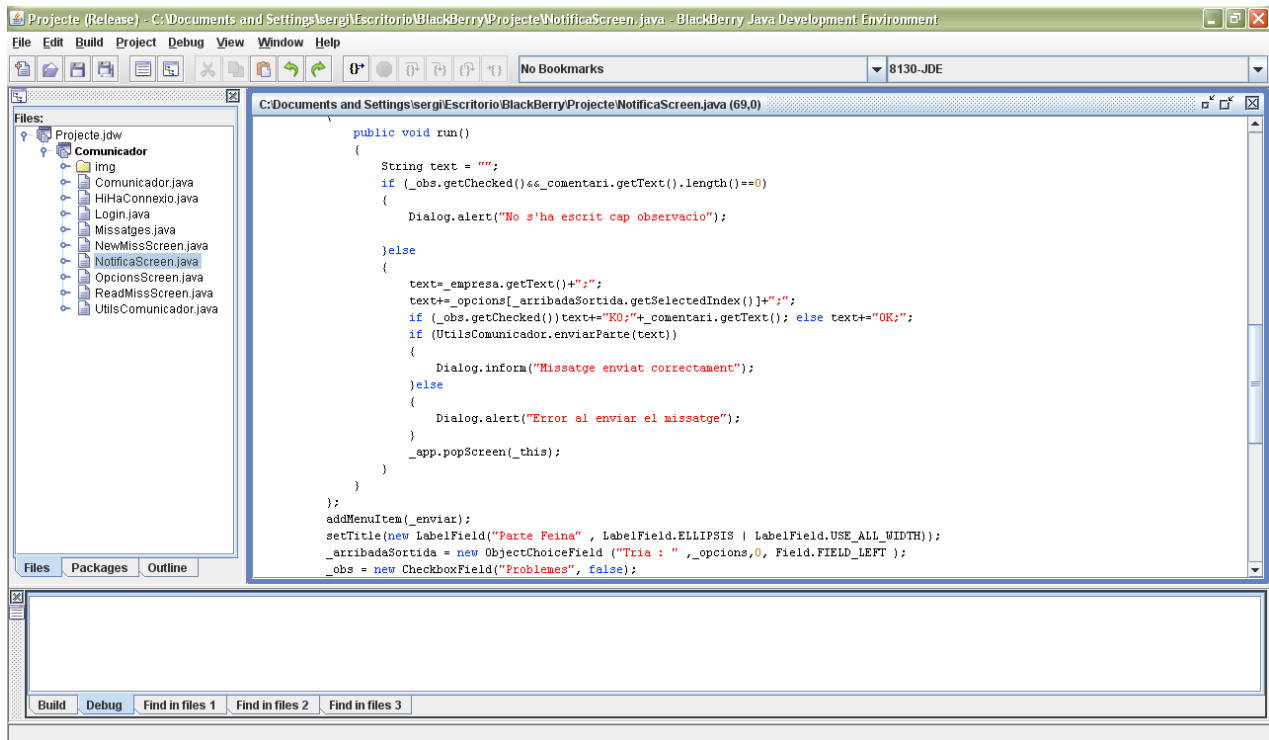


Figura 4 Entorn de programació de *BlackBerry JDE 4.5*

L'entorn en sí no aporta cap millora ni cap element distintiu. Presenta coloració de codi per tal de millorar la visibilitat. Al quadre de l'esquerra es poden veure les classes del nostre projecte i els altres arxius que conté. Els veritables avantatges de l'entorn de *BlackBerry* estan en el seu emulador i totes les facilitats que dóna per parametritzar gairebé qualsevol esdeveniment.

Amb l'emulador es poden simular trucades entrants, cobertura de les xarxes i altres. Com es pot veure a la Figura 5 s'ha desactivat la opció "*In Coverage*" (dins del quadre emergent) i això fa que l'emulador no tingui cobertura, tal i com es pot observar a la cantonada dreta superior de la imatge, on s'observa el símbol de l'antena quan no hi ha cobertura (Yx).

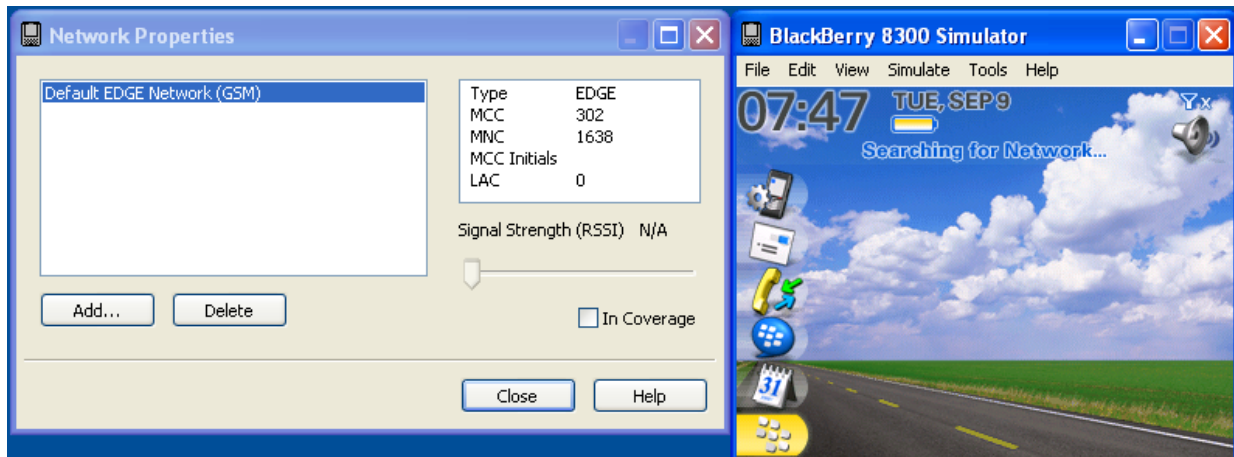


Figura 5 Desactivació de la cobertura de l'emulador per BlackBerry

A més a més, l'emulador proporciona exemples de diferents models per comprovar com queden estèticament les nostres aplicacions en les diferents modalitats de pantalla o la simplicitat d'utilització de l'aplicació amb els diferents teclats. Els emuladors són totalment funcionals, inclosos els botons de volum, rodetes i creuetes de navegació.



Figura 6 Representació de diferents models de BlackBerry a l'emulador

Per altra banda, *BlackBerry JDE* també inclou un simulador de connexions que permet connectar-nos a Internet, mentre que l'emulador del dispositiu creu que està connectat a través de la connexió de dades de telefonia. En resum, on ajuda més en el desenvolupament és amb l'emulador ja que l'entorn per si sol no aporta cap millora.

Un altre fet remarcable i d'agrair és que en un sol paquet s'inclou tot, l'entorn de programació, l'emulador i la resta d'eines, fet que en els entorns de les altres aplicacions no passa. Per contra, un dels majors desavantatges és la falta d'opció de disseny visual. Les pantalles es creen a partir de codi, "empenyent" els elements quan ens trobem en una pantalla (p.e. empenyent una imatge de fons, un quadre de text i un desplegable i ja tenim la pantalla de enviar nou missatge).

3.2 Visual Studio 2008

El Visual Studio 2008 és l'última versió de la suite de programació de Microsoft. Inclou suport per a diferents llenguatges, en el nostre cas el C# (aprofito aquest projecte també per aprendre aquest llenguatge fins ara desconegut per a mi). A diferència del *BlackBerry JDE*, Visual Studio 2008 no inclou SDK per desenvolupar ni tan sols les imatges per l'emulador.

El primer pas un cop ja és té instal·lat el Visual Studio, és descarregar i instal·lar l'SDK.^{xix} S'ha d'anar en compte amb la versió a instal·lar, ja que l'"Standard" és per telèfons intel·ligents i la versió "Professional" per a PocketPCs. Un cop fet això s'ha de baixar i instal·lar les imatges de l'emulador^{xx}.

La part de programació és molt semblant a la de qualsevol entorn. Tenim diferents pestanyes per tal de tenir més d'un arxiu obert (al centre de la Figura 7) i a la dreta tenim l'arbre de tots els arxius del nostre projecte.

Un dels avantatges de que proporciona *Visual Studio* i els SDK és la opció de poder crear les diferents vistes de les pantalles dels nostres programes de manera visual, arrossegant elements del "Toolbox" de l'esquerra dins la vista del dispositiu. Si cliquem en algun dels elements ja situats a la imatge del dispositiu anirem directament a la zona de codi que faci referència al propi element.

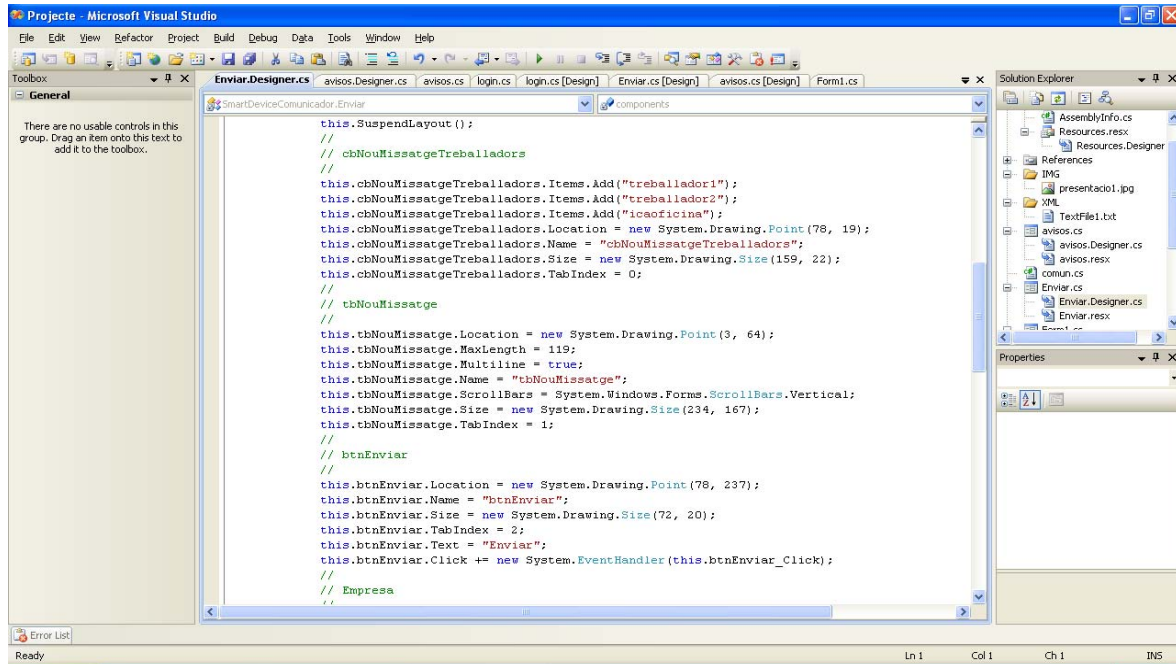


Figura 7 Entorn de programació de Visual Studio 2008

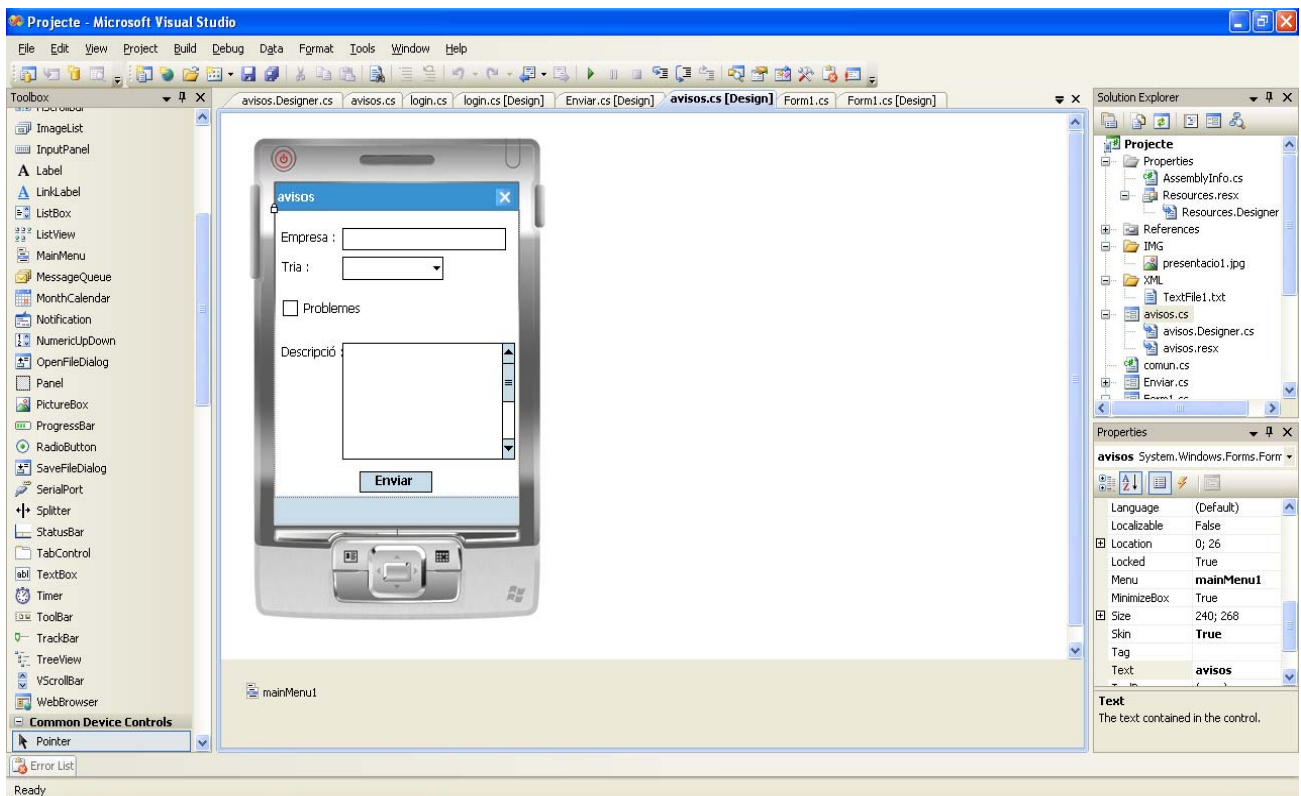


Figura 8 Vista de disseny d'una pantalla de Windows Mobile

Aquesta manera de programar simplifica moltes tasques, com per exemple per crear un fil d'execució que vagi realitzant consultes de missatges nous, només s'ha d'arrossegar l'element "timer" a l'aplicació i configurar-lo amb el quadre "Properties" (cantonada inferior a la dreta de la Figura 8). Amb aquests passos ja s'ha creat el fil i només faltaria incorporar-hi el codi.

L'SDK també ens proporciona eines de configuració i de control d'esdeveniments GSM que ens ajudaran a provar les nostres aplicacions simulant esdeveniments reals, com trucades entrants, SMS, etc.

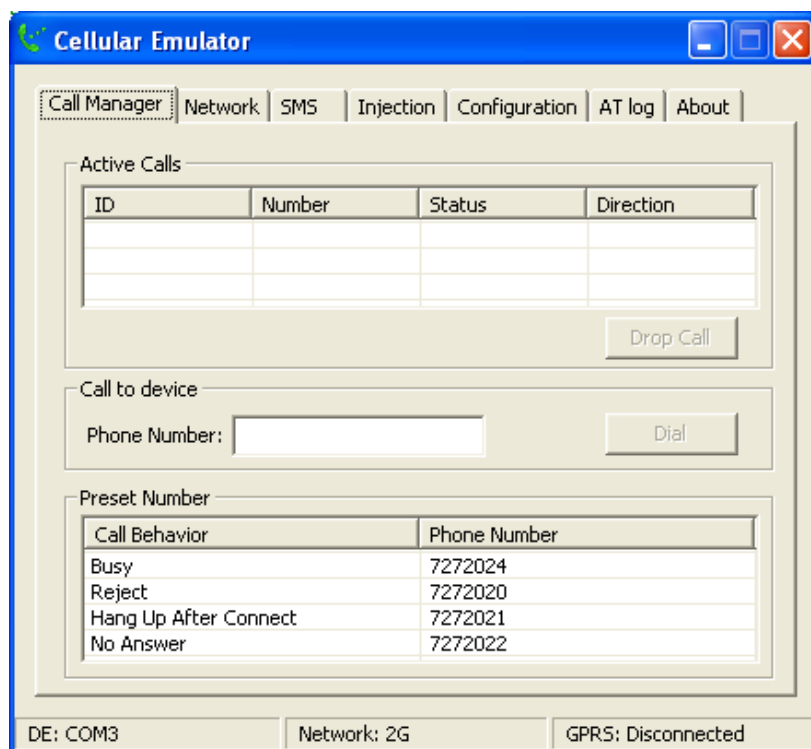


Figura 9 Emulador de xarxa GSM

Depenent de l'SDK i de les imatges que hàgim instal·lat tindrem uns aparells simulats o uns altres. Per exemple, en el nostre programa hem fet servir els de la versió professional. A la imatge de l'esquerra de la Figura 10 tenim la versió per dispositius sense telèfon, i a la dreta la versió amb telèfon. A simple vista no semblen gaire diferents. Cal fixar-se en la icona de cobertura de xarxa i els botons de penjar/despenjar només presents en la imatge de la dreta.



Figura 10 Models “Classic” i “Professional” de l'emulador de dispositius amb Windows

Encara que semblin iguals, és important saber quin dels dos estem fent servir ja que interiorment sí que hi ha diferències. Si el nostre programa fa servir connexió via telèfon i ens equivoquem d'emulador fent servir el “*Classic*” en comptes del “*Professional*”, aquesta no funcionarà.

Si instal·lem l'SDK de la versió estàndard tindrem disponibles les versions de l'emulador sobre telèfons intel·ligents. Aquestes versions són més reduïdes i hi ha disponibles diferents models per poder provar-les en diversos entorns, ja sigui de pantalla vertical o horitzontal (Figura 11).



Figura 11 Exemples de telèfons intel·ligents

A diferència de *BlackBerry*, on l'emulador representa models reals de dispositius, aquests models d'emuladors són genèrics ja que Microsoft només ven el sistema operatiu en comptes de fabricar o vendre dispositius.

3.3 Android (ECLIPSE)

Per poder programar aplicacions per *Android*, *Google* recomana fer servir l'entorn de programació *Eclipse*. *Eclipse* és una plataforma de desenvolupament gratuïta que corre sobre *Java*. Es fa servir tant per programar en *Java* com amb molts altres llenguatges, com *C* o *C++*. És gratuïta i es pot trobar per Internet. En aquest projecte he fet servir la versió *Ganymede*^{xxi}.

Un cop hem aconseguit Eclipse, cal instal·lar l'SDK d'*Android*^{xxii} i tenir present on s'instal·la ja que en el següent pas s'hi ha d'instal·lar el plugin d'Eclipse^{xxiii}. La versió actual de l'SDK d'*Android* és la Beta 0.9

Eclipse a nivell de codi s'assembla molt a l'entorn de *BlackBerry*. Al cap i a la fi, l'entorn de *BlackBerry* es basa en Eclipse. La part de creació de codi també funciona amb pestanyes i es pot tenir més d'un arxiu obert. Al quadre de l'esquerra de la Figura 12 es mostra l'estructura d'arxius del nostre projecte.

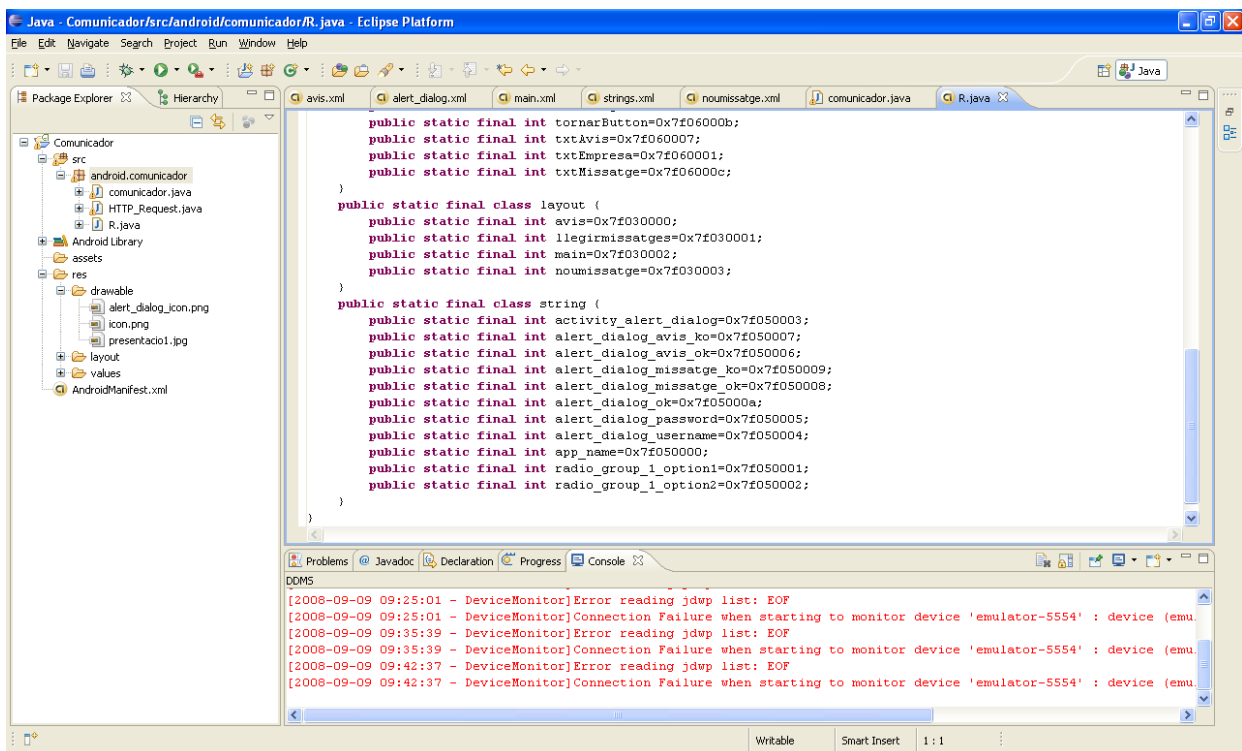


Figura 12 Vista de programació amb l'arxiu R.java

L'estructura d'un programa d'*Android* és diferent a l'estructura d'un programa Java. Per exemple l'arxiu R.java. Aquest arxiu conté identificadors dels valors que es guarden dins la carpeta "values" o "layout". Dins d'aquestes carpetes es guarden arxius XML que contenen cadenes de text o vectors de valors o les estructures de disseny per poder ser referenciats en qualsevol part del programa.

El desenvolupament d'aplicacions per *Android* també aporta una millora respecte el que hem vist per exemple amb *BlackBerry*. el disseny de les pantalles d'*Android* es fa mitjançant arxius XML (Figura 13).

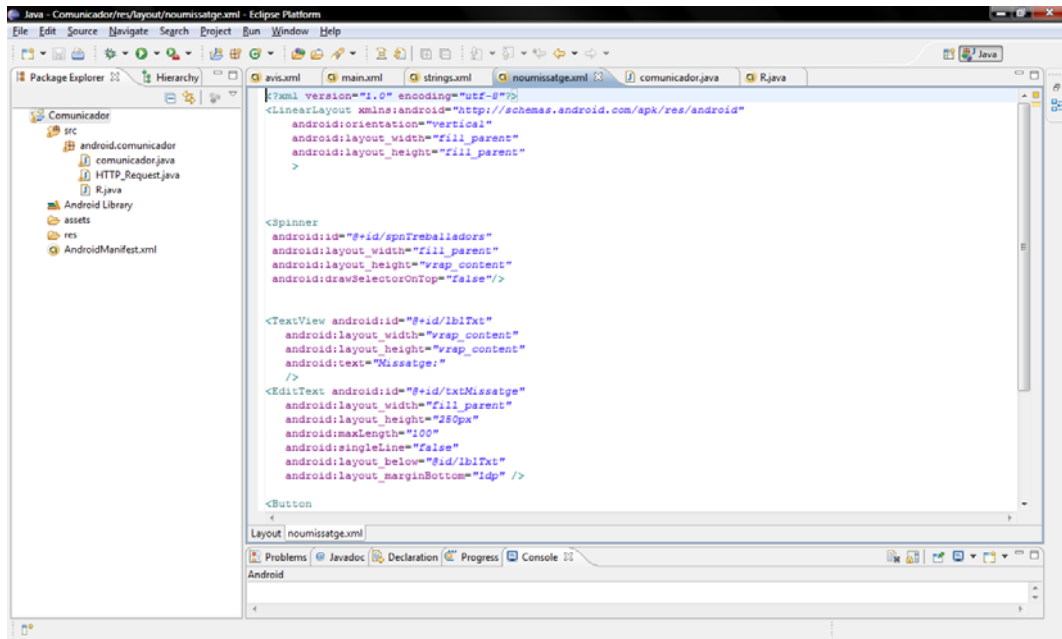


Figura 13 Imatge d'una pantalla amb un arxiu XML

Android aporta també una visió de la pantalla que estem dissenyant sense haver d'executar l'aplicació (Figura 14), fet que no és possible amb *BlackBerry*. No és tan intuïtiu com *Visual Studio* però és una millora.

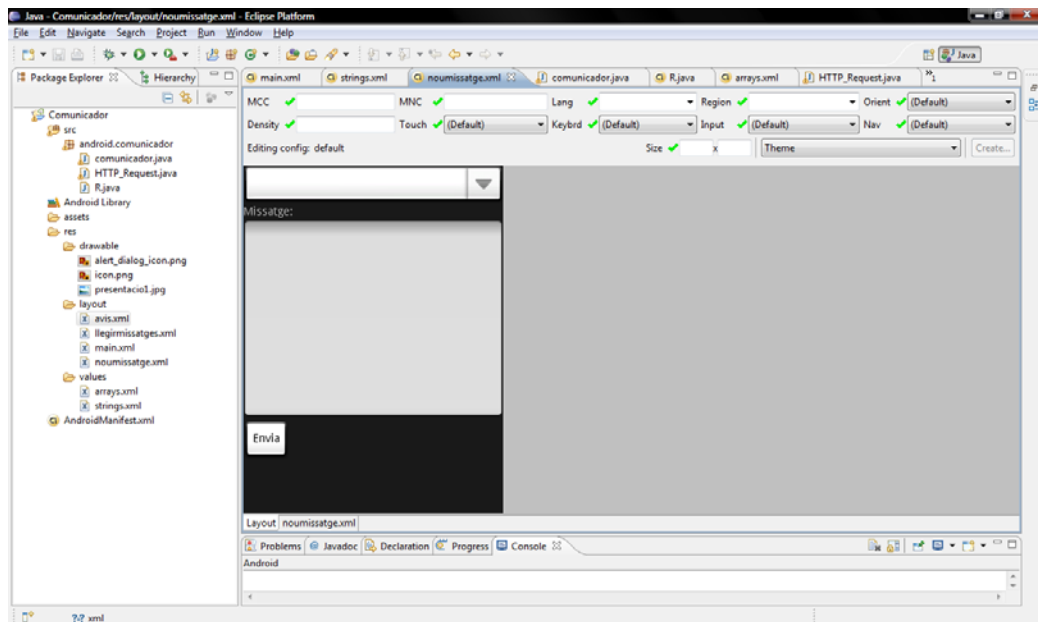


Figura 14 Vista de la pantalla en la fase de disseny amb el *Visual Studio*

I finalment l'emulador. De la mateixa manera que amb el Visual Studio, el model que es mostra no és cap model real sinó un model que representa tot el que pot ser *Android*, amb GPS, pantalla tàctil, etc.

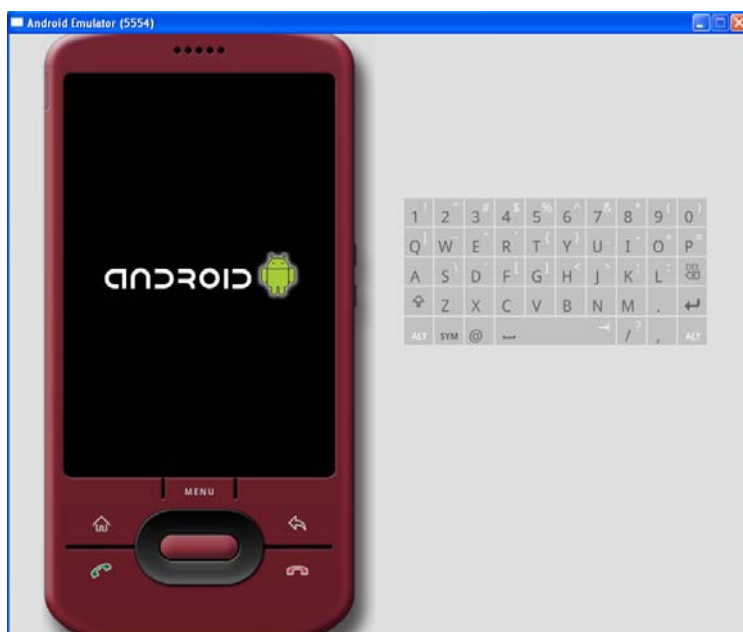


Figura 15 Emulador d'Android

El fet que l'emulador sigui tan complet i que representi un aparell amb tots els serveis del mercat pot ser una fulla de doble tall, ja que quan apareguin els dispositius reals amb aquest sistema operatiu i que potser no tinguin totes les funcionalitats, pot decebre als usuaris finals.

4 Descripció de l'aplicació

Com que la intenció és fer una aplicació que demostrï la capacitat dels diferents dispositius mòbils i que serveixi per valorar els diferents entorns de programació, les diferents APIs, etc., evitarem al màxim les complicacions donades per altres elements com per exemple dissenyar un servidor de comunicacions.

L'aplicació en qüestió consisteix en un sistema de comunicació remot entre empleats enfocat a empreses on els seus empleats es desplacen constantment i poden canviar de destí. Un exemple serien les empreses de serveis de reparació. L'aplicació a desenvolupar serviria com a mitjà de control per saber on es troba cada empleat i saber l'estat de les incidències en qualsevol moment.

4.1 Requeriments funcionals

Dividirem els requeriments funcionals en dos blocs, la part dels treballadors desplaçats i la part de l'empleat (en direm responsable o supervisor) que es queda a l'empresa. Els treballadors desplaçats serien els usuaris del dispositius mòbils i el supervisor seria el que faria servir l'entorn de sobretaula. Com que el projecte s'enfoca principalment als dispositius mòbils, no ens plantejarem una aplicació molt complexa per la part de sobretaula.

Treballadors desplaçats:

- S'han de poder comunicar amb els altres empleats de l'empresa a través de missatges curts.
- Ha d'haver-hi una funcionalitat que els permeti comunicar si han arribat o sortit de l'empresa on han sigut destinats.
- S'ha de poder consultar els missatges (en el cas que no hi hagi connexió s'ha de poder consultar els missatges antics).

Treballador responsable:

- S'ha de poder comunicar amb els empleats desplaçats.
- Ha de poder consultar els missatges antics.
- Ha de poder veure on es troben els treballadors desplaçats.
- A la vista de l'estat dels treballadors desplaçats ha de poder filtrar els missatges per empleats.

4.2 Requisits no funcionals

De la mateixa manera que en els requisits funcionals, també separarem els no funcionals entre els dels treballadors desplaçats i del treballador responsable.

Treballador desplaçat:

- Requisits de comunicació:
 - ⊕ Caldrà detectar si hi ha xarxa. En el cas que es faci servir la xarxa a través de telefonia mòbil s'ha de mostrar un avís per pantalla. En el cas del Pocket PC només cal detectar si estem connectats.
- Requisits de plataforma:
 - ⊕ *BlackBerry*: Al no ser pantalla tàctil s'ha evitar al màxim la navegació en l'aplicació per botons en pantalla i controlar la navegació mitjançant els menús.
 - *Windows Mobile* i *Android*: Aprofitar les opcions tàctils i utilitzar botons en pantalla per millorar-ne la usabilitat.
- Requisits d'errors:
 - ⊕ Controlar errors de comunicació i mostrar missatges a l'usuari.
- Requisits de disponibilitat:
 - ⊕ En cas de no haver-hi connexió no permetre enviar ni missatges ni avisos.

Treballador Responsable:

- Requisits de plataforma:
 - ⊕ Serà una aplicació d'entorn de sobretaula.
- Requisits de comunicació:
 - ⊕ Com que es tractarà d'una aplicació de sobretaula no es tindrà en compte que es pugui quedar sense connexió.

4.3 Diagrama casos d'us

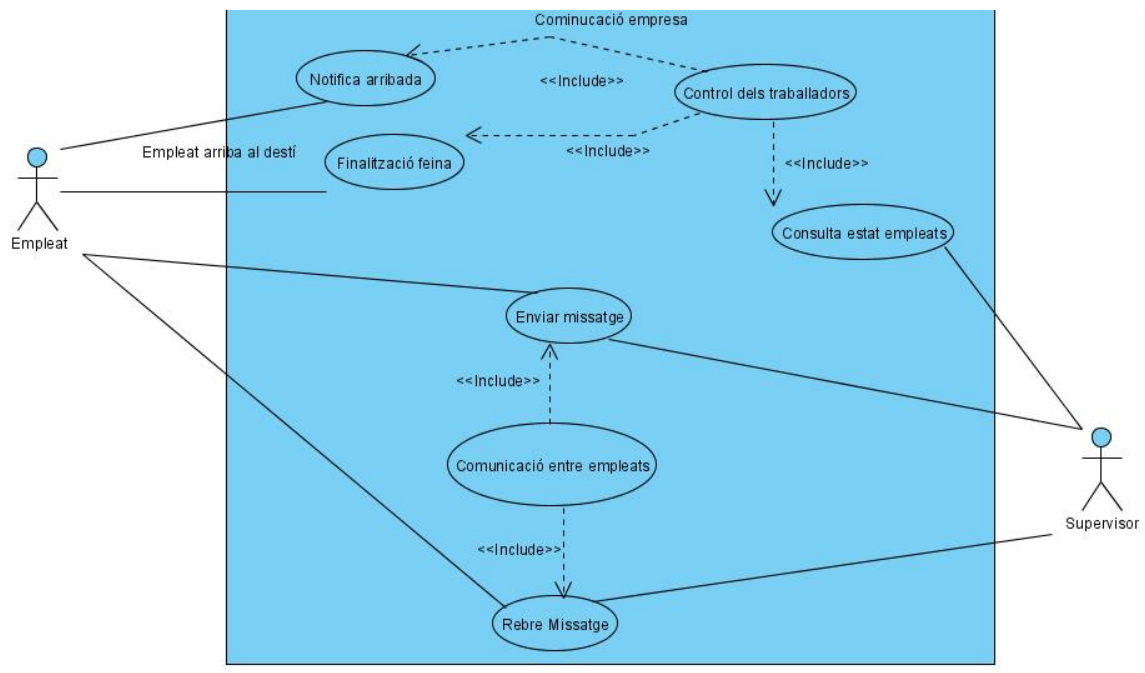


Figura 16 Diagrama de casos d'ús

L'usuari Supervisor és únic. En canvi de l'usuari Empleat n'hi haurà més d'un.

4.4 Explicació del casos d'us

Codi	Descripció	Usuari
E01	Notifica Arribada	Empleat
E02	Finalització Feina	Empleat
E03	Envia missatge	Empleat
E04	Rep Missatge	Empleat
E05	Consulta Missatges Antics	Empleat
S01	Consulta Estat	Supervisor
S02	Envia Missatge	Supervisor
S03	Rebre Missatge	Supervisor
S04	Consulta Missatges Antics	Supervisor

Taules detallades dels casos d'us:

- **E01 Notifica Arribada**

Identificador	E01
Nom	Notifica Arribada
Autor	Sergi Torralba
Resum	És l'acció que ha de realitzar un treballador al arribar a l'empresa on ha estat destinat
Actor(s)	Treballador
Pre-condicions	Arribar a la feina. Haver arribat però amb problemes.
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none"> 1. Al arribar a la feina l'empleat comunica a quina empresa ha arribat. 2. Quan marxi d'aquesta feina haurà de notificar-ho mitjançant "Finalització Feina".
Fluxos Alternatius	En cas d'error se li notificarà a l'usuari mitjançant una Alerta i se li recomanarà tornar-ho a intentar més endavant.

- **E02 Finalització Feina**

Identificador	E02
Nom	Finalització Feina
Autor	Sergi Torralba
Resum	És l'acció a realitzar quan un treballador deixa una feina, ja sigui perquè l'ha finalitzat o per què ha hagut de marxar per una altra raó.
Actor(s)	Treballador
Pre-condicions	Haver finalitzat la feina o tenir alguna raó per marxar.
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none">1. Un cop finalitzada la feina comunicar-ho a l'empresa.2. Si encara disposa de temps, anar a un altre empresa.
Fluxos Alternatius	En cas d'error se li notificarà a l'usuari mitjançant una Alerta i se li recomanarà tornar-ho a intentar més endavant.

- **E03 Envia missatge**

Identificador	E03
Nom	Envia missatge
Autor	Sergi Torralba
Resum	L'empleat desitja enviar un missatge a qualsevol dels altres empleats de l'empresa
Actor(s)	Treballador
Pre-condicions	Cap
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none">1. Enviar missatge2. Confirmació de missatge enviat correctament
Fluxos Alternatius	Error al enviar el missatge, es comunicarà l'error mitjançant una alerta i es recomanarà tornar-ho a intentar més tard

- **E04 Rebre missatges**

Identificador	E04
Nom	Rebre missatge
Autor	Sergi Torralba
Resum	Rebre missatge d'algun treballador de l'empresa o del supervisor
Actor(s)	Treballador
Pre-condicions	Que un altre treballador o el supervisor hagi enviat un missatge
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none"> 1. S'avisarà amb un avis l'arribada d'un missatge. 2. Llegir el missatge
Fluxos Alternatius	<ol style="list-style-type: none"> 1. No llegir el missatge

- **E05 Consultar missatges antics**

Identificador	E05
Nom	Consultar missatges antics
Autor	Sergi Torralba
Resum	Es podrà consultar els missatges antics desats al dispositiu mòbil
Actor(s)	Empleat
Pre-condicions	Voler consultar algun missatge antic.
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none"> 1. Llegir missatges.
Fluxos Alternatius	Cap

Taules detallades dels casos d'us:

- **S01 Notifica Arribada**

Identificador	S01
Nom	Consulta Estat
Autor	Sergi Torralba
Resum	L'usuari supervisor desitja consultar on estan els treballadors
Actor(s)	Supervisor
Pre-condicions	Que els treballadors hagin comunicat el seu estat
Post-condicions	Cap
Flux normal	Consulta l'estat de tots els treballadors o d'un en concret.
Fluxos Alternatius	Cap

- **S02 Envia missatge**

Identificador	S02
Nom	Envia missatge
Autor	Sergi Torralba
Resum	El supervisor desitja enviar un missatge a qualsevol dels altres empleats de l'empresa
Actor(s)	Supervisor
Pre-condicions	Cap
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none">1. Enviar missatge2. Confirmació de missatge enviat correctament
Fluxos Alternatius	Error al enviar el missatge, es comunicarà l'error mitjançant una alerta i es recomanarà tornar-ho a provar més tard

- **S03 Rebre missatges**

Identificador	S03
Nom	Rebre missatge
Autor	Sergi Torralba
Resum	Rebre missatge d'algun treballador de l'empresa o del supervisor
Actor(s)	Supervisor
Pre-condicions	Que un treballador hagi enviat un missatge
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none"> 1. S'avisarà amb un avis l'arribada d'un missatge. 2. Llegir el missatge
Fluxos Alternatius	<ol style="list-style-type: none"> 1. No llegir el missatge

- **S04 Consultar missatges antics**

Identificador	S04
Nom	Consultar missatges antics
Autor	Sergi Torralba
Resum	Es podrà consultar els missatges antics desats al dispositiu mòbil
Actor(s)	Supervisor
Pre-condicions	Voler consultar algun missatge antic.
Post-condicions	Cap
Flux normal	<ol style="list-style-type: none"> 1. Llegir missatges.
Fluxos Alternatius	Cap

4.5 Comunicacions

En el desenvolupament d'aquest treball s'ha fet servir un protocol de comunicacions públic. No és el més adient però ha servit per estalviar temps i per poder centrar-me en el tema del projecte que consisteix en l'avaluació dels diferents dispositius mòbils. S'ha fet servir el protocol de *TWITTER* ja que és un protocol molt establert i difós, fet que ha facilitat tot la cerca d'informació. Tot i així, només el farem servir per les demostracions, ja que no està pensat específicament per l'ús que se'n fa en aquesta aplicació. Fer servir Twitter com a protocol de comunicacions té varies limitacions:

- La mida màxima dels missatges: Twitter va ser ideat com un servei de microblog, és a dir un blog amb l'excepció que els missatges que s'envien tenen la mida d'un SMS (el nombre màxim de caràcters que permet Twitter són 140).
- La no encriptació de les dades: les dades s'envien en text pla.

L'avantatge de fer servir Twitter és que ens ha permès separar els missatges personalitzats dels avisos. Twitter està pensat per anar canviant d'estat i notificar-ho a tothom o a un grup reduït de persones autoritzades.

S'ha creat tres usuaris, treballador1 treballador2 i icaoficina. Els usuaris treballador1 i treballador2 representaran als treballadors desplaçats. El treballador1 disposarà d'un dispositiu *BlackBerry* i el treballador2 disposarà d'un *PocketPC* amb *Windows Mobile 6*. L'usuari icaoficina és el que farà servir el programa de sobretaula ja que representa el supervisor i està a l'oficina. En aquest cas, tots els empleats estan autoritzats a veure els canvis d'estat del altres usuaris, però només l'usuari icaoficina realitza consultes dels estats. És a dir, els treballadors només canvien d'estat, no el consulten. També hem posat com a condició que només les persones autoritzades puguin conèixer el nostre estat (actualitzacions privades).

Pel què respecta els missatges directes, Twitter també té un protocol d'enviament que ens ha facilitat molt la feina. De fet, Twitter controla per separat tot el què involucra la missatgeria directe.

Un dels elements que al principi vaig creure que facilitaria la feina però que al final va donar algun mal de cap, va ser que tota la comunicació amb Twitter es fa a través de

HTTP. Això fa que una tasca tan simple com codificar els missatges en un entorn de Java2e (que és amb el que estic més acostumat), s'hagi de fer a mà, ja que en entorns compactes no esta definida la funció de URLEncode o similars.

5 El Comunicador

Encara que el projecte tracta sobre dispositius mòbils hem cregut adient fer també una aplicació de sobretaula per obtenir un projecte rodó, ja que el plantejament de l'exemple constava de dues vessants, els treballadors per una banda (amb dispositius mòbils) i per l'altre el supervisor (que estaria a l'oficina).

Per l'aplicació de sobretaula s'ha optat per fer una aplicació amb Visual Studio.

Primer explicarem com s'ha solucionat cada problemàtica de l'aplicació per dispositius mòbils. En alguns exemples no es posaran captures d'*Android* degut a no haver aconseguit que hi hagués connexió. Per tant, només n'hi posaré en els exemples que siguin de disseny.

5.1 Dispositius mòbils

Les diferències entre les diferents plataformes ja es noten a les mateixes pantalles d'inici.



Figura 17 Pantalles de presentació de les aplicacions per dispositius mòbils

Al tenir una pantalla gran i estar tot enfocat a la pantalla tàctil he optat pels botons en la pantalla de presentació d'*Android* (imatge esquerra en la Figura 17).

La pantalla per *BlackBerry* (imatge central de la Figura 17) mostra el menú desplegat (que no ho està sempre). *BlackBerry* al no tenir pantalla tàctil està enfocat a fer servir botons de teclat. Per aquesta raó la majoria de navegacions es faran mitjançant menús.

A l'igual que *Android*, *Windows Mobile* també disposa de pantalla tàctil. Encara que sembli una incongruència, l'aplicació per *Windows Mobile* funciona a base de menús, tal i com es pot observar a la Figura 17 (imatge de l'esquerra, es mostra el menú desplegat). La raó és el "look & feel" de menús que s'aproximen més a *Windows Vista*.

Un dels temes que també s'havia de tenir en compte en el plantejament de l'aplicació era saber si el fet de fer servir la nostra aplicació causarà una despesa de connexió.



Figura 18 Pantalles sobre la connexió amb BlackBerry.

Per *BlackBerry* el que s'ha plantejat és comprovar primer si hi ha connexió via WIFI. En el cas que no n'hi hagi, es mostra un missatge a l'usuari demanant permís per connectar-se a la xarxa (Figura 18). Si l'usuari no es vol connectar, se li mostra un altre missatge informant-lo que sense connexió tindrà funcionalitat limitada.



Figura 19 Detecció de xarxa amb Windows Mobile

Els *PocketPC* no estan tan enfocats a la connexió com en poden estar els dispositius *BlackBerry*. Per aquesta raó la connexió es fa prèviament a l'execució de l'aplicació, com si es fes servir un mòdem en un ordinador de sobretaula. Així doncs, només s'informa amb la frase "Connectats a la xarxa" o "No connectats a la xarxa" (Figura 19). En el cas de no estar connectats, el menú conté les opcions que necessiten connexió deshabilitades.

En el cas d'haver-hi connexió a Internet, ja sigui mitjançant WIFI o per GSM, l'aplicació permet enviar missatges.



Figura 20 Pantalles de nou missatge de dispositius mòbils.

En aquesta funcionalitat s'han tingut en compte controls per evitar errors, com ara comprovar que s'ha seleccionat un destinatari o que el missatge no està buit. A la Figura 21 es mostra un dels missatges d'error per als tres casos.



Figura 21 Missatges de alerta en missatges buits.

Pel què fa a la programació dels missatges d'alerta hi ha una diferència important entre *BlackBerry* i *Windows Mobile* amb *Android*. Amb els dos primers és molt més simple aconseguir un missatge d'alerta que amb el darrer.

BlackBerry:

```
Dialog.alert("No hi ha missatge");
```

Windows Mobile:

```
MessageBox.Show("Si us plau escriu un missatge.");
```

Android:

```
return new AlertDialog.Builder(comunicador.this)
    .setIcon(R.drawable.alert_dialog_icon)
    .setTitle(R.string.alert_dialog_missatge_notext)
    .setPositiveButton(R.string.alert_dialog_ok, new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int whichButton)
        {
        }
    }).create();
```

Android dóna molta llibertat de configuració, però això fa que calguin més de cinc línies per a qualsevol operació, mentre que amb les altres plataformes es faci amb només una.

En el cas de què el missatge s'envii correctament, s'avisava a l'usuari amb un missatge informatiu (Figura 22).



Figura 22 Missatges de confirmació en la aplicació per dispositius mòbils.

De la mateixa manera també avisem a l'usuari si no ha sortit bé l'enviament del missatge amb un missatge d'error.

Per poder veure els missatges, tant per llegir l'acabat de rebre com per consultar els antics, els mostrem tots en bloc (Figura 23).



Figura 23 Pantalla de llegir missatges dels dispositius mòbils.

No s'ha pensat en tractar els missatges per separat ja que en aquest projecte no s'ha plantejat el fet de treballar sobre ells (ni esborrar-los ni el plantejament de resposta directa, etc.).

Pel què respecta a la comunicació d'arribada/sortida entre desplaçat i oficina (avisos), també s'ha dissenyat diferents pantalles, les quals s'il·lustren a la Figura 24. Per què l'avis s'envii, s'ha d'haver omplert el camp "empresa" i, si s'ha marcat la casella de "Problemes", s'obliga a omplir el camp establert per a la descripció del problema. Si no es compleix algun d'aquests requisits, es mostra a l'usuari un missatge emergent per informar-lo del què cal omplir.



Figura 24 Pantalles d'enviament d'Avisos dels dispositius mòbils

Una altra funcionalitat que s'ha introduït a l'aplicació, és que realitza peticions periòdiques al servidor per comprovar si hi ha missatges nous. En cas afirmatiu, es mostra un missatge a l'usuari informant-lo de la situació.



Figura 25 Avis de nou missatge en aplicacions de dispositius mòbils.

Per *BlackBerry* vaig aconseguir que es mostrés el missatge encara que l'aplicació estigués corrent al rerefons. El que no vaig aconseguir és que l'aplicació es posés al front. En canvi per *Windows Mobile* si l'aplicació no està al front i forcem un missatge per l'usuari, automàticament tota l'aplicació va al front.

5.2 L'aplicació de sobretaula

S'ha desenvolupat una aplicació compacte amb *Visual Studio 2008*. Aquesta aplicació està pensada per mostrar com connectar els treballadors desplaçats que disposen de dispositius mòbils amb la oficina.

L'aplicació presenta una petita navegació per pestanyes a la part esquerra, i a la part dreta, sigui quina sigui la pestanya seleccionada, apareixen els missatges que ha rebut l'usuari "icaoficina" (Figura 26).

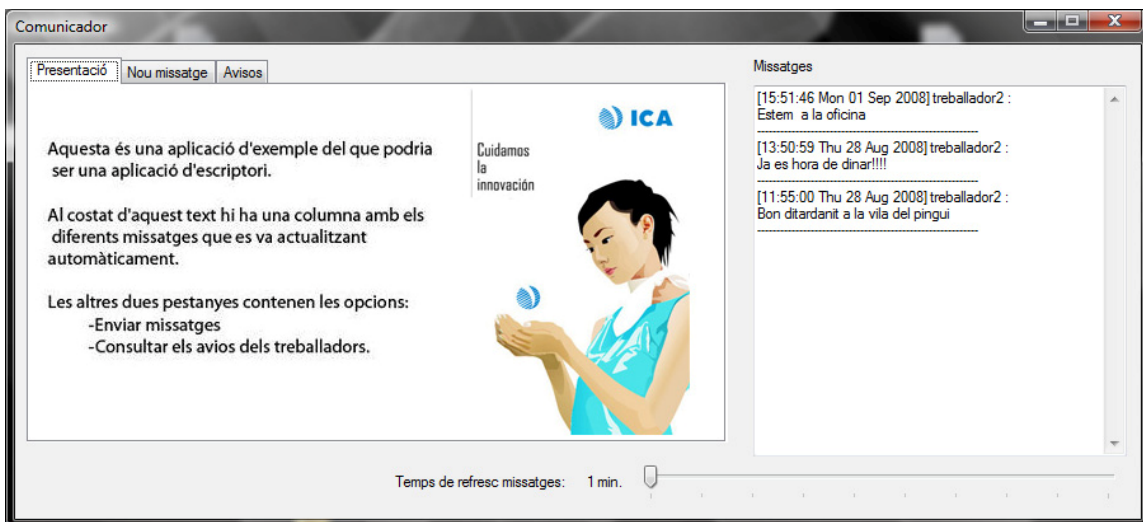


Figura 26 Pantalla de presentació de l'aplicació de sobretaula

Quan l'usuari rep un missatge se li mostra una avís per pantalla per si un cas no se n'ha adonat.

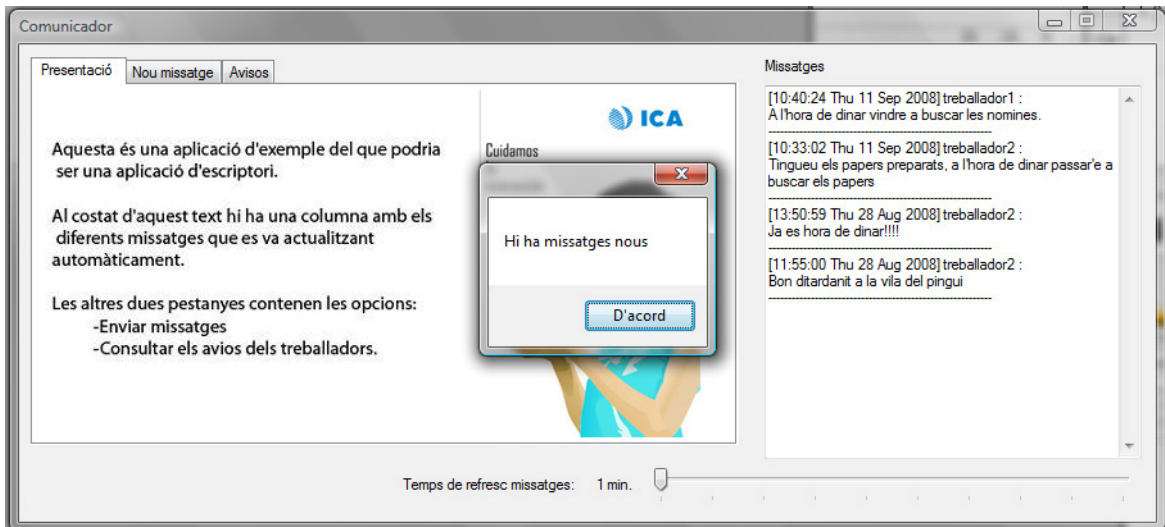


Figura 27 Avis de nou missatge a l'aplicació de sobretaula.

Aquest avís apareix sempre, estigui per la pestanya que estigui navegant l'usuari.

La pestanya central és la que es fa servir per enviar missatges. Consta d'un desplegable amb els treballadors de l'empresa i d'un quadre de text per escriure-hi el missatge.

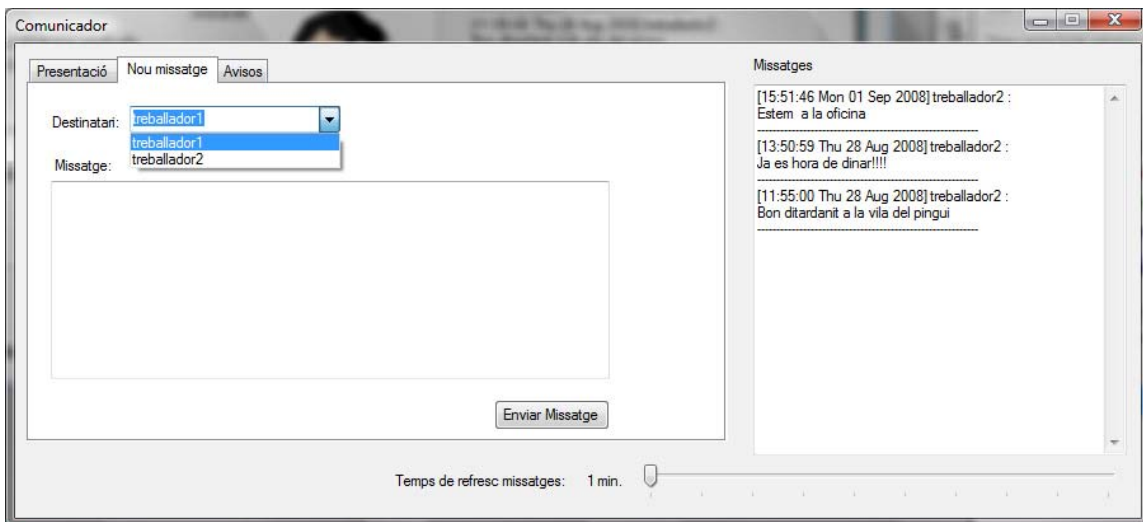


Figura 28 Pantalla de nou missatge de l'aplicació de sobretaula.

De manera anàloga als dispositius mòbils, també s'han fet validacions abans de realitzar els enviaments. Es comprova que s'hagi triat un destinatari i que el missatge no estigui buit (Figura 29).

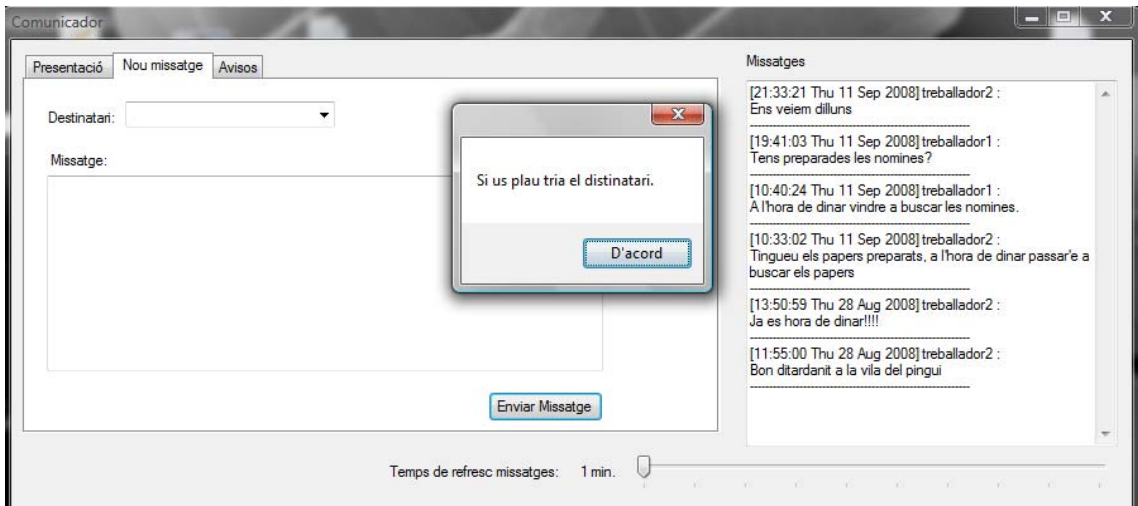


Figura 29. Pantalla de "falta destinatari" de l'aplicació de sobretaula

A diferència dels treballadors, l'usuari supervisor no es mou de la oficina. Per aquesta raó no ha de notificar avisos, però és la persona a la qual van dirigits els que envien tots els desplaçats. Per tant, a la pestanya "Avisos" se li mostren els avisos que van enviat els treballadors. Aquells avisos amb incidències presenten el fons vermell.

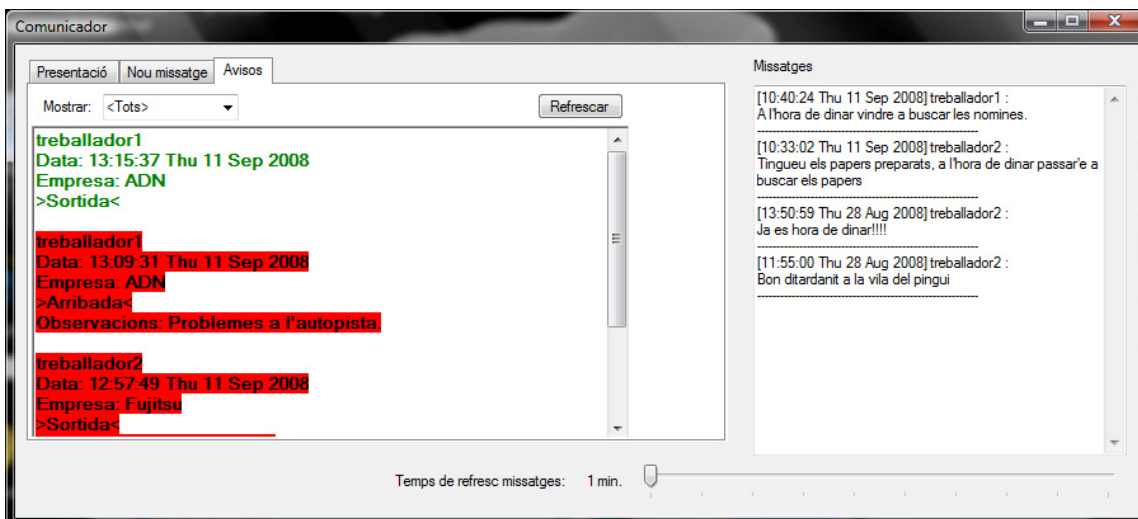


Figura 30 Pantalla d'Avisos de l'aplicació de sobretaula

La pestanya "Avisos" també té un desplegable que permet filtrar els avisos segons l'empleat. Així doncs podem mostrar els avisos d'un empleat en concret, tal i com es pot veure a la Figura 31.

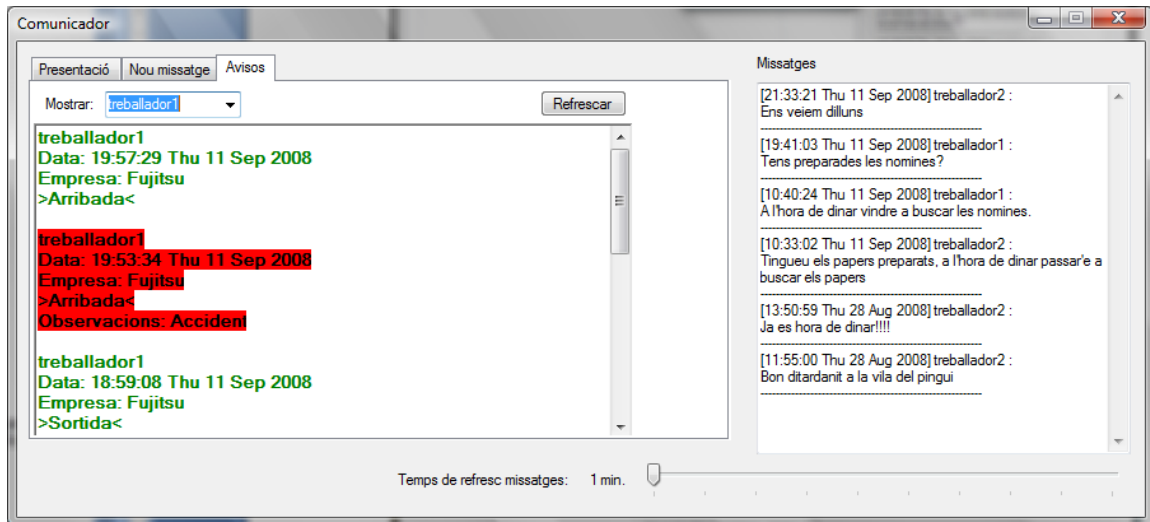


Figura 31 Pantalla d'avisos fent servir filtratge per usuari de l'aplicació de sobretaula.

6 Adaptació a les diferents tecnologies i avaluació econòmica

En aquest apartat es vol quantificar els costos de desenvolupament, no només econòmics sinó també tenir en compte què li caldria conèixer a un programador o quines facilitats tindria segons els seus coneixements.

6.1 Adaptació a les diferents tecnologies

Per posar-me com exemple, de programació per a dispositius mòbils no en sóc pas cap expert, exceptuant un curs de programació Java per dispositius mòbils a la Universitat Autònoma de Barcelona (on s'explicaven els MIDlets). RIM proporciona unes eines pròpies pel desenvolupament de programes per *BlackBerry*, que consisteixen bàsicament en un entorn de programació juntament amb l'emulador. Malgrat tot, el curs m'ha sigut útil ja que els programes desenvolupats per aquests dispositius en essència són MIDlets que poden fer servir funcionalitats esteses proporcionades per les llibreries de *BlackBerry*.

El desenvolupament d'aplicacions per a *Windows Mobile* té l'avantatge de fer servir la tecnologia .NET. Gràcies a això, ja s'hagi fet servir Visual Basic com C# el resultat final és el mateix. Aquest fet permetria poder reciclar programadors d'altre llenguatges amb un mínim de formació. A més també és molt similar el seu entorn de disseny d'aplicacions amb el de les aplicacions per escriptori. El que és important tenir en compte és el fet que el Compact Framework és reduït i algú acostumat a entorns web o de sobretaula trobarà a faltar algunes classes o funcions.

En el desenvolupament per *Android*, el llenguatge que es fa servir és Java, en comptes del JavaMe que és el que es va pensar per dispositius mòbils. Això fa que no s'hagi de tenir tants miraments en la programació ja que no hi ha tantes restriccions. Si que s'ha de tenir en compte que, tot i ser Java, l'estructura del programa no és la mateixa que la d'un applet. L'*Android* té una estructura pròpia pensada per millorar i facilitar el traspàs d'informació. Per això es requereix un cert temps d'aprenentatge per tal d'acostumar-se a aquest nova manera de pensar.

En resum, la creació d'aplicacions per dispositius mòbils no és complicada però s'ha de tenir en compte les diferents opcions (la tinença o no de pantalla tàctil, la tinença o no de teclat QWERTY, etc.) i les limitacions (tot i que cada cop són més potents, no són ordinadors de sobretaula).

6.2 Avaluació econòmica

A diferència de com aconseguir les eines de programació que es fan servir a nivell personal, en una empresa cal estar al màxim dins de la legalitat. Per aquesta raó farem un petit estudi del cost d'aquestes eines o les claus de signatura d'aplicacions en el cas que sigui necessari (tant el desenvolupament per *iPhone* com per aplicacions en *Symbian* no s'inclouen en aquest estudi ja que al no haver-les fet servir no entrem a valorar si serien econòmicament viables):

- Per desenvolupar aplicacions per *BlackBerry*, RIM posa a disposició dels desenvolupadors les eines necessàries de manera gratuïta a la seva web. Només cal pagar per obtenir una clau de registre de les aplicacions que facin ús d'elements possiblement perillosos pel dispositiu, com per exemple les classes pròpies de *BlackBerry* que fan referència a la memòria estàtica. En aquest cas fan pagar un 20\$^{xxiv} per les claus que permetin signar les aplicacions. Aquestes claus són necessàries per signar les aplicacions. És a dir, sense aquestes claus no es poden instal·lar les nostres aplicacions (si fan ús de les classes restringides de *BlackBerry*) en dispositius físics, tot i que no tindrem cap problema per instal·lar-les i provar-les a l'emulador.
- El cas del Visual Studio és diferent, els kits de desenvolupament (SDK) són gratuïts, en canvi l'eina de desenvolupament no ho és ^{xxv}. Tot i que Microsoft posa a disposició dels programadors versions gratuïtes de la seva "suite" Visual Studio (les versions "Express") per desenvolupar aplicacions per a dispositius mòbils cal tenir-ne la versió professional. El preu d'aquesta versió és aproximadament d'uns 1000€^{xxvi}. En el meu cas, al ser estudiant universitari Microsoft, mitjançant la iniciativa DreamSpark, m'ha proporcionat la versió requerida de manera totalment gratuïta.^{xxvii} Val a dir que Visual Studio serveix per programar en molts tipus

d'entorns, no només per dispositius mòbils (l'aplicació d'exemple d'aquest projecte està feta també amb el Visual Studio).

- *Android*, amb la filosofia oberta amb la què ha nascut, no es planteja posar costos al desenvolupament de les aplicacions. L'entorn de programació recomanat és Eclipse que és gratuït, i tant el kit de desenvolupament com el plugin que automatitza les configuracions dels programes en *Android* també ho són.

En resum, la opció més econòmica seria primerament l'*Android*, després *BlackBerry* i per finalitzar *Windows Mobile*. Però no tot és tan bonic, *Android* de moment només disposa d'una versió Beta; encara no hi ha cap dispositiu al mercat que el faci servir, i hi ha la possibilitat de que al sortir-ne algun dispositiu al mercat, aquest no sigui tant atractiu com l'emulador. Per altra banda, *BlackBerry* és una única marca i centrar-se en el desenvolupament per *BlackBerry* i deixar de banda *Windows Mobile* (que només a Espanya hi ha disponibles a prop de 40 models diferents de telèfons) o PocketPcs amb connexió telefònica^{xxviii} seria obviar un part molt amplia del mercat.

7 Conclusions i ampliacions

En aquest apartat intentaré explicar les principals conclusions que he extret d'haver fet aquest projecte.

7.1 Conclusions

Quan vaig començar a fer aquest projecte va significar un repte, ja que tot i la simplicitat que presentava aparentment l'aplicació, el fet de tractar amb entorns reduïts i molt diferents entre ells mateixos n'augmenta molt la complexitat.

El fet d'haver de descartar dues plataformes importants en aquests moments, com són l'*iPhone* i *Symbian* (que és en la que es basen els models de telèfon intel·ligents de la marca Nokia), pot semblar que deixi el treball una mica esbiaixat. Malauradament, en aquests moments el que interessa és aprofitar al màxim els recursos de què es disposa i no entra dins d'aquest pla invertir en maquinari nou (en el cas d'*Apple*) ni en invertir massa temps en aconseguir fer funcionar un entorn poc estable (el cas de *Symbian*).

Per la part de *BlackBerry*, en resum l'empresa RIM posa a disposició dels desenvolupadors de manera gratuïta un entorn de programació complet i molt estable. El desenvolupament d'aplicacions per aquesta plataforma no aporta cap novetat però això facilita la creació d'aplicacions ja que es basa en un entorn conegut i estès però amb algunes millores.

Per la part de *Windows Mobile* l'entorn de programació és molt complet. Per contra, aquest entorn no és gratuït, tot i que si es vol realitzar nous desenvolupaments les eines suplementàries que es necessiten són gratuïtes.

Pel què respecta l'*Android*, el fet d'utilitzar un entorn com l'Eclipse, entorn molt conegut per el desenvolupament de programari, ajudarà i molt a què surtin moltes aplicacions. El problema està en què en aquests moments *Android* encara es troba en una fase de desenvolupament i s'introdueixen canvis dia a dia, s'amplien o redueixen funcionalitats i es canvien temes d'estructura. Aquest fet fa que desenvolupar programari per aquesta

plataforma sigui més aviat per tal de conèixer-la i que quan surti la versió definitiva els programadors tinguin un rodatge i es realitzin aplicacions més ràpidament.

Sobre l'aplicació d'exemple:

S'ha complert els objectius tant de l'aplicació de *BlackBerry* com la de *Windows Mobile*. Totes dues són funcionals i compleixen amb tots els requisits. Malauradament no s'han pogut provar en un entorn real, ja que al enfocar el programa a les versions més noves no disposàvem encara de dispositius compatibles.

En canvi, en l'aplicació per *Android* no he aconseguit la comunicació per Internet tot i estar tot implementat. No he pogut aïllar el problema concret, però tot apunta que sigui en gran part degut a l'emulador i no al meu desenvolupament. Tot i així he aconseguit una aplicació amb *Android* amb la que simulo respostes per veure'n el seu comportament.

7.2 Ampliacions

Donant per suposat que una ampliació seria aconseguir que l'aplicació d'*Android* funcionés, hi ha altres temes més a ampliar.

Primer de tot les comunicacions. L'ús de Twitter ha sigut per poder tenir comunicacions sense necessitat de crear un servidor des de zero. Entenem que per comunicacions d'empresa s'ha de poder garantir un mínim de privacitat, de seguretat, i control d'errors, ja que fent servir Twitter no tenim la seguretat de que el servidor respongui en el temps adient i que s'actualitzi de la manera correcte (durant les proves ha passat que hem enviat un missatge i no hem rebut la actualització). Però per crear un servidor cal tenir en compte diferents aspectes, com ara la plataforma on correrà el servidor (entorn Windows, entorn Unix), o també si el controlarà el client o ho oferirem com un servei i serà la nostra empresa qui tindrà el control del servidor.

A l'abastar més d'una plataforma ha fet que no fos possible aprofundir en excés en l'aplicació en si. L'aplicació, tot i fer el que s'ha plantejat, es pot optimitzar molt. Es pot disminuir el volum de dades que s'envien/reben minimitzant costos. Es podria implementar un servei d'estat del missatge, sabent si el destinatari l'ha llegit. També es podria afegir la funcionalitat d'esborrar missatges, ja que és una cosa que no es va

plantejar però que seria interessant. Tot i que la quantitat de dades que ocupa cada missatge no és gaire, la memòria dels dispositius mòbils és limitada i es podria arribar a omplir.

Annex 1: Índex de taules i figures

Taula 1 Nombre d'unitats de telèfons mòbils de les diferents marques.....	3
Taula 2 Especificacions dels diferents models de <i>Carbide.c++</i>	12
Taula 3 Comparativa de les diferents versions de <i>Windows 6</i>	17
Figura 1 Percentatges de possessió de <i>Symbian</i> (Font www.symbian.com)	11
Figura 2 Esquema de l'estructura del <i>Visual Studio .NET</i>	19
Figura 3 Esquema de l'arquitectura d' <i>Android</i>	23
Figura 4 Entorn de programació de <i>BlackBerry JDE 4.5</i>	28
Figura 5 Desactivació de la cobertura de l'emulador per <i>BlackBerry</i>	29
Figura 6 Representació de diferents models de <i>BlackBerry</i> a l'emulador	29
Figura 7 Entorn de programació de <i>Visual Studio 2008</i>	31
Figura 8 Vista de disseny d'una pantalla de <i>Windows Mobile</i>	31
Figura 9 Emulador de xarxa GSM.....	32
Figura 10 Models "Classic" i "Professional" de l'emulador de dispositius amb <i>Windows</i> ...	33
Figura 11 Exemples de telèfons intel·ligents	34
Figura 12 Vista de programació amb l'arxiu <i>R.java</i>	35
Figura 13 Imatge d'una pantalla amb un arxiu <i>XML</i>	36
Figura 14 Vista de la pantalla en la fase de disseny amb el <i>Visual Studio</i>	36
Figura 15 Emulador d' <i>Android</i>	37
Figura 16 Diagrama de casos d'ús	41

Figura 17 Pantalles de presentació de les aplicacions per dispositius mòbils	53
Figura 18 Pantalles sobre la connexió amb BlackBerry.....	54
Figura 19 Detecció de xarxa amb Windows Mobile	55
Figura 20 Pantalles de nou missatge de dispositius mòbils.....	56
Figura 21 Missatges de alerta en missatges buits.	56
Figura 22 Missatges de confirmació en la aplicació per dispositius mòbils.....	57
Figura 23 Pantalla de llegir missatges dels dispositius mòbils.	58
Figura 24 Pantalles d'enviament d'Avisos dels dispositius mòbils.....	59
Figura 25 Avis de nou missatge en aplicacions de dispositius mòbils.....	59
Figura 26 Pantalla de presentació de l'aplicació de sobretaula	60
Figura 27 Avis de nou missatge a l'aplicació de sobretaula.....	61
Figura 28 Pantalla de nou missatge de l'aplicació de sobretaula.	61
Figura 29. Pantalla de "falta destinatari" de l'aplicació de sobretaula	62
Figura 30 Pantalla d'Avisos de l'aplicació de sobretaula.....	62
Figura 31 Pantalla d'avisos fent servir filtratge per usuari de l'aplicació de sobretaula.....	63

Bibliografia i Referències

Bibliografia i adreces web per el desenvolupament

- Microsoft Mobile Development Handbook, *Andy Wigley*, ed. Microsoft Press
ISBN:9780735623583

- *Android Essentials*, *Chris Haseman*, ed. Apress.
ISBN-13 (electrònic): 978-1-4302-1063-4

- Documentació per programar aplicacions per BlackBerry:

<http://na.blackberry.com/eng/support/docs/subcategories/?userType=21&category=BlackBerry+Java+Application+Development&subCategory=BlackBerry+Java+Development+Environment&url=%2Fsupport%2Fdocs%2Fsubcategories%2F&versionId=46>

APIs:

<http://msdn.microsoft.com/en-us/library/bb158486.aspx>

<http://www.blackberry.com/developers/docs/4.5.0api/index.html>

<http://code.google.com/Android/documentation.html>

Referències

i

<http://www.canalys.com/pr/2008/r2008021.htm>

ii

<http://www.xatakamovil.com/2008/04/17-comparativa-tarifas-internet-movil-operadores-tradicionales>

<http://www.xatakamovil.com/2008/04/18-comparativa-tarifas-internet-movil-operadores-low-cost>

iii

Technical Requirement: Intel processor-based Mac running Mac OS X Leopard

<http://developer.apple.com/iphone/program/>

iv

System Requirements

Windows XP

Microsoft Windows XP Service Pack 2.

Microsoft Visual Studio 2005, Standard Edition or above (***Express Editions are not supported***).

<http://www.microsoft.com/downloads/details.aspx?familyid=06111A3A-A651-4745-88EF-3D48091A390B&displaylang=en>

v

<http://www.symbian.com/about/overview/ownership/ownership.html>

vi

<http://www.symbian.com/phones/>

vii

Document on hi ha una comparació entre el desenvolupament d'aplicacions JavaME i aplicacions natives en C++ per Symbian

http://developer.symbian.com/main/downloads/papers/Native_And_Java_ME_Dev_On_SymbianOS_%20v1.1.pdf

viii

http://www.forum.nokia.com/main/resources/tools_and_sdks/carbide_cpp/

ix

Technical Requirement: Intel processor-based Mac running Mac OS X Leopard

<http://developer.apple.com/iphone/program/>

x

2. Usos permitidos y restricciones de la licencia.

A. Uso único. Esta licencia le permite instalar, utilizar y ejecutar una (1) copia del Software Apple en un único ordenador de marca Apple al mismo tiempo. ***Usted se compromete a no instalar, utilizar o ejecutar el Software Apple en ningún ordenador que no sea de marca Apple o permitir que otras personas lo hagan.*** Esta licencia no permite que el Software Apple se encuentre en más de un ordenador al mismo tiempo ni que se halle disponible a través de una red, en donde pueda ser usado por múltiples ordenadores a la vez.

<http://images.apple.com/legal/sla/docs/macosx105.pdf>

xi

<http://www.intomobile.com/2008/07/02/palm-treo-700wx-does-windows-mobile-6.html>

<http://www.istartedsomething.com/20070808/palm-treo-wm6-release/>

xii

<http://www.sonyericsson.com/cws/products/mobilephones/overview/x1?cc=es&lc=es>

xiii

http://blogs.msdn.com/blogfiles/jasonlan/WindowsLiveWriter/WindowsMobile6whatsthedifferencebetweent_9738/clip_image001.png

xiv

http://pocketpccentral.net/windows_mobile_7_brief.htm

xv

Aquestes són només alguns exemples:

<http://www.error500.net/gphone-rumores>

<http://www.rediff.com/money/2007/aug/24gphone.htm>

xvi

<http://googleblog.blogspot.com/2007/11/wheres-my-gphone.html>

xvii

http://www.openhandsetalliance.com/oha_members.html

xviii

<http://na.blackberry.com/eng/developers/downloads/jde.jsp>

xix

<http://www.microsoft.com/downloads/details.aspx?familyid=06111A3A-A651-4745-88EF-3D48091A390B&displaylang=en>

xx

<http://www.microsoft.com/downloads/details.aspx?familyid=38C46AA8-1DD7-426F-A913-4F370A65A582&displaylang=en>

xxi

<http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/ganymede/R/eclipse-java-ganymede-win32.zip>

xxii

http://code.google.com/android/download_list.html

xxiii

<http://code.google.com/android/intro/installing.html#installingplugin>

xxiv

<http://na.blackberry.com/eng/developers/downloads/api.jsp>

xxv

<http://www.microsoft.com/downloads/details.aspx?FamilyId=06111A3A-A651-4745-88EF-3D48091A390B&displaylang=en#Requirements>

xxvi

http://www.optize.es/servlet/MICROSOFT_VISUAL_STUDIO_2008_PROFESSIONA_341443_optize.html

xxvii

<https://downloads.channel8.msdn.com/>

xxviii

<http://www.microsoft.com/spain/windowsmobile/smartphone/default.mspx>