



**Universitat Autònoma
de Barcelona**

Aplicació de catàlegs virtuals per a tablets amb sistema operatiu Android

Memòria del projecte

d'Enginyeria Tècnica en

Informàtica de Sistemes realitzat per

Xavier Rigau Jimenez

i dirigit per

Daniel Guimarans Serrano

Escola d'Enginyeria

Sabadell, Juny de 2012

El sotasignat, Daniel Guimarans Serrano,
professor de l'Escola d'Enginyeria de la UAB,

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva direcció per

Xavier Rigau Jimenez

I per a que consti firma la present.
Sabadell, Juny de 2012

Signat: Daniel Guimarans Serrano

El sotassinat, Joan Vicent Orenca Serisuelo,
de Tempos 21 (Grupo AtoS),

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva supervisió per

Xavier Rigau Jimenez

I per a que consti firma la present.
Sabadell, Juny de 2012

Signat: Joan Vicent Orenca Serisuelo

Full de resum – Projecte de Fi de Carrera de l'Escola d'Enginyeria

Títol del projecte: Aplicació de catàlegs virtuals per a tablets amb sistema operatiu Android	
Autor: Xavier Rigau Jimenez	Data: Juny de 2012
Tutors: Daniel Guimarans Serrano i Joan Vicent Orenge Serisuelo	
Titulació: Enginyeria Tècnica en Informàtica de Sistemes	
Paraules clau <ul style="list-style-type: none">• Català: catàleg, tablet, Android• Castellà: catálogo, tablet, Android• Anglès: catalog, tablet, Android	
Resum del projecte <ul style="list-style-type: none">• Català:<p>L'objectiu d'aquest projecte de final de carrera a l'empresa és el de desenvolupar una eina de catàlegs virtuals per a dispositius del tipus "tablet". Es desenvoluparà només la part client (la del tablet) i per el moment només serà compatible amb tablets amb la versió 3 d'Android o superior, tot i que no es descarta que en properes versions s'ampliarà per a que funcioni també en dispositius mòbils amb versions anteriors. La part servidora ja es té a l'empresa.</p><p>Aquesta eina ha de servir per a que les empreses l'utilitzin com a suport de ventes, de tal manera que es puguin estalviar els costos que suposa la creació i constant renovació d'un catàleg en paper.</p>• Castellà:<p>El objetivo de este proyecto de final de carrera en la empresa es el de desarrollar una herramienta de catálogos virtuales para dispositivos de tipo "tablet". Se desarrollará solo la parte cliente (la del tablet) y por el momento solo será compatible con tablets con la versión 3 de Android o superior, aun así no se descarta que el próximas versiones se ampliará para que funcione también en dispositivos móviles con versiones anteriores. La parte servidora ya se tiene en la empresa.</p><p>Esta herramienta ha de servir para que las empresas la usen como soporte de vendas, de tal manera que se puedan ahorrar los costes que supone la creación y constante</p>	

renovación de un catálogo en papel.

- Anglès:

The goal of this end of degree project at the company is to develop a virtual catalogs tool for tablet devices. Only the client part (the application that will run on tablets) will be developed and actually it will be compatible just with tablet devices running Android 3 or later, but maybe in future versions it will also be compatible with phones running older Android versions. The server part has been already created by the company.

This tool must be something that business can use to give sales support, so they can save the costs of creating and constantly updating a paper catalog.

Agraïments

Abans de començar amb el projecte, vull agrair a la gent que ha participat o col·laborat en aquest projecte:

A l'Alex De Fuenmayor, la Carlota Royuela i la Margarida Fraide per a crear un disseny molt original i diferenciat.

A Ricard Sala i Joanvi Orenga per a trobar una idea que vam poder convertir en aquest projecte.

A l'empresa en general (Tempos 21), per a brindar-me els recursos necessaris.

A la meva parella, Lolivi, per la seva paciència i per distreure'm en els meus mals moments.

També m'agradaria donar les gràcies al meu pare per tota l'ajuda que m'ha brindat durant aquests mesos. Els teus coneixements em fan millorar dia rere dia. I gràcies per estar sempre quan et necessito.

Per últim, he d'agrair als meus tutors l'ajuda que m'han brindat. A Joanvi, que sempre ha tingut un moment per mi, a pesar de la seva càrrega de treball. I a Daniel Guimarans, que de la mateixa manera, sempre ha trobat els moments per a contestar tots els meus dubtes. Gràcies als dos per guiar-me en aquest projecte de forma admirable.

Índex

1. Introducció	13
1.1. Objectius i motivacions	13
1.2. Estudi de Viabilitat	14
1.2.1. Viabilitat tècnica	14
1.2.2. Viabilitat econòmica	16
1.3. Pla del projecte.....	18
1.3.1. Calendari dels recursos	19
1.3.2. Calendari del projecte	20
1.3.3. Dependències.....	20
1.3.4. Quadre de tasques del projecte	21
1.3.5. Calendari temporal.....	22
1.4. Anàlisi de requeriments	22
1.4.1. Requeriments funcionals.....	22
1.4.2. Requeriments no funcionals.....	23
2. Descripció del projecte.....	25
2.1. Contextualització	25
2.1.1. Informació sobre Android	25
2.1.2. Informació sobre dispositius tipus tablet.....	27
2.1.3. Informació sobre l'e-commerce	28
2.2. Restriccions del sistema	29
2.3. Fonaments i definicions.....	29
2.4. Disseny de l'aplicació	32
2.4.1. Disseny de la interfície d'usuari.....	32
2.4.2. Anàlisi i disseny de l'estructura de classes.....	37
2.5. Implementació	40
2.5.1. Llibreries	40
2.5.2. Persistència de dades.....	41
2.5.3. Comunicació de dades.....	41

2.5.4. Parallax.....	41
2.5.5. Carro de la compra	42
2.5.6. Orientació de la pantalla	42
2.6. Resultat.....	43
3. Conclusions	47
4. Bibliografia.....	49
5. Resum de figures.....	51

1. Introducció

1.1. Objectius i motivacions

L'objectiu d'aquest treball és el de crear una aplicació de catàlegs virtuals per a tabletetes electròniques ("tablets") amb sistema operatiu Android. Aquesta aplicació es crearà amb un conveni de pràctiques remunerades empresa-universitat a Tempos 21 (Grupo AtoS), on porto realitzant pràctiques des de l'Octubre de 2010 com a programador Android.

Tempos 21 és una empresa dedicada a la consultoria de mobilitat, per tant hi ha recursos suficients per a dur el projecte a terme. El responsable de que realitzarà la funció de tutor a l'empresa és en Joan Vicent Orenga, cap de desenvolupament de Tempos 21.

Un cop acabada aquesta aplicació, s'utilitzarà a mode de demostrador per part de l'equip comercial de Tempos 21 per a mostrar-lo a possibles nous clients com a part de les solucions de l'empresa, per tant primerament, no se'n farà un ús diari, tot i que si s'acaba venent a algun client, l'aplicació ha d'estar preparada per a un ús intensiu.

Amb la realització d'aquest projecte m'agradaria assolir nous coneixements i competències, com ara la metodologia a seguir per a la planificació de projectes, ser capaç de gestionar recursos humans i aprendre a desenvolupar per a tablets amb Android, ja que encara no he tingut la oportunitat de realitzar cap projecte amb aquest target de dispositius.

1.2. Estudi de Viabilitat

A continuació s'estudia la viabilitat tècnica i econòmica del projecte per a saber si és rentable dur-lo a terme. S'avaluaran les diverses opcions que hi ha per a crear l'aplicació de catàlegs virtuals i es decidirà la que es consideri que s'adapta millor a les nostres necessitats.

1.2.1. Viabilitat tècnica

A continuació s'exposen les diverses alternatives que hi havia a l'hora de triar la forma de crear l'aplicació de catàlegs virtuals:

Alternativa 1: Empresa externa que realitzi la aplicació.

Contractar una empresa suposa un cost més elevat i a més, es vol crear una eina molt adaptable i fàcilment modificable que es pugui usar en aplicacions pròpies, cosa que complicaria aquesta alternativa, ja que les persones que crearien l'aplicació serien alienes a l'empresa Tempos 21.

Alternativa 2: Crear una web específica per a dispositius tipus "tablet".

Aquesta alternativa arribaria a més dispositius (iPad, tablets amb Windows, etc.), però requereix més temps de desenvolupament, ja que s'haurien d'utilitzar tecnologies que no conec (HTML5, plataformes de webs multi dispositiu, etc.), per tant requeriria també un temps de formació, i a més no es podria integrar amb aplicacions natives.

Alternativa proposada: Desenvolupar l'aplicació a l'empresa.

Aquesta opció serà poc costosa per a l'empresa tant en temps com en diners, ja que és una tecnologia que conec. La contrapartida és que els tablets Android, no són tant fàcils de fer servir per l'usuari final, així que hi haurà d'haver un esforç més elevat durant el disseny de la interfície d'usuari per a que aquesta sigui molt fàcil i intuïtiva de fer servir.

Comparativa de les solucions proposades

Aquesta és la taula comparativa de les opcions descrites:

<i>Alternativa</i>	<i>Costos adquisició</i>	<i>Costos adaptació</i>	<i>Nous recursos</i>	<i>Complexitat</i>	<i>Nivell Integració</i>	<i>Formació</i>
1 (app. externa)	<i>Alts (15.000 – 20.000 €)</i>	<i>Alts</i>	<i>Adaptable</i>	<i>Baixa</i>	<i>Baix</i>	<i>Partners</i>
2 (web)	<i>0 €</i>	<i>Es desconeix</i>	<i>No cal</i>	<i>Alta</i>	<i>Baix</i>	<i>Alta</i>
3 (proposada)	<i>0 €</i>	<i>Pressupost projecte/Mitjans</i>	<i>No cal</i>	<i>Mitjana</i>	<i>Alt</i>	<i>Baixa</i>

Figura 1 Taula de solucions proposades

Com podem veure la primera opció queda descartada ja que no es permet una fàcil integració i el cost d'adquisició és alt.

La segona, tampoc permet una fàcil integració i a més, requereix un temps de formació del qual no es disposa.

Per tant, **es durà a terme l'opció proposada (opció 3)**, ja que és la que millor s'adapta a les necessitats bàsiques del projecte i l'empresa ha arribat a la conclusió que és la més viable de les tres.

1.2.2. Viabilitat econòmica

Aquest apartat conté un resum amb els costos econòmics que tindrà la realització del projecte.

Recursos del projecte

Aquesta és la taula de recursos humans del projecte:

Nom del recurs	Tipus	Capacitat màxima	Tassa estàndard
Xavier Rigau	<i>Trabajo</i>	100%	15,00 €/hora
Cap de Projecte	<i>Trabajo</i>	100%	47,00 €/hora
Dissenyador	<i>Trabajo</i>	100%	28,00 €/hora
Responsable de proves	<i>de Trabajo</i>	100%	22,00 €/hora

Figura 2 Taula de recursos del projecte

Com podem veure, comptarem amb un Cap de Projecte (el tutor del projecte a l'empresa), un dissenyador/a, un responsable de proves i un desenvolupador (jo).

Estimació cost del personal

Aquesta és la taula de costos dels recursos humans:

<i>Recurs</i>	<i>Cost</i>	<i>Treball</i>
<i>Xavier Rigau</i>	5178 €	345'2 hores
<i>Cap de projecte</i>	564 €	12 hores
<i>Dissenyador</i>	1232 €	44 hores
<i>Responsable de proves</i>	950'4 €	43'2 hores
TOTAL	7924'4 €	444'4 hores

Figura 3 Taula de costos dels recursos humans

Estimació cost dels recursos

Aquesta és la taula de costos dels recursos materials:

<i>Recurs</i>	<i>Cost amortització</i>	<i>Cost unitari</i>	<i>Període amortització</i>	<i>Període utilització</i>
<i>Amortització PC programador</i>	66'67 €	800 €	36 m.	3 m.
<i>Amortització MS Office</i>	20'8 €	250 €	36 m.	3 m.
<i>Amortització MS Project</i>	30 €	360 €	36 m.	3 m.
<i>Amortització "tablet" Motorola Xoom</i>	33'33 €	400 €	36 m.	3 m.
<i>Amortització PC Dissenyador</i>	50 €	1800 €	36 m.	1 m.
<i>Amortització Photoshop</i>	5'5 €	200 €	36 m.	1 m.
TOTAL	206'36 €			

Figura 4 Taula de costos dels recursos físics

La resta d'aplicacions que s'utilitzen per a dur a terme aquest projecte són gratuïtes, per tant no suposen cap cost a amortitzar. Aquestes eines són:

- **Eclipse i Plugin Android Development Tools (ADT).** Eina per a la programació de diferents llenguatges i plugin que facilita la creació d'aplicacions per Android en Eclipse.
- **Android Software Development Kit (SDK) Tools.** Conjunt d'elements (documentació, tutorials, eines, emuladors, etc.) que permeten crear millors aplicacions per a Android.
- **Mantis Bug Tracker.** Servei que permet tenir una gestió d'errors de les aplicacions a mode de tiquets.
- **Servidor de Subversion.** Repositori de control de versions de les aplicacions i projectes que permet la col·laboració de més d'un membre a un mateix projecte.

Resum i anàlisi cost

Havent trobat tots els costos, obtenim el següent resum:

<i>Tipus de cost</i>	<i>Quantitat</i>
<i>Cost de desenvolupament del projecte</i>	<i>7924'4 €</i>
<i>Cost d'amortització del material</i>	<i>206'36 €</i>
<i>TOTAL</i>	<i>8130'76 €</i>

Figura 5 Taula re resum de costos

Per tant, segueix sent viable, ja que els costos són acceptables per al tipus de projecte que es realitzarà.

1.3. Pla del projecte

Nota: Al començar aquest projecte em vaig marcar l'objectiu d'entregar-lo a la convocatòria de Febrer, però per motius aliens a la realització del projecte l'entrega es realitza a la convocatòria de Juny. És per això que la planificació del projecte només es té en compte el primer quadrimestre del curs acadèmic.

Aquest és el pla d'aquest projecte, per tant s'hi recull el conjunt d'activitats que permeten desenvolupar, executar i controlar el projecte.

El pla del projecte inclou les tasques a realitzar amb la durada de cadascuna d'elles i el calendari del projecte.

S'ha utilitzat un metodologia lineal en l'elaboració del pla i per al desenvolupament s'utilitzarà una metodologia de software àgil i iteratiu com es Scrum, de tal forma que el desenvolupament no començarà fins que el disseny no estigui aprovat i diàriament s'aniran fent reunions curtes (entre 3 i 5 minuts) amb el Cap de Projecte, que realitzarà la funció de Scrum Master. També hi haurà reunions setmanals (cada divendres, amb durada d'entre 15 a 20 minuts), on també hi assistirà el dissenyador.

Com he comentat anteriorment, per a la planificació del projecte s'utilitzarà el Microsoft Project (eina de pagament), per al control de versions del codi font s'utilitzarà un servidor de Subversion i per a la gestió d'errors es farà servir el Mantis Bug Tracker (ambdues eines gratuïtes).

1.3.1. Calendari dels recursos

Un cop dividit el projecte en tasques, assignem aquestes tasques als recursos humans del projecte. El calendari resultant és el següent:


Id		Nombre del recurso	Trabajo				
1		Xavier Rigau	345,2 horas				
	<i>Identificador</i>	<i>Nombre de tarea</i>	<i>Unidades</i>	<i>Trabajo</i>	<i>Retraso</i>	<i>Comienzo</i>	
	3	Estudi de l'estat de l'art	100%	18 horas	0 días	lun 17/10/11	
	4	Planificació	100%	12 horas	0 días	vie 21/10/11	
	5	Pluja d'idees	100%	4 horas	0 días	mié 26/10/11	
	6	Estudi de viabilitat	25%	4 horas	0 días	vie 28/10/11	
	12	Anàlisis Funcional i Tècnic	100%	24 horas	0 días	jue 03/11/11	
	13	Disseny de la BBDD	100%	16 horas	0 días	vie 11/11/11	
	14	Implementació	100%	144 horas	0 días	jue 17/11/11	
	15	Documentació del codi font	100%	16 horas	0 días	vie 06/01/12	
	20	Implementació	100%	16 horas	0 días	mié 01/02/12	
	21	Documentació	100%	8 horas	0 días	mar 07/02/12	
	25	Redacció de la memòria	75%	36 horas	0 días	lun 16/01/12	
	28	Preparació de l'exposició	100%	24 horas	0 días	vie 10/02/12	
	7	Revisio necessitats servidor	100%	4 horas	0 días	jue 27/10/11	
	16	Pla de proves	100%	8 horas	0 días	jue 12/01/12	
	17	Bancs de proves	25%	7,2 horas	0 días	lun 16/01/12	
	29	Exposició oral PFC (08/02/2012 - 14/02/2012)	100%	0 horas	0 días	mié 08/02/12	
	22	Pla de proves	100%	4 horas	0 días	jue 09/02/12	
2		Cap de Projecte	12 horas				
	<i>Identificador</i>	<i>Nombre de tarea</i>	<i>Unidades</i>	<i>Trabajo</i>	<i>Retraso</i>	<i>Comienzo</i>	<i>Fin</i>
	5	Pluja d'idees	100%	4 horas	0 días	mié 26/10/11	mié 26/10/11
	7	Revisio necessitats servidor	100%	8 horas	0 días	mié 26/10/11	jue 27/10/11
3		Dissenyador	44 horas				
	<i>Identificador</i>	<i>Nombre de tarea</i>	<i>Unidades</i>	<i>Trabajo</i>	<i>Retraso</i>	<i>Comienzo</i>	
	10	Disseny interacció/wireframes	100%	20 horas	0 días	mié 26/10/11	
	5	Pluja d'idees	100%	4 horas	0 días	mié 26/10/11	
	11	Disseny Visual	100%	20 horas	0 días	lun 31/10/11	
4		Responsable de proves	43,2 horas				
	<i>Identificador</i>	<i>Nombre de tarea</i>	<i>Unidades</i>	<i>Trabajo</i>	<i>Retraso</i>	<i>Comienzo</i>	<i>Fin</i>
	17	Bancs de proves	100%	19,2 horas	0 días	vie 13/01/12	mar 17/01/12
	23	Bancs de proves	100%	24 horas	0 días	jue 09/02/12	mar 14/02/12

Figura 6 Calendari de tasques

1.3.2. Calendari del projecte

Resum del calendari del projecte:

- **Calendari del projecte:** El projecte es desenvoluparà d'octubre de 2011 a febrer de 2012.
- **Data començament:** 17 d'octubre de 2011
- **Data finalització:** 8 de febrer de 2012

1.3.3. Dependències

Excepte en dos casos puntuals, totes les fases es desenvolupen utilitzant un model lineal, per tant:

- Cada fase no es comença fins que no s'ha completat la fase anterior.
- En la fase de desenvolupament es preveu un model àgil de tal manera que el disseny, el desenvolupament i el test segueixin un model iteratiu.

1.3.4. Quadre de tasques del projecte

Aquesta és la taula extreta del MS Project amb l'inici i final de les tasques d'aquest projecte:

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1	☐ PFC Xavi Rigau	89,5 días	lun 17/10/11	vie 17/02/12	
2	☐ Estudis previs	12,5 días	lun 17/10/11	mié 02/11/11	
3	Estudi de l'estat de l'art	3 días	lun 17/10/11	lun 31/10/11	
4	Planificació	1,5 días	vie 21/10/11	mar 25/10/11	
5	Pluja d'idees	4 horas	mié 26/10/11	mié 26/10/11	
6	Estudi de viabilitat	2 días	vie 28/10/11	mié 02/11/11	5
7	Revisio necessitats servidor	1 día	mié 26/10/11	jue 27/10/11	5
8					
9	☐ Aplicació Android	64,6 días	mié 26/10/11	mié 25/01/12	
10	Disseny interacció/wireframes	2,5 días	mié 26/10/11	vie 28/10/11	5
11	Disseny Visual	2,5 días	lun 31/10/11	mié 02/11/11	10
12	Anàlisis Funcional I Tècnic	3 días	jue 03/11/11	jue 10/11/11	10FC-3 días
13	Disseny de la BBDD	2 días	vie 11/11/11	mié 16/11/11	12
14	Implementació	18 días	jue 17/11/11	jue 05/01/12	13
15	Documentació del codi font	2 días	vie 06/01/12	mié 11/01/12	14
16	Pla de proves	1 día	jue 12/01/12	vie 13/01/12	15
17	Bancs de proves	5 días	vie 13/01/12	mié 25/01/12	16
18					
19	☐ Aplicació del servidor	9,5 días	mié 01/02/12	mar 14/02/12	
20	Implementació	2 días	mié 01/02/12	lun 06/02/12	7
21	Documentació	1 día	mar 07/02/12	mié 08/02/12	20
22	Pla de proves	4 horas	jue 09/02/12	jue 09/02/12	21
23	Bancs de proves	3 días	jue 09/02/12	mar 14/02/12	22
24					
25	Redacció de la memòria	6 días	lun 16/01/12	mar 31/01/12	16
26	Entrega memòria del PFC	0 días	mar 31/01/12	mar 31/01/12	25
27					
28	Preparació de l'exposició	3 días	vie 10/02/12	vie 17/02/12	26
29	Exposició oral PFC (08/02/2012 - 14/02/2012)	0 días	mié 08/02/12	mié 08/02/12	

Figura 7 Taula de durada de les tasques

1.3.5. Calendari temporal

A continuació hi ha una imatge del diagrama de Gantt de les activitats:

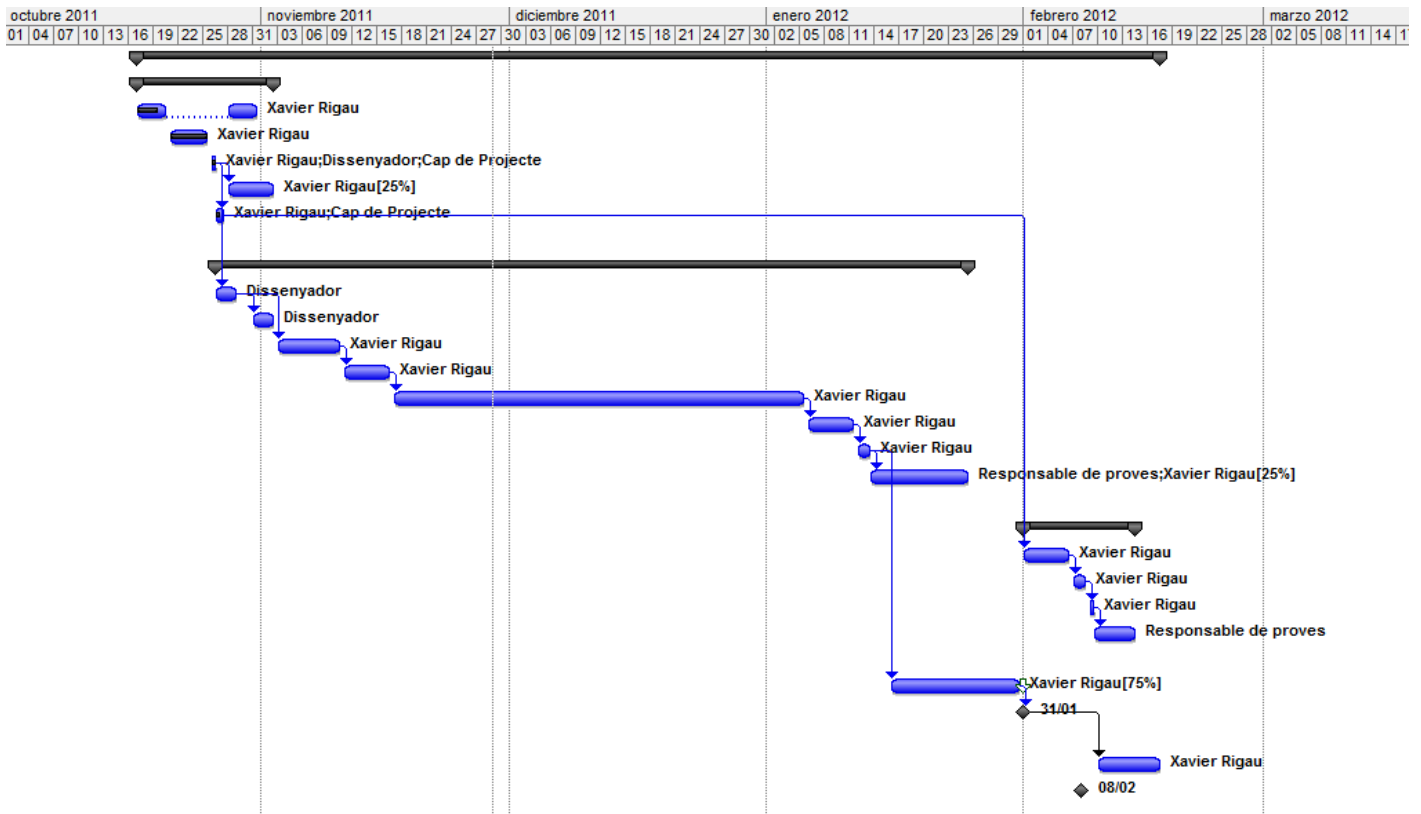


Figura 8 Diagrama de Gantt d'activitats

1.4. Anàlisis de requeriments

Seguidament veurem l'anàlisi de requeriments funcionals i no funcionals de l'aplicació de catàlegs virtuals que s'han proposat. Com que serà un demostrador, aquestes llistes de requeriments podran ser modificades posteriorment (per a clients que estiguin interessats a comprar aquest catàleg, per exemple), però per a la realització d'aquest projecte es mantindran els descrits a continuació.

1.4.1. Requeriments funcionals

L'aplicació haurà de servir per a suport de vendes, per tant els requisits funcionals són:

- S'haurà de poder navegar per un catàleg de productes.
- Cal poder veure els detalls dels productes.

- El catàleg de productes s'ha de poder modificar i veure aquests canvis a l'aplicació.
- Els productes estaran dividits per categories.
- Ha de funcionar sense tenir connexió a Internet.
- L'usuari tindrà una llista de la compra a la que podrà afegir o treure productes.
- Cerca de productes i filtrat de productes.

1.4.2. Requeriments no funcionals

Dels requisits demanats, s'han identificat que els següents són no funcionals:

- La navegació de productes serà diferent depenent si el tablet està en mode vertical o horitzontal.
- El venedor podrà navegar pels productes en una llista d'elements.
- El disseny ha de ser atractiu a la vista, mantenint el branding de MobiRet (colors grisos).
- Al iniciar l'aplicació es veurà una "Splash screen" amb el logotip de l'aplicació.
- El mode d'afegir productes a la llista de la compra ha de ser del tipus arrossegar i deixar o "drag and drop" amb els productes.
- La barra de cerca apareixerà a la part superior de la pantalla.

2. Descripció del projecte

2.1. Contextualització

El món d'aplicacions mòbils és molt canviant, i (com en la informàtica en general) cada cop més s'intenta abaratir costos i temps de desenvolupament, tot creant mòduls flexibles que permetin, mitjançant petites modificacions i adaptacions, la creació ràpida d'aplicacions.

Així, s'ha decidit que el resultat d'aquest projecte ha de ser un codi font moduable, que sigui fàcilment integrable i que es pugui adaptar amb poc esforç a diferents aplicacions.

Aleshores, l'empresa vol que es desenvolupi aquest projecte com un mòdul que pugui funcionar sol, però que alhora es sigui capaç d'integrar-se amb altres aplicacions de forma que es vendrà com a solució per a catàlegs virtuals amb un cost d'adaptació molt baix.

Actualment l'empresa no disposa de cap producte amb aquestes característiques, i tot i que ja s'han creat aplicacions amb catàlegs de productes, tots han sigut específics per una aplicació en concret. Per tant, aquests catàlegs són poc aprofitables i s'ha de reescriure molt de codi cada vegada.

Aquesta aplicació formarà part d'un producte creat per Tempos 21 anomenat MobiRet (Mobile Retail), que és una solució del tipus e-commerce, principalment enfocada al sector mòbil. MobiRet s'ha implantat a Condis Supermercats recentment, demostrant el seu gran potencial, per tant, la creació d'un nou mòdul a MobiRet enfocat al món tablet servirà per a millorar el conjunt MobiRet significativament.

2.1.1. Informació sobre Android

Android és un conjunt de programari per a telèfons mòbil que inclou un sistema operatiu, programari intermediari i aplicacions. Google Inc. va comprar el desenvolupador inicial del programari, Android Inc., el 2005. El sistema operatiu per a mòbils d'Android es basa en una versió modificada del nucli Linux. Google i altres membres de la Open Handset Alliance van col·laborar en el desenvolupament i llançament de l'Android. L'Android Open Source Project (AOSP) té l'objectiu de mantenir i continuar desenvolupant l'Android. El sistema operatiu Android és la plataforma per a smartphones amb més vendes.

Android té una gran comunitat de desenvolupadors que escriuen aplicacions ("apps") que estenen les funcions dels dispositius. Hi ha actualment unes 300.000 aplicacions disponibles per a l'Android. Google Play, prèviament conegut com a Android Market, és el dipòsit d'aplicacions en línia oficial de Google, tot i que també es poden descarregar per llocs web de tercers. Els desenvolupadors escriuen principalment en Java, controlant el dispositiu mitjançant llibreries de Java desenvolupades per Google.

La presentació de l'Android el 5 de novembre de 2007 es va anunciar amb la fundació de la Open Handset Alliance, un consorci de 80 empreses de maquinari, programari i telecomunicacions amb l'objectiu de crear estàndards oberts per a dispositius mòbil. Google va llançar gran part del codi de l'Android sota la Llicència Apache, una llicència de programari lliure oberta.



Figura 9 Logotip d'Android

Característiques remarcables d'Android:

- Màquina Virtual de Java Dalvik. Cada procés corre en la seva màquina virtual i és independent de la resta.
- Possibilitat d'escriure aplicacions en codi natiu C/C++.
- Gran capacitat per a multitasca.
- Gràfics 2D i 3D en OpenGL ES 1.0 i 2.0.
- Bases de dades en SQLite.
- Capacitat d'executar scripts en altres llenguatges (Python, PHP, etc.) mitjançant SL4A (Scripting Layer for Android).
- Característiques extra segons el dispositiu:
 - Sensor GPS i/o AGPS.
 - Connectivitat 2G, 3G, Bluetooth, NFC i/o Wifi.
 - Càmera darrera i/o frontal.
 - Acceleròmetre, giroscopi, brúixola, etc.
 - Sensor de llum i/o d'humitat.
 - Altres (en el futur aniran apareixent més complements, tot i que Android es pot connectar a dispositius tipus Arduino per a augmentar les possibilitats que un dispositiu d'aquestes característiques ens ofereix).

2.1.2. Informació sobre dispositius tipus tablet

Una tauleta tàctil (de l'anglès: tablet o tablet computer) és un tipus d'ordinador portàtil, més gran que un smartphone o una PDA, sent una evolució directe d'ambdós fins el punt d'emprar el mateix sistema operatiu. Forma una única peça amb una pantalla tàctil (resistiva o multi-touch) amb la qual s'interactua primàriament amb els dits (multi-touch) o amb un estilet (passiu o actiu), sense necessitat de teclat físic ni ratolí . Aquests últims són reemplaçats per un teclat virtual i, en determinats models, per un mini-trackball integrat en una de les vores de la pantalla.

El terme pot aplicar-se a una varietat de formats que difereixen en la posició de la pantalla respecte a unes tecles o botons. El format estàndard es del tipus pissarra (slate) i no té teclat integrat encara que pot es pot connectar a un teclat sense fils (per ex., Bluetooth) o mitjançant un cable USB (molts sistemes operatius reconeixen directament teclats i ratolins USB). Un altre format és el portàtil convertible, que disposa d'un teclat físic que gira sobre una frontissa o pot lliscar sota la pantalla. Un tercer format, anomenat híbrid (com el HP Compaq TC1100), disposa d'un teclat físic, però pot separar-se'n per comportar-se com una pissarra. Finalment els Booklets inclouen dues pantalles, almenys una d'elles tàctil, mostrant-hi un teclat virtual.

Els primers exemples del concepte tauleta d'informació es van originar principalment com a prototips i idees conceptuals, dels quals el més prominent va ser el Dynabook d'Alan Kay en l'any 1972. Els primers dispositius electrònics portàtils basats en el concepte van aparèixer a la fi del segle 20. Durant la dècada del 2000 Microsoft va llançar el Microsoft Tablet PC que va tenir relativament poc èxit encara que va aconseguir crear un nínxol de mercat en hospitals i negocis mòbils (per ex., Forces de venda). Finalment el 2010 Apple Inc presenta el iPad, basat en el seu reeixit iPhone, aconseguint un gran èxit comercial en proveir per fi de la interfície adequada.



Figura 10 Diferents dispositius tablet

Avui dia les tauletes utilitzen majoritàriament un sistema operatiu dissenyat amb la mobilitat en ment (iOS d'Apple, l'open Android, el Windows Phone i el minoritari Symbian provenen del camp dels smartphone, on es reparteixen el mercat; MeeGo i HP webOS provenen del món PDA) deixant de banda els que empren Windows Tablet PC, pensats més amb l'ordinador d'escriptori en ment.

2.1.3. Informació sobre l'e-commerce

El comerç electrònic, també conegut com e-commerce (electronic commerce en anglès), consisteix en la compra i venda de productes o de serveis a través de mitjans electrònics, com ara Internet i altres xarxes informàtiques. Originalment el terme s'aplicava a la realització de transaccions mitjançant mitjans electrònics com ara l'Intercanvi electrònic de dades, però amb l'adveniment de la Internet i la World Wide Web a mitjans dels anys 90 va començar a referir-se principalment a la venda de béns i serveis a través d'Internet, usant com a forma de pagament mitjans electrònics, com ara la targeta de crèdit.

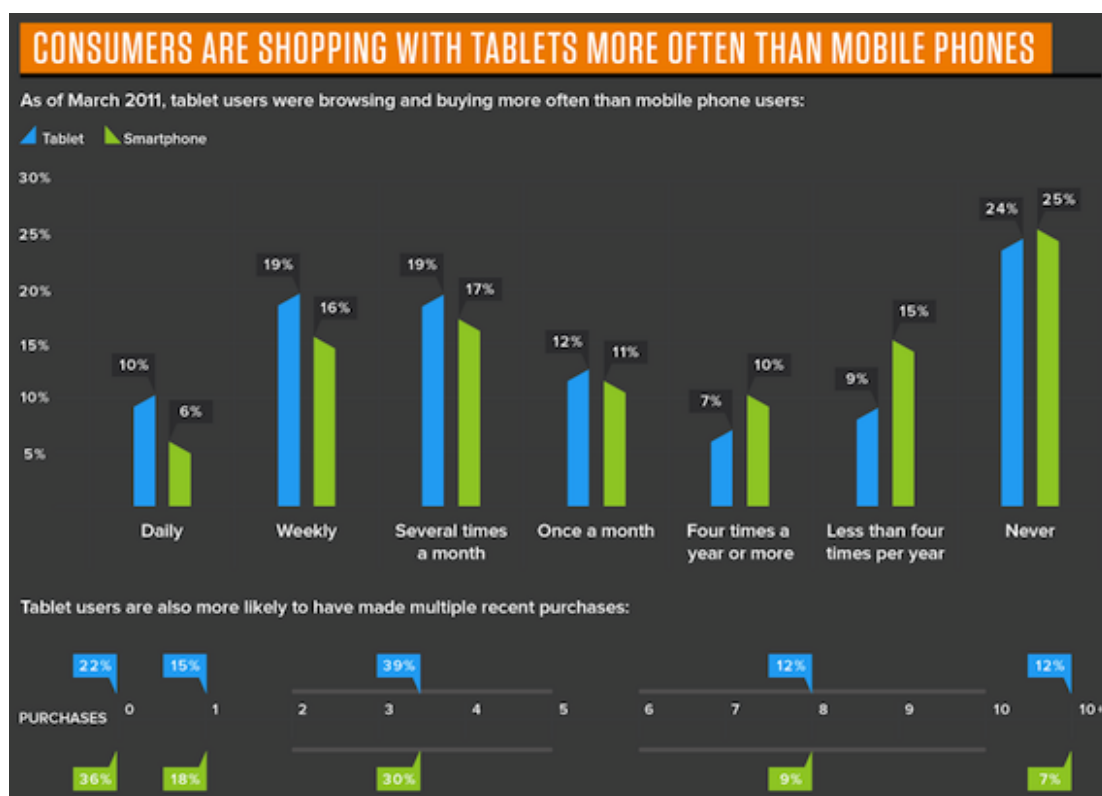


Figura 11 Gràfica comparativa de les compres des de telèfons i des de tablets

La quantitat de comerç duta a terme electrònicament ha crescut de manera extraordinària a causa de Internet. Una gran varietat de comerç es realitza d'aquesta manera, estimulant la creació i utilització d'innovacions com la transferència de fons electrònica, l'administració de cadenes de subministrament, el màrqueting a Internet, el processat de transaccions en línia (OLTP), l'intercanvi electrònic de dades (EDI), els sistemes d'administració de l'inventari i els sistemes automatitzats de recollida de dades. La major part del comerç electrònic consisteix en la compra i venda de productes o serveis entre persones i empreses, però un percentatge considerable del comerç electrònic consisteix en l'adquisició d'articles virtuals (software i derivats en la seva majoria), com ara l'accés a contingut "premium" d'un lloc web.

2.2. Restriccions del sistema

Abans de començar, cal definir quines seran les restriccions que tindrà el sistema, per a tenir-los en compte durant el desenvolupament. Aquesta és la llista resultant de restriccions:

- L'aplicació ha de córrer en tots els dispositius tablet que portin sistema operatiu Android 3.x (Honeycomb) o Android 4.x (Ice Cream Sandwich).
- Les proves i demostracions es realitzaran en un tablet Motorola Xoom amb Android Honeycomb 3.1, per tant els dissenys es faran basats en aquest dispositiu.

2.3. Fonaments i definicions

Tot seguit esmentaré i definiré alguns dels components d'Android més importants que faig servir per a poder entendre el codi font de l'aplicació de catàlegs virtuals més fàcilment:

- **Activity.** Aquest és el component més utilitzat en el desenvolupament natiu per Android. Bàsicament, representa una pantalla de la nostra aplicació. Ho podem definir com a un "element" que l'usuari pot fer en un moment determinat dintre de la nostra aplicació. Una Activity o Activitat té el seu cicle de vida que va canviant segons certs factors.

Aquesta imatge descriu els diferents estats d'una Activity:

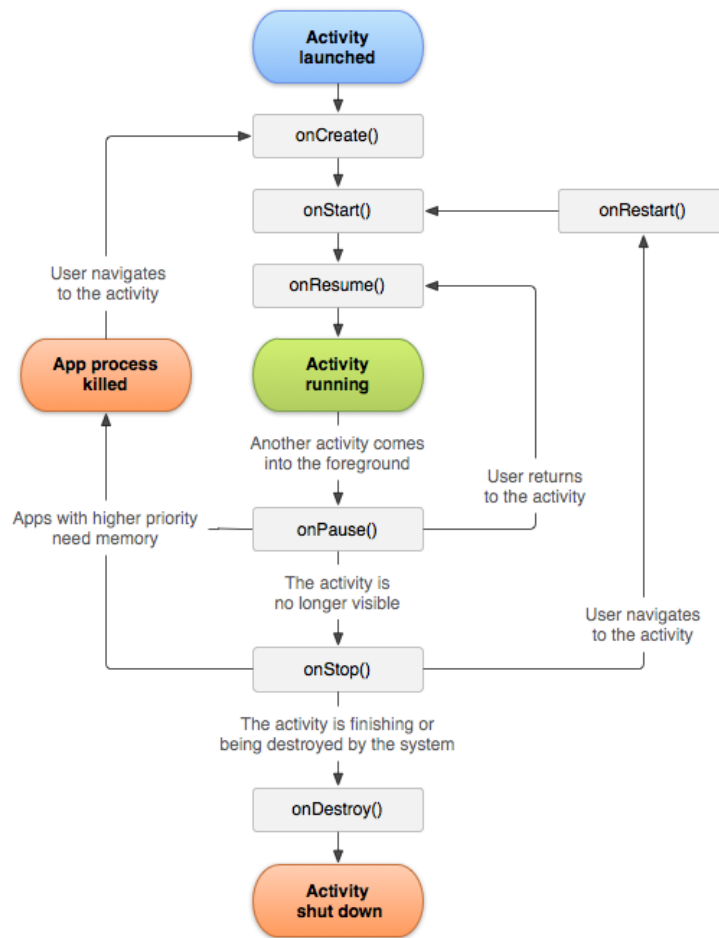


Figura 12 Cicle de vida d'una Activity

- **Fragment.** Un Fragment es pot traduir en “una part” d'una Activity. Es van introduir a partir d'Android 3.0, tot i que hi ha una llibreria de compatibilitat que permet utilitzar-los en versions anteriors. Estan pensats per a modelitzar el codi, sobre tot en pantalles grans on hi ha diversos components gràfics i no gràfics en la mateixa pantalla. Gràcies als Fragments, tenim elements amb funcionalitats i codi reutilitzable, per tant, cal intentar que un Fragment es centri en tenir una funcionalitat concreta i no més. Una Activity pot contenir un o més Fragments i aquests poden tenir la seva pròpia vista (llistes, mapes, etc.) o no (accés a Bases de Dades, tasques en segon pla, etc.). Un Fragment també té un cicle de vida i és semblant al d'una Activity amb petits canvis.
- **Intent.** Podem definir un Intent com “la descripció d'una operació que es durà a terme”. Aquesta operació pot ser de diversos tipus, però el més comú és usar-lo per a obrir Activities. Aquest permet el pas d'informació d'una Activitat a una

altra d'una forma molt fàcil (però també limitada). Normalment s'utilitzarà per a passar objectes de tipus simple o objectes serialitzables.

- **SharedPreferences.** Permet desar informació d'un nombre limitat de tipus d'objectes (enters, cadenes de text, llistes de cadenes de text, etc.) d'una forma molt senzilla a mode de preferències. També permet editar aquests valors de preferències, esborrar-los i afegir-ne de nous.
- **Canvas.** El Canvas és un element bàsic de dibuixat per pantalla. Ens ofereix uns mètodes per a dibuixar imatges, punts, línies, formes, etc. I tot el que dibuixem en un Canvas es veurà per pantalla. Les Views utilitzen aquest element per a dibuixar-se i aparèixer per pantalla, així com per actualitzar el seu estat (per quan l'usuari prem la vista, o l'arrossega, etc.)
- **View.** Classe base que representa un component visual que es pot mostrar per pantalla mitjançant un Canvas. Una View (o Vista) ocupa un rectangle en la pantalla i ha de ser capaç de dibuixar-se i reaccionar en funció de la interacció que hi té l'usuari. Tots els components que es mostren per pantalla son una extensió de View (ListView, ScrollView, ViewGroup, Layouts, etc.) amb funcionalitats pròpies que els diferencien als uns dels altres. Un ListView és un element molt utilitzat que permet mostrar un llistat d'elements servits per una font qualsevol que es defineix a través dels ListAdapters.
- **ListAdapter o Adapter.** Aquest element actua de pont entre un ListView i les dades que es mostraran en aquest ListView. L'Adapter és responsable de recuperar aquestes dades i crear les files del llistat amb la informació corresponent per a cadascuna d'elles.
- **AsyncTask.** És una Tasca Asíncrona que el programador defineix. És a dir que ens permet realitzar una tasca de mitja-llarga durada sense interferir amb el funcionament de l'aplicació mentre aquesta tasca es duu a terme. D'aquesta manera impedim que el fil d'execució de l'aplicació es vegi afectat i no sofreixi interrupcions. Quan acaba la AsyncTask, podem desencadenar accions (mostrar els resultats en un llistat, mostrar un missatge d'error o de completat, etc.).
- **AndroidManifest.** És un fitxer que té tota aplicació Android en el qual es defineixen els components que tindrà l'aplicació, els elements de hardware als que necessita accedir, la compatibilitat de versions, etc. En el nostre cas definim que necessitem accedir a Internet, per exemple, igual que hem de definir totes les Activitats de les quals es compona la nostra aplicació.
- **Garbage Collector.** És un mecanisme que autogestiona la memòria lliure d'un sistema operatiu o entorn d'execució. S'encarrega d'alliberar memòria (de

finalitzar objectes) quan detecta que el nivell de memòria ocupada d'una aplicació ha augmentat. Aleshores busca objectes que no estiguin utilitzant-se o elements que puguin produir un consum excessiu de memòria i els treu de memòria. En Android el Garbage Collector s'executa en el mateix fil d'execució que la resta de la interfície d'usuari, per tant cal anar en compte, ja que alhora de crear components que tindran que tenir una velocitat de dibuixat alta, com per exemple els jocs, si s'executa el Garbage Collector causarà pauses d'alguns milisegons durant l'execució.

- **SQLite.** Sistema Gestor de Bases de Dades que porta incorporat Android (igual que moltes altres plataformes mòbils). Ofereix un gran potencial amb un consum de recursos baix, així que és una bona solució per a Bases de Dades simples.



Figura 13 Logotip d'SQLite

2.4. Disseny de l'aplicació

El disseny de l'aplicació, es basa en dues parts fonamentals que es realitzaran en el següent ordre:

1. Disseny de la interfície d'usuari. Es farà tenint en compte els requeriments abans descrits, a més de tenir presents els principis de disseny ergonòmic, interacció natural d'usuari i experiència d'usuari en tablet.
2. Anàlisi i disseny de l'estructura de classes. Per a realitzar aquesta part, partim de l'anterior disseny descrit al punt 1.

2.4.1. Disseny de la interfície d'usuari

Un dissenyador de Tempos 21 ha dissenyat la part gràfica de l'aplicació. S'ha introduït un element que pot complicar una mica més la implementació, però que alhora, millorarà la experiència d'usuari i farà que el catàleg sigui molt més atractiu. Com que els clients dels venedors que portin els catàlegs virtuals també tindran que interactuar amb el catàleg, el dissenyador ha decidit que en mode horitzontal, el catàleg tindrà un aspecte visual molt atractiu i amb menys opcions, i en mode vertical, l'aspecte visual serà l'estàndard per a aquest tipus d'aplicacions, però amb més opcions i una navegació molt més ràpida i fàcil que en mode horitzontal.

Es compona de 6 pantalles:

1. **Splash Screen.** Aquesta és la primera pantalla que veurem al obrir l'aplicació de catàlegs virtuals. Aquest element s'acostuma a utilitzar en moltes aplicacions, perquè ajuda a incrementar el la marca (o "branding") de l'aplicació, ja que, fonamentalment, aquesta pantalla només contindrà un fons amb colors de l'aplicació, el nom de l'aplicació i possiblement el logotip de l'empresa. Mentre mostrem aquest splash screen, aprofitem a més per a carregar les dades del catàleg, comprovar si tenim la última versió del llistat de productes, etc.
2. **Llistat de categories.** Un cop deixem de mostrar el splash screen (quan hem acabat d'inicialitzar l'aplicació), apareix la llista de categories. Aquí només podem fer dues accions:
 - a. Entrar en una categoria.
 - b. Realitzar una cerca per paraules.
3. **Llistat de productes.** Ambdues accions ens porten a la mateixa pantalla que és la llista de productes trobats en funció dels paràmetres de la cerca (en un cas seran paraules clau i en l'altre cas un identificador de categoria). El contingut d'aquesta pantalla dependrà de l'orientació del tablet:
 - a. Mode vertical (o "portrait"): A més de veure un llistat amb els resultats de productes de la cerca que hem realitzat, tenim una altra llista que funciona a mode de llista de la compra o carro de la compra (al qual hi podem afegir, editar o treure productes) i per últim, una part de filtres amb barra de cerca per text que ens permetrà refinar el nombre de resultats obtinguts.
 - b. Si tenim el tablet en mode horitzontal (o "landscape") ens apareix un carrusel a mode de galeria d'imatges amb les fotografies i els noms dels productes que tindriem com a resultats de cerca. Aquest carrusel serà un Parallax (veure apartat "Parallax" en la secció "Implementació").
4. **Detall de producte.** Quan fem click a un dels productes veurem el detall del producte. Aquesta pantalla conté la informació referent al producte que hem seleccionat, com per exemple la descripció i imatges d'aquest producte. Igual que a l'apartat anterior, pot aparèixer en pantalla de dues maneres:
 - a. **Vertical.** Ens apareix una finestra flotant que es desplega des d'un dels costats de la pantalla.

- b. **Horizontal.** En aquest cas també apareix com a finestra flotant a sobre del contingut que estem veient. A més, la informació que s'hi mostra és la mateixa que en el cas anterior però organitzada de forma diferent. Com a diferència del mode vertical, en aquest cas apareix un botó per a anar a la pantalla de comparar productes.
5. **Comparar productes.** Aquest apartat ens permet comparar 4 productes dividits en 4 columnes, tot veient la informació resumida d'aquests en la mateixa pantalla. En qualsevol moment podem fer polsar els botons que apareixen en la part inferior de cada columna per a modificar els productes que es compararan en aquest apartat.
6. **Selecció de productes a comparar.** Aquí, veiem tota la llista de productes i podem seleccionar-ne un per a que ens aparegui en la pantalla de comparació de productes. Un cop seleccionat el producte, aquesta pantalla es tanca i la comparativa ja s'ha actualitzat automàticament amb la nostra selecció.

A continuació hi ha els dissenys de les pantalles de l'aplicació:

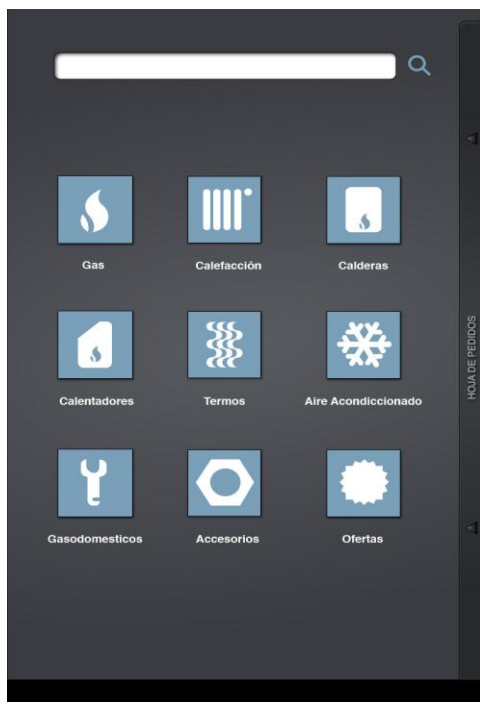


Figura 15 Vista de categorías



Figura 14 Llista de productes



Figura 18 Vista del carro de la compra



Figura 19 Detall del producte (vertical)



Figura 16 Vista del Parallax 1

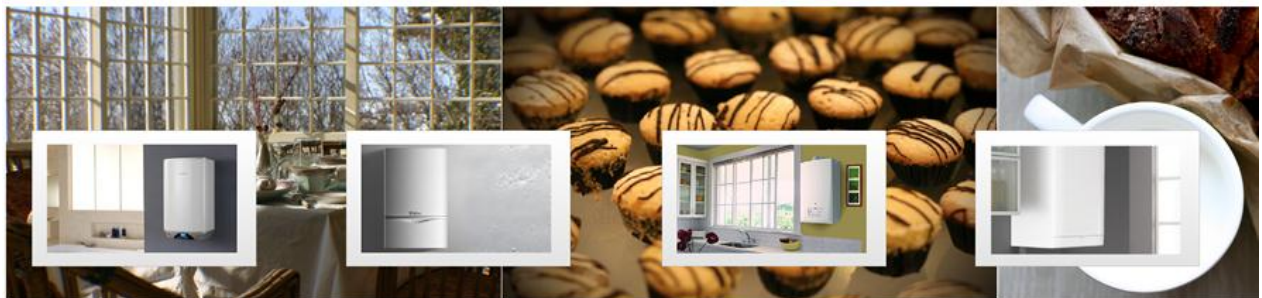


Figura 17 Vista del Parallax 2



Figura 20 Detall del producte (horitzontal)



Figura 21 Comparativa de productes

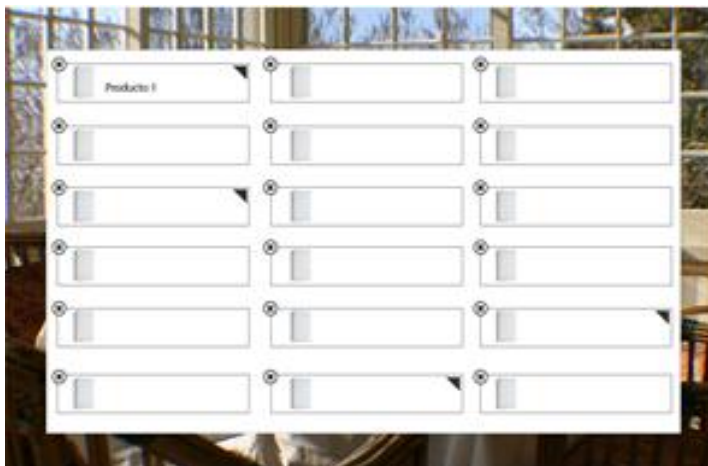


Figura 22 Triar productes a comparar

Un cop explicat tot aquest procés, ja podem crear un diagrama de pantalles amb la navegació de l'aplicació. El diagrama resultant és el següent:

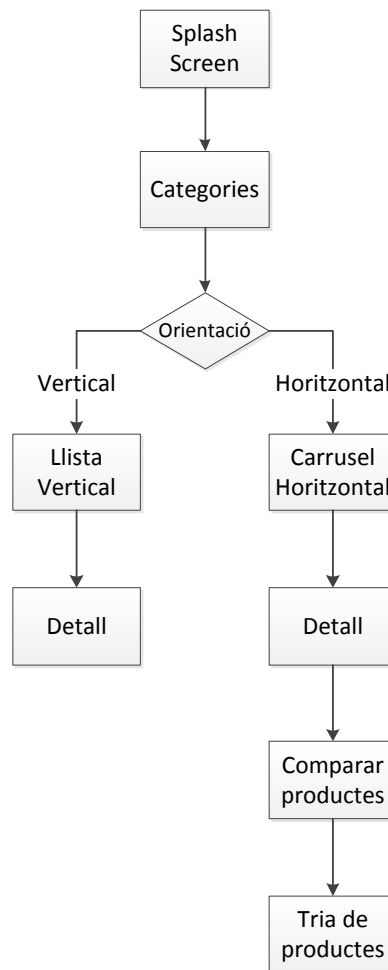


Figura 23 Diagrama de pantalles

2.4.2. Anàlisi i disseny de l'estructura de classes

S'ha decidit que el funcionament de l'aplicació serà el següent:

- Hi ha un fitxer JSON en una adreça pública d'Internet anomenat timestamp.json, que ens diu quina és la data de la última versió de la llista de productes, a més de la l'adreça web que té el fitxer JSON amb els productes d'aquesta llista.
- Quan s'obri l'aplicació de catàlegs virtuals, es mostrarà la pantalla d'splash screen i es descarregarà el timestamp esmentat al punt anterior. Ara cal

comprovar si aquest timestamp és diferent al de que té l'aplicació (si es que en té).

- Tant si no tenim timestamp (primer cop que s'obre l'aplicació) com si és diferent al del servidor es procedirà a descarregar la nova llista de productes, tot invocant la nova adreça URL que ens indiqui el fitxer de timestamp.json. En cas que el timestamp no hagi canviat respecte l'anterior vegada que s'ha obert l'aplicació no durem a terme cap acció.
- Un cop descarregat tot el llistat nou de productes, es procedirà a esborrar el contingut de la taula de productes de la Base de Dades per a desar la nova llista de productes. Quan hagi acabat es mostrarà la pantalla amb la llista de categories de productes.
- En aquests moment, l'usuari pot fer click en una de les categories, o pot realitzar una cerca per paraules. Si fa una d'aquestes accions, es crea un objecte Search, i en el primer cas (veure una categoria) se li assigna el codi de la categoria; en el segon cas (per paraules clau), al objecte se li assigna l'String amb el text que ha cercat l'usuari i s'obre l'Activity de la llista de resultats, tot passant-li aquest objecte Search a través de l'Intent.
- La pantalla de la llista de resultats rep l'objecte de cerca i realitza una consulta a la Base de Dades amb els valors dels camps que tingui l'objecte Search. El resultat d'aquesta consulta és el que es mostrarà al ListView a través de l'Adapter de productes en cas que el tablet estigui en mode vertical o es mostrarà en forma del Parallax de productes si estem en mode horitzontal.
- L'usuari tindrà la possibilitat de mantenir polsat un producte amb el dit i arrossegar-lo a la llista de la compra de productes. Aleshores, la llista de la compra s'actualitzarà per a fer aparèixer el nou producte automàticament.
- En el mode vertical, l'usuari pot aplicar i modificar els filtres de la cerca de productes. Aquests canvis modificaran els camps de l'objecte Search, i de la mateixa manera que al punt anterior, la llista de productes s'anirà actualitzant a mida que aquests filtres són modificats.
- Quan l'usuari faci click a un dels productes, es mostrarà el detall del producte.
- L'usuari podrà modificar el seu carro de la compra mentre estigui en mode vertical.
- Quan estigui al mode horitzontal, al detall del producte hi apareixerà un botó que servirà per anar a la pantalla de comparar productes, on hi apareixeran els 4 productes a comparar. A més, podrà canviar cadascun d'aquests productes per altres en qualsevol moment amb una pantalla de selecció de productes.

Elements bàsics de l'aplicació

A continuació hi ha un breu resum de les classes més importants i amb la part més crítica de la funcionalitat de l'aplicació:

- **Product.** Aquesta classe s'utilitza a l'aplicació per a representar un producte del nostre catàleg. Conté tota la informació relacionada amb aquest. A més, conté les anotacions que necessita l'ORMLite (descrita a l'apartat "Llibreries" de la secció "Implementació"), ja que el que dessem a Base de Dades són objectes d'aquesta classe.
- **Search.** Utilitzem aquesta classe a mode de cerques en l'aplicació. És un objecte serialitzable que podem passar d'una Activitat a una altra a través d'un Intent. Conté els camps que pot contenir un filtre de cerca de productes. Considerem que cada cop que obtenim una llista de productes de la base de dades, és una cerca. És a dir que utilitzem els objectes Search per a anar de la pantalla de selecció de categoria a la llista dels productes, quan l'usuari cerca productes per paraules clau i quan realitza un filtrat (per exemple filtrat per preu) sobre la llista que ja està veient per pantalla.
- **SplashActivity.** Primer cal assegurar-se de que hi ha connexió a Internet, ja que si no n'hi ha, passem directament a la pantalla de selecció de categories. Si hi ha connexió, realitzem la comprovació de si hi ha una nova llista de productes o no i actualitzem la Base de Dades i si és així, descarreguem aquesta nova llista, esborrem les dades dels productes que ja tenim de la Base de Dades i hi dessem aquesta nova llista.
- **ProductsActivity.** Activitat que rep un objecte de tipus Search a través de l'Intent que la executa. Aquest objecte conté la informació necessària per a obtenir una llista de Products segons els paràmetres de cerca que l'usuari ha introduït. Segons la orientació de la pantalla, posarà un Fragment o un altre, passant-li aquest llistat de productes.
- **ProductListFragment.** Aquest Fragment és el que apareixerà per pantalla quan la ProductsActivity detecta la orientació vertical de la pantalla. Es mostren els productes que ha recuperat la ProductsActivity en un ListView. A més permet filtrar aquests productes (realitzar noves cerques).
- **ParallaxFragment.** El ParallaxFragment es mostrarà quan la ProductsActivity detecti la orientació horitzontal de pantalla. En aquest cas, es mostra en una vista de Parallax enlloc d'un ListView. A més, no permet filtrar els resultats que es mostren per pantalla.

- **ShoppingList.** Aquesta classe representa la llista de la compra. Aquest és un llistat de productes que conté per a cada producte, un número que representa una quantitat, la talla de cada producte i el preu total dels elements de la llista de la compra.

2.5. Implementació

Aquest apartat descriu com s'han implementat les diverses parts importants de l'aplicació en detall. També he descrit les llibreries que utilitzo per a reaprofitar codi existent. Com s'ha comentat anteriorment, les eines que fem servir per a desenvolupar són totalment gratuïtes:

- Eclipse (projecte open-source, eina gratuïta).
- ADT Plugin per l'Eclipse (open-source, plugin gratuït).
- Android SDK Tools (open-source, conjunt d'eines gratuïtes).

2.5.1. Llibreries

Aquesta és la llista de llibreries de tercers que s'utilitzen en l'aplicació de catàlegs virtuals i que m'han ajudat en el desenvolupament del projecte:

- **ORMLite.** Llibreria d'Object Relational Mapping (o ORM. Això significa mapejar objectes a files d'una taula de Base de Dades) sobre l'SQLite d'Android. Ens estalvia molt la feina ja que podem crear Data Access Objects (o DAO. Són els objectes que ens permeten modificar les dades directament a la Base de Dades mitjançant una interfície senzilla). Així, mantenim la llista de productes del catàleg.
- **Gson.** Aquesta és una llibreria creada per Google i que ens permet treballar amb el format JSON des de Java. Permet tenir serialitzat d'objectes Java a JSON i desserialitzat de JSON a Java. Tot això ho fa mitjançant la reflexió de Java, per tant no cal fer servir anotacions, ja que la llibreria no ho necessita.
- **PFC ParallaxView.** Aquesta llibreria la he creat per aquest projecte en concret. Facilita la feina alhora de crear un ParallaxView (veure apartat "Parallax" en d'aquesta mateixa secció), ja que conté dues classes amb part de la funcionalitat Parallax implementada.
- **Classe Panel.** Aquesta classe es un component que actua com a tirador on hi ha el carro de la compra i el detall del producte. Tot i que Android ja ens proveeix d'un element visual que fa aquesta funció, aquest només funciona

d'esquerra a dreta i de baix a dalt i es necessitava que el tirador anés de dalt a baix i de dreta a esquerra, per tant no ens serveix.

2.5.2. Persistència de dades

Com ja he comentat anteriorment, la persistència de dades es fa utilitzant la llibreria ORMLite, que s'encarrega d'interactuar amb l'SQLite 3 que ja té Android incorporada. D'aquesta manera, amb molt poc codi tenim una base de dades amb una taula de productes (on hi emmagatzemem la llista de productes cada vegada que hi ha un canvi en la llista de productes).



Figura 24 Esquema d'ORMLite

2.5.3. Comunicació de dades

Per a la comunicació de dades, s'utilitza el format estàndard JSON, ja que és suficientment senzill i flexible. A més, comparat amb l'XML, el JSON és molt més ràpid de transferir per internet ja que no conté els tags de cada camp i existeixen llibreries que faciliten encara més quan s'ha de treballar amb JSON, com el cas de Gson, que ens permet agafar un objecte de Java i convertir-lo a JSON mitjançant una crida a un mètode de la llibreria. Per tant, tant el fitxer de timestamp com el fitxer de productes que tenim al servidor estan en format JSON. Aleshores, quan descarreguem la llista de productes, mitjançant el Gson obtenim la llista d'objectes Product sense tenir que llegir i convertir tot l'String que rebem cap a objectes. Ho fa automàticament.

2.5.4. Parallax

Per a crear el Parallax, faig servir una llibreria que he creat per a aquest projecte en concret, amb la intenció de poder-la fer servir més endavant, si fa falta.

Bàsicament, un Parallax és una vista que permet fer scroll i té varies capes, de tal forma que quan fem scroll (en aquest cas és horitzontal), cadascuna de les capes es mou a diferent velocitat, simulant així un efecte de profunditat i de perspectiva. Aquest efecte s'utilitza molt sovint als videojocs. Si pensem en un joc de tipus plataforma (per exemple Super Mario Bros.), hi ha un fons que pot ser, per exemple, un cel amb muntanyes (aquest es mou més a poc a poc i es va repetint, fa l'efecte que està molt lluny), després hi ha una segona capa, com podrien ser uns arbres que es mouen una mica més ràpid (fa l'efecte que estan molt més a prop que el fons) i per últim, hi ha

l'escenari per on el personatge va avançant durant el joc (aquest escenari es mou a la velocitat que avança el jugador). Això ens dona un efecte de profunditat sense tenir que utilitzar motors gràfics 3D ni cap tecnologia més complexa.

Per a poder implementar aquesta funcionalitat, he tingut que crear una Vista pròpia que dibuixi directament sobre el Canvas. El sistema de vistes d'Android no era la millor solució per a aquest problema, ja que la eficiència i rapidesa de dibuixat era molt important per a donar una bona experiència d'usuari. El Garbage Collector de la màquina virtual Dalvik produeix una pausa en el thread de dibuixat d'uns mili segons cada cop que s'executa, per tant s'ha d'evitar això en la mida del possible.

Aleshores, la solució a la que he arribat ha sigut:

- Crear una classe ParallaxView que és una subclasse de View. Aquesta classe representa el contenidor de les capes del Parallax. Té referències als objectes de les diferents capes i controla que cadascuna d'elles es dibuixi quan toca, canviï la seva posició d'scroll quan l'usuari toca la pantalla, etc.
- Crear una classe ParallaxLayer que representa una capa del Parallax i conté informació sobre la posició del scroll actual i la mida de la capa, a més de tenir mètodes abstractes. Cada capa del Parallax haurà d'heretar aquesta classe i implementar els mètodes abstractes per a que tot funcioni correctament.

2.5.5. Carro de la compra

Per a la llista del carro de la compra, s'utilitza el patró Singleton, ja que hem de conservar i anar modificant la mateixa llista (el mateix objecte de llista) durant tota l'aplicació. Com que en Android no podem passar llistes a d'una Activity a una altra, quan volem que una llista sigui accessible des de qualsevol punt de l'aplicació, s'acostumen a crear llistes estàtiques i públiques que s'inicialitzen al obrir l'aplicació i s'hi va accedint a elles durant l'execució de les Activities. Per tant, el patró Singleton és perfecte, ja que tenim la instància del objecte en qüestió com a variable estàtica i com que el constructor de la classe és privat no podem instanciar més objectes, però sempre podem obtenir la instància única amb el mètode getInstance().

2.5.6. Orientació de la pantalla

Un dels problemes més complex que m'he trobat és el de fer que la pantalla per a veure els productes sigui radicalment diferent segons si el tablet està en mode horitzontal o vertical. En Android el que s'acostuma a fer sovint és que es crea una vista d'una pantalla per al mode vertical i se'n crea una altra per al mode horitzontal de

la mateixa pantalla, però amb la informació reorganitzada. És a dir que normalment, no canvia la quantitat d'informació que es mostra d'un mode a l'altre, simplement es canvia l'estructura de la vista per a que quedi bé i no falti ni sobri espai de pantalla.

En aquest cas, com que el contingut de la pantalla vertical és totalment diferent del contingut de la pantalla horitzontal, he arribat a una solució amb algunes limitacions però que funciona correctament, tenint en compte que és millor separar la funcionalitat d'aquestes pantalles per a simplificar el codi. Descric breument a continuació aquesta solució:

- Tindrem 1 Activity i 2 Fragments. L'Activitat, gestionarà els girs de pantalla i els Fragments, n'hi ha un per al mode de pantalla vertical (V) que tindrà el llistat i un altre per al mode horitzontal (H) que conté el Parallax.
- L'Activitat s'encarrega d'escoltar als canvis de orientació de la pantalla del dispositiu. Quan aquesta s'obre, comprova quina és la orientació actual de la pantalla i posa com a visible el Fragment que sigui necessari. Així com detecta un canvi, treu el Fragment actual i posa el que es correspongui amb la nova orientació.

2.6. Resultat

A continuació hi ha unes captures de pantalla mostrant el resultat de l'aplicació finalitzada. Com podem veure s'assimila molt al disseny inicial, tot i que certes parts han sofert petits canvis degut a les revisions amb el dissenyador:

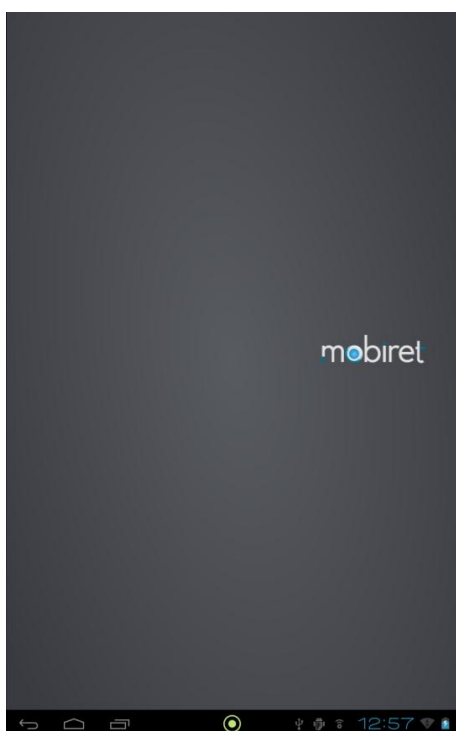


Figura 25 Splash screen



Figura 26 Vista de categorías



Figura 27 Lista de productos



Figura 28 Vista del detall (vertical)

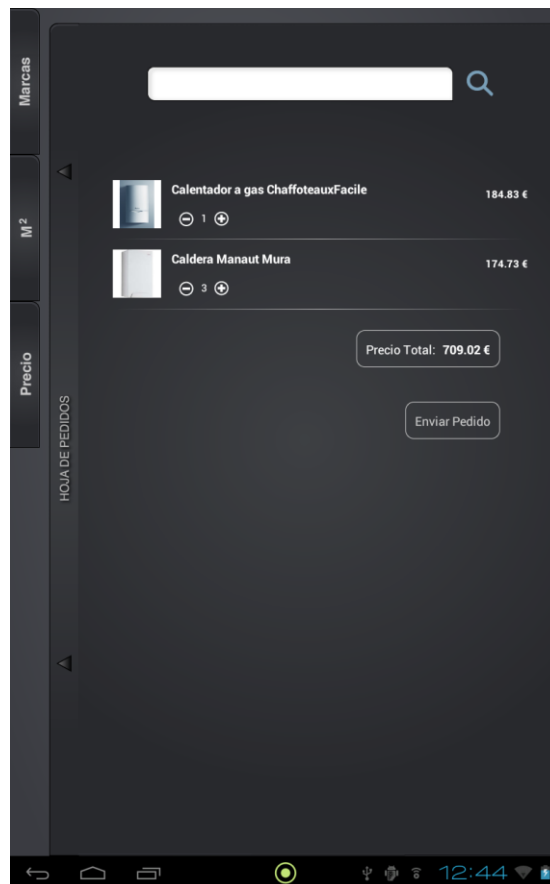


Figura 29 Lista de carro de la compra



Figura 32 Vista de Parallax 1



Figura 30 Vista de Parallax 2

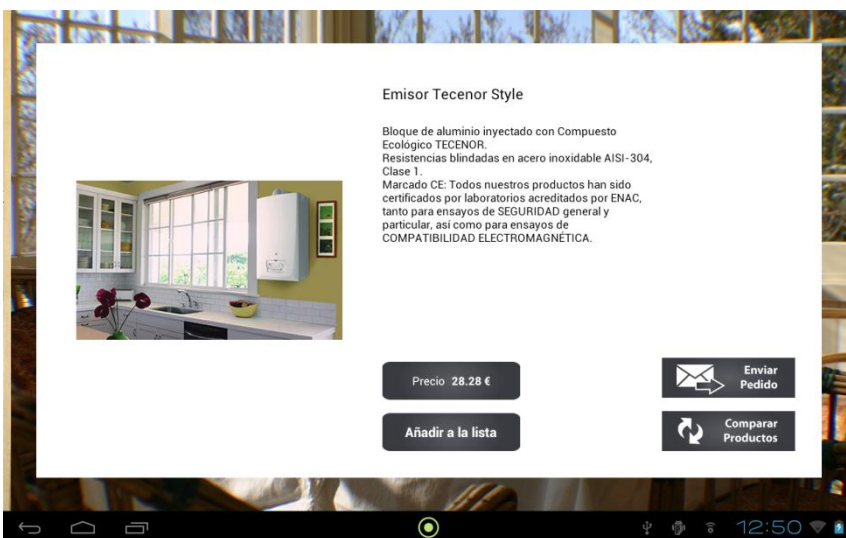


Figura 31 Detall de producte (horizonta)

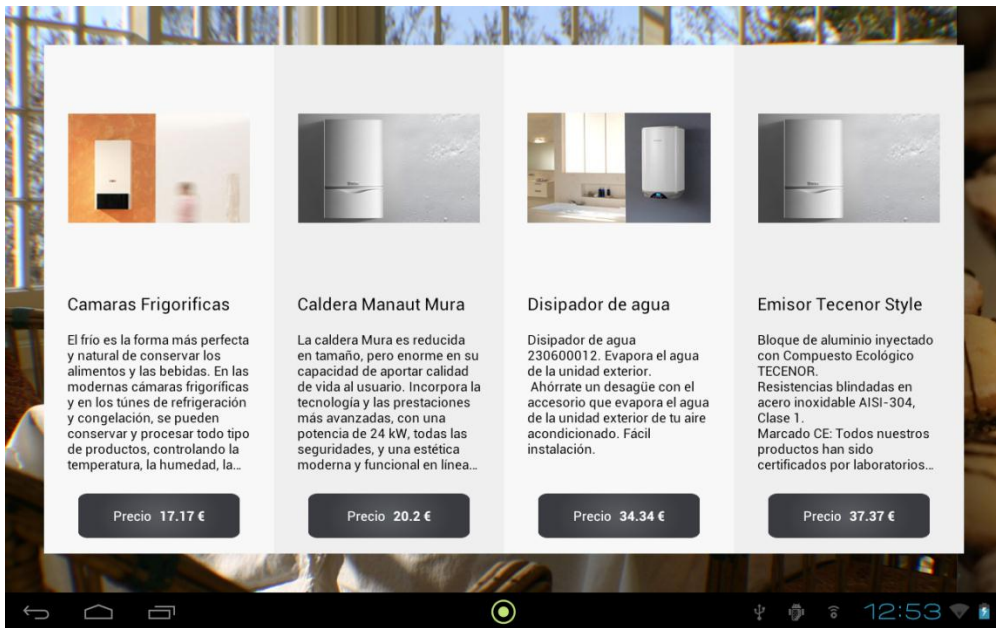


Figura 34 Comparació de productos



Figura 33 Selector de productos a comparar

3. Conclusions

Aquest projecte m'ha servit per a millorar els meus coneixements sobre la plataforma Android. A més he après a crear aplicacions per a dispositius amb pantalles més grans que els telèfons mòbils, com són els tablets. Això és un dels beneficis més interessants que en puc treure d'aquest projecte, ja que el món tablet, està creixent i cada cop hi ha més gent que n'utilitza, alhora que apareixen nous models i els preus baixen, convertint-se en una tecnologia més assequible.

També he trobat molt interessant el treballar amb l'eina de gestió de projectes Microsoft Project, que fins a la realització del projecte no havia utilitzat mai. Crec que un dels punts positius que en puc treure de la realització d'aquest projecte és el haver tingut que gestionar el projecte tot sencer: crear el llistat de tasques a realitzar, realitzar un Planning temporal a partir d'aquestes tasques, tenir en compte l'assignació dels recursos, etc. M'ha brindat l'oportunitat d'adquirir aquests coneixements que no s'aprenen en el dia a dia d'un estudiant i m'ha ajudat a tenir una noció de la gestió de projectes, ja que abans no coneixia aquest món.

A més hi ha hagut situacions en què he tingut que afrontar problemes que mai m'havia trobat i arribar a solucions pel meu compte. Aquests són alguns exemples d'aquests problemes i com els he solucionat:

- **Parallax.** Al principi, per a solucionar el problema del Parallax, vaig crear una classe que representava una capa del Parallax, i en essència era un contenidor de vistes que anava col·locant els seus fills l'un al costat de l'altre i gestionava el seu scroll de tal forma que quan arribava movent els seus fills cap al principi o cap al final segons la posició del scroll. Això comportava que calia carregar totes les vistes en memòria al inici, per tant no era eficient i a més, el Garbage Collector s'executava cada cert temps i això feia que durant unes fraccions de segon s'aturés el fil d'execució. Tot plegat causava una mala experiència d'usuari. La nova solució causa una major ocupació de la memòria del dispositiu, però no causa pauses durant l'scroll, per tant ha millorat l'experiència d'usuari.
- **Orientació de pantalla.** La primera solució a la que vaig arribar per al problema amb el canvi de pantalla amb la orientació va ser la d'utilitzar una sola Activitat i crear dues vistes: una per a que es mostrés amb la orientació de pantalla vertical i una altra per a l'orientació horitzontal. Els problemes que hi havia amb aquesta solució eren que el codi font que es creava, tenia una part

comuna per a les dues pantalles, però després havia de tenir part del codi font específic per a controlar la lògica del Parallax i una altra part per a gestionar la lògica del ListView. Per tant, teníem un codi molt poc reutilitzable. Amb la solució actual resollem aquest problema, creant Fragments totalment reutilitzables que es centren a dur a terme una funció en concret.

També m'agradaria comentar les possibles millores que es podrien dur a terme per a millorar aquesta aplicació en un futur:

- Crear un servidor web amb una Base de Dades de productes. Aquesta Base de Dades es podria modificar, importar, exportar, etc. amb un formulari web i el tablet accediria a aquest servei web enlloc d'un fitxer JSON.
- Les actualitzacions de productes podrien arribar mitjançant la tecnologia Push, és a dir que el servidor avisa al dispositiu quan hi ha una nova versió del llistat productes per a que aquest es descarregui la nova llista de productes, enlloc de consultar cada vegada el fitxer de Timestamp.
- El Parallax es podria fer en codi natiu C o C++ per a que sigui més eficient i així es podria gestionar millor la memòria quan carreguem imatges de molta resolució.
- Integració amb sistemes de pagament tipus Paypal per a poder realitzar la compra al moment, enlloc d'utilitzar un sistema simple de comandes.

Resumint, crec que ha sigut una experiència molt positiva i amb cert èxit, doncs s'han assolit els objectius proposats i tot i així, encara hi ha la possibilitat de millorar aquest projecte amb els suggeriments proposades anteriorment.

4. Bibliografia

- Android (Wikipedia). [en línia]: <http://ca.wikipedia.org/wiki/Android>
- Tauleta tàctil (Wikipedia). [en línia]:
http://ca.wikipedia.org/wiki/Tauleta_t%C3%A0ctil
- Comerç electrònic (Wikipedia). [en línia]:
http://es.wikipedia.org/wiki/Comercio_electr%C3%B3nico
- Introducció a ORMLite (ORMLite). [en línia] <http://ormlite.com/>
- Introducció a GSON (GSON). [en línia] <http://code.google.com/p/google-gson/>
- Application Fundamentals (Android Developers). [en línia]
<http://developer.android.com/guide/components/fundamentals.html>
- Documentació Activity (Android Developers). [en línia]
<http://developer.android.com/reference/android/app/Activity.html>
- Documentació Fragment (Android Developers). [en línia]
<http://developer.android.com/reference/android/app/Fragment.html>
- Documentació Intent (Android Developers). [en línia]
<http://developer.android.com/reference/android/content/Intent.html>
- Documentació Shared Preferences (Android Developers). [en línia]
<http://developer.android.com/reference/android/content/SharedPreferences.html>
- Documentació Adapter (Android Developers). [en línia]
<http://developer.android.com/reference/android/widget/Adapter.html>
- Garbage Collection (Wikipedia). [en línia]
[http://en.wikipedia.org/wiki/Garbage_collection_\(computer_science\)](http://en.wikipedia.org/wiki/Garbage_collection_(computer_science))

5. Resum de figures

Figura 1 Taula de solucions proposades.....	15
Figura 2 Taula de recursos del projecte.....	16
Figura 3 Taula de costos dels recursos humans.....	17
Figura 4 Taula de costos dels recursos físics.....	17
Figura 5 Taula re resum de costos.....	18
Figura 6 Calendari de tasques.....	19
Figura 7 Taula de durada de les tasques.....	21
Figura 8 Diagrama de Gantt d'activitats.....	22
Figura 9 Logotip d'Android.....	26
Figura 10 Diferents dispositius tablet.....	27
Figura 11 Gràfica comparativa de les compres des de telèfons i des de tablets.....	28
Figura 12 Cicle de vida d'una Activity.....	30
Figura 13 Logotip d'SQLite.....	32
Figura 14 Llista de productes.....	34
Figura 15 Vista de categories.....	34
Figura 16 Vista del Parallax 1.....	35
Figura 17 Vista del Parallax 2.....	35
Figura 18 Vista del carro de la compra.....	35
Figura 19 Detall del producte (vertical).....	35
Figura 20 Detall del producte (horitzontal).....	36
Figura 21 Comparativa de productes.....	36
Figura 22 Triar productes a comparar.....	36
Figura 23 Diagrama de pantalles.....	37
Figura 24 Esquema d'ORMLite.....	41
Figura 25 Splash screen.....	43
Figura 26 Vista de categories.....	43
Figura 27 Llista de productes.....	44
Figura 28 Vista del detall (vertical).....	44

Figura 29 Llista de carro de la compra	44
Figura 32 Vista de Parallax 2	45
Figura 31 Detall de producte (horitzontal)	45
Figura 30 Vista de Parallax 1	45
Figura 33 Selector de productes a comparar	46
Figura 34 Comparació de productes	46

A Sabadell, 26 de juny de 2012

Xavier Rigau Jimenez