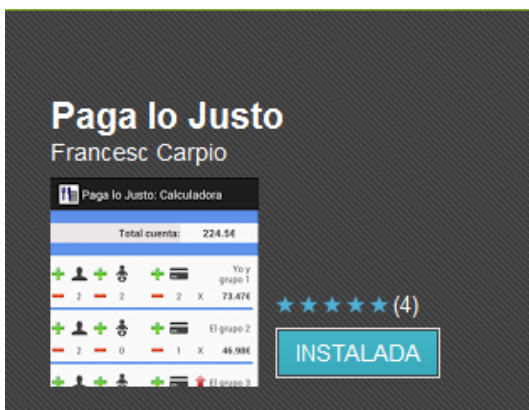


Treball Final de Carrera

Desenvolupament d'aplicacions per dispositius mòbils

MEMÒRIA



Alumne: Francesc Carpio López

Consultor: Albert Grau Perisé

UNIVERSITAT OBERTA DE CATALUNYA

E.T. Informàtica de Sistemes

Curs 2012-2013



*Dedicat als meus fills César i Néstor, i a la meva
companya María Luisa.*

Les hores robades seran justament retornades.

Índex de continguts

1	Justificació i objectius del projecte	5
2	Planificació inicial i real	7
a.	Llistat de tasques.....	7
b.	Diagrama de GANTT	8
3	Documentació de l'anàlisi	9
3.1	Usuaris i context d'ús	9
3.1.1	Introducció	9
3.1.2	Observació i investigació contextual.....	9
3.1.3	Enquestes	10
3.1.4	Entrevistes en profunditat	11
3.1.5	Anàlisi competitiva	11
3.1.6	Conclusió: perfils d'usuaris.....	13
3.2	Disseny conceptual.....	13
3.2.1	Escenaris.....	13
3.2.2	Fluxos d'interacció.....	14
3.3	Prototipatge	15
3.3.1	Criteris de disseny: Principis heurístics	15
3.3.2	Prototipus d'alta fidelitat estàtic.....	16
3.3.3	Classificació favorits	18
3.3.4	Historial despeses.....	19
3.4	Avaluació	19
3.4.1	Qüestionari sobre el test.....	19
3.4.2	Llistat de tasques a realitzar	20
3.4.3	Resultat de l'avaluació	20
4	Implementació del producte.....	22
4.1	Arquitectura de l'aplicació	22
4.1.1	Capa d'usuari i presentació	22
4.1.2	Capa lògica	24
4.1.3	Capa de dades	25
4.2	Entorn de treball	26
4.3	El codi	28
4.4	La depuració del programa	29



4.5	La configuració	31
4.6	Comentaris i documentació	32
4.6.1	Comentaris del codi.....	32
4.6.2	Documentació de les classes	33
4.7	Control de versions	34
4.8	Proves.....	35
4.8.1	Proves de funcionalitat i robustesa.....	37
4.8.2	Proves d'acceptació de l'usuari.....	40
4.9	Ajuda	42
4.10	Publicació i model de negoci.....	44
5	Conclusions	47
6	Bibliografia	49



1 Justificació i objectius del projecte

L'objectiu principal del projecte ha estat la creació d'una aplicació per a dispositius mòbils compatibles amb el sistema operatiu Android.

El treball ha consistit en desenvolupar i implementar totes les fases d'una aplicació informàtica: des del seu naixement com a idea de programa, fins a la seva publicació en el Market de Google per a que qualsevol usuari la pugui descarregar, instal·lar i utilitzar en el seu dispositiu. Per tant, hem hagut de treballar totes les etapes del cicle de vida d'una aplicació, excepte el manteniment i l'extinció final de l'aplicació.

El resultat ha estat el programa "Paga lo Justo". Aquest, consisteix en una aplicació senzilla i acotada, però amb suficients detalls com per a fer immersió profitosa en la programació per a Android i les tecnologies mòbils: disseny d'interfícies, programació i utilització de llibreries, consulta de dades tant locals com remotes o comunicació amb servidors, etc.

Tot aquest treball, què hem fet i com, el resumim a continuació responent a les següents qüestions:

Quin és el nostre producte?

A partir de l'observació i investigació contextual hem volgut resoldre un problema de la vida real: el repartiment entre diferents comensals del compte total d'un restaurant, tenint en compte que no tots han de pagar la mateixa quantitat, ja sigui perquè paguen en parella o família, o perquè han consumit més o menys que d'altres. Així doncs, **el nostre producte és una calculadora per a la divisió del total del compte d'un restaurant.**

Com s'han definit les funcions que ha de realitzar?

Per a la definició dels requeriments de l'aplicació s'han realitzat enquestes, entrevistes, així com l'anàlisi de la competència, creant d'aquesta manera els **perfils d'usuari**. Un cop definit el problema s'han establert les característiques funcionals que haurà d'acomplir la nostra aplicació.

L'aplicació ha de dividir el compte de forma justa sense introduir masses dades, únicament el import final. A més d'altres funcionalitats secundàries com resum de despeses o valoració de restaurants.

Quines són les seves característiques de disseny?

Per a la realització del disseny de la nostra aplicació ens hem basat en els principis del **Disseny Centrat en l'Usuari (DCU)**. En totes les etapes del projecte hem tingut present a l'usuari: des del disseny del prototipus fins a l'avaluació final mitjançant les proves d'acceptació. Això ens ha fet realitzar canvis i modificacions continues en el nostre programa, enriquint-lo i facilitant la seva comprensió i ús.

Un exemple d'aquesta continua realimentació amb l'usuari ha estat el propi nom de l'aplicació, que va ser inicialment "Solomon Restaurant", després "Restaurant in Group" i finalment i definitivament "Paga lo Justo".

Quines tecnologies s'han utilitzat per a la seva implementació?

La nostra eina principal per a la realització de l'aplicació ha estat l'entorn integrat de desenvolupament (IDE), de codi obert Eclipse, juntament amb el kit de desenvolupament de programari (Android SDK), i les eines pel desenvolupament (Android ADT).

El codi de l'aplicació s'ha fet tenint en compte diferents criteris definits com a "bones pràctiques", entre els quals destaquen:

- La utilització d'unitats relatives per permetre l'adaptació a diferents grandàries de pantalles.
- La separació de la interfície gràfica i el codi de l'aplicació.
- Externalitzar recursos per facilitar canvis globals com pot ser la implementació de diferents idiomes.
- Utilitzar les eines facilitades per l'SDK oficial, molt optimitzades i provades com: traces en depuració, documentació del codi, gestió de la base de dades, etc.

Com es pot instal·lar l'aplicació en un dispositiu compatible?

Per a la publicació de l'aplicació s'ha triat Google Play, abans conegut com a Android Market, des de on qualsevol usuari amb dispositiu compatible amb Android podrà descarregar-se i instal·lar-se l'aplicació amb total garantia i seguretat.

Tanmateix, a part de l'anàlisi, el disseny, la implementació i les proves que requereix un projecte d'aquest tipus, no hem deixat al marge el seu context principal: el treball de final de carrera. És per això que hem hagut de treballar altres aspectes com la generació de tota la documentació associada al projecte i els lliuraments establerts com la present memòria.

2 Planificació inicial i real

La planificació plantejada al inici del treball s'ha seguit amb força normalitat gràcies tant al treball continu i constant com als lliuraments de les diferents PACs que han pautat molt el ritme de treball.

La única desviació remarcable vers la planificació inicial ha estat l'allargament de les últimes etapes en el desenvolupament de l'aplicació, principalment en les proves d'acceptació de l'usuari.

Les proves amb usuaris reals ens han fet modificar certes parts del programa així com afegir més ajudes a l'usuari. A més, aquestes modificacions requerien de noves comprovacions per assegurar que no hi hagués cap nou error. Tot i això, és pot considerar que l'endarreriment ha estat lleu permetent acomplir amb les dates pactades.

a. Llistat de tasques

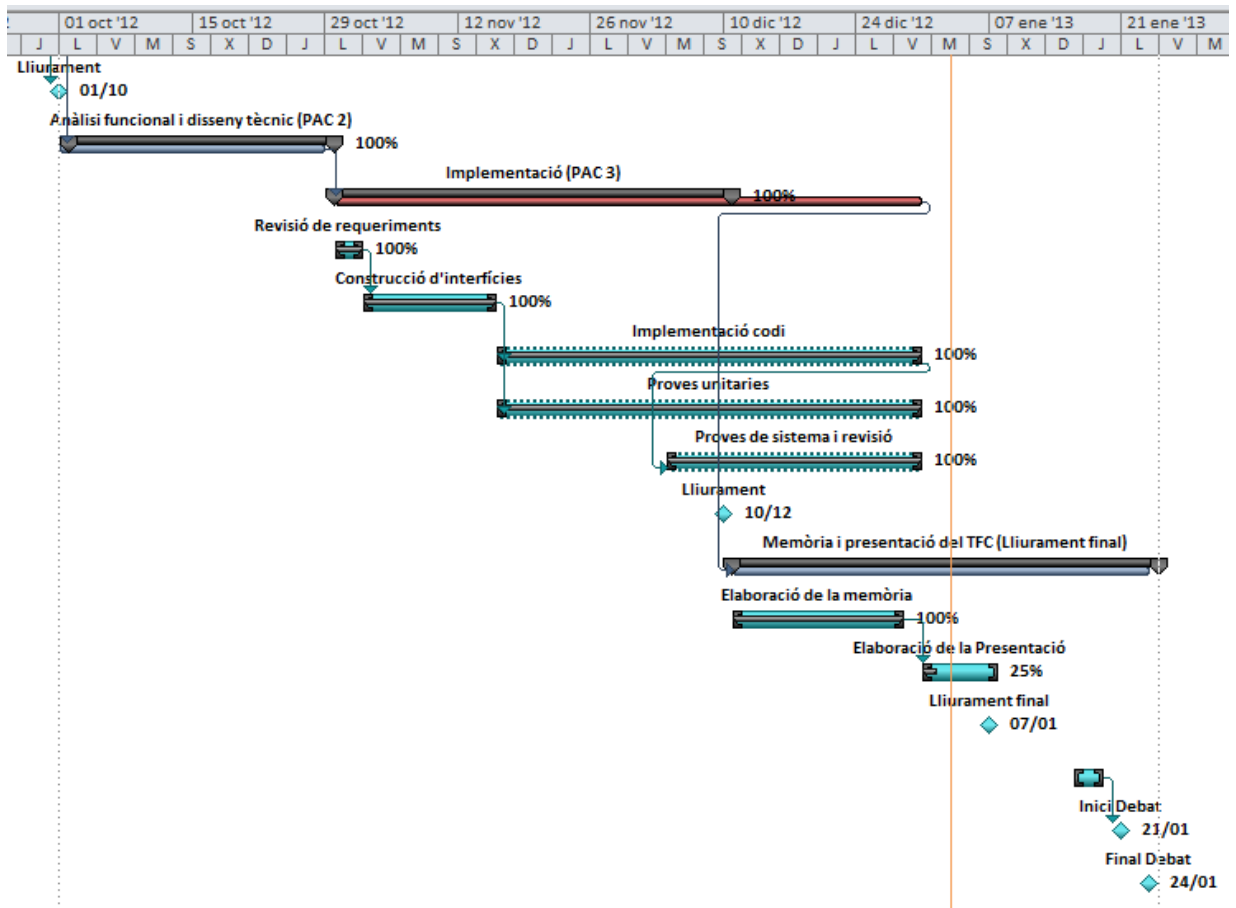
		Nombre de tarea	Duración	Comienzo	Fin
1		TFC "Paga lo Justo" per a Android	92 días	mié 19/09/12	jue 24/01/13
2	✓	Definició del projecte	7 días	mié 19/09/12	jue 27/09/12
3	✓	Pluja d'idees	4 días	mié 19/09/12	lun 24/09/12
4	✓	Anàlisi de mercat	3 días	mar 25/09/12	jue 27/09/12
5	✓	Pla de treball (PAC 1)	6 días	lun 24/09/12	lun 01/10/12
6	✓	Elaboració	5 días	lun 24/09/12	vie 28/09/12
7	✓	Lliurament	0 días	lun 01/10/12	lun 01/10/12
8	✓	Anàlisi funcional i disseny tècnic (PAC 2)	20 días	lun 01/10/12	lun 29/10/12
9	✓	Anàlisi funcional	3 días	lun 01/10/12	mié 03/10/12
10	✓	Casos d'ús	2 días	jue 04/10/12	vie 05/10/12
11	✓	Disseny conceptual	3 días	lun 08/10/12	mié 10/10/12
12	✓	Preparació entorn de treball	2 días	jue 11/10/12	vie 12/10/12
13	✓	Construcció de prototipus	6 días	lun 15/10/12	lun 22/10/12
14	✓	Elaboració del document	4 días	mar 23/10/12	vie 26/10/12
15	✓	Lliurament	0 días	lun 29/10/12	lun 29/10/12

		Nombre de tarea	Duración	Comienzo	Fin
16	✓	Implementació (PAC 3)	30 días	mar 30/10/12	lun 10/12/12
17	✓	Revisió de requeriments	3 días	mar 30/10/12	jue 01/11/12
18	✓	Construcció d'interfícies	10 días	vie 02/11/12	jue 15/11/12
19	✓	Implementació codi	32 días	vie 16/11/12	dom 30/12/12
20	✓	Proves unitaries	32 días	vie 16/11/12	dom 30/12/12
21	✓	Proves de sistema i revisió	20 días	mar 04/12/12	dom 30/12/12
22	✓	Lliurament	0 días	lun 10/12/12	lun 10/12/12
23		Memòria i presentació del TFC (Lliurament final)	33 días	mar 11/12/12	jue 24/01/13
24	✓	Elaboració de la memòria	14 días	mar 11/12/12	vie 28/12/12
25		Elaboració de la Presentació	6 días	lun 31/12/12	lun 07/01/13
26		Lliurament final	0 días	lun 07/01/13	lun 07/01/13
27		Preparació del Debat	3 días	mié 16/01/13	vie 18/01/13
28		Inici Debat	0 días	lun 21/01/13	lun 21/01/13
29		Final Debat	0 días	jue 24/01/13	jue 24/01/13



b. Diagrama de GANTT

En la següent captura és pot observar el diagrama de GANTT en el qual s'ha minimitzat els grups de tasques per poder adjuntar-ho al document.



3 Documentació de l'anàlisi

3.1 Usuaris i context d'ús

3.1.1 Introducció

La nostra aplicació vol resoldre els problemes generats als usuaris a l'hora de pagar un compte conjunt en un restaurant, cercant com a objectiu principal que la repartició de les despeses estigui distribuïda de la manera més justa possible entre tots els pagadors.

L'anàlisi realitzat ens ha permès conèixer què volen i **què no volen** els futurs usuaris de l'aplicació. Així, centrant-nos en les prioritats dels usuaris s'ha pogut concretar el disseny de les funcionalitats principals de la nostra aplicació, així com els trets característics del nostre prototipus.

Les tècniques triades per a la investigació, i la corresponent justificació d'aquesta elecció, es defineixen a continuació:

- **Observació i investigació contextual:** Ens ha permès definir conceptualment el problema, estudiar diferents casuístiques i observar com actuen els agents involucrats.
- **Enquestes:** Ens ha permès definir els perfils d'usuari i conèixer quines són les seves principals característiques. També ens ha resultat útil per prioritzar les funcionalitats generals de l'aplicació.
- **Entrevistes en profunditat:** A partir dels perfils d'usuari definits a partir de les enquestes, ens ha facilitat la concreció detallada de les funcionalitats l'aplicació. A més, ens ha permès conèixer quines són les expectatives d'usabilitat i ergonomia d'aquests usuaris.
- **Anàlisi competitiva:** S'ha analitzat l'aplicació "Napkin Calculator", la qual té un objectiu similar a l'aplicació que volem implementar. El nostre objectiu és estudiar el seu funcionament, trobar les seves debilitats i les seves fortaleces, i observar com els usuaris interaccionen amb ella.

A continuació descriurem el detall dels resultats que s'han obtingut per a cadascuna de les tècniques emprades.

3.1.2 Observació i investigació contextual

Les observacions s'han realitzat en diferents establiments, en múltiples sopars, dinars, i diferents situacions.

Les observacions i conclusions extretes han estat les següents:

- La majoria dels restaurants o bars no accepten o se'ls fa incòmoda el fet de que cada usuari pagui individualment la seva part. L'argumentació per part del personal del restaurant acostuma a ser:



- Massa temps per calcular el compte, provoca esperes i cues a d'altres clients del restaurant.
- Sovint queden coses oblidades del compte que ningú se'n fa càrrec. Poden quedar sense pagar o el personal del restaurant decideix carregar-ho al compte de l'últim pagador.
- Quan hi ha un grup on cadascú es paga la seva part, s'acostuma a fer a parts iguals. Aquesta situació pot derivar en què algú no estigui d'acord en la forma de pagament. La solució passa per calcular en detall el que ha pres cadascú, provocant tensió entre els integrants del grup.
- Quan, a més, en un grup hi ha parelles o famílies el problema anterior s'agreuja i el càlcul, si s'arriba a fer, és complicat.
- En la majoria dels casos, per fer la divisió del pagament, sigui senzilla o al detall, s'utilitza la calculadora del mòbil. Sempre hi ha algú entre els components del grup que s'ofereix a fer-ho.

3.1.3 Enquestes

S'ha realitzat un model d'enquesta amb l'objectiu principal d'establir perfils d'usuari. L'enquesta s'ha realitzat a 34 individus d'edats compreses entre 20 i 50 anys de tots dos sexes. Es pot trobar a l'annex 1 el model d'aquesta enquesta així com els resultats obtinguts.

Sobre els resultats obtinguts en l'enquesta som conscients de les seves limitacions, ja sigui pel nombre insuficient de mostres com per una pobre estratificació de la població. Així i tot, els resultats es poden considerar prou orientatius i útils per poder seguir amb el desenvolupament de la nostra aplicació.

De l'anàlisi del resultat de l'enquesta s'han extret les següents conclusions:

- El principal SO utilitzat per les persones que han fet l'enquesta ha estat **Android** amb un **63%** del total.
- Més del **90%** dels individus enquestats pensen que la millor forma de **repartir** el total del compte d'un restaurant és a **parts iguals**.
- Un **82%** dels enquestats ha utilitzat algun cop **la calculadora del mòbil** per fer el càlcul del **repartiment**.
- De l'estudi de les enquestes, s'ha pogut establir una segmentació per grups d'edat en les següents qüestions:

	Menors de 35 anys	Majors de 35 anys
Majoritàriament paguen:	Individualment o Parella	Família
Rebre ofertes:	Normalment no.	Sí, 50% habituals, 50% qualsevol
Informació pagaments:	Més del 50%	Menys del 50%

3.1.4 Entrevistes en profunditat

Per a les entrevistes en profunditat, hem triat 4 usuaris de mòbil. La selecció dels individus s'ha realitzat tenint en compte els segments d'edat que hem diferenciat a través de les enquestes i tenint en compte els diferents coneixements i ús de dispositius mòbils intel·ligents. Hem batejat als enquestats amb els noms d'usuari JE, JI, AE, AI, amb les següents característiques:

- Usuari **JE**: És un usuari menor de 35 anys i expert en l'ús de dispositius mòbils intel·ligents, tant Android com iPhone.
- Usuari **JI**: És un usuari menor de 35 anys i amb coneixement bàsic en dispositius mòbils intel·ligents.
- Usuari **AE**: És un usuari major de 35 anys i expert en l'ús de dispositius mòbils intel·ligents, tant Android com iPhone.
- Usuari **AI**: És un usuari major de 35 anys i amb coneixement bàsic en dispositius mòbils intel·ligents


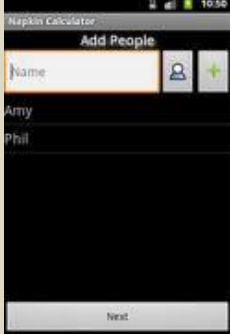


Les entrevistes s'han realitzat sense seguir un guió massa estricte, deixant que els usuaris aportin idees més obertes sobre el problema de repartir un compte d'un restaurant i com una aplicació ens podria facilitar el resultat.

A continuació es detallen les conclusions més importants extretes de les entrevistes:

- De manera general, es pensa que la millor forma de repartir el compte és a parts iguals. Encara que en casos més específics i si és de forma senzilla, també es valora molt positivament realitzar alguna ponderació. Exemples:
 - Quan hi ha nens, la seva part no pot ser igual que una d'adult.
 - En un grup, quan hi ha algú que no veu alcohol, perquè per exemple li toca conduir, hauria de pagar menys.
- Si es fes un programa per fer el càlcul hauria de ser senzill i ràpid per obtenir un resultat. No s'han de donar masses detalls.
- En general, si fos gratuïta, la instal·larien en el seu mòbil per provar-la.
- L'aplicació hauria de ser gratuïta, a canvi no es valora negativament mostrar publicitat.
- En general la idea de tenir un llistat amb el preu mig per persona de cada restaurant és força interessant per l'usuari, així com la valoració del restaurant. Per un altre banda, introduir molta informació del restaurant: telèfon, adreça, etc., és veu poc útil perquè aquesta informació o se sap ho es pot trobar fàcilment.
- Mantenir un històric de les meves despeses es veu com una opció secundària.

3.1.5 Anàlisi competitiva

En la cerca d'aplicacions de referència que es va realitzar en la pac anterior: pla de treball, vam trobar un programa per calcular el import per persona per un compte conjunt. A més de testejar el programa de forma individual, hem fet provar també als usuaris entrevistats l'aplicació per a què ens donessin el seu parer. A continuació es mostren els resultats i les conclusions de les proves.

Napkin Calculator	Observacions	Oportunitats	Febleses
	La interfície no és gaire atractiva ni professional.	Dissenyar una interfície més atractiva.	Disposem de poc temps per fer la implementació. No som experts en disseny.
	S'ha de donar d'alta a cada comensal. Es massa pesat i poc útil per a grups grans.	No realitzar aquesta opció. Millorem facilitat i rapidesa.	Perdrem detall.
	S'ha d'afegir articles per a cada comensal, indicant el nom i els preus.	Fer opcional afegir articles, ja que molts volen fer el càlcul a parts iguals. En cas d'afegir articles, que siguin generals i amb un preu establert. Minimitzar la introducció de dades per part de l'usuari.	Perdem detall.
	El preu ha pagar sempre es per persona. No es té en compte parelles ni famílies. Es té en compte les propines.	Donar el preu per individu, grup o família a pagar. Opció extra propina, pot ser no implementada en la primera versió del programa.	No hem tingut en compte establir un preu o percentatge per propina, ja que a la nostre zona no hi ha costum.
Altres observacions	La utilització de l'aplicació és a mode d'assistent "wizard".	L'assistent és més intuïtiu, però fa que l'usuari vagi més lent i es cansi abans del programa.	No tindrem temps per fer masses interfícies.
	No hi ha cap més valor afegit.	Podem oferir valoracions i estadístiques bàsiques sobre preus del nostres restaurants. Facilitar informació a l'usuari.	Temps d'implementació de la nostra aplicació molt limitat.

3.1.6 Conclusió: perfils d'usuaris

De l'estudi d'usuari i context d'ús hem pogut determinar dos tipus de perfils. Aquests perfils no fan una interacció gaire diferenciada sobre l'aplicació, ja que aquesta és molt senzilla i està molt acotada. La principal diferència que ens permet fer la divisió en dos perfils d'usuari recau en les preferències generals de l'usuari. A continuació definim els dos perfils amb les seves característiques principals:

- Perfil "**menor de 35**"
 - Està acostumat a utilitzar aplicacions mòbils. Es desenvolupa bé pels menús contextuais, quina funció estàndard té cada botó o quan es prem un botó durant un cert temps, arrossegar botons, etc.
 - No li agrada rebre cap tipus de publicitat.
 - Li agrada tenir dades del seu històric personal.
 - Paga la seva part de forma individual o en parella.

- Perfil "**major de 35**"
 - Utilitza aplicacions mòbils de forma ocasionalment. No té interioritzades totes les funcions estàndards d'utilització: menús contextuais, arrossegar per eliminar, etc.
 - Prefereix rebre publicitat útil o dirigida als seus interessos.
 - No li dóna tanta importància a l'històric personal.
 - Paga la seva part en grup familiar.

3.2 Disseny conceptual

3.2.1 Escenaris

3.2.1.1 Personatges

A partir dels perfils d'usuari definits hem creat dos personatges per situar millor el nostre escenari.

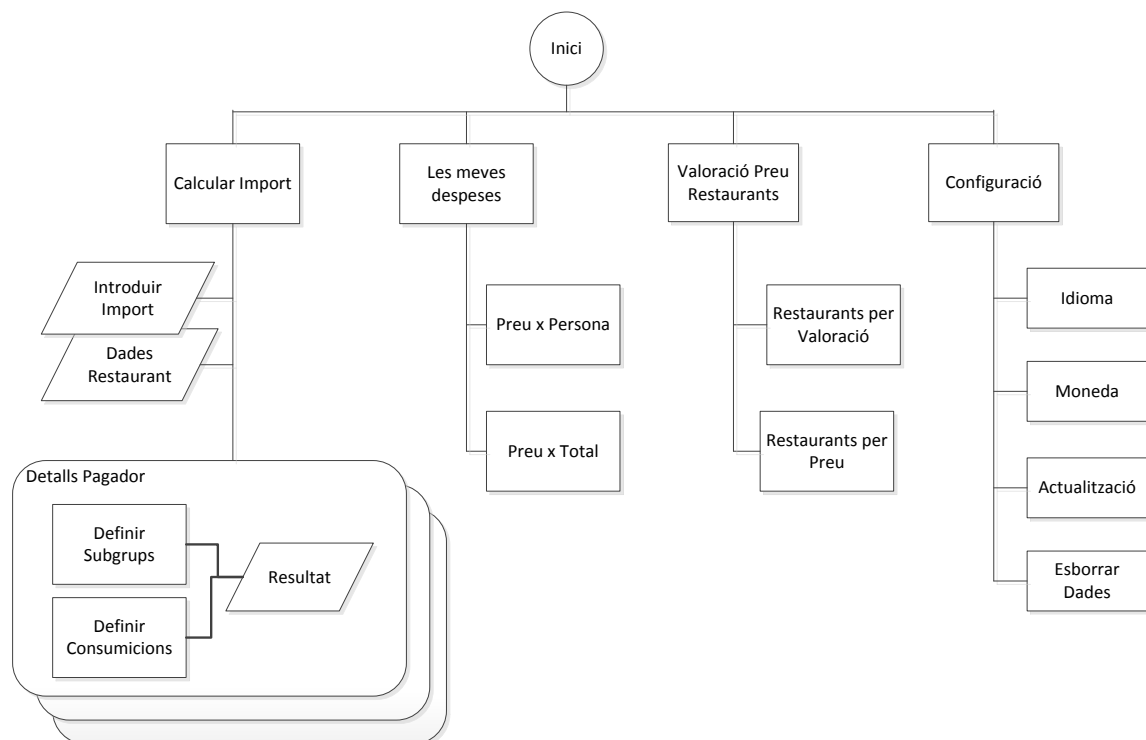
Perfil	Nom	Edat	Habilitats	Objectius
Menor 35	Joana	26	Facilitat per a la utilització d'aplicacions mòbils.	-Calcular el repartiment en un grup normalment compost per pagadors individuals o en parella. -Detallar la quota de pagament de cada grup: qui ha begut alcohol, postres, etc. -Visualitzar històric de despeses. -Visualitzar preus i valoració restaurant.
Major 35	Martí	41	No coneix totes les opcions d'interacció amb les aplicacions.	-Calcular el repartiment per a diferents grups: parelles, individual i família. -Visualitzar històric de despeses. -Visualitzar preus i valoració restaurant.

3.2.1.2 Definició d'escenaris

Escenaris	Persona	Context	Tasques i funcionalitats
Escenari 1a	Joana	Es troba en un restaurant i acaben de donar el compte total.	-Ha d'introduir el import total a l'aplicació. -Ha d'introduir informació del restaurant: nom, valoració. -Ha de seleccionar el número d'individus pel repartiment. -Opcionalment, detallarà diferències entre individus: els que han pres postres, 2n plat, etc. -Obtindrà el preu que ha de fer càrrec cada pagador.
Escenari 1b	Martí	Es troba en un restaurant i acaben de donar el compte total.	-Ha d'introduir el import total a l'aplicació. -Ha d'introduir informació del restaurant: nom, valoració. -Ha de seleccionar els diferents subgrups dins del grup principal: famílies amb nens, parelles, etc. -Obtindrà el preu que ha de fer càrrec cada pagador.
Escenari 2a	Joana	Pot trobar-se en qualsevol lloc però vol informació sobre les seves despeses i/o restaurants habituals.	-Pot d'accedir a l'apartat de despeses. -Vol saber les seves despeses anteriors. -Vol saber la tendència de preus dels restaurants. -Pot d'accedir a l'apartat de restaurants per triar on anirà a dinar. -Vol saber quins són els restaurants més valorats. -Vol saber quins són els restaurants més econòmics.
Escenari 2b	Martí	Pot trobar-se en qualsevol lloc però vol informació sobre les seves despeses i/o restaurants habituals.	-Ha d'accedir a l'apartat de restaurants per triar on anirà a dinar. -Vol saber quins són els restaurants més valorats. -Vol saber quins són els restaurants més econòmics. -Vol informació sobre ofertes.

3.2.2 Fluxos d'interacció

A continuació es mostra el diagrama de flux d'interacció de l'aplicació:



3.3 Prototipatge

3.3.1 Criteris de disseny: Principis heurístics.

Pel disseny del prototipus ens hem basat en els principis d'avaluació heurística desenvolupats pel reconegut consultor en interacció persona-ordinador Jakob Nielsen. Aquest, postula en 10 premisses els principals criteris per a l'avaluació heurística d'interfície d'usuari.

A continuació s'enumeren els 10 principis d'avaluació publicats en el llibre *Usability Engineering* de Jakob Nielsen i com els tingut en compte pel disseny del nostre prototipus.

Principi heurístic	Tractament en prototipus
Visibilitat de l'estat del sistema:	A la part superior de la pantalla, mantenir el títol de l'aplicació i la informació sobre l'activitat que s'està realitzant.
Concordança entre el sistema i la realitat:	Utilitzar missatges i mostrar la informació en llenguatge pla i directe.
Control i llibertat per l'usuari:	Facilitat en la navegació per a les diferents activitats. Tornar sempre a l'activitat anterior evitant mostrar masses diàlegs.
Consistència i estàndards:	Utilitzar sempre les mateixes paraules i icones per representar les mateixes coses.
Prevenició d'errors:	Evitar que l'usuari pugui introduir dades errònies o deixar-se dades en blanc necessàries.
Reconèixer abans que recordar:	Utilitzar icones fàcils de reconèixer. No donar res per entès.
Flexibilitat i eficiència d'ús:	Evitar de fer un assistent (<i>wizard</i>), complet, que facilitant les tasques a l'usuari novell posaria nerviós a un usuari estàndard. Possibilitat de fer un assistent pel primer cop d'utilització i després anul·lar-ho.
Disseny minimalista:	Minimitzar imatges, moviments i text que no aporten res per a l'usuari o l'aplicació.
Facilitar a l'usuari a reconèixer, diagnosticar i recuperar errors:	Mostrar informació a l'usuari en pantalla inferior de tasques que s'han de realitzar i poden crear errors pèrdua d'integritat.
Ajuda i documentació:	Facilitar una pantalla d'ajuda a l'usuari. Possible creació d'un assistent que expliqui la funcionalitat de l'aplicació.

3.3.2 Prototipus d'alta fidelitat estàtic

3.3.2.1 Menú principal

0. Menú d'inici



Descripció

- Pantalla principal de l'aplicació.
- Ens permet accedir a la resta de funcionalitats del programa:
 - . Calculadora: funció principal per calcular la part de cadascú.
 - . Favorits: Per veure la informació dels restaurants favorits i d'una classificació per preu.
 - . Estadístiques: Historial de despeses d'usuari.
 - . Configuració: Personalització i configuració bàsica de l'aplicació

3.3.2.2 Funció calculadora

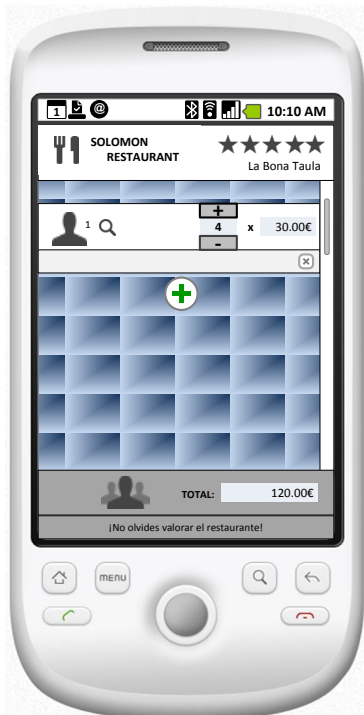
0.1.1. Calculadora: Informació general



Descripció

- Ens permet introduir les principals dades que necessitem pel nostre objectiu:
 - . Nom del restaurant: L'utilitzem per la classificació i valoració de restaurants.
 - . Import compte: Definim el import del total del compte del restaurant.

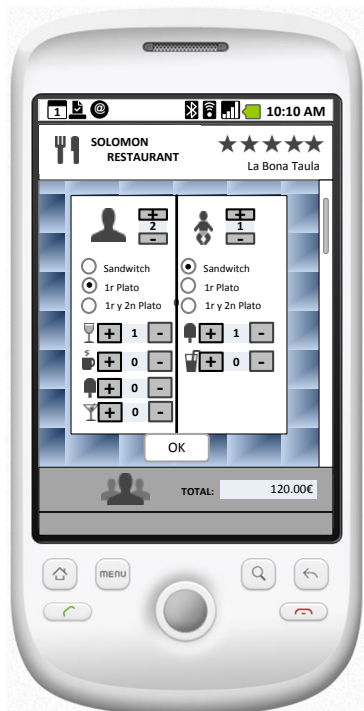
0.1.2. Calculadora: Resultat repartició



Descripció

- Ens permet detallar la informació de grups de pagadors i ens mostra el resultat per grup.
- . Permet valorar el restaurant.
- . Definir quans subgrups hi ha. En el nostre cas serien 4 individus pagadors.
- . Podem afegir nous grups de pagadors.
- . Mostra informació del total.
- . Mostra informació de les tasques pendents. En el nostre cas encara no s'ha valorat el restaurant.
- . Podrem accedir als detalls del subgrup.

0.1.3. Calculadora: Detall subgrup

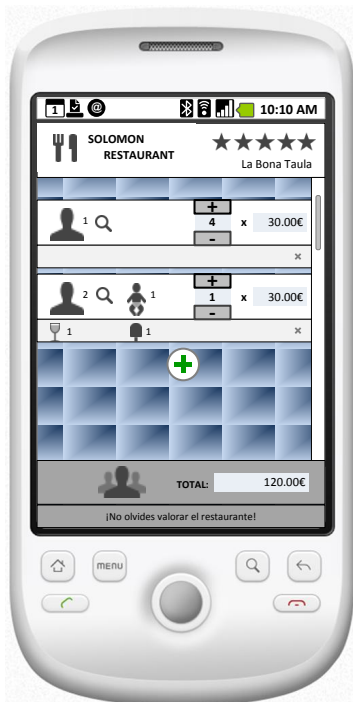


Descripció

- Ens permet detallar la informació de grups de pagadors i ens mostra el resultat per grup.
- . Permet valorar el restaurant.
- . Definir quans subgrups hi ha. En el nostre cas serien 4 parelles.
- . Podem afegir nous grups de pagadors.
- . Mostra informació del total.
- . Mostra informació de les tasques pendents. En el nostre cas encara no s'ha valorat el restaurant.
- . Podrem accedir als detalls del subgrup.

0.1.2. Calculadora: Resultat repartició

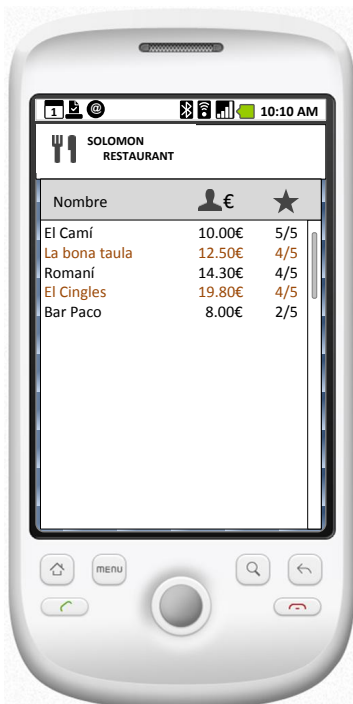
Descripció



- Tornem al resultat de la repartició després d'afegir un nou subgrup.
- Com exemple s'ha afegit un nou subgrup format per una parella i un nen, que a més tenen begudes i postres. Així tenim 4 individus pagadors i una parella amb nens.
- Els resultats per subgrup s'haurien d'actualitzar al nou import (en el nostre prototip no hem canviat aquests valors).
- El total no ha de variar.

3.3.3 Classificació favorits

0.2.1. Favorits

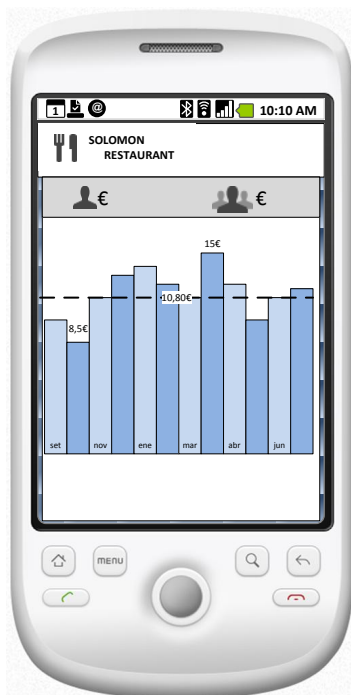


Descripció

- Mostra informació de les nostres entrades pels càlculs realitzats: preu per persona i valoració.
 - . Permetrà ordenar pel preu per persona.
 - . Permetrà ordenar per la valoració feta.

3.3.4 Historial despeses

0.3.1. Despeses



Descripció

- Mostra l'historial de despeses per l'últim any des del mes anterior.
- . Permetrà veure la tendència del preu per persona.
- . Permetrà veure la despesa total de cada mes.

3.4 Avaluació

3.4.1 Qüestionari sobre el test

Per a la realització del qüestionari ens hem basat en l'avaluació heurística, centrant les nostres preguntes en els principis plantejats per Jakob Nielsen, per així assegurar, encara que no sigui per experts, que hem seguit aquests criteris per a la realització del nostre prototipus.

Una part de les preguntes s'han fet per fer una valoració quantitativa i d'altres s'han plantejat de forma qualitativa i suficientment obertes perquè l'usuari pugui aportar les seves impressions lliurement amb l'objectiu de poder aplicar modificacions i millores. Les preguntes qualitatives es realitzarien en forma d'entrevista.

A continuació es descriuen el recull de preguntes:

Part I: Qüestionari quantitatiu

1. Ha trobat informació suficient en els diferents menús i pantalles per poder realitzar les tasques de manera satisfactòria?

Molt

Suficient

Poc

2. Ha pogut realitzar sense dificultats els problemes plantejats per a cada tasca?

Molt bé

Amb dificultats

No he pogut

**3. Creu que el temps per a obtenir els resultats de les tasques ha estat adequat?**

Molt

Suficient

Poc

4. S'ha perdut en algun moment entre els menús i pantalles de l'aplicació?

Mai

A vegades

Molt

5. L'aplicació ha donat cap error incomprensible o ha fet un funcionament incorrecte?

Sí

No

Part II: Qüestionari qualitatiu?

1. Troba que l'aplicació és pràctica? S'entenen totes les seves funcionalitats?
2. El disseny de l'aplicació, entre les diferents pantalles, és consistent, es veu com un únic programa o una suma d'eines?
3. Les icones i les accions que es poden fer són prou explicatives? Necessiten informació extra?

3.4.2 Llistat de tasques a realitzar

Les tasques que els usuaris haurien de realitzar s'han definit amb l'objectiu de testear totes les funcionalitats de l'aplicació a més de proves de robustesa per poder trobar possibles errors en l'entrada de dades.

Tasca	Nom	Descripció del procés
1	Càlcul simple	Es demanaria a l'usuari que a partir d'un import, calculés el què hauria de pagar cadascú per un grup format per 2 individus i 4 parelles. L'usuari haurà de valorar el restaurant.
2	Càlcul complex	Es demanaria a l'usuari que a partir d'un import, calculés el què hauria de pagar cadascú per un grup format per 1 individu que s'ha pres cafè i copa, 1 individu que ha demanat 2n plat, 2 parelles sense cap extra, 1 parella que ha demanat postra i dues famílies de 2 adults i 2 nens en el qual els nens han demanat gelat. L'usuari no valorarà el restaurant.
3	Consulta Despeses	L'usuari hauria de consultar les despeses acumulades i seleccionar per cost per persona i total.
4	Consulta Restaurants	L'usuari consultaria el registre de restaurants i hauria d'ordenar per preu per persona i per valoració.
5	Configuració	L'usuari accediria al panell de configuració de l'aplicació. Haurà d'esborrar les dades.
6	Proba Robustesa	Es demanarà a l'usuari que introdueixi informació incompleta o errònia per al càlcul.

3.4.3 Resultat de l'avaluació

Hem volgut realitzar l'avaluació del nostre prototipus estàtic amb alguns dels usuaris amb els que vàrem fer les entrevistes amb profunditat, així ells han pogut avaluar si l'aplicació que es vol

implementar podrà resoldre els problemes plantejats en l'entrevista. Es pensa que així es pot tenir un cert grau d'iterativitat en el disseny del nostre prototipus.

Per realitzar l'avaluació s'ha imprès a escala 1:1 els models estàtics i se li ha demanat als usuaris que provessin de realitzar les tasques encomanades fent la navegació de forma manual, intercanviant les impressions per a cada estat de l'aplicació, i explicant-li de forma verbal com actuaria el programa.

Posteriorment s'ha realitzat el test i l'entrevista qualitativa. Les conclusions estretes i les millores plantejades, han estat les següents:

Arguments Usuari	Proposta de millora
Algunes pantalles no donen la suficient informació a l'usuari: pantalla introducció import, pantalla detalls consum.	Afegir títols que descriguin el procés que s'està realitzant.
El títol de l'aplicació no és representatiu.	Es planteja canviar de títol de l'aplicació. Hi ha dues propostes: Restaurant in Group i Restaurant Partium.
Es perd la visió de l'import total en la pantalla de càlcul.	Es pot posar aquesta informació a dalt en comptes d'estar al peu com ara.
No s'entén com es fa la divisió. Un cop explicada és veu molt fàcil de fer, però de primeres no.	Fer un assistent que ajudi a l'usuari a fer servir per primer cop el programa. Després deixar-ho opcional ja que quan se sap fer, l'assistent és innecessari i lent.
Al menú principal és millor que aparegui la icona d'ajuda i deixar la de configuració per que aparegui quan es premi el botó de menú.	Es valora fer aquest canvi. Es podria canviar pel botó de l'assistent.
Ens hem adonat que les despeses personals no estan contemplades, ja que són per tot el grup.	Identificar a la pantalla de càlcul al subgrup propietari del mòbil per desar les seves despeses.

4 Implementació del producte

4.1 Arquitectura de l'aplicació

L'arquitectura de la nostra aplicació es basa en el model de programació per capes, és a dir, la separació de la interfície d'usuari, la lògica de l'aplicació i les dades.

Aquesta arquitectura de 3 capes ens ha resultat molt flexible a l'hora de fer noves adaptacions i canvis. A més, ens facilita tenir una estructura molt endreçada i entenedora, la qual ens permet entendre la lògica de l'aplicació molt més ràpidament.

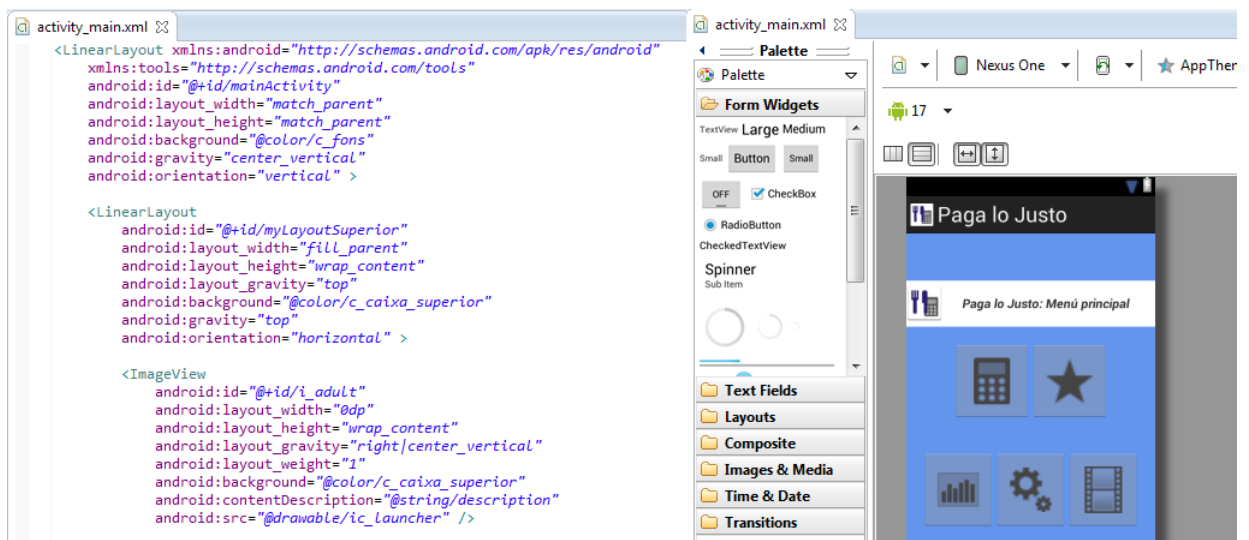
A continuació es descriuran les característiques principals per a cada capa del nostre model:

4.1.1 Capa d'usuari i presentació

En la capa d'usuari s'han definit els dissenys de cada pantalla de la nostra aplicació. Aquests dissenys s'han realitzat mitjançant la definició dels seus elements en XML, el qual permet obtenir de forma directa la vista prèvia, essent molt còmode pel dissenyador.

El disseny en XML està considerat com una bona pràctica a l'hora de fer aplicacions en Android ja que separa de forma natural la capa lògica de la interfície d'usuari. Per altre banda, cal comentar que per alguns objectius hem tingut que realitzar en temps d'execució alguns dissenys de pantalla, per exemple, per dibuixar els gràfics, ja que d'altre manera no és possible degut a la necessitat de crear-los en temps d'execució.

Figura 1: Definició interfície usuari en XML



A més de les pantalles, ha estat una pràctica habitual en el projecte la definició de s'han definit en XML components formats per diferents controls o vistes, com poden ser els elements d'un control ListView.

Figura 2: Exemple de definició d'un element generic de un ListView


```

<ImageView
    android:id="@+id/i_nen"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical|right"
    android:layout_weight="2"
    android:contentDescription="@string/description"
    android:src="@drawable/baby" />

<TextView
    android:id="@+id/foratNens"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:layout_weight="1"
    android:gravity="right|center_vertical" />

<ImageView
    android:id="@+id/btnAddSubGrup"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_gravity="center_vertical|center_horizontal"
    android:layout_weight="1"
    android:clickable="true"
    android:contentDescription="@string/description"
    android:src="@drawable/edit_add" />

```

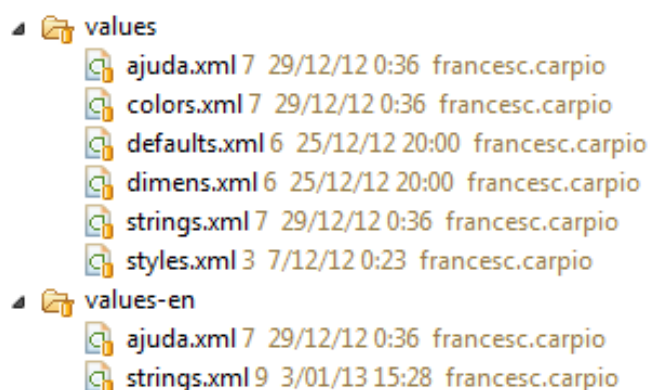


Finalment, cal remarcar la utilització del fitxers dels recursos que defineixen diferents tipus de valors per a l'aplicació: text, mides, colors, etc.

Utilitzar els fitxers de recursos de forma correcte ens a permès dos objectius importants:

- Mantenir un estil per a totes les pantalles de l'aplicació, fitxers: *colors.xml*, *dimens.xml* i *styles.xml*.
- Separar tots els textos que apareixen de la lògica del programa facilitant l'adaptació de diferents idiomes per a l'aplicació de manera senzilla i ràpida, fitxers: *ajuda.xml*, *strings.xml* i *defaults.xml*

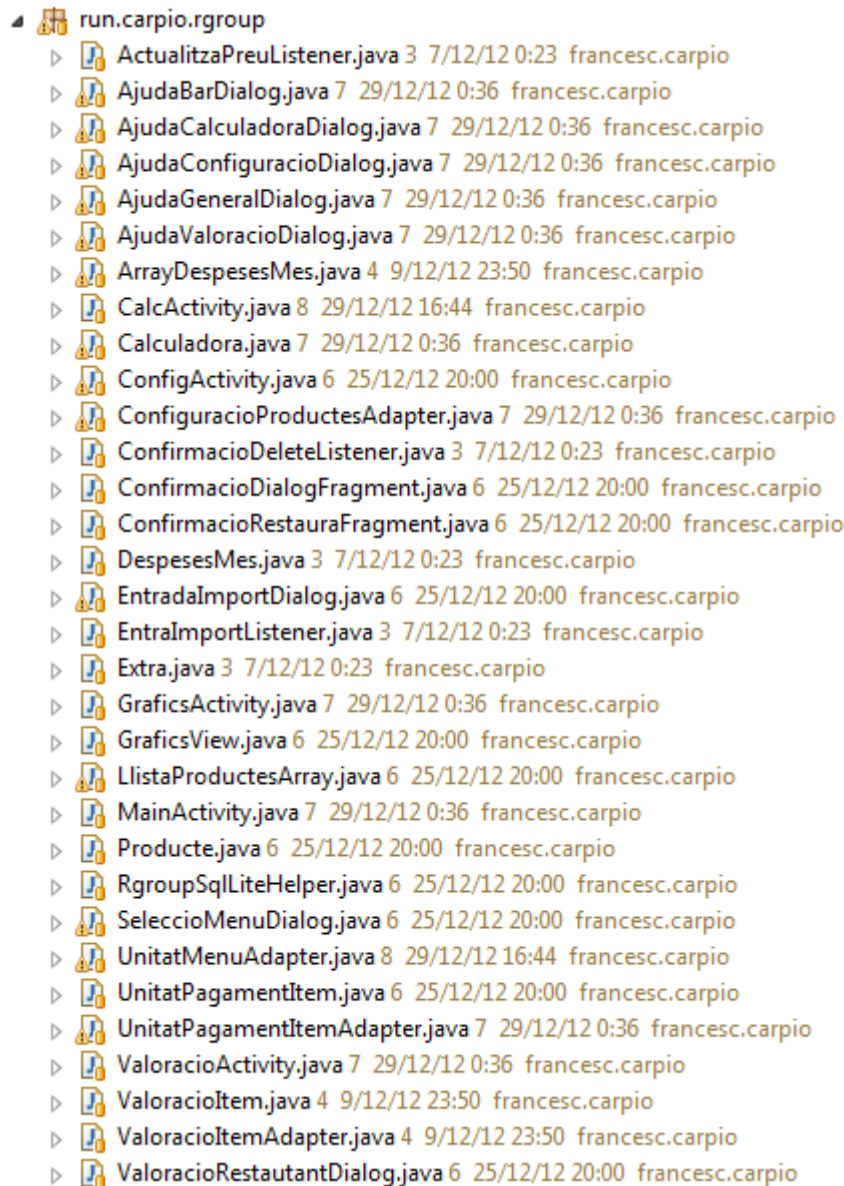
Figura 3: Fitxers de recursos



4.1.2 Capa lògica

La capa lògica està formada per totes les classes en codi Java que implementen les diferents tasques que ha de realitzar l'aplicació.

Figura 4: Classes que formen part de la capa lògica de l'aplicació



Per a cada disseny de pantalla o *Layout* s'ha associat una classe que proporciona la lògica d'aquesta pantalla: càrrega de configuració, interacció amb l'usuari, operacions, etc. Així mateix s'ha fet per a cada conjunt de controls o *GroupView*, com els elements del llistats o vistes compostes.

Dins d'aquesta capa cal destacar la classe *Calculadora* que realitza totes les operacions pel càlcul de la divisió proporcional per a cada subgrup.

4.1.3 Capa de dades

La capa de dades ens proporciona la gestió de la comunicació amb la base de dades SQLite del sistema.

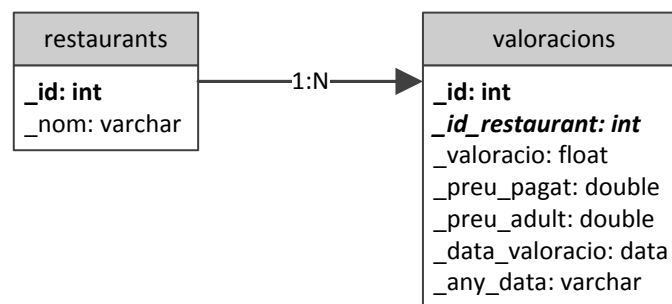
Les operacions principals que realitza són tres:

- La creació de la base de dades i les taules en cas de no existir.
- La inserció de dades a les taules.
- Les lectures de dades a les taules.

La base de dades de l'aplicació té com a funció desar les dades de valoració dels restaurants per part de l'usuari, així com guardar dades sobre el consum.

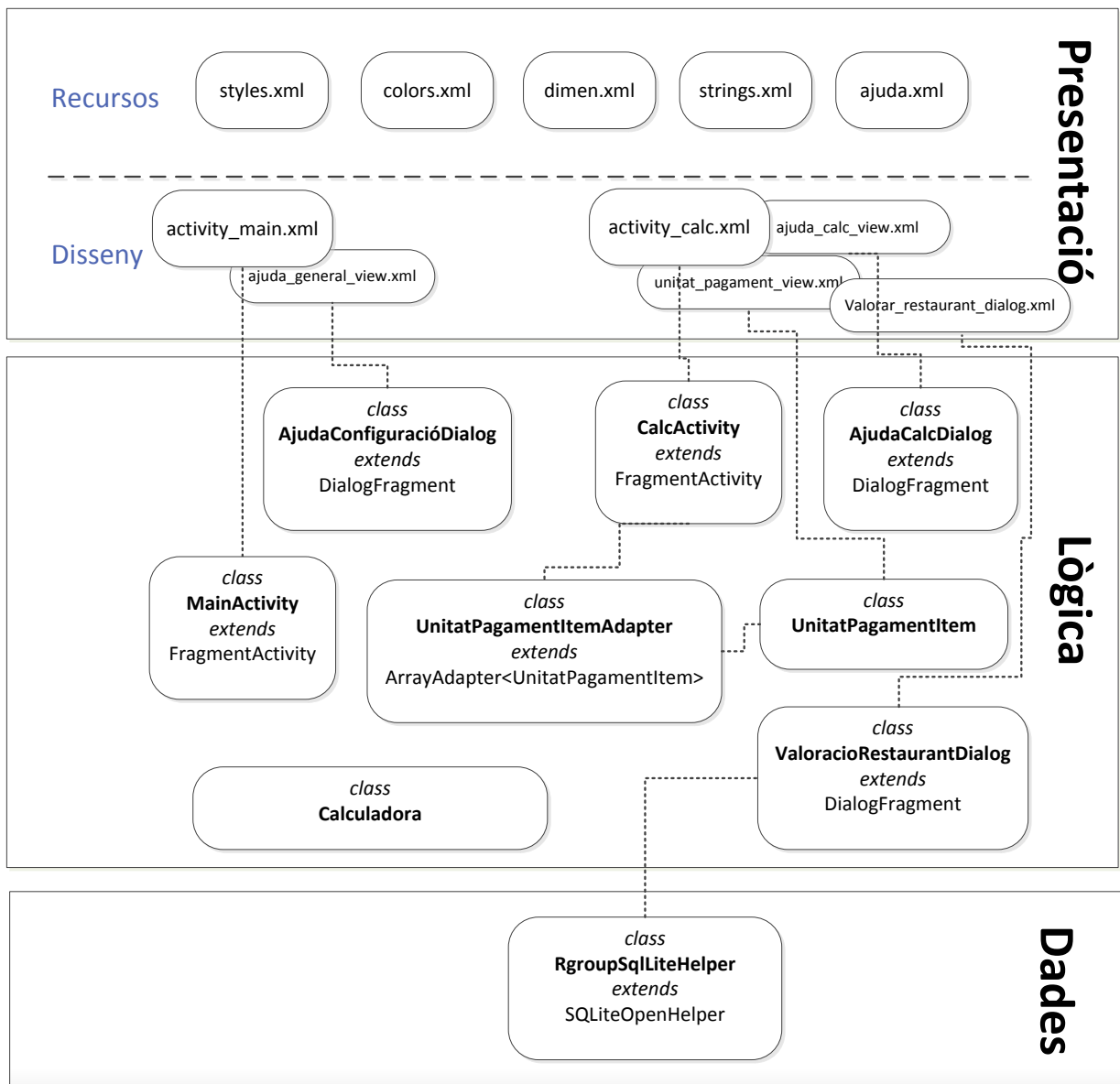
S'ha separat en dues taules per permetre la indexació del nom del restaurant amb una entrada de text de tipus *AutoCompleteTextView*. També la separació de en dues taules compleix més amb el paradigma del model entitat-relació.

Figura 5: Base de dades de l'aplicació



A continuació es mostra el diagrama de l'arquitectura reduït amb els principals elements de l'aplicació:

Figura 6: Elements principals de l'arquitectura de l'aplicació



4.2 Entorn de treball

El nostre entorn de treball ha estat principalment l'IDE Eclipse en la seva versió JUNO, amb els paquets SDK i ADT d'Android.

L'entorn proporcionat per Eclipse és molt complet facilitant al programador i al dissenyador una eina indispensable per a la implementació de qualsevol aplicació per a Android.

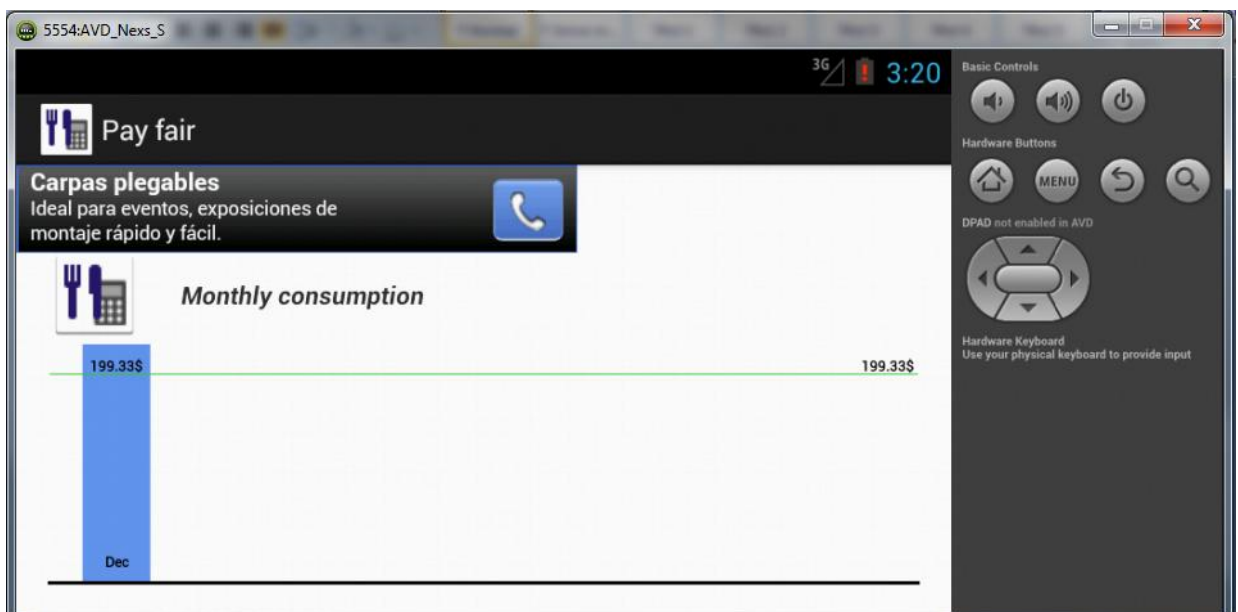
Les eines per depurar l'aplicació s'han utilitzat intensament amb excel·lents resultats, sobretot quan s'ha utilitzat un dispositiu físic per ha fer aquestes depuracions, ja que el rendiment del programa era molt superior que al utilitzar un dispositiu virtual.

Figura 7: Eclipse JUNO amb SDK i ADT d'Android



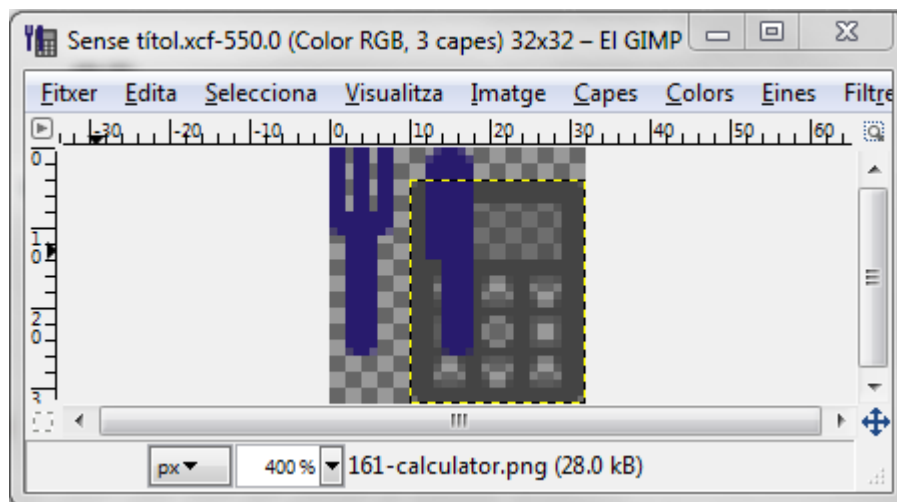
Els dispositius virtuals (AVD) han estat útils per algunes proves com el disseny d'interfícies quan volíem aplicar diferents grandàries o el canvi d'idioma. Tot i això, hem trobat que el sistema s'enrellentia molt amb l'ús dels AVD.

Figura 8: AVD Nexus S. Pantalla horitzontal i idioma anglès.



Altres eines utilitzades pel desenvolupament de l'aplicació han estat el Gimp i l'Inkscape, principalment per a la creació d'alguna icona o fons, encara que, la majoria d'icones s'han estret de la llibreria *glyphish-icons*.

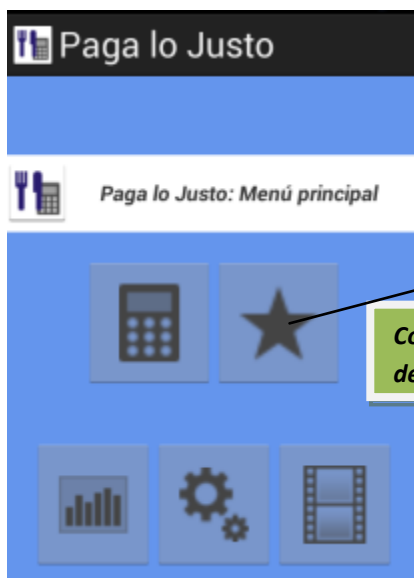
Figura 9: Edició d'Icones amb GIMP



4.3 El codi

Com ja s'ha fet esmena a l'arquitectura, la majoria de les classes del projecte implementen la lògica de cada pantalla, que normalment es resumeix en gestionar la interacció amb l'usuari.

Per exemple, per el disseny *activity_main.xml*, s'ha implementat la classe *MainActivity.java*. Per a aquest tipus de classes no hem tingut que preocupar-nos gaire de l'eficiència del codi, ja que les accions són mínimes i l'única dificultat ha estat la part de disseny.



Control View vinculat a una acció definida en la classe Java

```

/**
 * Obre l'activity valoració
 *
 * @param view
 *         Control que fa l'acció
 */
public void obrirValoracions(View view) {
    Intent i = new Intent(this, ValoracioActivity.class);
    this.startActivity(i);
}

/**
 * Obre l'activity configuració
 *
 * @param view
 *         Control que fa l'acció
 */
public void obrirConfiguracio(View view) {
    Intent i = new Intent(this, ConfigActivity.class);
    this.startActivity(i);
}

```

No ha estat així en d'altres pantalles com pot ser la principal de la calculadora. Per aquesta part, s'ha tingut en compte aspectes d'eficiència, ja que hem de pensar que un dispositiu mòbil té uns recursos limitats, i no podem sobrecarregar el rendiment del programa. Si ho féssim, l'usuari patiria temps de

resposta massa llargs, provocant la seva frustració i, en conseqüència, deixar d'utilitzar la nostra aplicació. Un cas on hem pogut comprovar aquesta limitació del dispositiu, ha estat en la creació d'entrades de proves per la base de dades. Hem vist com aquesta operació podia trigar entre 30 a 40 segons, donant la sensació que l'aplicació no responia. S'ha de dir que per aquest cas no s'ha optimitzat res del codi perquè aquesta funcionalitat és només per fer test i depuració.

Per aquestos casos el que hem fet ha estat estalviar al màxim en la creació de nous objectes així com evitar transaccionalitats amb l'objectiu que anés executant el codi mentre l'usuari ha de fer una acció com l'entrada d'una dada.

Un altre pràctica de treball ha estat minimitzar l'ús de les clàusules `try {} catch {}`, ja que consumeixen més recursos. Per fer-ho, hem preferit comprovar en temps d'execució els valors vàlids mitjançant condicionals.

Figura 10: Evitem l'ús de `try-catch` amb condicionals

```
//marge top i botton
float alcaMaxim = totalY - alcaProp * 2;

//Comprovem si s'ha instanciat l'array
if(this.arrayDespesesMes == null){
    canvas.restore();
    return;
}

String moneda = getResources().getString(R.string.moneda);
double despesa = 0;
for(int i = 0; i < arrayDespesesMes.size(); i++){
    //dibuixar mesos
```

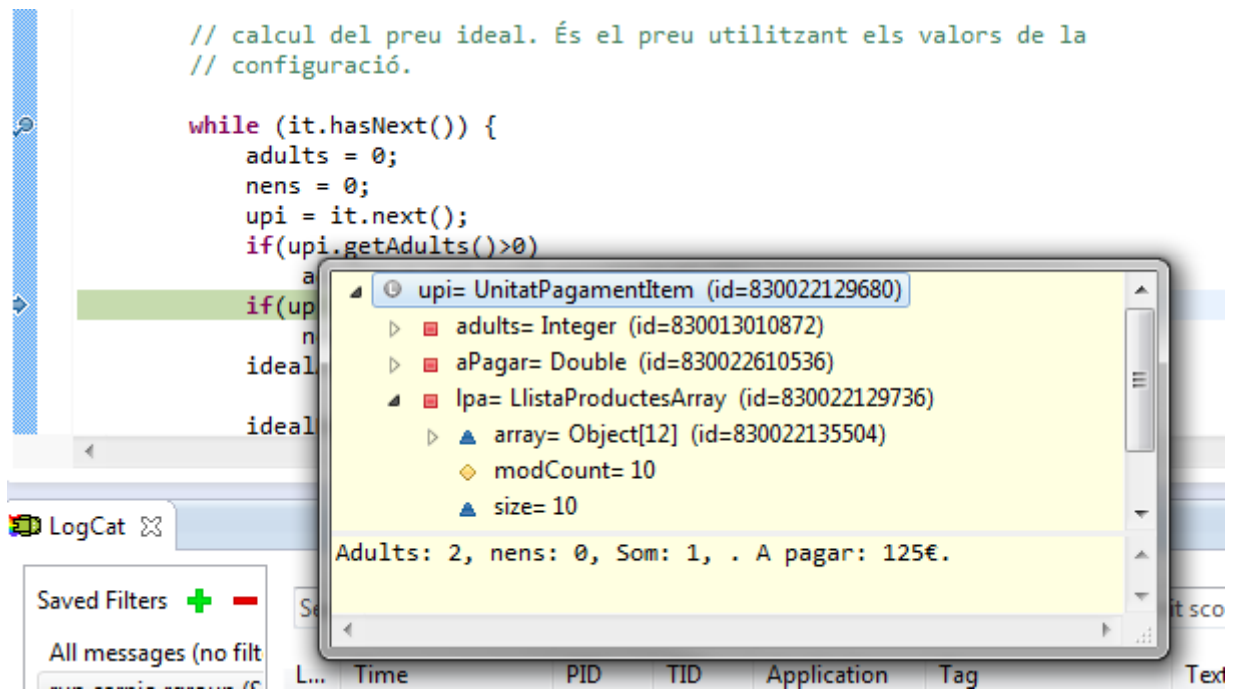
4.4 La depuració del programa

Un dels punts més importants en el desenvolupament és la depuració del programa. Les eines proporcionades per l'Eclipse són molt potents en aquest aspecte.

Hem utilitzat intensivament l'ús de la perspectiva *debug*, la depuració per passos, l'exploració d'objectes, etc.

L'únic inconvenient trobat al utilitzar aquestes eines ha estat la baixada general del rendiment del nostre equip. Això ha provocat pèrdua de productivitat en la creació del producte fins que es va començar a utilitzar dispositius físics externs per a la depuració. Aquest fet a millorat de forma substancial el ritme de treball en la creació del nostre projecte.

Figura 11: Perspectiva debug d'Eclipse



Un altre tècnica utilitzada per a la depuració ha estat la utilització de la classe `android.util.Log`, que ens ha permès la creació de traces pel nostre programa de forma molt senzilla i suficient pel nostre propòsit.

Figura 12: Utilització de les traces

```

import android.os.Bundle;
//Classe per a crear traces
import android.util.Log;
...
//Etiqueta per a la traça
private static final String TAG = "ConfiguracioProductesAdapter";
...
//Escriu traça nivell Debug
Log.d(TAG,"Abans: " + menuAdult);
if(menuAdult.length()>0)
    menuAdult+=",";

```

Figura 13: Visualització de les traces

Level	Time	PID	TID	Applicati...	Tag	Text
I	01...	786	786	run.c...	Choreographer	Skipped 643 frames! The application may be doing its main thread.
D	01...	786	786	run.c...	ConfiguracioProductesAdapter	Abans: primerplat,refresc,postre,cafe
D	01...	786	786	run.c...	ConfiguracioProductesAdapter	Despres: primerplat,refresc,postre,cafe,segonplat
D	01...	786	786	run.c...	Producte	Adults: 0, nens: 0, Nom: Starter, . Preu: 5.5€, l lse, Menu Adult: false
D	01...	786	786	run.c...	nouProducte	Adults: 0, nens: 0, Nom: Starter, . Preu: 5.5€, l

4.5 La configuració

Dins del desenvolupament de l'aplicació, hem donat una gran importància a la part de configuració d'usuari, ja que ens permet crear un programa molt flexible.

Per una banda tenim les dades per defecte de l'aplicació, allotjades al fitxer de recursos *default.xml* i que són independents de l'idioma.

Analitzarem els valors que hi ha en el fitxer *default.xml*, ja que és de màxima importància per al nostre programa.

- Primer definim quines *claus* o productes formen part per defecte del menú d'adult o del menú infantil.

```
<!-- Definició de menus -->
<string name="menu_adult">primerplat,refresc,postre,cafe</string>
<string name="menu_infantil">entrepà,refresc,gelat</string>
```

- Definim una llista amb les claus dels productes. Aquest *array* es carrega en temps d'execució. Això vol dir que podem canviar el número de productes que apareixen al menú dinàmicament només canviant aquest *array*. Cal dir que aquestes claus no es presenten per pantalla, pertanyen a la lògica interna.

```
<!-- Claus pels productes -->
<string-array name="claus_menu">
  <item>entrant</item>
  <item>primerplat</item>
  <item>segonplat</item>
  <item>entrepà</item>
  <item>refresc</item>
  <item>alcohol</item>
  <item>postre</item>
  <item>gelat</item>
  <item>copa</item>
  <item>cafe</item>
</string-array>
```

- A continuació es mostren els preus dels productes. L'*array* ha de coincidir en grandària i posició amb l'anterior, ja que el programà associarà a cada clau del menú un preu.

```
<!-- Preus pels productes -->
<string-array name="preus_menu">
  <item>5.5</item>
  <item>7</item>
  <item>10</item>
  <item>4</item>
  <item>1.5</item>
  <item>2.20</item>
  <item>3.5</item>
  <item>2</item>
  <item>5</item>
  <item>1.3</item>
</string-array>
```

Des del programa es carregaran els ítems del menú de forma dinàmica, a més de la traducció per a l'idioma per defecte que s'allotja en el fitxer *strings.xml*:

```
Resources r = activity.getResources();
//claus_menu : son les claus de cada article del menu
claus_menu = r.getStringArray(R.array.claus_menu);
//preus_menu : els preus per defecte de cada article
preus_menu = r.getStringArray(R.array.preus_menu);
//noms_menu : el nom en l'idioma del article
noms_menu = r.getStringArray(R.array.menus);
```

En conclusió, podem canviar totalment el menú i el seu idioma només fent modificacions en els fitxers de recursos *default.xml* i *strings.xml*

Finalment, l'usuari podrà modificar la disposició del menú, quantitat i preu ja que desem les preferències de forma permanent al dispositiu amb l'ús de la classe *android.content.SharedPreferences*, i en tot moment es poden carregar les dades per defecte que es troben en el fitxer *default.xml* i mai l'usuari les modifica.

```
//Carregem les preferències de l'usuari
SharedPreferences preferences = activity.getSharedPreferences(
    "preferencies", mode);

//Recuperem el valor per defecte
valorDefault = activity.getResources()
    .getString(R.string.menu_infantil);

//Si l'usuari no ha canviat les seves preferències
//agafem les que hi ha per defecte
String menuInfantil = preferences.getString("menu_infantil",
    valorDefault);
```

4.6 Comentarís i documentació

Bàsicament el codi s'ha documentat utilitzant dos tipus de comentaris: els comentaris al codi i la documentació de la classe.

4.6.1 Comentarís del codi

Els comentaris al codi s'han introduït per facilitar la comprensió de les parts més complexes de la programació.

És per aquest motiu que s'ha intentat només afegir els comentaris més necessaris, i s'ha treballat més en utilitzar noms de variables i mètodes aclaridors, amb una estructura clara, que faciliti al programador el seu enteniment amb el mínim de comentaris extra.

Figura 14: Introducció mínima de comentaris al codi.

```
// preu per defecte si l'usuari no defineix cap
String preu = "0.00";

// mostrem el dialog per introduir l'import
FragmentManager fm = getSupportFragmentManager();
EntradaImportDialog eid = EntradaImportDialog.newInstance(preu);
eid.show(fm, "dlg_edit_name");

// definim les unitats de pagament i la seva visualització
myListView = (ListView) findViewById(R.id.myListView);
unitatItems = new ArrayList<UnitatPagamentItem>();
int resID = R.layout.unitat_pagament_view;
upAdapter = new UnitatPagamentItemAdapter(this, resID, unitatItems,
    myListView);
upAdapter.setCalcActivity(this);
myListView.setAdapter(upAdapter);
```

4.6.2 Documentació de les classes

Per a la documentació de les classes hem utilitzat l'estàndard Javadoc. Per aquest tipus de documentació s'ha tingut cura de comentar totes les classes, mètodes i atributs públics.

La motivació a realitzar la feina de documentar en Javadoc ha estat perquè és clarament més útil per entendre la lògica global del programa aquest tipus de documentació que els comentaris al codi, que depenen més de les destreses del programador. Per aquest motiu creiem necessari presentar el Javadoc de l'aplicació juntament amb tot el codi.

Figura 15: Índex documentació de "Paga lo Justo"

The screenshot shows the Javadoc index for the 'Paga lo Justo' application. The interface is divided into two main sections: a list of classes on the left and a detailed view of the 'Package run.carpio.rgroup' on the right.

Package run.carpio.rgroup

Interface Summary

Interface	Description
ActualitzaPreuListener	Interfície per poder enviar esdeveniments en l'actualització del preu total
ConfirmacioDeleteListener	Interfície que permet interactuar l'acció entre el DialogFragment i l'Activity
EntralImportListener	Permet la comunicació entre l'Activity i el Dialog

Class Summary

Class	Description
AjudaBarDialog	Defineix un dialog amb fajuda per a la pantalla "consum mensual"
AjudaCalculadoraDialog	Defineix un dialog amb fajuda per a la pantalla "Calculadora"
AjudaConfiguracioDialog	Defineix un dialog amb fajuda per a la pantalla "Configuració"
AjudaGeneralDialog	Defineix un dialog amb fajuda general
AjudaValoracioDialog	Defineix un dialog amb fajuda per a la secció valoració
ArrayDespesesMes<DespesesMes>	Array de despeses mes
CalcActivity	Activity on és realitza el càlcul de la divisió del Total
Calculadora	Realitza els càlculs necessaris per fer la divisió proporcional del pagament.
ConfigActivity	Activity que permet actuar sobre la configuració bàsica de l'aplicació. - Esborrar dades de la base de dades. - Restar els preus i el menú.
ConfiguracioProductesAdapter	Implementa la lògica per a cada element de la configuració d'un producte.
ConfirmacioDialogFragment	Dialog per confirmar l'esborrament de tots els registres de la base de dades.

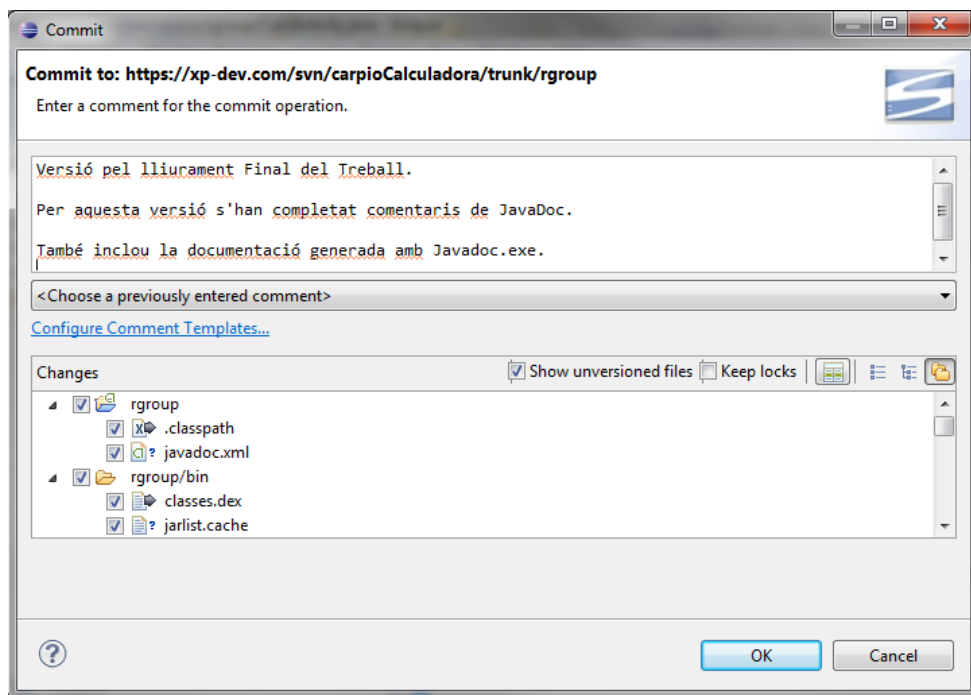
4.7 Control de versions

Per al control de versions hem fet servir el programari lliura sota llicència Apache/BSD de *Subversion*.

Aquest sistema ens ha proporcionat garanties a l'hora d'anar incorporant funcions a la nostra aplicació amb seguretat, per no perdre dades, o més important, per no desbaratar codi vell correctament implementat, amb codi nou erroni.

El programa s'ha instal·lat com a plugin en el IDE Eclipse, complementat pel portal *xp-dev.com* com a repositori on-line.

Figura 16: Actualització del repositori de versions des de Eclipse



Pel repositori Web *xp-dev.com* hem accedit a un compte gratuït que té, entre d'altres limitacions, que només es poden gestionar dos projectes. Pel nostre cas ha estat suficient .

Subversion també ens ha permet compartir el nostre treball amb el consultor de forma continua, encara que no instantània ja que només cometíem o fèiem les nostres aportacions quan s'havien modificat forces funcionalitats, ja que com a backup hem utilitzat altres mètodes.

Figura 17: Gestió de versions des de *xp-dev.com*

The screenshot shows the Subversion repository interface for 'carpioCalculadora'. At the top, it displays the repository name and URL: `https://xp-dev.com/svn/carpioCalculadora`. Below this, there are navigation tabs for 'Explore', 'Commits', 'Settings', and 'Import & Export'. A warning message states 'This repository has no backups' with a link to 'Upgrade your account to fix these warnings, or use backup'. The main content area is divided into two sections: 'carpioCalculadora /' and 'Commits for carpioCalculadora/'.

Entry	Size	Revision	Updated
branches		1	Sat 01 Dec, 2012 00:57:22 +0000
tags		1	Sat 01 Dec, 2012 00:57:22 +0000
trunk		9	Thu 03 Jan, 2013 14:28:23 +0000

Revision	Author	Committed	Message
9	francesc.carpio	Thu 03 Jan, 2013 14:28:23 +0000	Afegim icona al menú principal. Tot traduït a l'anglès.
8	francesc.carpio	Sat 29 Dec, 2012 15:44:56 +0000	Canvi Selector Color Item pagament. Tot Anglès.
7	francesc.carpio	Fri 28 Dec, 2012 23:36:43 +0000	Ajuda amb vídeo i traducció a l'anglès.
6	francesc.carpio	Tue 25 Dec, 2012 19:00:32 +0000	S'ha realitzat diferents proves de funcionalitat i robustesa, modificant S'afegeix l'ajuda i el menú contextual per accedir directament a l'ajuda.

4.8 Proves

Degut a que la finalitat del projecte ha estat arribar a publicar l'aplicació, durant tot el desenvolupament del mateix s'han fet diferents tipus de proves. Des de proves unitàries, de funcionalitat i finalment d'acceptació.

Per a les proves unitàries ens hagués agradat utilitzar les biblioteques de JUnit, però ho vàrem desestimar per falta de temps. Tot i això, pels processos més delicats de el programa, hem fet proves unitàries manualment, sense ser tant completes i pràctiques com les que es farien en JUnit, però suficient pel nostre objectiu.

Per a les proves de funcionalitat i robustesa es va definir primerament el conjunt de la descripció de les proves. Posteriorment es varen fer i, depenent del resultat, es va implementar l'acció de millora.

Tant les proves unitàries com les de funcionalitat i robustesa es vàrem fer per mi mateix ja que no disposàvem de més recursos, és a dir, de programadors o d'un equip d'experts per fer-les.

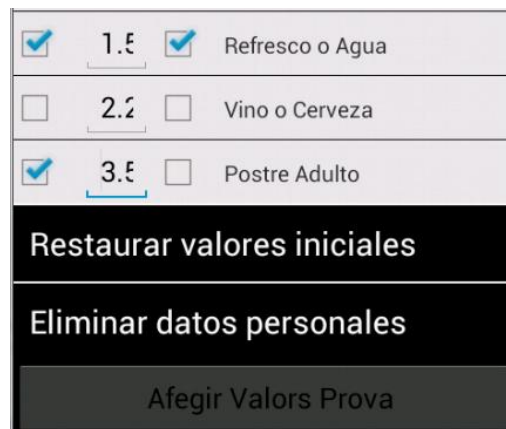
Per exemple, per provar la pantalla de valoració de restaurants i els gràfics dels consums mensuals, s'ha afegit una opció al menú de configuració que només apareix quan al fitxer de recursos indiquem com a cert (*true*), fer el test de l'aplicació.

Figura 18: Activem l'opció test al fitxer *defaults.xml*

```
<string name="test">true</string>
```

D'aquesta manera apareixerà a la pantalla de configuració una nova opció que executa el codi per introduir valors aleatoris a la base de dades.

Figura 19: Opció "afegir valors de prova" des de la pantalla configuració.



Afegir els valors de proves ens ha permet comprovar el correcte funcionament de les pantalles de valoració i consum mensual.

Figura 20: Pantalla "valoració"

Nombre	€	★
Casa Paco	22.06€	5/5
La Cueva De Paco	7.43€	5/5
Restaurant -1053798788	13.26€	4/5
Restaurant -1129413435	14.06€	1/5
Restaurant -1137524798	8.16€	0/5
Restaurant -1149968420	25.67€	0/5
Restaurant -1208595142	18.46€	3/5
Restaurant -1230794276	16.55€	2/5
Restaurant -123345230	7.94€	4/5
Restaurant -1242071982	10.53€	1/5
Restaurant -1307674879	9.31€	2/5

Figura 21: Pantalla "consum mensual"



Finalment, les proves d'acceptació es varem fer amb l'ajut de diferents usuaris finals, que vam provar totes les funcionalitats de l'aplicació i van donar les seves idees de millora, moltes de les quals s'han tingut present i s'han implementat.

Una qüestió remarcable a l'hora d'implementar les accions de millora és que posteriorment s'han hagut de fer noves proves, creant un efecte de depuració o millorament global de l'aplicació, però amb un cost superior en dedicació superior a l'estimat en principi.

A continuació es detallen les principals proves realitzades i les accions preses.

4.8.1 Proves de funcionalitat i robustesa

Conjunt Prova 1: Navegació pantalles		
Descripció	<i>Comprovar la navegació per totes les pantalles de l'aplicació, tant per entrar com per anar cap endarrere.</i>	
Secció	Tota l'aplicació	
Observacions	Mesures Correctores	Validat
Totes les accions realitzades han resultat satisfactòries.	Cap	<input checked="" type="checkbox"/>

Conjunt Prova 2: Menús d'ajuda		
Descripció	<i>Comprovar l'accés i els continguts de tots els diàlegs d'ajuda que té el programa, així com la prova de visualització del vídeo d'ajuda amb el dispositiu.</i>	
Secció	Tota l'aplicació	
Observacions	Mesures Correctores	Validat
Totes les accions realitzades han resultat satisfactòries.	Cap	<input checked="" type="checkbox"/>
La despesa total del mes no correspon a la introduïda en el primer ítem		
Prova:	Navegació per l'aplicació	
Observació	Mesures Correctores	Validat
Tot correcte, tant l'accés com després tornar a l'activity anterior.		<input checked="" type="checkbox"/>
La despesa total del mes no correspon a la introduïda en el primer ítem		

Conjunt Prova 3: Càlcul del repartiment		
Descripció	<i>Comprovar el càlcul de la repartició entre diferents pagadors. Comprovar la robustesa per a l'entrada de dades.</i>	
Secció	Calculadora	
Observacions	Resultats, Mesures Correctores	Validat
Es comprova la robustesa per a l'entrada de dades incorrectes: nombres negatius, decimals, nombres incorrectes amb lletres o signes, deixar-ho en blanc.	La robustesa és correcte perquè s'ha triat un control View que només permet l'entrada decimal positiva. Per una entrada en blanc l'import s'ha establert a 0. Es poden introduir unitats de pagament sense cap problema, així quan es canvia l'import l'operació és directe.	<input checked="" type="checkbox"/>
Comprovar el resultat de l'operació per a diferents unitats de pagament i comparant-lo amb càlculs realitzats manualment.	El càlcul sempre ha resultat correcte.	<input checked="" type="checkbox"/>
Afegir, eliminar unitats de pagament.	Tot correcte.	<input checked="" type="checkbox"/>

	Mai es permet eliminar quan només queda una única unitat de pagament. Al arribar al màxim d'unitats, definides en el fitxer default.xml, no es creen més i apareix un missatge informant a l'usuari.	
Modificació del menú estàndard adult i infantil.	Hem trobat que a l'afegir un producte en el menú infantil, encara que no hi hagi cap nen a la unitat de pagament, apareix la fletxa indicant que el menú és més car.	✘
	Hem trobat també que al anar cap endarrere amb la tecla return, les dades modificades sempre queden permanents i dóna l'efecte que s'haurien de cancel·lar.	
	Hem corregit l'error i ja no apareix la fletxa.	✔
	No s'ha fet cancel·lació de modificacions realitzades. Es contemplarà per futures versions.	✔
Conjunt Prova 4: Valoració del restaurant des de calculadora		
Descripció	<i>Comprovar l'entrada de dades pel nom del restaurant</i>	
Secció	Calculadora	
Observacions	Resultats, Mesures Correctores	Validat
Es comprova la robustesa per a una entrada en blanc. Es comprova funció tecles acceptar i cancel·lar	La robustesa és correcta. Quan no s'introdueix un nom, s'indica a l'usuari de l'error. Al acceptar desm les dades a la BBDD, de la valoració, el nom i les despeses. Al cancel·lar o retornar, es cancel·la l'acció.	✔

Conjunt Prova 5: Visualització de la valoració		
Descripció	<i>Comprovar si les dades mostrades són correctes. Comprovar si s'ordena adequadament per a cada columna.</i>	
Secció	Valoració	
Observacions	Resultats, Mesures Correctores	Validat
Es comproven les dades de la valoració: nom, preu per persona i valoració. Per a les proves s'ha creat un mètode per introduir valors aleatoris a la BBDD.	El preu per persona no és correcte.	✘
	L'operació del càlcul del preu mig per persona no era correcte. S'ha modificat l'operació de la classe Calculadora.	✔
Es comproven l'ordenació per columnes.	Tot correcte.	✔

Conjunt Prova 6: Visualització Despeses Mensuals

Descripció	<i>Comprovar la visualització del gràfic en tots dos modes: landscape i portrait. Comprovar les dades correctes.</i>		
Secció	Gràfic despeses.		
	Observacions	Resultats, Mesures Correctores	Validat
	Es comprova la visualització del gràfic en el dos modes. Per a les proves s'ha creat un mètode per introduir valors aleatoris a la BBDD.	Per a valors petits, el gràfic de barres es mostra per sota de l'eix x.	<input type="checkbox"/>
		S'han modificat el càlcul dels eixos i de les proporcions per fer el gràfic.	<input checked="" type="checkbox"/>
	Es comprova que els valors siguin correctes.	El valor que es mostra no és el de la primera unitat de pagament de la calculadora.	<input type="checkbox"/>
		S'ha comprovat que el valor és el total de la primera unitat de pagament i quan hi ha més d'un pagador d'aquesta unitat l'import no coincideix. Hem corregit aquest error.	<input checked="" type="checkbox"/>

Conjunt Prova 7: Configuració de l'aplicació			
Descripció	<i>Comprovar els paràmetres de configuració del menú. Comprovar robustesa. Comprova l'eliminació de dades personals.</i>		
Secció	Configuració		
	Observacions	Resultats, Mesures Correctores	Validat
	Es comproven les modificacions del menú per a la preferència d'usuari.	Els canvis són correctes.	<input checked="" type="checkbox"/>
	Es comprova la eliminació de les dades personals i valors per defecte.	Tot correcte.	<input checked="" type="checkbox"/>
	Es comprova la robustesa per a dades incorrectes del menú d'instal·lació.	L'aplicació queda fora de servei.	<input type="checkbox"/>
		No s'han fet millores en aquest aspecte. Es podria informar a l'usuari que cal reinstal·lar l'aplicació per dades corruptes al fitxer de configuració. Es contemplarà per futures versions.	<input checked="" type="checkbox"/>

Conjunt Prova 8: Vídeo Tutorial			
Descripció	<i>Comprovar el correcte accés al vídeo tutorial</i>		
Secció	Vídeo		
	Observacions	Resultats, Mesures Correctores	Validat
	Es comprova l'acció d'obrir vídeo des de pantalla principal.	Tot correcte	<input checked="" type="checkbox"/>

4.8.2 Proves d'acceptació de l'usuari

Les proves d'acceptació han consistit en donar a provar a diferents usuaris el programa en situacions reals.

Hem observat la forma d'ús del programa per part dels usuaris i hem considerat les observacions fetes per aquests amb l'objectiu d'implementar noves millores a l'aplicació.

El resultat de les observacions i les accions preses, són les següents:

Observacions sobre l'ús de l'aplicació per part dels usuaris:

- **Observació:** Pràcticament la majoria dels usuaris no utilitzen la tecla menú del dispositiu, per tant no poden accedir a l'ajuda.

- **Millora:** Afegim un ítem al menú principal per accedir directament a l'ajuda.

Figura 22: Afegim Ítem



- **Observació:** L'usuari no intueix que es pot modificar el menú individualment de cada unitat de pagament.
- **Millora:** Afegim un canvi de color a cada ítem del menú, perquè l'usuari provi l'acció "itemLongClick"

Figura 23: Canvi de color quan l'usuari toca l'ítem



- **Observació:** L'usuari quan vol afegir un producte al menú, o només pren el *checkbox* i no afegeix cap unitat o afegeix una unitat però no pren el *checkbox*.

Això provoca que els canvis no tinguin efecte sobre el preu del menú.

- **Millora:** *Automatitzem les dues accions, quan afegim una unitat activarem el "checkbox" o quan activem el "checkbox" afegim una unitat.*

Figura 24: Al modificar el checkbox afegim una unitat al menú.

Adult Menu	Products	Children Menu
+ 0 - <input type="checkbox"/>	Starter	+ 0 - <input type="checkbox"/>
+ 1 - <input checked="" type="checkbox"/>	Frist dish	+ 0 - <input type="checkbox"/>
+ 1 - <input checked="" type="checkbox"/>	Second dish	+ 0 - <input type="checkbox"/>
+ 0 - <input type="checkbox"/>	Sandwich	+ 0 - <input type="checkbox"/>
+ 1 - <input checked="" type="checkbox"/>	Soda or water	+ 0 - <input checked="" type="checkbox"/>
Accept		

- **Observació:** L'usuari vol poder crear les diferents distribucions de pagament sense introduir el import. Així, quan ve el compte ja està tot preparat i només cal introduir aquesta dada.
- **Millora:** *Permetrem la cancel·lació de l'entrada del import perquè a continuació es puguin definir les unitats de pagament.*

Figura 25: Creació d'unitats de pagament sense import inicial

Total Account:						
+	+	+				I and group 1
- 2	- 0	- 1	X			0.0\$
+	+	+				Group 2
- 1	- 1	- 2	X			0.0\$

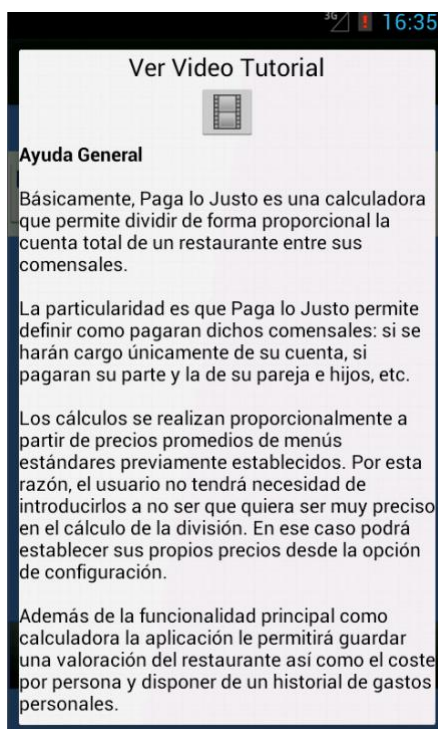
- **Observació:** L'usuari no entén amb facilitat com es realitza el càlcul. Un cop s'explica el primer cop ho troba molt senzill i pràctic, però en la majoria dels casos cal aquesta primera explicació.
 - **Millora:** S'ha creat una ajuda en vídeo d'ajuda per facilitar a l'usuari com ha de utilitzar la calculadora.

4.9 Ajuda

Com s'ha tractat en el punt 3 d'aquest mateix document, el disseny de l'aplicació s'ha realitzat utilitzant tècniques pel DCU i tenint present els *principis heurístics*.

Per tant, l'ajuda a l'usuari per un programa senzill hauria de ser mínima. Nosaltres hem plantejat aquesta ajuda explicant breument que podem fer en cada pantalla.

Figura 26: Pantalla *dialog* ajuda general



Per accedir a l'ajuda l'usuari ha de prémer la tecla menú del dispositiu.

Figura 27: Accedir a l'ajuda des de la tecla menú



Durant el transcurs de les proves d'acceptació amb els usuaris ens hem adonat d'alguns problemes que fan d'aquesta ajuda poc útil per l'usuari:

- Pocs usuaris utilitzen la tecla de menú del dispositiu, així, mai se'n adonen que existeix aquesta ajuda.
- Quan la troben no l'acaben de llegir o fer-li gaire cas.
- L'ajuda complementa el coneixement de les diferents parts del programa, la majoria de les quals ja les entén l'usuari sense cap ajuda. El problema és la comprensió de la utilització de la calculadora. La majoria necessiten d'una explicació bàsica, tipus demostració, per entendre el funcionament i per començar a utilitzar-la amb comoditat.

És per totes aquestes observacions que s'ha afegit un vídeo que mostra el funcionament de la calculadora, ja que és l'objectiu principal de l'aplicació, deixant, d'altre banda, com a secundària la resta de l'ajuda més tradicional.

Figura 28: Accés directe al vídeo d'ajuda



4.10 Publicació i model de negoci

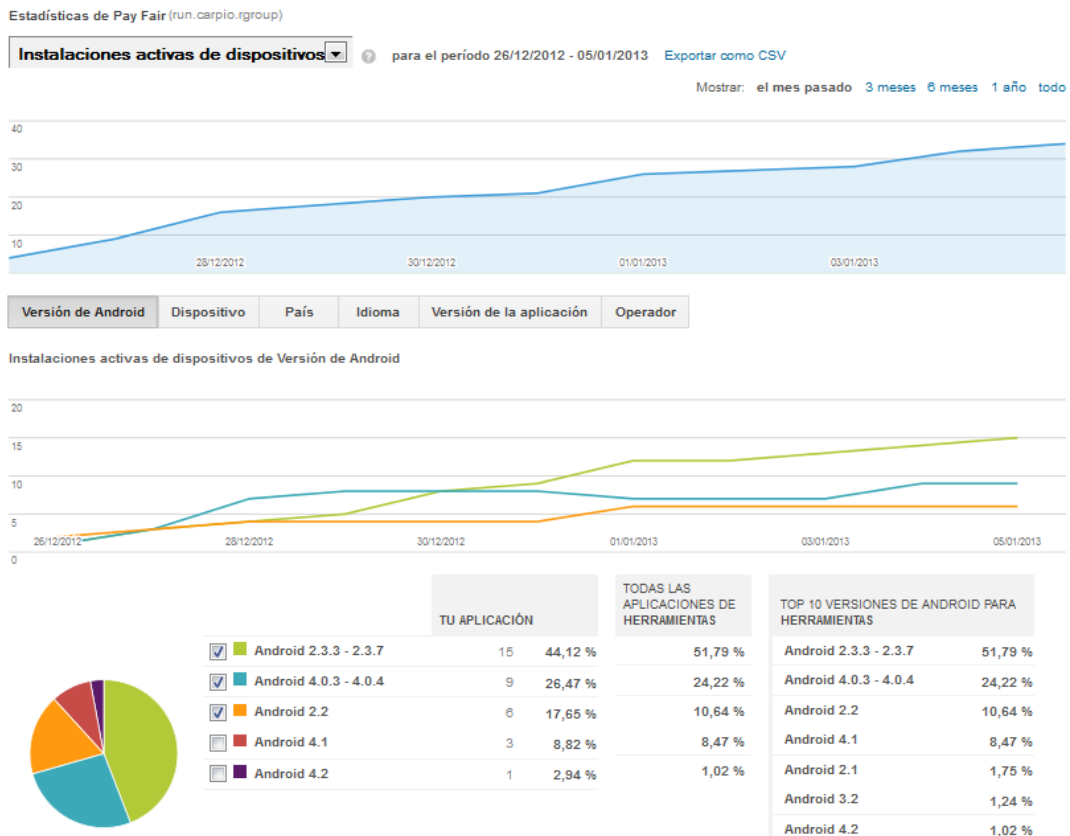
Hem triat la botiga de programari Google Play per publicar la nostra aplicació per ser el distribuïdor per referència d'aplicacions per a sistemes basats en Android.

Figura 29: Publicació de "Paga lo Justo" en Google Play

The screenshot shows the Google Play Developer Console interface. At the top, it displays the user's name 'Francesc Carpio' and email 'francesc.carpio@gmail.com', along with navigation links for 'Inicio', 'Ayuda', 'Android.com', and 'Cerrar sesión'. A 'Probar el nuevo diseño' button is also visible. The main header includes the Google Play logo and 'ANDROID DEVELOPER CONSOLE'. Below this, the user's profile 'Francesc Carpio' is shown with their email 'sitnova@gmail.com' and links to 'Editar perfil' and 'Administrar cuentas de usuario'. There are two yellow boxes: 'Cómo subir aplicaciones' with a 'Dismiss' button and 'Ubicaciones admitidas para distribuir aplicaciones' also with a 'Dismiss' button. A section titled 'Todas las listas de aplicaciones de Android de Google Play' lists the application 'Paga lo Justo 1.4' (Aplicaciones: Herramientas) with a 5-star rating, 51 total users, and 34 active installations. It also shows 'Gratis', 'Errores', and a 'Publicada' status with an 'Anunciar esta aplicación' button. A 'Subir aplicación' button is at the bottom right.

Encara que existeix un cost per donar-se d'alta com a desenvolupador amb dret per pujar programes, les avantatges que proporciona aquesta plataforma són molt superiors a d'altres: facilitat d'ús, gestió de les meves aplicacions i sobretot, la facilitat que té l'usuari final d'obtenir l'aplicació.

Figura 30: Estadístiques proporcionades per Google Play de l'aplicació



S'han considerat altres com *GetJar* o *Apps de Amazon*, però s'han desestimat com a primera opció perquè els usuaris, principalment de dispositius mòbils, no tenen preinstal·lat el gestor per accedir als programes que s'hi troben en aquestes botigues.

Tot i això, si es volgués continuar explotant l'aplicació, seria convenient publicar-la en més botigues com les citades, amb l'objectiu d'aprofitar el creixement en la venda de dispositius com *Kindle Fire*, que sí proporcionen la gestió d'obtenció d'aplicacions de la seva botiga de forma directa.

Model de negoci

Com a model de negoci s'ha triat la incorporació de publicitat en l'aplicació.

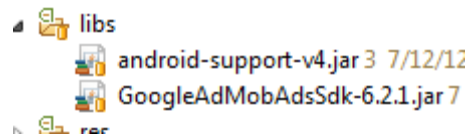
Primerament, s'ha comprovat que per a totes les pantalles existeix espai suficient per a incorporar aquesta publicitat sense perjudici greu a l'ús que pugui fer l'usuari.

Figura 31: Bloc de publicitat a l'peu de la pantalla



Per a la introducció de la publicitat, s'ha utilitzat les eines de programació proporcionades per AdMob de Google.

Figura 32: Llibreria AdMob de Google afegida a l'aplicació



La incorporació per a cada pantalla ha estat senzill. Gràcies a que hem utilitzat posicionaments i grandàries relatives s'ha pogut afegir aquest *layout* sense interferir en el nostre programa.

Figura 33: Mètode per a la incorporació del Banner Publicitari

```

public void creaAdMob() {
    // Crear la adView
    adView = new AdView(this, AdSize.BANNER, this.getResources().getString(
        R.string.admob_id));

    LinearLayout layout = (LinearLayout) findViewById(R.id.calcActivity);

    // Añadirle la adView
    layout.addView(adView);

    // Iniciar una solicitud genérica para cargarla con un anuncio
    adView.loadAd(new AdRequest());
}

```

S'ha triat AdMob de Google per a la gestió de la publicitat, pràcticament pels mateixos motius de la selecció de Google Play per a la publicació del programa: facilitat d'implementació, gran número de clients i per la seva gestió senzilla i clara.

Figura 34: Gestió dels ingressos d'AdMob de "Paga lo Justo"



En cap moment s'ha considerat definir un preu per l'aplicació ja que frenaria la seva distribució, com hem pogut comprovar en les proves d'acceptació: quan hem demanat als usuaris si pagarien per obtenir l'aplicació, la resposta ha estat sempre negativa.

5 Conclusions

El disseny centrat en l'usuari (DCU)

Vull destacar el treball realitzat en les primeres fases del projecte sobre el *disseny centrat en l'usuari*: el treball d'investigació contextual, enquestes, anàlisi competitiva, principis heurístics, prototipus, etc. Ha estat una tasca molt enriquidora que sempre ha estat present durant tota la implementació del projecte.

Aquesta feina ens ha permès tenir una idea molt clara del que volíem aconseguir des de un principi, centrant els nostres esforços en la visió del producte final i en com l'usuari respondria a aquest producte.

Així puc dir que l'èxit del projecte, que per a mi ha estat finalitzar-ho en els terminis establerts i acomplint amb les expectatives plantejades en la planificació, ha estat, sense cap dubte, gràcies a la feina realitzada en aquesta fase del projecte.

L'entorn de desenvolupament Eclipse

L'Eclipse ha demostrat ser un gran entorn de programació professional, més tenint en compte que és un projecte de codi obert i amb llicència lliure, actualment sota Eclipse Public License (EPL).

L'entorn Eclipse s'ha integrat perfectament amb el SDK i l'ADT d'Android, i ens ha permès fer, entre d'altres, les següents tasques:

- Implementar una programació molt estructurada, diferenciant molt clarament la capa lògica amb la interfície d'usuari.
- Gestionar els recursos de l'aplicació.
- Fer una depuració exhaustiva de l'aplicació.
- Fer comprovacions amb dispositius virtuals.
- Mantenir un repositori de versions, creació de la documentació, etc.

L'únic aspecte negatiu ha estat l'alt consum de recursos de l'entorn, sobre tot quan hem utilitzat el dispositiu virtual per fer les proves i la depuració. Aquest problema l'hem pogut reduir utilitzant dispositius físics reals per a fer les tasques de depuració o les proves.

La gestió del temps

Com hem comentat anteriorment, l'èxit del projecte ha estat principalment per una bona feina en la fase d'anàlisi que en fet basant-nos en el *Disseny Centrat en l'Usuari*. Altre aspecte que vull remarcar i que també ha estat molt important per a la finalització del projecte és la gestió del temps.

Durant tot el projecte les hores diàries dedicades han estat molt constants i els lliuraments de les PACS ens han servit per ajustar la nostra planificació.

Una qüestió que ens ha provocat cert neguit ha estat l'endarreriment a l'hora de començar a *picar codi*, per dos qüestions principals:

- Primer per la feina feta a l'anàlisi que ha quedat completament justificada.
- La segona ha estat pel desconeixement previ de la tecnologia utilitzada per a la programació de l'aplicació, ja que mai havia programat per a Android.

Aquest fet ens ha provocat dedicar molt temps a l'aprenentatge d'aquesta tecnologia ja que no vaig voler començar l'aplicació fins no tenir clar com funcionava. En aquesta fase d'aprenentatge vaig seguir molts tutorials i analitzar diferents exemples fins a tenir clar el que volia aplicar a la meu programa.

Aquesta decisió la puc valorar positivament perquè després d'adquirir els coneixements necessaris, el desenvolupament del programa va ser molt més ràpid de l'esperat i amb una qualitat bastant acceptable, és a dir, vaig estalviar esborranys i proves.

Finalment, sobre la gestió del temps, m'agradaria comentar com les hores dedicades a la interfície d'usuari han estat més que les de programació de backend. Podríem dir que les hores de backend són més productives, almenys per a les meves habilitats personals.

Línies de treball futur

Com a línies de treball futur o millores en l'aplicació vull destacar tres: un en l'àmbit tècnic, un altre en l'àmbit qualitatiu i finalment un sobre el model de negoci.

- En l'àmbit tècnic m'hagués agradat utilitzar la biblioteca de *JUnit* per a la realització de proves unitàries. No ho vàrem fer per a la previsió temporal que tenim massa ajustada, però seria una molt bona decisió fer-ho per assegurar sempre la màxima qualitat del programa.
- En l'àmbit qualitatiu voldria millorar la interfície gràfica, ja que les meves limitacions en aquest aspecte en marquen un sostre ben definit i necessitaria la col·laboració d'un expert per fer-la més atractiva visualment i per a la millora de la pantalla calculadora: fer-la més intuïtiva, millorar el seu aspecte o apropar-la més ha una visió real del restaurant i no l'abstracció feta.
- Sobre el model de negoci es podria millorar aprofitant les dades que podríem recavar amb l'usuari per proporcionar una publicitat més directe, com pot ser per la de generació d'ofertes de menús per a grup de restaurants habituals. També veuríem necessari la publicació del nostre programa per arribar a més usuaris ja que al mercat hi ha moltes aplicacions que volen resoldre el mateix problema i la nostra s'hauria de posicionar sobre elles pels seu tret característic ja que actualment és la única que fa el càlcul sense haver d'introduir els preus per producte o comensal, només amb el preu total.

6 Bibliografia

Meier, Reto (2009). "Professional Android Application Development". Ed. Wrox Wiley.

Gargenta, Marco (2011). "Learning Android". O'Reilly.

Garreta Domingo, Muriel; Mor Pera, Enric (2011). "Disseny centrat en l'usuari". Barcelona: FUOC.

Nielsen, Jakob. "Jakob's column on Web usability". En línia: <http://www.useit.com>
[data de consulta: 11/2012]

Vogel, Lars. "Android Development Tutorial". En línia:
<http://www.vogella.com/articles/Android/article.html>
[data de consulta: 11/2012]