

The background features a decorative graphic consisting of three concentric blue circles of varying sizes, each with a lighter blue outer ring. These circles are positioned in the top right, middle right, and bottom right corners. Thin blue lines extend from the top left towards the circles, creating a sense of movement or connection.

Treball Final de Carrera - Final

Desenvolupament aplicacions dispositius mòbils (iOS)

Iban Martín Herrera
08/01/2014

Consultors: Ignasi Lorente Puchades
Jordi Almirall López

Índex

1. Introducció	5
1.1. Tema escollit	5
1.2. Motivacions i objectius	5
1.3. Funcionalitats	6
2. Planificació del projecte	6
2.1. Descripció de les activitats	7
2.2. Recursos	7
2.3. Viabilitat	7
2.4. Incidències i riscos	8
2.5. Rols	8
3. Usuaris i context d'ús	9
3.1. Plantejament i desenvolupament	9
3.2. Resultats i conclusions	9
3.3. Perfils	10
4. Disseny conceptual	10
4.1. Escenaris d'ús	11
4.2. Fluxos d'interacció	12
5. Prototipatge	12
5.1. Sketches	12
5.2. Prototipus horitzontal d'alta fidelitat	13
6. Avaluació	17
6.1. Preguntes als usuaris	17
6.2. Tasques a realitzar per l'usuari	18
6.3. Preguntes sobre les tasques realitzades	18
6.4. Resultats de l'avaluació	18
7. Arquitectura de l'aplicació i cassos d'ús	19
8. Implementació de l'aplicació	20
8.1. Mètodes de tercers utilitzats	21
8.2. Problemes trobats i solucions	24
8.3. Comparació amb el prototipatge	32
9. Conclusions	42
9.1. Millores futures	42

Index de il·lustracions

Imatge01 - Flux d'interacció	12
Imatge02 - Sketch	13
Imatge03 - Prototip Login	14
Imatge04 - Prototip Visualització Mes	14
Imatge05 - Prototip Visualització Setmana	15
Imatge06 - Prototip Visualització Dia	15
Imatge07 - Prototip Visualització Llista	16
Imatge08 - Prototip Nova Tasca	16
Imatge09 - Prototip Modificació Tasca	17
Imatge10 - Prototip Eliminar Tasca	17
Imatge11 - Arquitectura aplicació	19
Imatge12 - Web documentació OpenAPI	21
Imatge13 - Calendari Tapku	22
Imatge14 - Calendari Tapku Modificat	22
Imatge15 - Classes Tapku	23
Imatge16 - Llicència Tapku	23
Imatge17 - Web Documentació Mètode GET	24
Imatge18 - Web Documentació descarrega Exemple1	25
Imatge19 - Classes Open API	26
Imatge20 - Login ideal	26
Imatge21 - Login modificat	27
Imatge22 - Login modificat web	28
Imatge23 - Login modificat drets	28
Imatge24 - Login modificat final	29
Imatge25 - Classe .h	30
Imatge26 - Codi Get	31
Imatge27 - Codi Push, Put i Delete	31
Imatge28 - Codi Connections	31
Imatge29 - Codi Resposta	32
Imatge30 - Comparació estructura	32
Imatge31 - Comparació Login	33
Imatge32 - Comparació Visualització Mes	34
Imatge33 - Comparació Visualització Setmana	35
Imatge34 - Comparació Visualització Dia	36
Imatge35 - Comparació Visualització Llista	37
Imatge36 - Visualització Nova Tasca	38
Imatge37 - Visualització d'una Tasca	39
Imatge38 - Eliminació d'una Tasca	40
Imatge39 - Canvis en tasques	41
Imatge40 - Millora Calendari	42

Agraïments

Vull agrair a molta gent que sens dubte ha influït en mi positivament i que gràcies a això he estat capaç de arribar fins aquí.

Per començar volia agrair a tots els consultors al llarg de la carrera que sempre han ajudat al alumne, en especial i concretament en aquesta assignatura a Xavi Aracil per tota la ajuda sobre l'Open API, que sense ell hauria estat totalment impossible que la meva aplicació s'hagués comunicat amb aquest i per tant com resultat, no tindria els objectius complets i també a Ignasi Lorente per resoldre alguns dubtes que han anat sorgint al llarg d'aquest semestre.

Volia agrair també a tota la meva família, ja que sempre han estat al costat i sempre m'han animat a complir els reptes que m'he plantejat i en especial als meus pares, que són els que fa molts anys em van ajudar a començar aquesta carrera i que sense ells no seria el que sóc avui.

I per finalitzar vull agrair molt especialment a la meva dona, ja que ha estat molts dies al costat meu donant-me tot el suport que podia, per a que superés una a una totes les assignatures. Gràcies a ella, m'ha fet encara millor del que era i gràcies a ella, estic escrivint ara aquestes paraules, que no hagués estat possible si no estigués acabant la carrera.

Gràcies!

1. Introducció

El Treball Final de Carrera té com objectiu la de realitzar un projecte d'una grandària que ocupa tot un semestre i que avalua els coneixements, aptitud i agilitat adquirits al llarg de la carrera. Encara que aquest consta de realitzar una aplicació per a iPhone i que no hi ha cap assignatura centralitzada en aquest camp de la informàtica, sí que hi han moltes assignatures que ens serveix per encarar el projecte.

El present document és el resultat de la feina realitzada al llarg d'aquest semestre per la realització del treball final de carrera i que a la vegada recull molta informació sobre aquest mateix.

He escollit aquest tema de aplicacions per iPhone, perquè a principis de l'any 2013, vaig realitzar un curs de Programació d'aplicacions i jocs per iPhone i iPad, amb la finalitat de trobar feina en aquest camp, però encara sense èxit. D'altra banda, vaig descobrir que és un camp que m'apassiona encara més del que m'imaginava. La realització d'aquest projecte, em dona la oportunitat d'ampliar aquests coneixements.

1.1. Tema escollit

El tema seleccionat és el de Calendari, ja que de entre els que podíem escollir, és el que em sembla més entretingut i que em permetrà aprendre molt sobre les funcionalitats que pot aportar el iPhone en aquest apartat i que és molt diferent al que vaig fer al curs mencionat en el punt anterior.

1.2. Motivacions i objectius

La meua motivació principal és el disseny. Fins ara, la meua feina sempre ha estat la de Tècnic de Sistemes. En la última feina era Gestor de BBDD i la veritat és que estava molt bé. Entre les meves tasques, de vegades tenia que fer formularis per introduir dades de nous clients, que em va permetre descobrir que m'agradava moltíssim fer nous dissenys. Al acabar la feina gràcies a la crisi, vaig tenir el temps lliure que em permetria realitzar un curs de Disseny Web, en el que vaig aprendre molt i finalment tenia clar que és al que em vull dedicar a partir d'ara.

D'altra banda també m'agrada molt tot sobre les noves tecnologies. Per circumstàncies de cercar nova feina, vaig tenir la sort de poder fer un curs de Programació d'aplicacions i jocs per iPhone i iPad, i que això va confirmar el que pensava ja fa temps. Vull realitzar nous dissenys i si els puc aplicar per els mòbils d'última generació, encara millor.

Els meus objectius són tant clars com les meves motivacions. Vull aprendre més i més. Si puc aprendre més programació i disseny i poder aplicar-lo en les noves tecnologies. Per tant, vull aprofitar la realització d'aquest projecte per realitzar aquest objectius:

- Aprendre més sobre la programació per a mòbils, concretament per iPhone.
- La realització d'un calendari. Fins al dia d'avui, no he fet res semblant, així que per a mi és un gran repte i que penso realitzar fins l'últim detall.
- Adquirir els coneixements necessaris, com per exemple els temes de connexions a diferents entorns, més concretament per aquest projecte la connectivitat amb l'API de la UOC.
- L'utilització de la llicència de desenvolupador de Apple, ja que fins ara no he tingut la oportunitat de fer-ho.
- Per últim, però no menys important, la realització d'un projecte tant important, complex i de grans dimensions com aquest.

1.3. Funcionalitats

En aquest apartat nombro quines seran les funcionalitats de l'aplicació calendari. He de comentar que aquestes són susceptibles a canvis al llarg del semestre, així com afegir algunes de noves. Les funcionalitats són les següents:

- Visualització del calendari per mes, setmana o dia.
- Acció de moure's entre els mesos, setmanes o dies, segons la visualització seleccionada.
- Visualització detallada d'una tasca. Tipus, descripció i data.
- Acció de afegir, eliminar o modificar una tasca.
- Visualització d'un llistat de tasques a partir del dia actual.

2. Planificació del projecte

La planificació de les diferents tasques durant la realització del projecte, són les detallades en la següent taula:

Tasca	Data començament	Dies de feina
Proposta activitat: calendari	16/09/2013	7
PAC1	23/09/2013	8
Lectura materials	23/09/2013	2
Anàlisi Pla de Treball	25/09/2013	1
Elaboració Pla de Treball	26/09/2013	5
Entrega PAC1	01/10/2013	0
PAC2	02/10/2013	28
Anàlisi	02/10/2013	5
Disseny	07/10/2013	15
Prototip	22/10/2013	8
Entrega PAC2	29/10/2013	0
PAC3	30/10/2013	34
Implementació i tests	30/10/2013	30
Realització vídeo Demo	29/11/2013	4
Entrega PAC3	03/12/2013	0
Lliurament Final	04/12/2013	35
Realització memòria	04/12/2013	25
Realització vídeo presentació projecte	29/12/2013	10
Entrega Lliurament Final	08/01/2013	0
Debat Virtual	22/01/2014	3

Les tasques marcades en negreta són inamovibles, ja que són dates límit. La resta de dates no són tan estrictes i per tant es poden modificar segons el progrés de la realització del projecte.

2.1. Descripció de les activitats

A continuació tenim una breu descripció de les activitats que es realitzaran en la preparació del projecte:

- Proposta: En aquesta tasca, el consultor proposa unes quantes activitats, en el que el alumne, escull una per a realitzar-la com el seu projecte. El consultor dona el OK si està d'acord amb la proposta del alumne. En el meu cas, he escollit la activitat del calendari.
- Fase 1: Aquesta tasca és una petita part i començament del projecte, en el que es realitza la primera part d'aquest document, on es mostra principalment les motivacions, funcionalitats i planificació del que serà la realització del projecte durant tot el semestre.
- Fase 2: En aquesta part del projecte, es realitza l'anàlisi, el disseny i la realització d'un petit prototip del que serà la interfície d'usuari de la activitat escollida.
- Fase 3: Aquesta és la tasca més important del projecte. La implementació d'aquest, comporta la programació del codi i les proves pertinents per comprovar la solidesa i fluïdesa de la activitat. A més per completar l'entrega d'aquesta part, s'ha de realitzar un petit vídeo demostratiu en el que es vegi les funcionalitats del programa, tal com l'utilitzaria l'usuari final.
- Fase final: Tasca en la que es fa la memòria, aquest document, de la realització del projecte i conjuntament en la entrega, es realitza també un vídeo més detallat de cada una de les funcionalitats de la activitat, amb l'ajuda d'una presentació en PowerPoint.

2.2. Recursos

Els recursos que s'utilitzaran per la realització d'aquest projecte, són els següents:

- Hardware: Utilitzaré una màquina virtual, Mac OS X 10.6.6 amb 2Gb de Ram, processador de 2 nuclis a 3GHz i 30 Gb de HDD. Aquesta serà executada per la màquina local, amb Windows 7 64bits, 4 Gb de Ram, processador AMD Phenom 6 nuclis a 3GHz i 1 Tb de HDD.
- Software: En aquesta màquina virtual, tinc instal·lat el Xcode 3.2.6.
- Referent al mòbil, faré servir un iPhone 3G. Encara que per realitzar proves més instantànies, l'Xcode porta un emulador de iPhone/iPad, que encara que té les seves limitacions, serveix per comprovar la correcta funcionalitat d'aquest projecte.

2.3. Viabilitat

La avaluació d'aquest projecte, es reparteix en tres apartats:

- Viabilitat tècnica: aquest projecte és tècnicament viable, ja que totes les funcionalitats previstes per la realització completa d'aquest, han estat analitzades i comprovat que són possibles de realitzar.
- Viabilitat operativa: des de la perspectiva operativa, aquest projecte també és viable, ja que al tenir disponible el calendari personal de les tasques a realitzar, de les assignatures matriculades i altres relacionades amb la teva Universitat, a la mà, és totalment operativa i després de fer el estudi previ de la seva acollida, es confirma que així ho serà.
- Viabilitat econòmica: aquesta viabilitat no s'estudia per aquest projecte, ja que és un treball per la Universitat. De no ser així, s'hauria d'estudiar les hores que es treballarien per la realització total del projecte i segons el sou de la empresa contractada i les hores previstes, tindríem el cost total d'aquest i per tant, valorar la seva viabilitat econòmica.

2.4. Incidències i riscos

En aquest apartat s'estudia la possibilitat que sorgeixin incidències al llarg de la realització del projecte i a partir d'aquest estudi, es valoren els riscos que s'assumeixen. Les incidències que poden sorgir poden ser les següents:

- Tenir una averia tant al Hardware com al Software a la màquina local. Al tenir una màquina virtual de MAC, aquesta depèn totalment de la màquina local. Pot donar-se el cas que per qualsevol problema que tingui la màquina local, aquesta no pugui executar la màquina virtual. (El mateix pot passar si falla el software de la màquina virtual).
- Necessitat de canviar de versió de OS X i/o de Xcode, per un tema de compatibilitats amb la programació necessària de l'aplicació. Com per exemple la connectivitat amb l'API UOC: al no tenir cap coneixement en aquest camp, no puc valorar el risc que comporta.
- Disponibilitat personal, pot donar-se el cas, encara que molt poc probable, que durant la realització del projecte, no tingui la disponibilitat prevista i necessària, provocada tant per una baixa física, malaltia i/o problemes personals.

D'altra banda, per minimitzar qualsevol risc amb qualsevol pèrdua de dades relacionades amb el projecte (documents, links d'informació, codi, etc...) faré còpia de seguretat en un PenDrive exclusiu, durant tot el semestre.

2.5. Rols

En aquest apartat es veuen tres classes de persones que adopten diferents perfils, detallats a continuació:

- Consultors: Aquests adoptaran dos tipus de perfils diferents:
 - El primer és el de consultor de l'assignatura, ja que amb el seu ajut ens ofereix als alumnes la informació necessària per la realització de les tasques i la resolució dels dubtes que tinguem.
 - L'altre perfil és el de client final, ja que aquest valorarà el projecte en sí i la seva satisfacció de la feina realitzada. (Fruit d'aquesta part, l'alumne obté la nota de la assignatura).
- L'alumne: Aquest adopta diferents perfils per la realització total del projecte:
 - Director del projecte: és el cap del departament que realitza el projecte, aquest gestiona les funcions de cada empleat del seu departament.
 - Analista: és el encarregat de fer l'anàlisi previ del projecte.
 - Dissenyador: és el encarregat de fer el disseny de les interfícies de l'aplicació.
 - Programador: és el encarregat de programar les funcionalitats que tindrà l'aplicació.
 - Tester: és el encarregat de trobar falles al posar a prova cada una de les funcions de que disposa l'aplicació. Informar d'aquestes falles al programador, per a que les resolgui satisfactòriament. Una vegada resoltes, tornar a provar l'aplicació completament per trobar noves falles i així fins no trobar cap.
 - Documentació: és el encarregat de fer la memòria i vídeos demostratius del projecte.
- Familiars i amics: aprofitant la confiança que tinc amb cada un dels membres, els hi demano que facin de tester, per així tenir més possibilitats de trobar errors a l'aplicació.

3. Usuaris i context d'ús

Segons els mètodes d'indagació coneguts, he seleccionat dos: Anàlisi competitiva i entrevistes en profunditat. D'aquesta manera s'analitza una aplicació o varies semblants a l'aplicació que tenim com objectiu fer i amb l'afegit de les entrevistes en profunditat, podrem afegir funcionalitats i/o fer-la més intuïtiva, creativa, etc...

3.1. Plantejament i Desenvolupament

Anàlisi competitiva

Tal com he mencionat abans, l'elecció de l'anàlisi competitiva és per analitzar una altra aplicació que tingui molta semblança amb la idea que tenim de la nostra, per així tenir-la com exemple i tenir un començament del que volem fer. Després d'analitzar-la es veu fins a quin punt és semblant a la nostra i podem aprofitar aquest anàlisi al nostre favor.

L'aplicació existent és el mateix calendari que porta el iPhone per defecte. Aquesta aplicació, al obrir-se representa el mes actual, en el que es pot navegar i veure el pròxim mes o l'anterior d'una forma molt senzilla, a més té funcionalitats per veure en detall un sol dia o un llistat de les tasques ordenades segons en el dia que estan programades. D'altra banda, hi ha la funció d'afegir una tasca en el dia que es té seleccionat en aquest moment i una altra funció per activar/desactivar diferents calendaris. En aquest mateix apartat, es poden editar diferents calendaris, de manera que quan es crea una tasca, aquesta pertany a un d'aquests calendaris.

És una aplicació molt senzilla, però de gran utilitat, ja que es pot fer servir com una agenda i el més important, com passa amb totes les aplicacions de mòbil, és que tens totes aquestes funcionalitats a la mà.

Entrevistes en profunditat

Aquest mètode d'indagació ens permet saber amb detall, que és el que necessita l'usuari final en l'aplicació. Això inclou tant les funcionalitats, com que sigui fàcil d'utilitzar i que sigui el més intuïtiva possible. Les entrevistes són diferents als qüestionaris, mentre que en aquest mètode només es fan preguntes amb unes respostes obligatòries, les entrevistes ens permet tenir una conversació i és més probable descobrir uns interessos del usuari final que ni tant sols ens havien plantejat.

L'entrevista sí ha de tenir una pauta a seguir, amb l'objectiu de aconseguir una conversació fluida i maximitzar la informació que es vol obtenir. La pauta a seguir és la següent: començant per saber que és el que han de tenir totes les aplicacions de mòbil tipus calendari, seguir amb les funcionalitats i l'aspecte que han de tenir (alguns aspectes són molestos, amb la conseqüència que l'usuari no utilitzi la nostra aplicació) i quan i on la farà servir.

3.2. Resultats i Conclusions

Després de seguir els mètodes d'indagació trobem que l'aplicació serà molt semblant a la del calendari del iPhone. Era d'esperar, ja que les aplicacions tipus calendari no són molt diferents les unes de les altres.

Després de fer les entrevistes sí que és veritat que la funcionalitat de tenir diferents calendaris no és factible, ja que aquesta aplicació només és per l'entorn de la UOC i no és el nostre objectiu, barrejar les activitats de la UOC amb altres en aquesta aplicació. D'altra banda, sí que es demana, poder diferenciar les tasques de cada assignatura i la possibilitat de afegir tasques que no siguin d'una assignatura en concret, però sí relacionada amb l'estudi d'alguna d'aquestes, com per exemple, cercar documentació sobre un llenguatge de programació.

3.3. Perfils

Després de fer l'anàlisi i veure qui utilitzarà l'aplicació, només hi ha un únic perfil d'usuari que anomenaré 'usuari standard'. Sí que és veritat que es pot diferenciar els alumnes dels consultors i/o tutors, que tenen perfils diferents a la UOC, però com es veu en el punt anterior, es poden afegir tasques que no siguin d'una assignatura i per tant no s'ha de fer cap funcionalitat exclusiva per aquests altres perfils. D'altra banda tenim els usuaris que tenen molta experiència en utilitzar aplicacions de mòbil i els que no en tenen tanta, però degut a la senzillesa de l'aplicació, no veig tampoc motiu per fer diferents perfils.

Usuari Standard

Tal com he comentat abans, l'usuari standard és tant el alumne com un consultor i/o un tutor i també és un usuari que no requereix coneixements avançats, per moure's per les diferents funcionalitats de l'aplicació. D'altra banda tenim que la demografia és majoritàriament de Catalunya, però per la experiència que porto tots aquests anys a la UOC, sóc conscient que hi ha usuaris arreu del món. Finalment, comentar que aquest usuari, té l'interès d'utilitzar aquesta aplicació, per tenir a la mà el seu propi calendari de la UOC i poder accedir a ella en qualsevol moment, motivació per la qual utilitzarà com una eina més a la UOC.

El context d'ús d'aquest usuari, indica que al ser una aplicació de smartphone es pot utilitzar en qualsevol moment i en qualsevol lloc. A part d'aquesta evidència, puc afegir i gràcies al mètode d'indagació de les entrevistes en profunditat, que l'aplicació serà consultada principalment al sortir de la feina (al estar en el transport públic de tornada a casa) i/o a casa, en el moment que l'usuari es posi en el seu moment d'estudi. D'altra banda, per afegir informació al calendari, s'utilitzarà bàsicament a casa en el moment d'estudi, amb la intenció de programar futures tasques.

Les tasques a realitzar per els usuaris, són les següents:

- Afegir una nova tasca: en la creació d'una nova tasca, haurà de posar el nom identificatiu, seleccionar un color també de identificació i assignar dia i hora de la mateixa. Optativament, pot posar una descripció més detallada.
- Modificar una tasca: de la mateixa manera que es crea una nova tasca, aquesta es pot modificar.
- Eliminar una tasca: eliminar la tasca seleccionada de forma permanent.
- Visualitzar les tasques: aquestes es poden visualitzar de diferents formes, com pot ser de forma única i amb tot detall, visualitzar les tasques de un dia en concret, en forma de llista a partir del dia actual i visualitzar-les per mesos o setmanes. En totes les pantalles de visualitzacions excepte la de detall de la tasca, només es veurà els noms de les tasques i concretament en la visualització per mes, es veurà el dia del color de la tasca (en el cas de haver més d'una tasca un mateix dia, el color serà vermell).

Finalment, les característiques identificades als mètodes d'indagació son: imatge clara en tota l'aplicació que identifiqui en tot moment que és el calendari de la UOC, les múltiples formes de visualitzar les tasques i la senzillesa en la edició de les tasques, dit d'altra manera, que sigui molt intuïtiva.

4. Disseny conceptual

A continuació es detallen uns escenaris d'ús, que serviran per veure la utilitat de les funcionalitats de l'aplicació en diferents situacions que es poden donar i els fluxos d'interacció per detallar l'estructura d'aquesta.

4.1. Escenaris d'ús

Escenari 1

Jack és un estudiant de la UOC que treballa des de fa més de 5 anys en una multinacional com a Tècnic de Sistemes, després de la seva jornada de feina, torna cap a casa seva. És dilluns i sap que aquesta setmana li toca entregar la PAC que va estar preparant la setmana anterior, però no recorda quin dia té com a límit per entregar-la. Així que entra en la estació de metro, aprofita per consultar aquest dubte mentre espera a que arribi. Agafa el mòbil, entra en l'aplicació i de primeres veu que el dijous de la setmana actual el té marcat en blau, pitja en el dia i confirma les seves sospites, la data límit per fer la entrega de la PAC és el dijous i torna a casa tranquil·lament, ja que encara no ha de entregar-la.

- Perfil: usuari standard
- Context: sortida de la feina
- Objectius: confirmar data límit d'entrega de PAC
- Tasques: visualització tasca
- Necessitats d'informació: data de la tasca
- Funcionalitats: detall tasca
- Desenvolupament: consultar l'aplicació per veure la data límit de la PAC

Escenari 2

Sam és una jove estudiant de la UOC i no té feina. Aprofitant que té més temps que la majoria dels estudiants i pot dedicar tot el seu temps per estudiar, està matriculada de 6 assignatures amb la conseqüència que té moltes PACs i molta feina d'estudi. Cada vegada que comença un nou dia i abans de posar-se a estudiar qualsevol de les assignatures, consulta l'aplicació per veure el llistat de tasques que té pendents des del dia actual i poder gestionar el seu temps de la millor manera possible.

- Perfil: usuari standard
- Context: cada matí
- Objectius: veure feina pendent
- Tasques: visualització del llistat de tasques
- Necessitats d'informació: identificar tasques pendents
- Funcionalitats: llistat de tasques
- Desenvolupament: consultar l'aplicació per veure el llistat de tasques

Escenari 3

Alex és un estudiant de la UOC i té feina des de fa un parell d'anys en una empresa petita de Gestor de BBDD. Encara no ha començat el semestre, però sap que en l'assignatura de programació s'utilitza el llenguatge de JAVA i fa molt temps que no el toca. Per tant de no trobar-se en problemes d'última hora al fer la assignatura i realitzar la primera PAC, s'assigna la tasca de fer un repàs de JAVA aquest mateix cap de setmana. Agafa el mòbil, obre l'aplicació, selecciona el dissabte de la mateixa setmana i crea la nova tasca per no oblidar-se de fer repàs de JAVA.

- Perfil: usuari standard
- Context: preparant l'inici del semestre
- Objectius: crear una tasca com recordatori
- Tasques: creació d'una tasca
- Necessitats d'informació: recordatori de la tasca
- Funcionalitats: creació d'una tasca
- Desenvolupament: obrir l'aplicació i crear una tasca un dia en concret

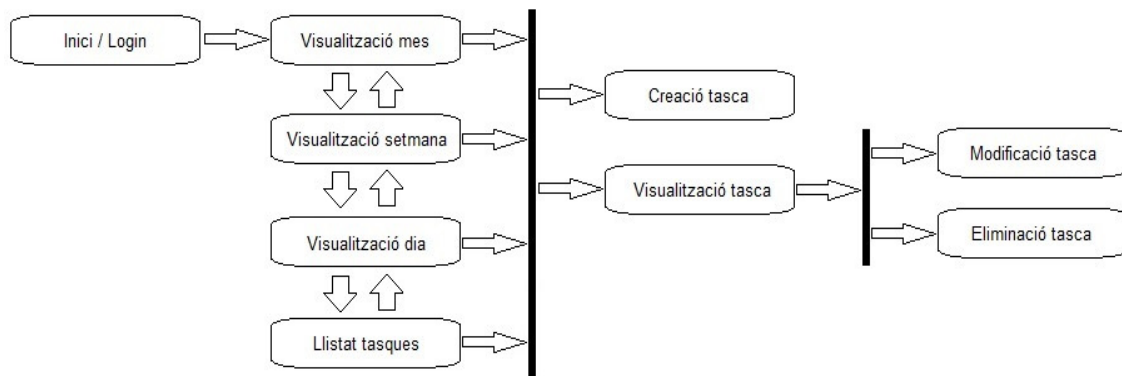
Escenari 4

Joan és un estudiant de la UOC amb 4 assignatures matriculades l'actual semestre i una empresa li ha contractat fa molt poc de programador web en PHP. En Joan es troba amb una situació molt delicada, ja que les assignatures de les que està matriculat tenen molt volum de feina i ara amb la nova feina es troba que no té tant temps com preveia per realitzar totes les assignatures. Després de pensar una bona estona, pren la decisió de deixar dues assignatures ja que té el risc de suspendre les 4 assignatures, per falta de preparació. Així que agafa el seu mòbil, obre l'aplicació i elimina les tasques relacionades amb les dues assignatures que ha decidit deixar, de forma que el volum de feina que tenia abans es redueix a la meitat.

- Perfil: usuari standard
- Context: nova feina, prendre decisió de deixar dues assignatures
- Objectius: eliminar les tasques de dues assignatures
- Tasques: eliminar tasques
- Necessitats d'informació: veure tasques per eliminar les de dues assignatures
- Funcionalitats: eliminar tasca, varies vegades
- Desenvolupament: obrir l'aplicació i eliminar varies tasques

4.2. Fluxos d'interacció

Degut a que només hi ha un perfil, ja que tothom (sigui un alumne o un consultor/tutor) pot fer les mateixes accions, també hi ha un únic flux d'interacció:



Imatge01 - Flux d'interacció

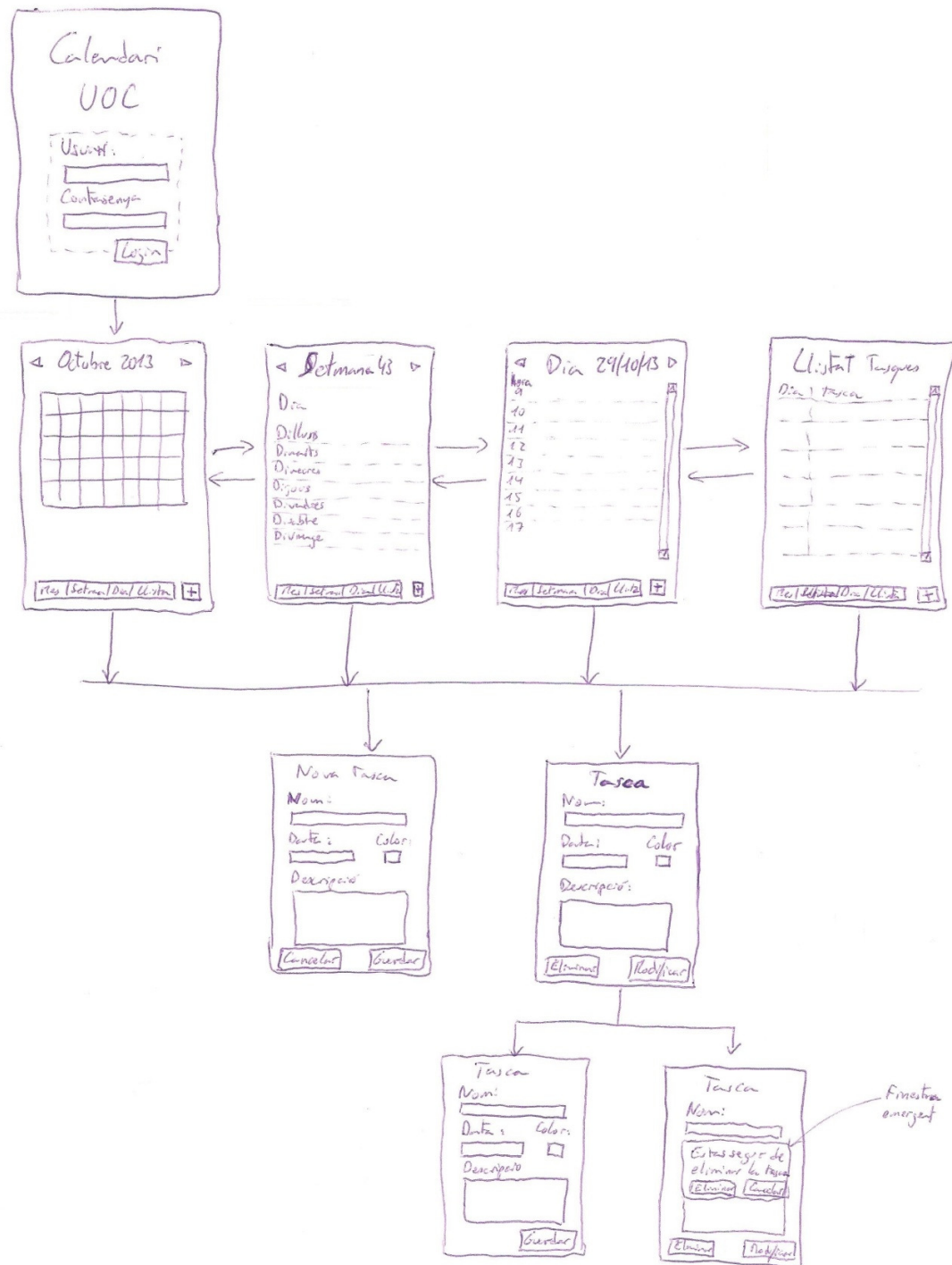
5. Prototipatge

Des del començament de l'elaboració d'aquest projecte, he tingut en compte les avaluacions heurístiques en tot moment i que continuaré de la mateixa manera fins la finalització d'aquest.

Com exemple, fruit de la investigació, els mètodes d'indagació i les entrevistes en profunditat per aquest projecte, he obtingut el resultat d'una aplicació que permet de forma molt senzilla a que el usuari arribi als seus objectius de forma gairebé immediata. En molt poques accions, l'usuari pot realitzar totes les tasques.

5.1. Sketches

A continuació s'exposa els sketches fets a mà i després escanejats per adjuntar-lo en aquest document. Prenent com ha model el flux d'interacció, el resultat és el següent:



Imatge02 - Sketch

5.2. Prototipus horitzontal d'alta fidelitat

Per fer el prototipus horitzontal d'alta fidelitat, he utilitzat el programa Pencil, el que ha permès que faci les pantalles que tindrà l'aplicació, de forma molt exacte respecte al que tinc pensat fer.

Comentar que els colors, l'estil de les línies, etc... potser diferenciarà una mica respecte al següent prototipus, ja que segons es crea i comprova el disseny al dispositiu, aquest es va millorant, ja sigui per un color massa fort, la seva brillantor o que alguna línia no es vegi suficient. Aquests canvis seran per resoldre un disseny modern i també agradable.



Imatge03 - Prototip Login

Inici / Login

Aquesta és la pantalla que veurem tal com s'inicia l'aplicació.

Comentar també, que els smartphones en el cas de no tancar les aplicacions, la sessió tampoc no es tanca i per tant aquesta pantalla només es veurà cada vegada que s'inicia l'aplicació des del seu inici.



Imatge04 - Prototip Visualització Mes

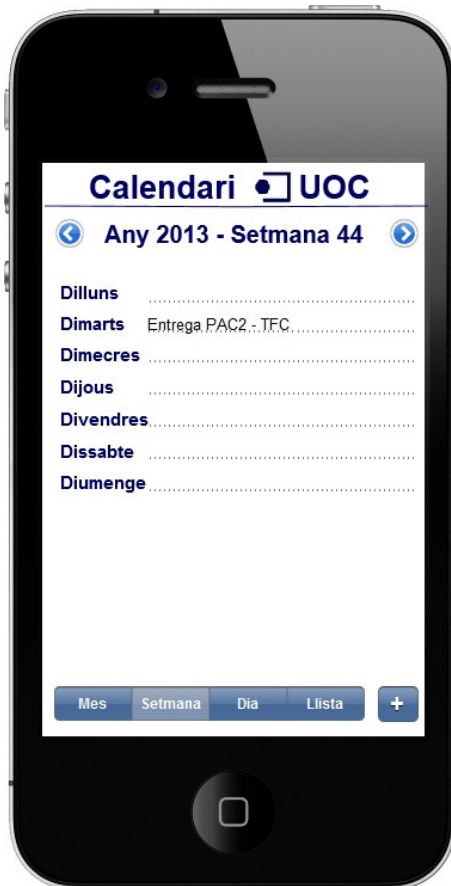
Visualització per mes

Aquesta és la pantalla que es veu al iniciar l'aplicació després del Login i en el cas de estar en una altra pantalla, al pitjar en el botó 'Mes' del menú de sota.

Com es veu a tots dos costats del nom del mes, hi han els botons per anar al anterior o al següent mes.

De primeres es veuen diferenciats 3 dies. La casella de color taronja significa que hi ha una tasca programada (que encara que ja ha passat, la tasca continua existint), el dia més enfosquit és el dia actual i el dia 29 és d'un color més clar, per diferenciar que s'ha pitjat en ell i que a sota es veuen els noms de les tasques programades per aquell dia, si hi ha moltes, es podrà desplaçar aquest llistat verticalment. Al pitjar en una tasca es veurà el detall d'aquesta en la pantalla de 'Visualització de tasca'. Finalment hi ha el botó '+' per afegir una nova tasca.

Aquest és un exemple de millora de disseny. Segurament al veure aquests colors en el dispositiu, seran un motiu per canviar el color base del calendari.



Imatge05 - Prototip Visualització Setmana

Visualització per setmana

Pitjant en el botó 'Setmana' del menú de sota, ens surt la pantalla de les visualitzacions per setmana.

Aquí podem veure que també disposem dels dos botons als costats del número de la setmana, per anar a la anterior o la següent.

Al igual que en la 'Visualització per mes', en aquesta pantalla veiem la setmana completa seleccionada i en el cas que hi hagi una tasca, es veurà en el dia que està programada. En el cas que es pitgi en una tasca, anirem directament a la pantalla de 'Visualització de tasca' per veure-la amb més detall i en el cas que es pitgi en un dia, anirem a la pantalla de 'Visualització per dia'. D'altra banda, també hi ha el botó '+' per afegir una nova tasca.



Imatge06 - Prototip Visualització Dia

Visualització per dia

Pitjant en el botó 'Dia' del menú de sota, ens surt aquesta pantalla, del dia que estigui seleccionat en aquell moment.

Com en les pantalles anteriors, també disposem dels dos botons als costats del dia/mes/any, per anar al anterior o següent dia.

En aquesta pantalla es veuen les hores del dia en format llista, si hi ha alguna tasca per aquest dia, es mostrarà en la hora programada. Com no hi ha lloc suficient a la pantalla, hi ha una barra vertical per desplaçar-se per totes les hores del dia (des de les 0h. fins a les 24h.). Al igual que en la pantalla de 'Visualització per Setmana' al pitjar en una tasca, anirem directament a la pantalla de 'Visualització de tasca' per veure-la amb més detall i de la mateixa manera que en les altres pantalles de visualitzacions, també hi ha el botó '+' per afegir una nova tasca.



Imatge07 - Prototip Visualització Llista

Visualització per llistat de tasques

Finalment, al pitjar en l'últim botó del menú de sota 'Llista', ens surt aquesta pantalla i es veuran totes les tasques programades a partir del dia actual, en forma de llista (Dia | Nom de la tasca) i en ordre ascendent.

En cas que la llista sigui molt llarga, tenim la opció de utilitzar la barra vertical per desplaçar la pantalla còmodament.

Com en les anteriors pantalles de visualització, al pitjar en una tasca, anirem directament a la pantalla de 'Visualització de tasca' per veure-la amb més detall i com sempre, també hi ha el botó '+' per afegir una nova tasca.



Imatge08 - Prototip Nova Tasca

Nova tasca

Aquesta pantalla és la que surt, al pitjar en el botó '+' en qualsevol de les pantalles anteriors de visualitzacions.

Com es veu de forma clara, podem incloure el nom de la tasca, la data i hora en que la volem programar, un color que serveixi de distintiu per a que es vegi en la pantalla de 'Visualització per mes' i una descripció per posar el que vulguem.

A sota, podem en tot moment cancel·lar i tornar a la pantalla en la que estàvem abans o guardar la tasca, en que l'aplicació confirmarà que els camps necessaris tenen informació i al igual que l'acció de cancel·lar, tornarem en la pantalla en la que estàvem abans de fer la nova tasca.

Comentari: Al pitjar en la data, sortirà la eina del iPhone que visualitza per més per seleccionar un dia i per la selecció de la hora sortirà un UIPickerView del iPhone.

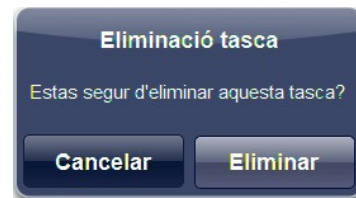


Imatge09 - Prototip Modificació Tasca

Visualització de Tasca

Aquesta pantalla és la de visualització de la tasca amb tots els camps d'aquesta. Veiem que és exactament igual a la de creació de una nova tasca, però evidentment sense poder fer cap canvi. Aquí és on anirem, cada vegada que pitgem en una tasca en qualsevol de les pantalles de visualització anteriors.

A diferència de la de creació de una nova tasca, tenim els dos botons de sota, en el que podrem modificar la tasca que estem visualitzant o eliminar-la. En aquest últim cas, enlloc d'anar a una altra pantalla, ens sortirà una pantalla emergent, en la que ens demanarà confirmació de la eliminació de la tasca:



Imatge10 - Prototip Eliminar Tasca

6. Avaluació

En aquesta ultima fase, els usuaris avaluaran el prototipus realitzat per aquest projecte i així corregir els problemes que sorgeixin i/o detectar millores que es puguin realitzar abans de començar a fer l'aplicació final.

6.1. Preguntes als usuaris

En aquest apartat veurem les preguntes que farem als usuaris que provaran el prototipus abans de veure'l i realitzar tasques sobre aquest. Les preguntes són les següents:

- Què ha de tenir tota aplicació tipus calendari de tasques?
- Quina informació no pot faltar en cada tasca que es crea?
- Hi ha alguna funcionalitat o informació que voldries afegir a una típica aplicació calendari?
- I al contrari, eliminaries alguna funcionalitat en aquest tipus d'aplicacions?

6.2. Tasques a realitzar per l'usuari

Ara, el que es demana al usuari és que faci unes quantes tasques amb el prototipus, per veure la reacció del usuari i millorar el prototipus en base a la experiència d'aquest apartat:

- Entrar en l'aplicació i 'navegar' per diferents mesos.
- Provar les diferents pantalles de visualitzacions (mes, setmana, dia i llista)
- Crear una nova tasca i veure-la després en la pantalla anterior a la creació de la tasca.
- Modificar una tasca i després eliminar-la definitivament.

6.3. Preguntes sobre les tasques realitzades

Finalment, després que el usuari hagi provat el prototipus, cal preguntar-li sobre la seva experiència al fer les tasques anteriors. Després es corregeix el prototipus, per solucionar els problemes detectats i fer les millores necessàries. Les preguntes són les següents:

- Has trobat alguna dificultat a l'hora de fer login?
- Et sembla correcte l'aspecte de les diferents pantalles de visualitzacions?
- Trobes interessant el menú de sota? és intuïtiu i suficientment intel·ligible?
- Quina dificultat trobes a l'aplicació en general?
- Utilitzaries aquesta aplicació per la teva organització de les tasques de la UOC? per què?

6.4. Resultats de l'avaluació

En aquest apartat vull comentar diferents aspectes que canviat o afegit en l'aplicació, després de fer la avaluació:

- He canviat l'aspecte i distribució de la pantalla de visualització per setmana, de forma que ara es veu en línies, ja que abans estava per caselles. La intenció era veure totes les tasques de un sol dia, en el cas que hi hagin varies. Com conseqüència, he afegit els dos següents canvis:

- En la pantalla de visualització per setmana, a la hora de representar varies tasques en un sol dia, ho faré amb el nom de 'Varies tasques' i no com estava pensat en un principi, que era posar-les una darrera de l'altra.
- Al pitjar en 'Varies tasques', anirem a la pantalla de 'Visualització per dia' de manera que es veuran totes les tasques d'aquest dia.

- El mateix passa amb els colors en la pantalla de visualització per mes. En cas que en un sol dia, hi hagi més d'una tasca amb diferent color, aquest dia es representarà amb el color vermell, ja que pot donar una impressió equivocada i que no quedi gens clar el significat d'un dia multicolor. D'altra banda la programació d'aquest resultat es pot complicar una mica en aquest aspecte. En resum, el color vermell estarà reservat per l'aplicació per representar aquests casos i per tant en la elecció del color en la programació de la tasca, no es podrà escollir el color vermell.

- He considerat afegir unes sigles en la programació de la tasca, per representar-les d'aquesta manera en cas de no trobar lloc suficient utilitzant la representació per els noms. És a dir, posem per cas que en un sol dia hi han tres tasques, per tant en la visualització per setmana, és més que probable que no es vegin els noms de les tres tasques, per tant, l'aplicació les representaria amb les sigles en lloc dels noms. Al veure-ho en un prototip i després de fer els altres canvis, he considerat descartar aquest canvi, ja que no ho veia del tot clar i que m'agrada més el sistema de posar el nom de 'Varies tasques' per aquests casos.

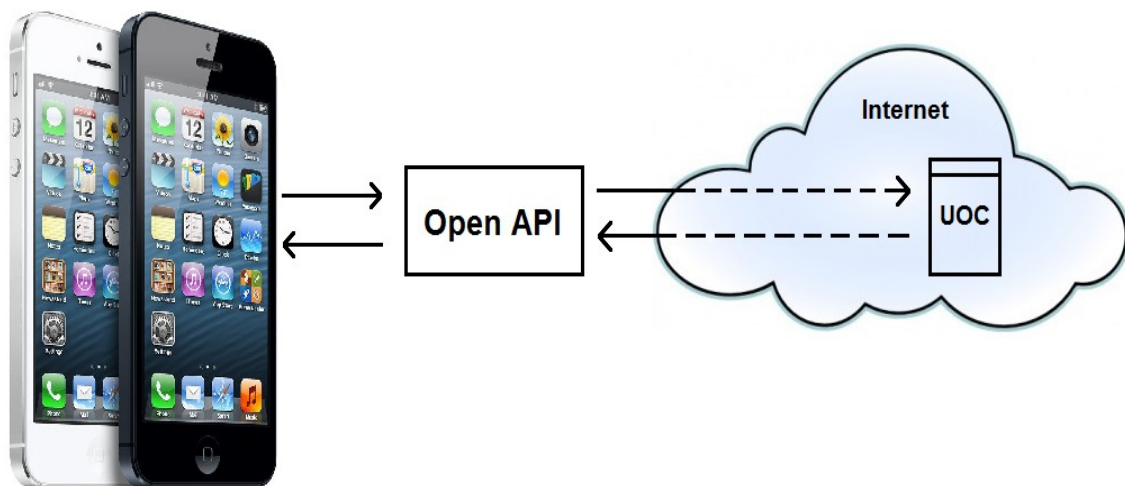
- Per últim, volia comentar que el botó '+' per la creació d'una nova tasca, m'he trobat en algun cas que de primeres no era tant clar i que l'usuari ha descobert per a què servia, una vegada l'utilitza. Per tant, estic considerant canviar-lo per una descripció tipus 'Nova tasca'.

Per acabar, comentar que aquests són tots els canvis i consideracions que he realitzat, després de les fases de prototipatge i avaluació.

7. Arquitectura y cassos d'ús

L'arquitectura de l'aplicació és molt senzilla. Només consta d'una part mòbil (iPhone) que és en la que s'ha desenvolupat l'aplicació i una altra part anomenada Open API, eina que ens facilita la UOC i que és la que ens dona l'accés a la UOC a través d'internet per accedir a les seves dades, en que l'aplicació consulta per després representar-les a la pantalla del dispositiu.

A continuació es representa gràficament aquesta arquitectura:



Imatge11 - Arquitectura aplicació

Per últim, comentar que l'aplicació s'ha desenvolupat en Xcode3 i s'han fet les proves en un iPhone 3G, lo que ens permet dir que l'aplicació es pot utilitzar en versions posteriors de iPhone, encara que no tingui les eines més modernes de avui dia.

Els següents cassos d'ús ens representen les possibilitats que ens ofereix l'aplicació, en diferents circumstàncies i després de provar-la per mi mateix:

Cas d'ús 1 - Fer Login: Obro l'aplicació i després de carregar, em trobo que fer Login és el primer que s'ha de fer. Si provo de posar un usuari inexistent o una contrasenya incorrecte, com és lògic m'avisa que les credencials són incorrectes i he de tornar-ho a intentar. Una vegada posat el usuari i la contrasenya correctes, em demana quin tipus d'autorització li dono a l'aplicació (lectura i escriptura) per accedir a les meves dades. Una vegada fet, em dona la benvinguda.

Cas d'ús 2 - Visualitzacions: Una vegada accepto la benvinguda, l'aplicació carrega una nova pantalla amb la visualització per mes. A sota hi ha una barra amb diferents opcions que utilitzo per veure les diferents propostes de visualitzacions de les tasques: visualització per mes, per setmana, per dia i per un llistat de les tasques pendents. En cada una d'aquestes visualitzacions puc moure'm entre els diferents mesos, setmanes i dies respectivament, en la visualització per el llistat de tasques no hi ha més opcions que el propi llistat. Un altre detall, és que al canviar entre les diferents vistes, aquestes no es reinicien, és a dir, al tornar a veure altres vistes, aquestes estan on l'havia deixat.

Cas d'ús 3 - Consultar una tasca: En cada pantalla de visualització, hi ha un llistat de tasques en la que es mostren en un llistat de tasques del dia seleccionat en cas de la visualització per mes, un llistat de cada dia de la setmana en el cas de la visualització per setmana, un llistat de cada hora del dia en el cas de la visualització per dia i un llistat de les tasques pendents en el cas de la visualització per llista. En totes les llistes, al pitjar en una tasca, canvia de pantalla per veure el detall d'aquesta, en la que no em deixa modificar cap camp i a sota em dona l'opció de modificar-la o eliminar-la.

Cas d'ús 4 - Crear una nova tasca: En la barra de sota, al costat de les diferents opcions de visualitzacions hi ha la opció de crear una nova tasca, al utilitzar-la, vaig a la pantalla de tasca on tots els camps estan vuits. Puc editar cada un dels camps i a sota tinc la opció de guardar la tasca o cancel·lar-la i no guardar res. Al guardar una nova tasca, surt una pantalla informant que s'ha guardat correctament i com conseqüència totes les pantalles de visualitzacions es reinicien per que s'han actualitzat, de forma que puc veure la nova tasca en les diferents pantalles.

Cas d'ús 5 - Crear una nova tasca complint les condicions: Al crear una nova tasca, s'han de complir un parell de condicions: el nom de la tasca no pot començar amb un numero i la tasca ha de tenir per força un dia i hora programats. En cas de no complir alguna d'aquestes condicions, surt un avis en el moment de intentar guardar la tasca informant del error i a continuació deixa corregir-los de forma que puguis guardar la tasca satisfactòriament.

Cas d'ús 6 - Modificar/Eliminar una tasca: Al utilitzar l'opció de consultar el detall d'una tasca, dona l'opció de eliminar o modificar una tasca. Al utilitzar l'opció de eliminar, surt un avis de si estic segur i al acceptar elimina per sempre aquesta tasca. D'altra banda, al utilitzar l'opció de modificar, puc modificar cada un dels camps de la tasca, que abans no podia i guardar aquests canvis. Després de qualsevol d'aquests canvis, l'aplicació s'actualitza de nou i es poden veure aquests canvis en les visualitzacions.

8. Implementació de l'aplicació

En aquest apartat explicaré els passos realitzats per la implementació de l'aplicació. En els següents apartats explicaré els diferents mètodes utilitzats que no he creat jo i que els he descarregat d'Internet, els problemes que m'he trobat al llarg de la creació de l'aplicació i les solucions pressos i finalment la comparació del prototipatge realitzat anteriorment amb l'aplicació final.

Primer de tot vaig tenir que informar-me de tot el referent al Open API de la UOC. Per això, disposava d'una única web que ens facilita alguna informació per poder veure que és capaç de fer el Open API i com utilitzar alguns dels mètodes dels que disposa.

La web és: <http://open-api.uoc.edu/>



Imatge12 - Web documentació OpenAPI

La informació que hi ha a la web és molt extensa i aclareix molts dubtes de que és el que fa, però a la hora de que és el que s'ha de fer i com, aquesta informació és molt limitada. Aquest és el més gran dels problemes amb els que m'he trobat i que gracies al autor que va realitzar la web, va poder ajudar-me en aquest punt. Tot això ja ho explicaré amb més detall al punt de "Problemes trobats i solucions".

Durant la compressió, utilització i realització del codi relacionat amb el Open API, en paral·lel realitzava l'aplicació en sí, amb els diferents apartats i funcionalitats. L'aplicació consta principalment de 5 pantalles, en el que he utilitzat diferents recursos com: UIView, UITableView, UITextView, UITextField, UIButton, NSDictionary, NSArray, NSMutableArray, UIImageView i UILabel.

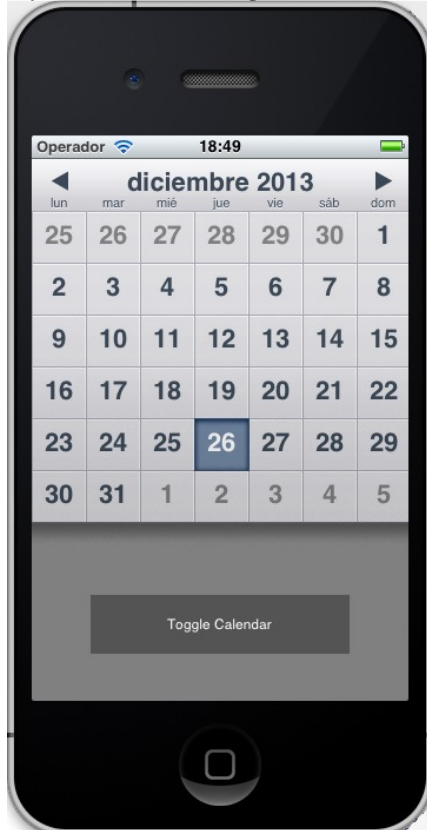
8.1. Mètodes de tercers utilitzats

He creat aquest apartat expressament per documentar una manca de recursos a l'hora de utilitzar el Interface Builder. Concretament en la part gràfica de la programació del Xcode, esperava trobar-me amb un recurs que pogués utilitzar una vista d'un calendari típic de iPhone i que no va ser així, per força tenia que programar-ho des de 0 i amb el temps del que disposava, vaig calcular que fer això em faria perdre tot el temps per realitzar la resta de l'aplicació.

Així que vaig trobar-me en què tenia que utilitzar un codi ja programat i implementar-ho a la meua aplicació. Cercant per Internet, vaig trobar el que necessitava i no només podia utilitzar aquest codi, sinó que a més la llicència em permet modificar-lo segons les meves necessitats. Aquest calendari s'anomena: Tapku Calendar.

La web on ho vaig descarregar és: <http://developinginthedark.com/posts/iphone-tapku-calendar-markers>

Aquesta és una imatge del calendari sense modificar res:



Imatge13 - Calendari Tapku

I aquesta és una imatge de com he modificat el calendari per personalitzar-ho al meu entorn:

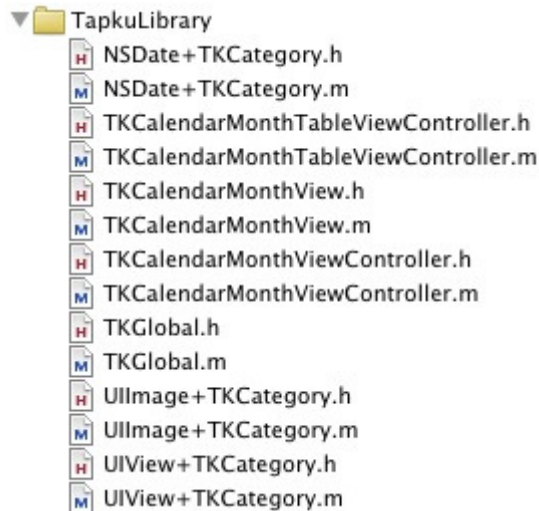


Imatge14 - Calendari Tapku modificat

Comentari: He esborrat la resta del contingut de la imatge per apreciar millor les diferències.

Principalment les modificacions que vaig fer són la mida total del calendari, ja que la mida original no em deixava espai per col·locar la resta d'elements, la mida de cada una de les cel·les i les imatges que hi ha al fons a cada una de elles i els noms dels dies.

Aquest calendari es tracta d'un objecte més (es crea i s'afegeix a la view que interessa). Les classes que componen aquest calendari són les següents:



Imatge15 - Classes Tapku

I la llicència que he comentat abans i que em permet tant utilitzar com modificar aquest codi està en totes les classes i el escrit és el següent:

```
//
//  TKCalendarMonthView.m
//  Created by Devin Ross on 6/10/10.
//
/*

tapku.com || http://github.com/devinross/tapkublibrary

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

*/
```

Imatge16 - Llicència Tapku calendar

8.2. Problemes trobats i solucions

M'he trobat amb diversos problemes que anomenaré a continuació:

Open API

El problema més gran amb el que m'he trobat a l'hora de fer l'aplicació, ha estat la mateixa eina que ens ofereix la UOC per connectar amb la seves dades, l'Open API.

Tal com he mencionat abans, hi ha la web de documentació en la que informa, a part d'altres coses, els mètodes dels que disposa, per així poder utilitzar-los en la nostra aplicació. D'altra banda, hi han uns petits exemples que es poden descarregar, utilitzar-los i veure el seu funcionament per aplicar-los en la nostra aplicació.

A la web <http://open-api.uoc.edu/documentacio/uoc-public-api/calendar-serviceimpl/> es pot veure com utilitzar per exemple el mètode GET:

/calendar/events

URI: [/api/v1/calendar/events](#)

GET

Get the events of the calendar. If there isn't any parameter it return all the events of the current week.

The user must have given the application the grant *READ* to use these operation.

Parameters

Name	Description	Type	Default Value
start	Start date in ISO8601 format (optional). If <i>end</i> parameter is not informed it returns the events of the next 30 days.	query	
end	End date in ISO8601 format (optional). If <i>start</i> parameter is not informed it returns the events of the previous 30 days.	query	

Response Body

element: [eventList](#)

EventList object with calendar's events.

Represents a list of events

Properties

Property	Type	Description
events	array of (event)	Event list.

Status Codes – Errors

The HTTP call return a status code depending on the result of the operation. These codes are listed below

Code	Description
200	Success Call
401	Not authorized. Maybe you have to authorize the user o the OAuth token has expired

Example

HTTP Call Example

```
GET http://denver.uoc.es:8080/webapps/uocapi/v1/calendar/events
```


I la descarrega del exemple 1 seria la següent:

branch: master openapidemos / ejemplo1 History

Changed sever name		
Xavi Aracil authored 2 months ago latest commit 89188f009a		
...		
Ejemplo1.xcodeproj	Revert "Commit inicial del ejemplo2 copiando el ejemplo1"	7 months ago
openApi	Changed sever name	2 months ago
Default-568h@2x.png	Carpeta por ejemplo	7 months ago
GTMHTTPFetcher.h	Carpeta por ejemplo	7 months ago
GTMHTTPFetcher.m	Carpeta por ejemplo	7 months ago
GTMOAuth2Authentication.h	Carpeta por ejemplo	7 months ago
GTMOAuth2Authentication.m	Carpeta por ejemplo	7 months ago
GTMOAuth2SignIn.h	Carpeta por ejemplo	7 months ago
GTMOAuth2SignIn.m	Carpeta por ejemplo	7 months ago
GTMOAuth2ViewControllerT...	Carpeta por ejemplo	7 months ago
GTMOAuth2ViewControllerT...	Carpeta por ejemplo	7 months ago
GTMOAuth2ViewTouch.xib	Carpeta por ejemplo	7 months ago
MenuViewController.h	Carpeta por ejemplo	7 months ago
MenuViewController.m	UIAlertView en el menú para las opciones no implementadas	7 months ago
ico114.png	Carpeta por ejemplo	7 months ago
ico57.png	Carpeta por ejemplo	7 months ago

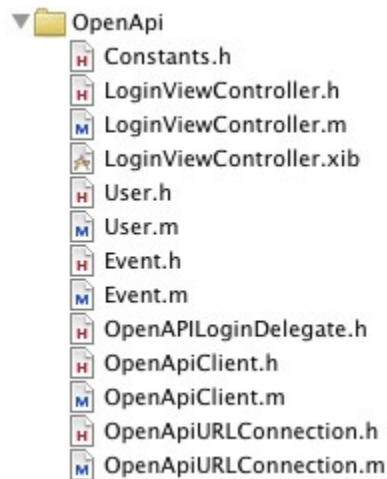
Imatge18 - Web Documentació descarrega Exemple1

Ara bé, tot això serveix sempre i quan utilitzis la versió del Xcode 4 o posterior i aquí és on vaig trobar amb el gran problema. Jo dispo de només del Xcode 3.2.6, per instal·lar-me el Xcode 4, és necessari que disposi de un S.O. més actual al Snow Leopard 10.6.6. que és el que tinc instal·lat a la màquina virtual, però no puc canviar-ho, per que si intento actualitzar-lo la màquina virtual deixa de funcionar. Així que em vaig trobar que per força tenia que utilitzar una versió antiga del Xcode i sense poder utilitzar gairebé res de la web de documentació.

Bé doncs, al trobar-me en una situació en que no podia avançar, vaig demanar ajuda al persona encarregada del Open API, en Xavi Aracil. Després de comentar-li aquest problema i el fet de no poder utilitzar les classes penjades dels exemples, varem trobar la solució. En lloc d'utilitzar les classes tipus GTMOAuth2 (tal com es veu a la imatge18), va crear un nou exemple específic per al Xcode 3.

Aquest exemple contenia els mètodes per fer les crides al Open API i altres que esperen la resposta d'aquest, al igual que els exemples amb GTMOAuth2, però funcionant amb Xcode3. Encara que el codi dels exemples de la web de documentació no els podia executar, vaig poder comparar els dos tipus de codi, veient que la similitud d'aquests era molt alta. D'aquesta forma, per fi vaig poder començar a programar aquesta part i que fins el moment no podia avançar, degut a les limitacions de un Xcode antic.

Al finalitzar la programació de gran part de l'aplicació les classes del Open API, queda de la següent manera:



Imatge19 - Classes Open API

Volia comentar també que les classes LoginView, User i Event, no serveixen exactament per realitzar les crides al Open API, si no més bé per fer el Login i tractar les dades que retorna el Open API d'una forma més còmode.

Login

Aquest no és exactament un problema, però la sensació, o dit d'altra manera, des del meu punt de vista sí que és un problema tal com vaig tenir que fer aquesta part de l'aplicació.

En un principi l'acció de fer login ha de ser amb dos simples passos, posar l'usuari i la contrasenya. Vaig preparar la part gràfica de manera que va quedar així:



Imatge20 - Login ideal

La idea era que al donar-li al botó 'Login', aquest executaria el codi per fer precisament aquesta acció. Al trobar-me amb el problema del Open API, vaig tenir que esperar a saber com fer-ho. Una vegada tenia la informació adequada, vaig adonar-me què el que havia fet, no servia per res, ja que per força, tinc que utilitzar una classe tipus 'Controller' amb una vista específica. Aquesta disposa dels mateixos camps user i contrasenya, i des de aquí es fa tant el Login, com la espera a la resposta del Open API. Una vegada, fet això, retorna a la pantalla anterior però amb les dades del login guardades en una variable, per utilitzar-la després en qualsevol moment en l'aplicació per fer altres consultes al Open API.

Per aquest motiu, vaig decidir no eliminar el que tenia fet i posar un TextView amb aquest avis i un botó, damunt dels camps que tenia preparats, de forma que visiblement es veuen els camps darrere d'aquest avis, quedant així:



Imatge21 - Login Modificat

Després de modificar aquest login, només vaig tenir que afegir el codi a la acció del botó 'continuar', per anar a l'altra vista forçada a utilitzar, per fer el Login i obtenir les dades d'accés.

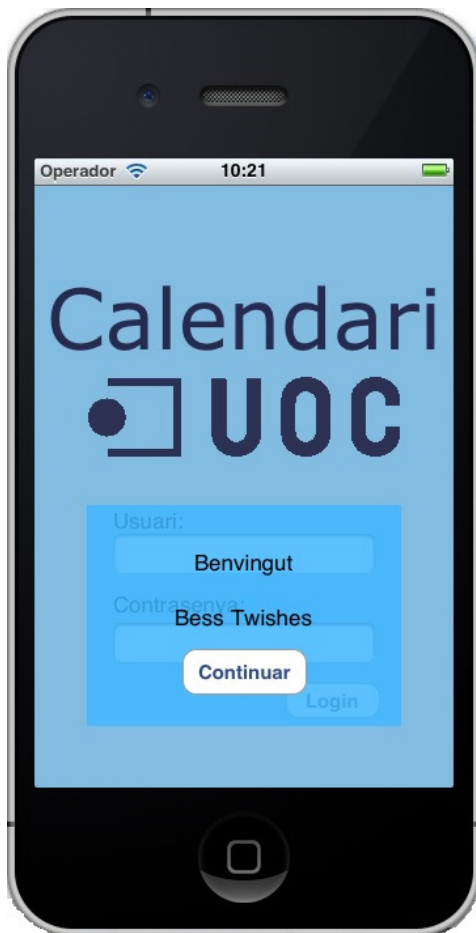
A continuació, adjunto captures de pantalla per veure la nova pantalla de Login i una següent pantalla per donar drets de lectura i escriptura:



Imatge22 - Login Modificat web



Imatge23 - Login Modificat drets



Imatge24 - Login Modificat final

Al tornar a la pantalla inicial del Login, ja es pot continuar per entrar a l'aplicació.

5 Subviews

Aquest és un altre gran problema que em vaig trobar i que a dia de avui encara no he trobat la solució que m'havia agradat aplicar.

L'aplicació consta, a part del Login, de cinc pantalles diferents en la que es pot visualitzar una en concret en qualsevol moment. Per navegar entre aquestes cinc pantalles, vaig crear un ViewController amb les cinc subviews i segons en quin botó de la ToolBar es pitjava, mostrava una pantalla o una altra, com és lògic.

Em vaig trobar amb el problema, concretament, quan vaig voler fer accions de canviar la subview que es mostrava des de una subview i no des de la ToolBar que vaig crear. Fins un punt, entenc que és lògic, ja que és el ViewController el que s'encarrega de canviar d'una subview a una altra, però estic convençut que hi ha alguna manera de fer la mateixa acció des de una subview.

Per aquest motiu, vaig estar molt temps cercant per Internet. Vaig fer moltes proves amb la informació que vaig trobar, vaig crear altres subviews, provar codi que en un principi feia exactament el que volia fer, però després de estar gairebé una setmana amb aquest problema, no vaig aconseguir el meu objectiu.

Així que vaig demanar ajuda al consultor Ignasi Lorente, per trobar una solució a aquest problema, però després d'explicar tot això per email, tampoc varem trobar la manera de solucionar-ho. Estava clar que no seria fàcil. Finalment al no trobar una solució, vaig decidir fer l'aplicació completament nova, però sense un ViewController i evidentment sense subviews.

Per tant vaig crear una única classe, amb totes les views juntes i crear la meua pròpia Toolbar, tenir els cinc botons en totes les views, que ocultava totes les views i mostrava una en concret, és a dir, feia les mateixes accions de un Toolbar, però sense utilitzar un d'aquests.

Aquesta solució comportava ajuntar tots els objectes en una mateixa classe (a la part gràfica era exactament el mateix, tenir tots els objectes junts en una mateixa view) de forma que l'ordre a l'hora de programar va ser el més important. Vaig crear seccions, amb comentaris, on es veia de forma molt clara les diferents views amb les seves declaracions de variables, releases, mètodes, etc...

Per exemple la classe .h va quedar així:

```

//////////////////////////////////// General //////////////////////////////////////
UIView *vLogin, *vMes, *vSetmana, *vDia, *vLlista, *vTasca;
NSDictionary *dadesLogin;
NSMutableArray *arrayTasques;
NSArray *idTasquesMes, *idTasquesSetmana, *idTasquesDia, *idTasques;

//////////////////////////////////// Vista Login //////////////////////////////////////
IBOutlet UIButton *bLogin, *bContinuar;
IBOutlet UITextView *tComentari;
IBOutlet UITextField *tUsuari;
IBOutlet UITextField *tContrasenya;
bool loginFet;

//////////////////////////////////// Vista Mes //////////////////////////////////////
IBOutlet UIButton *bMes_M, *bSetmana_M, *bDia_M, *bLlista_M, *bTasca_M;
TKCalendarMonthView *calendari;
IBOutlet UITableView *taulaDadesMes;
NSMutableArray *seccionsMes;
NSMutableArray *dadesMes;

//////////////////////////////////// Vista Setmana //////////////////////////////////////
IBOutlet UIButton *bMes_S, *bSetmana_S, *bDia_S, *bLlista_S, *bTasca_S;
IBOutlet UIButton *bDilluns, *bDimarts, *bDimecres, *bDijous, *bDivendres, *bDissabte, *bDiumenge;
IBOutlet UIButton *bDretaSetmana, *bEsquerreSetmana;
IBOutlet UILabel *etSetmana, *etDillunsSetmana;
IBOutlet UITableView *taulaDadesSetmana;
NSMutableArray *seccionsSetmana;
NSMutableArray *dadesSetmana;

//////////////////////////////////// Vista Dia //////////////////////////////////////
IBOutlet UIButton *bMes_D, *bSetmana_D, *bDia_D, *bLlista_D, *bTasca_D;
IBOutlet UIButton *bDretaDia, *bEsquerreDia;
IBOutlet UILabel *etDia;
IBOutlet UITableView *taulaDadesDia;
NSMutableArray *seccionsDia;
NSMutableArray *dadesDia;

//////////////////////////////////// Vista Llista //////////////////////////////////////
IBOutlet UIButton *bMes_L, *bSetmana_L, *bDia_L, *bLlista_L, *bTasca_L;
IBOutlet UITableView *taulaDadesLlista;
NSMutableArray *seccionsLlista;
NSMutableArray *dadesLlista;

```

Imatge25 - Classe .h

Bé, aquesta és la solució que vaig trobar, però si trobo la manera com fer-ho amb la idea inicial, faré de nou l'aplicació, ja que aporta molt més ordre i claredat, amb la creació de cada una de les classes, la part gràfica, utilització de una Toolbar, etc...

Mètodes Post, Put, Delete

Aquest apartat és part del problema del Open API, però vull dedicar-li un apartat únic per aquests mètodes, ja que em vaig trobar amb altres problemes a part d'aprendre com utilitzar-los.

A part d'aquests mètodes, també existeix el mètode 'Get' que tracta de demanar dades al Open API i tractar les dades rebudes després per utilitzar-les a l'aplicació. Aquest mètode és relativament senzill, ja que només es tracta d'una crida a la direcció: <http://oslo.uoc.es:8080/webapps/uocapi/api/v1/calendar/events>


```
#define kAPIUrl @"http://oslo.uoc.es:8080/webapps/uocapi/api/v1"

- (void) loadEvents{
    NSURL *eventsURL = [NSURL URLWithString:
        [NSString stringWithFormat:
            @"%@/calendar/events?access_token=%@", kAPIUrl, [self accessToken]]];

    OpenAPIURLConnection *connection = [[[OpenAPIURLConnection alloc] init] autorelease];
    connection.delegate = self;
    [connection getOpenAPIData:eventsURL];
}
```

Imatge26 - Codi Get

Una vegada mostrat com es fa la crida al mètode Get, les crides als Push, Put i Delete, són molt similars:

```
- (void) guardaNovaTasca:(NSString *) data {
    NSURL *eventsURL = [NSURL URLWithString:
        [NSString stringWithFormat:
            @"%@/calendar/events?access_token=%@", kAPIUrl, [self accessToken]]];

    OpenAPIURLConnection *connection = [[[OpenAPIURLConnection alloc] init] autorelease];
    connection.delegate = self;
    connection.sendJSON = YES;
    [connection postOpenAPIData:data url:eventsURL];
}

- (void) guardaCanvisTasca:(NSString *) data {
    NSURL *eventsURL = [NSURL URLWithString:
        [NSString stringWithFormat:
            @"%@/calendar/events?access_token=%@", kAPIUrl, [self accessToken]]];

    OpenAPIURLConnection *connection = [[[OpenAPIURLConnection alloc] init] autorelease];
    connection.delegate = self;
    connection.sendJSON = YES;
    [connection putOpenAPIData:data url:eventsURL];
}

- (void) eliminaTasca:(NSString *) identificador {
    NSURL *idEventURL = [NSURL URLWithString:
        [NSString stringWithFormat:
            @"%@/calendar/events/%?access_token=%@", kAPIUrl, identificador, [self accessToken]]];

    OpenAPIURLConnection *connection = [[[OpenAPIURLConnection alloc] init] autorelease];
    connection.delegate = self;
    connection.sendJSON = YES;
    [connection deleteOpenAPIData:idEventURL];
}
```

Imatge27 - Codi Push, Put i Delete

Al fer les crides, l'Open API les processa de la següent forma:

```
-(void) requestOpenAPIUrl:(NSURL *) url method:(NSString *) method data:(NSString *) data {
    [[UIApplication sharedApplication] setNetworkActivityIndicatorVisible:YES];
    NSMutableURLRequest *request = [[NSMutableURLRequest alloc] initWithURL:url];
    [request setHTTPMethod:method];
    if (self.sendJSON) { //Si el format de les dades està en json
        [request setValue:@"application/json; charset=UTF-8" forHTTPHeaderField:@"Content-Type"];
    }
    if (data) {
        [request setHTTPBody:[data dataUsingEncoding:NSUTF8StringEncoding]];
    }
    urlConnection=[[NSURLConnection alloc] initWithRequest:request delegate:self];
    receivedData = [[NSMutableData alloc] init];
}

-(void) getOpenAPIData:(NSURL *) url {
    [[UIApplication sharedApplication] setNetworkActivityIndicatorVisible:YES];
    NSURLRequest *request = [[NSURLRequest alloc] initWithURL:url];
    urlConnection=[[NSURLConnection alloc] initWithRequest:request delegate:self];
    receivedData = [[NSMutableData alloc] init];
}

-(void) postOpenAPIData:(NSString *) data url:(NSURL *) url {
    [self requestOpenAPIUrl:url method:@"POST" data:data];
}

-(void) putOpenAPIData:(NSString *)data url:(NSURL *)url {
    [self requestOpenAPIUrl:url method:@"PUT" data:data];
}

-(void) deleteOpenAPIData:(NSURL *) url {
    [self requestOpenAPIUrl:url method:@"DELETE" data:nil];
}
```

Imatge28 - Codi Connections

En aquest punt, ja només queda esperar la resposta en altres mètodes encarregats d'aquesta tasca:

```
#pragma mark -
#pragma mark NSURLConnection delegate methods
- (void)connection:(NSURLConnection *)connection didReceiveData:(NSData *)data{
    [receivedData appendData:data];
}

- (void)connection:(NSURLConnection *)connection didFailWithError:(NSError *)error{
    [[UIApplication sharedApplication] setNetworkActivityIndicatorVisible:NO];
    [self.delegate openApiURLConnection:self didFailWithError:error];
}

- (void)connectionDidFinishLoading:(NSURLConnection *)connection{
    [[UIApplication sharedApplication] setNetworkActivityIndicatorVisible:NO];
    NSString *response = [[NSString alloc] initWithData:receivedData encoding:NSUTF8StringEncoding];
    SBJsonParser *jResponse = [[SBJsonParser alloc] initWithData:response];
    NSDictionary *serverData = [jResponse objectForKey:@"serverData"];
    [self.delegate openApiURLConnection:self didFinishSuccessfully:serverData];
}
```

Imatge29 - Codi Resposta

Una vegada entès el funcionament dels mètodes del Open Api, és quan em vaig trobar amb el primer problema amb aquest: el mètode Push. Aquest mètode s'utilitza per guardar tasques noves, però resulta que aquest mètode no funcionava, al informar de les proves que vaig realitzar, es va saber que hi havia una incidència al Open API, que es va solucionar al cap d'uns dies. Després d'això, ja va funcionar correctament.

Després em vaig trobar amb el segon problema amb els mètodes Put i Delete, que serveixen concretament per modificar una tasca ja creada i eliminar-la, respectivament. Aquest problema, no està resolt, ja que aquests mètodes no estan implementats al Open API. Per tant, quan aquests mètodes estiguin implementats i funcionant, realitzaré els canvis necessaris i proves a l'aplicació. Aquesta part l'explicaré amb més profunditat en l'apartat '9. Millores futures'

8.3. Comparació amb el prototipatge

En aquest apartat, explicaré amb imatges la comparació del que va ser el prototipatge, amb el resultat final de l'aplicació.

La estructura de les imatges serà la següent:



Imatge30 - Comparació estructura

A continuació, començaré per la part del Login i després amb les pantalles de visualitzacions. Login:



Imatge31 - Comparació Login

Visualització per mes:



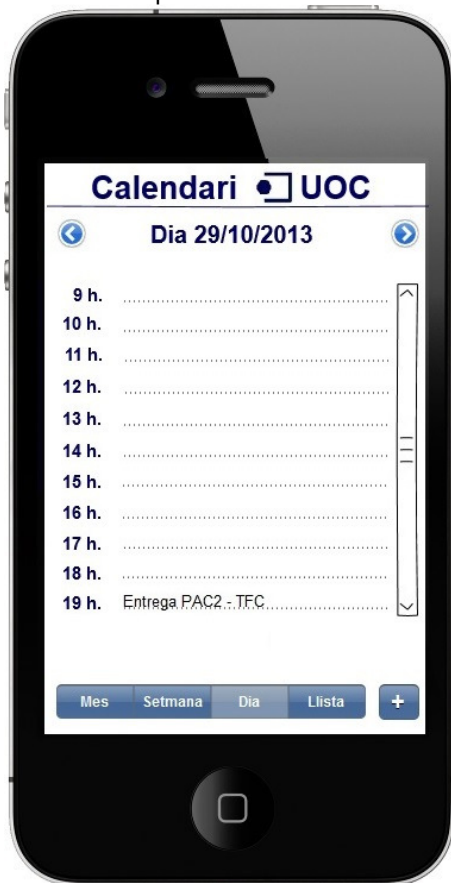
Imatge32 - Comparació Visualització Mes

Visualització per setmana:



Imatge33 - Comparació Visualització Setmana

Visualització per dia:



Imatge34 - Comparació Visualització Dia

Visualització per llista:



Imatge35 - Comparació Visualització LLista

Nota tasca:



Imatge36 - Nova Tasca

Visualització d'una tasca:



Imatge37 - Visualització d'una Tasca

Eliminació d'una tasca:



Imatge38 - Eliminació d'una Tasca

Aquesta és la comparació amb totes les pantalles que es van dissenyar al prototipatge, però en l'aplicació vaig afegir un parell més que indiquen les accions realitzades a les tasques:



Imatge39 - Canvis en tasques

9. Conclusions

Finalment, després d'acabar el projecte de fi de carrera, puc dir que he après molt sobre aquesta tecnologia de la programació per mòbils iPhone i per la experiència d'aquests mesos, sé que encara em falta molt per aprendre. En aquest projecte, he utilitzat els coneixements adquirits en el curs del l'any passat, trobar codi i poder reutilitzar-lo, he après a utilitzar l'Open API de la UOC, cosa que m'ajudarà algun dia que treballi en aquest àmbit ja que a tot arreu s'utilitzen connexions externes, d'una manera o una altra. D'altra banda, també he experimentat l'utilització de 'Desenvolupador d'Apple' per instal·lar l'aplicació al iPhone.

Per tot això, puc garantir que l'experiència adquirida durant el temps d'aquest projecte ha estat molt gratificant.

9.1. Millores futures

En aquest apartat explicaré les millores futures, encara que puc dir, que en resum són la resolució total del problemes esmentats en l'apartat corresponent.

La millora més important, és la de tenir accés als mètodes Put i Delete, per completar totes les funcionalitats de l'aplicació. Al no tenir-les actualment i veure l'aplicació tal com està ara, dona la sensació de que no està acabada, que falta molt per fer, però la veritat és que sí estan aquest mètodes implementats. Només falta que estiguin llestos en el Open API i fer les proves adients a l'aplicació, per comprovar què el que està fet, funciona correctament.

Una altra millora important des del meu punt de vista, és afegir els colors de les tasques programades, tal com vaig proposar en el prototipatge i no com està ara, que mostra un puntet a sota del numero del dia que està programada. Per fer això necessito més temps i que actualment no tinc. Això m'ho vaig proposar en la mateixa programació de l'aplicació, però amb els problemes que em vaig trobar i que no vaig preveure perdre tant temps, m'he obligat a deixar-ho per millores futures. A continuació ho explico amb un parell d'imatges:

Actual

enero 2014						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2

Millora

enero 2014						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2

Imatge40 - Millora Calendari

Per últim, encara que aquesta millora és menys important, és la de millorar l'aspecte general de l'aplicació. Dic que és menys important, ja que tal com l'he deixat actualment ja m'agrada, però quan tingui temps, vull dedicar-li més temps per millorar el seu aspecte i aconseguir que sigui més moderna i intuïtiva.