



Ecole d'ingénieurs et d'architectes de Fribourg
Hochschule für Technik und Architektur Freiburg



École supérieure d'ingénieurs de Beyrouth

Thèse de Bachelor :

ESIB@Pad

Rapport final

Auteur Elias Medawar
 elias.medawar@edu.hefr.ch

Responsables Internes Omar Abou Khaled Elena Mugellini
 omar.aboukhaled@hefr.ch elena.mugellini@hefr.ch

Responsable externe Dany Mezher
 dany.mezher@fi.usj.edu.lb

Experts Marc Wuergler Roland Marro
 marc.wuergler@gmail.ch marmor@fr.ch

Version 1

22 août 2011

Chapitre 1

A propos

1 Elias Medawar



24 ans

Tel : +41 76 409 0330

Célibataire

E-mail : elias.medawar@gmail.com

Suisse et Libanais

1.1 Compétences personnelles

Passionné, Flexible, Autonome

1.2 Formation

2008 à aujourd’hui

Bachelor bilingue(Français-Allemand) en Informatique

A l’Ecole d’ingénieurs et d’architectes de Fribourg et 1 semestre d’échange à la Hochschule Luzern

2004 à 2008

Maturité professionnelle et apprentissage en Informatique

Lycée Technique Baptiste Savoie de St-Imier

1.3 Informatique

	Maîtrisé	Connaissances
Programmation :	JAVA et JAVA EE, Objective-C(iPhone et iPad) , C#, PHP, Matlab	AS3, C , C++ , VHDL
Base de données :	Oracle , MySQL	SQL server
Méthodologie :	UML	Scrum
Bureautique :	Suite office, LATEX	

1.4 Activités professionnelles et projets

juin-août 2011

École supérieure des Ingénieurs de Beyrouth, Liban Travail de bachelor
Création d’une application iPhone/iPad qui permet aux étudiants et professeur de l’université d’accéder au système d’information de l’école.

Acquis : Gestion de projet et recherche de solutions innovantes.

2008 à aujourd’hui

Multimedia Information System Group Fribourg

Collaboration dans différents projets de recherche.

- MAGI System , test de l’interaction home-machine à l’aide des pensées.
- WiiNote, Annotation d’images à l’aide des manettes de la console Wii.

Acquis :Autonomie et recherche de solutions innovantes.

09/2010 - 02/2011

Tura AG, Rotkreuz projet de semestre à Luzern

Implémentation d’un serveur webDAV en php pour la plateforme YaOOS.

- Permet le mapage des données comme lecteur réseaux en local.

Acquis :Gestion de projet, analyse des besoins.

mars - aout 2008

Techniques Electroniques Moutier Sàrl

- Création d’un CMS pour la gestion dynamique du site de l’entreprise.

Acquis : Rigueur, travail d’équipe.

2006 à 2009

Création de sites web en free-lance

Acquis :Gestion des priorités, sens de la communication,résistance au stress.

1.5 Langues

Français : Langue maternelle.

Arabe : Parlé couramment, notions d’écriture.

Allemand : Courant (niveau B2-C1).

Anglais : Scolaire (niveau A2), bon vocabulaire technique.

Table des matières

1 A propos	3
1 Elias Medawar	5
1.1 Compétences personnelles	5
1.2 Formation	5
1.3 Informatique	5
1.4 Activités professionnelles et projets	5
1.5 Langues	6
2 Introduction	11
1 Contexte	13
2 Structure du rapport	13
3 Software Project Management Plan	15
1 Introduction	17
2 Organisation	17
2.1 Séances	17
2.2 Site internet	17
2.3 Communication	18
3 Planification	19
3.1 Plan global	19
3.2 Description des jalons	19
3.3 Planification détaillé	22
4 Processus technique	23
5 Gestion des risques	24
5.1 État le 01/06/2011	24
5.2 État le 04/07/2011	25
5.3 État le 08/08/2011	26
6 Gestion des configurations	26
7 Gestion de la documentation	27
8 Gestion des finances	28
8.1 Ressources humaines	28
8.2 Ressources matérielles	28
4 Software Requirement Specification	29
1 Introduction	31
2 Analyse	31
2.1 Aperçu global	31

2.2	Use case	32
2.3	Description des utilisateurs	32
2.4	Faisabilité	33
3	Spécification des exigences	33
3.1	Spécification des interfaces	33
3.2	Exigences fonctionnelles	35
5	Software Design Description	49
1	Introduction	51
1.1	But du chapitre	51
1.2	Aperçu du chapitre	51
2	Architecture du système	51
2.1	Architecture choisie	51
2.2	Discussion des alternatives d'architectures	52
2.3	Composants du système	53
3	Conception et Implémentation des composants	55
3.1	Compatibilité graphique iPhone,iPad	55
3.2	Système de cache	56
3.3	Accès aux données des services web	57
3.4	Base de données	61
3.5	Navigation	63
3.6	Settings	67
3.7	Map	69
3.8	News	73
3.9	Directory	77
3.10	Calendrier	82
3.11	ExamResult	85
6	Software Test Description	89
1	Introduction	91
1.1	Philosophie de test	91
2	Organisation des tests	92
2.1	Éléments à tester	92
2.2	Éléments à ne pas tester	92
2.3	Outils de test et environnement	92
3	Cas de test	92
3.1	Navigation	92
3.2	Paramétrage	93
3.3	Carte	94
3.4	News	94
3.5	Directory	95
3.6	Calendrier	95
3.7	Résultats des examens	96
4	Protocole de test	97
4.1	Protocole de test 1	97
4.2	Protocole de test 2	98
4.3	Protocole de test 3	100
4.4	Protocole de test 4	103
4.5	Protocole de test 5	106

7 Conclusion	111
1 Évolutions futures	113
1.1 Accès à toutes les données via les services web	113
1.2 Nouvelles fonctionnalités	113
1.3 Version de l'application sur d'autres appareils	113
1.4 Une version multilingue du projet.	113
1.5 Des cartes plus complètes des campus	113
1.6 Accessible pour d'autres universités	114
2 Impressions personnelles	114
3 Remerciements	114
4 Conclusion	115
8 Indexes	117
Glossary	121
9 Annexes	127
1 Imprimées	127
1.1 Annexe A - directive du projet	127
1.2 Annexe F Flyer	127
1.3 Annexe G Cahier des charges	127
1.4 Annexe H Journal de bord	127
2 le CD	127

Chapitre 2

Introduction

1 Contexte

Ce projet s'inscrit dans le cadre du travail de Bachelor pour l'obtention du diplôme de Bachelor à l'Ecole d'ingénieurs et d'architectes de Fribourg. Il est réalisé à l'Université Saint-Joseph au Liban.

Le potentiel des nouveaux appareils mobiles comme l'iPhone et l'iPad est énorme et il n'est jusqu'à présent pas exploité pour accéder au système d'information de l'USJ . A l'aide de ce projet, nous allons exploiter aux mieux les nouvelles technologies et permettre aux utilisateurs du campus de consulter simplement et rapidement des informations telles que le programme des cours, les notes des examens, le plan du campus ou encore l'annuaire du campus.

2 Structure du rapport

La documentation du projet a été faite selon la norme IEEE 829-1998 sur la documentation de projets informatiques. Cette norme regroupe les informations selon leurs types par documents.

Dans les équipes de développement d'une grande taille, chaque personne à un rôle dans le projet(Testeur, Architecte, Designer,Chef de projet, etc..). La norme permet en outre de lier le rôle d'un ingénieur à un document.Par exemple le testeur sera responsable du STD(Software Test Documentation), le chef de projet est responsable du SPMP(Software Project Management Plan), etc. Dans le cas présent, une seule personne travaille sur le projet, mais grâce à cette manière de documenter, le rôle de la personne qui est censée rédiger le document est clairement défini et les points importants qui y sont liés également. Ces différents rôles sont ensuite tous repris un à un et les tâches qui y sont associées sont effectuées selon ce modèle.

Pour une question de lisibilité, les différents documents ont été regroupés sous forme de chapitres pour former le rapport final.

Le chapitre 3 à partir de la page 15 regroupe toutes les informations concernant la gestion du projet. Ensuite, à partir de la page 29, vient le chapitre 4, qui contient toutes les informations sur l'analyse faite durant le projet. Les pages 49 à 89 rassemblent toute la documentation de la conception et de l'implémentation de l'application. Pour finir, à la page 89, tout ce qui concerne les tests et leurs résultats est expliqué. Une conclusion termine le rapport. Elle est suivie par les annexes et informations sur le contenu du CD.

Chapitre 3

Software Project Management Plan



1 Introduction

Ce chapitre regroupe toutes les informations relatives à la gestion du projet et du processus de développement. Ce chapitre est mis à jour régulièrement afin de permettre à tout moment d'avoir un aperçu de l'avancement du projet.

2 Organisation

L'annexe A (/Documentation/Annexes/A_Directives) contient les directives qui ont été distribuées au début du projet. Ces directives sont la ligne directrice concernant l'organisation du projet et sont complétées par ce document. Le cahier des charges de l'annexe G a été créé et validé en début de projet, il est la référence en ce qui concerne les objectifs à atteindre.

2.1 Séances

- Des séances hebdomadaires seront effectuées avec M. Dany Mezher, le responsable externe .
- Une séance est tenue si possible toute les 2-3 semaines via Skype avec Mme Mugellini et M. Abou Khaled, les responsables internes.
- Au moins une séance est organisée avec les experts M. Roland Marro et M. Marc Wuergler via Skype.

2.2 Site internet

Un site internet est mis en ligne et il est disponible à l'adresse suivante : <https://forge.tic.eia-fr.ch/projects/esibpad>

Mme Mugellini et M. Abou Khaled peuvent utiliser leur login AII pour accéder aux données.

Pour les autres personnes, des comptes ont été créés par le service informatique de l'école. Les comptes sont valides jusqu'au 31.12.2011.

M.Dany Mezher	
Username :	dany.mezher
Password :	voir mail
M.Marc Wuergler	
Username :	marc.wuergler
Password :	voir mail
M.Roland Marro	
Username :	roland.marro
Password :	voir mail

TABLE 3.1: Données de login pour les personnes externe à l'EIA-FR

Le site contiendra :

- le journal de bord :<https://forge.tic.eia-fr.ch/projects/esibpad/wiki>
- Les PVs :<https://forge.tic.eia-fr.ch/projects/esibpad/documents>
- La documentation sous format PDF :<https://forge.tic.eia-fr.ch/projects/esibpad/documents>

2.3 Communication

Le moyen de communication principal est l'e-mail.

Nom	email	Téléphone [Fixe, Mobile]
M. Würgler Marc	marc.wuergler@sunrise.ch	+41 26 660 03 04, +41 78 609 49 44
M. Marro Roland	marmor@fr.ch	+41 26 305 31 61
M. Dany Mezher	dany.mezher@fi.usj.edu.lb	+961 142 134 1,+961 700 100 30
Mme Elena Mugellini	elena.mugellini@hefr.ch	+41 26 429 68 70
M. Omar Abou Khaled	omar.aboukhaled@hefr.ch	+41 26 429 65 89
M. Elias Medawar	elias.medawar@edu.hefr.ch	+961 712 900 72, +41 764 090 330

TABLE 3.2: Résumé des adresses e-mail et des numéros de téléphone.

Des rendez-vous pour des vidéo-conférences seront organisés à l'aide de Skype.

3 Planification

3.1 Plan global

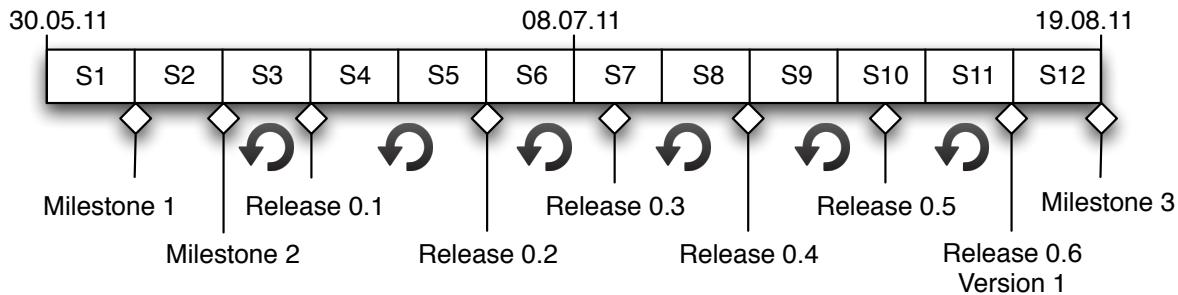


FIGURE 3.1: Vue détaillée du planning du projet

3.2 Description des jalons

Nom	Date	But à atteindre
Milestone 1	03.06.11	<ul style="list-style-type: none"> – Le cahier des charges est établi et validé. – La première version du SPMP est rédigée.
Milestone 2	10.06.11	<ul style="list-style-type: none"> – Pouvoir déployer une simple application sur l'iPhone et l'iPad. – Un environnement de développement local est mis en place, avec des Web Services de test ainsi que des données de test.
Release 0.1	17.06.11	<ul style="list-style-type: none"> – La page d'accueil de l'application avec les différents menus est réalisée. – La page de paramètres de l'application est réalisée.
Suite à la page suivante		

suite de la page précédente		
Nom	Date	But à atteindre
Release 0.2	01.07.11	<ul style="list-style-type: none"> – Afficher la carte du campus. a) La position actuelle de l'utilisateur sera détectée à l'aide du GPS de l'appareil et affichée sur la carte. b) L'utilisateur peut, à l'aide de la fonction "chercher" : trouver l'emplacement d'une salle ou le bureau d'une personne. c) Les informations de la carte sont enregistrées sur le serveur et peuvent être mises à jour à tout moment. Un système de cache évite de recharger la carte à chaque visite.
Release 0.3	13.07.11	<ul style="list-style-type: none"> – Permettre de consulter les nouvelles du campus. a) Si une nouvelle est liée à un lieu, permettre de l'afficher facilement sur la carte.
Release 0.4	22.07.11	<ul style="list-style-type: none"> – Permettre l'accès à l'annuaire de l'université. a) Quand on clique sur un numéro de téléphone, l'appel est lancé. b) Quand on clique sur une adresse mail, la fenêtre d'envoi de mail de l'appareil est ouverte.
Release 0.5	03.08.11	<ul style="list-style-type: none"> – Permettre aux professeurs et aux étudiants d'afficher leurs horaires. a) Quand on clique sur un cours, l'emplacement de ce dernier est affiché sur la carte. b) L'utilisateur peut sauvegarder son horaire sur l'appareil pour un accès offline.
Release 0.6 Version 1	03.08.11	<ul style="list-style-type: none"> – Permettre aux étudiants de consulter le résultat des examens. Cet objectif est conditionné par l'accord de l'administration et du service informatique de l'ESIB.
Suite à la page suivante		

suite de la page précédente		
Nom	Date	But à atteindre
Milstone 3	19.08.11	<ul style="list-style-type: none">– L'application est prête à être publiée sur l'App store.– La documentation est finie.– La présentation finale est prête.

TABLE 3.3: Description des jalons. Les date de releases sont considérées comme des jalons.

3.3 Planification détaillé

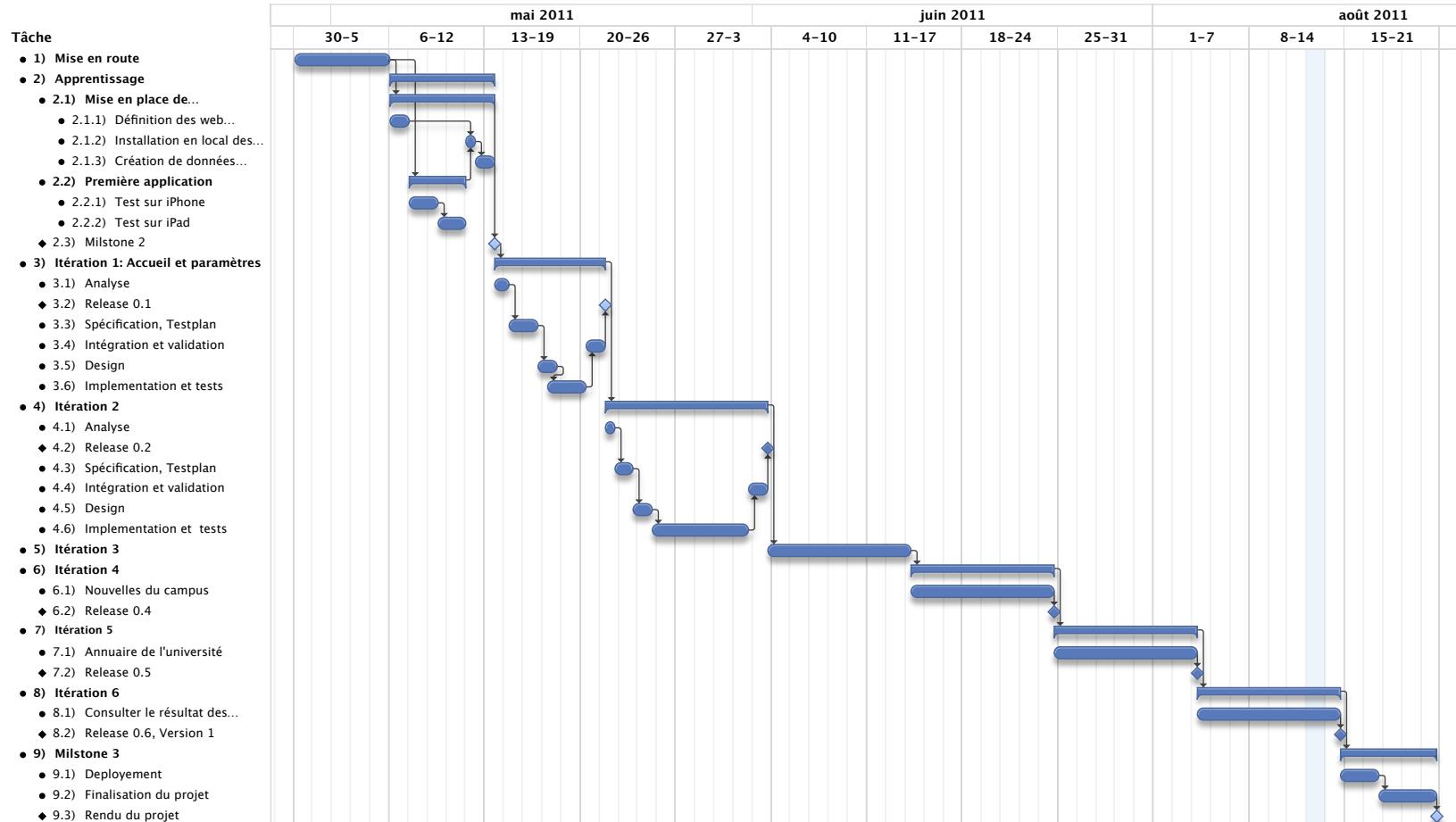


FIGURE 3.2: Vue globale du planning du projet avec les milestones

La planification détaillée de chaque itération n'est pas faite, elle sera faite au début de chaque itération si besoin est. Etant seul à travailler sur les tâches et vu que les itérations sont courtes, une planification plus détaillée est inutile. En début d'itération, une liste de tâches à faire (TODO) est faite avec une estimation du temps nécessaire pour atteindre l'objectif. Ce fonctionnement s'approche de la méthode de travail Scrum.

4 Processus technique

Sur la Figure 3.1 nous pouvons voir que nous allons travailler par itérations. Voici une définition plus détaillée de ce que l'on entend par itération

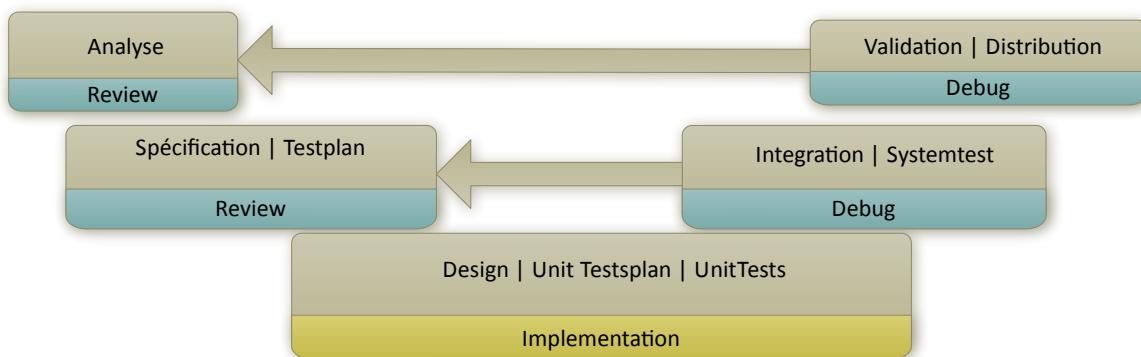


FIGURE 3.3: Illustration du modèle de développement en V qui est appliqué à chaque itération.

“Le modèle du cycle en V a été imaginé pour pallier au problème de réactivité du modèle en cascade. Ce modèle est une amélioration du modèle en cascade qui permet, en cas d'anomalie, de limiter un retour aux étapes précédentes. Les phases de la partie montante doivent renvoyer de l'information sur les phases en vis-à-vis lorsque des défauts sont détectés afin d'améliorer le logiciel. De plus, le cycle en V met en évidence la nécessité d'anticiper et de préparer dans les étapes descendantes les « attendus » des futures étapes montantes : ainsi les attendus des tests de validation sont définis lors des spécifications, les attendus des tests unitaires sont définis lors de la conception, etc. Le cycle en V est devenu un standard de l'industrie du développement de logiciel et de la gestion de projet depuis les années 1980. “[4]

Ainsi cette approche sera nommée une itération et elle sera répétée à chaque release pour arriver au but qui a été fixé.

5 Gestion des risques

Les divers risques qui mèneraient à un échec du projet sont résumés ici. Le but est de mettre à jour les risques régulièrement, et de faire qu'ils diminuent au plus vite.

5.1 État le 01/06/2011

Nr.	Risque	P	DC	I	Mesure
T1	Le peu d'expérience dans le développement Objective-C induit en erreur(sous estimation de la charge de travail, mauvaise architecture,etc) lors de la prise de décisions importantes au début du projet	3	3	9	Discuter les décisions avec des personnes ayant de l'expérience dans le domaine, prendre le temps d'apprendre les bases de l'Objective-C au début du projet et prévoir une tâche simple pour la première itération.
T2	Les services web ne sont pas encore opérationnels et peuvent retarder l'avancement du projet	3	3	9	Prévoir un environnement de développement en local avec des Web Services de test indépendants.
N1	La méthodologie de travail au sein de l' EIA-FR différent trop de celle de l'ESIB et les méthodes ne conviennent pas à l'une ou l'autre partie.	1	2	2	Organiser régulièrement des séances pour valider les décisions.
N2	Sous-estimation de la charges de travail, dépassement du temps mis à disposition.	2	2	4	Travailler par itération et se baser sur l'expérience acquise lors des itérations précédentes pour bien planifier les suivantes.Ne pas rester bloqué sur une étape sans demander de l'aide.

TABLE 3.4: Risques identifié au lancement du projet.

Légende :

Tx = Risque technique

Nx = Risque non technique

P = Probabilité (1 peu probable / 3 très probable)

DC = Dégât et conséquence (1 peu / 3 grave)

I = Importance ([1-2 petite][3-4 moyenne][5-9 sérieux])(P*DC)

Les risques T1 et T2 qui sont d'une grande importance ont été pris en considération pour la planification.

5.2 État le 04/07/2011

Nr.	Risque	P	DC	I	Mesure
T1	Le peu d'expérience dans le développement Objective-C induit en erreur(sous estimation de la charge de travail, mauvaise architecture,etc) lors de la prise de décisions importantes au début du projet	2	3	6	Ce risque à diminuer suite à l'expérience acquise durant la première itération.
T2	Les Web Services ne sont pas encore opérationnels et peuvent retarder l'avancement du projet	1	3	3	Diminution suite à la création d'un environnement de test stable et contrôlable. Une première version des web services a été mis en place par le service informatique de l'USJ .
N1	La méthodologie de travail au sein de l' EIA-FR différent trop de celle de l'ESIB et les méthodes ne conviennent pas à l'une ou l'autre partie.	1	2	2	Organiser régulièrement des séances pour valider les décisions.
N2	Sous-estimation de la charges de travail, dépassement du temps mis à disposition.	3	2	6	Suite au retard pris lors de la première itération ce risque augmente.

TABLE 3.5: Risques après la première itération.

Légende :

Tx = Risque technique

Nx = Risque non technique

P = Probabilité (1 peu probable / 3 très probable)

DC = Dégât et conséquence (1 peu / 3 grave)

I = Importance ([1-2 petite][3-4 moyenne][5-9 sérieux])(P*DC)

Les probabilités des risques T1 et T2 ont diminué après le premier mois de développement tandis que celle du risque N2 a augmenté. On peut constater que globalement les risques diminuent. Il faut travailler sur la planification pour diminuer au plus vite le risque N2.

5.3 État le 08/08/2011

Nr.	Risque	P	DC	I	Mesure
T1	Le peu d'expérience dans le développement Objective-C induit en erreur(sous estimation de la charge de travail, mauvaise architecture,etc) lors de la prise de décisions importantes au début du projet	0	3	0	Ce risque peut être considéré nul vu l'expérience acquise jusqu'à présent
T2	Les Web Services ne sont pas encore opérationnels et peuvent retarder l'avancement du projet	1	1	1	aucun changement.
N1	La méthodologie de travail au sein de l' EIA-FR différent trop de celle de l'ESIB et les méthodes ne conviennent pas à l'une ou l'autre partie.	0	2	0	Le système des 2 écoles n'est pas compatibles et la manière de travail convient aux 2 selon les différentes séances organisées et les commentaires émis. .
N2	Sous-estimation de la charges de travail, dépassement du temps mis à disposition.	1	2	2	Des heures de travail supplémentaires ont été faites pour rattraper le retard et on est dans le temps selon la planification.

TABLE 3.6: Risques après la première itération.

Légende :

Tx = Risque technique

Nx = Risque non technique

P = Probabilité (1 peu probable / 3 très probable)

DC = Dégât et conséquence (1 peu / 3 grave)

I = Importance ([1-2 petite][3-4 moyenne][5-9 sérieux])(P*DC)

A cette étape du projet, on peut dire que les risques d'échec sont quasi nuls, il faut toutefois garder un rythme de travail assez soutenu pour parvenir au bout à temps.

6 Gestion des configurations

Afin de garder des traces de l'évolution du projet, des versions des sources des documents seront sauvegardées sur un serveur SVN. La version courante du projet est hébergée à l'adresse suivante : <http://esibpad.googlecode.com/svn/trunk/>

https://github.com/eia-fr/ESIB_PAD/¹. Les releases seront stockées à l'emplacement suivant : <http://esibpad.googlecode.com/svn/tags/> https://github.com/eia-fr/ESIB_PAD/tree/

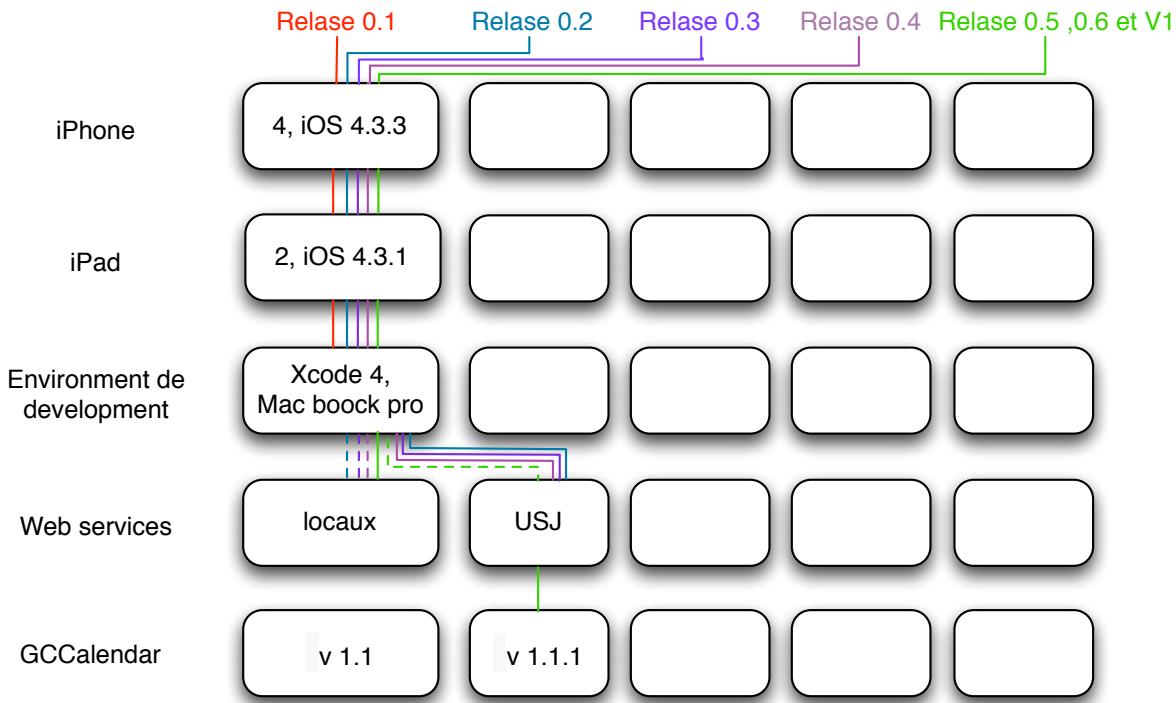


FIGURE 3.4: Compatibilité entre composants et versions du logiciel. Les traits continus représentent une compatibilité à 100 %, les traits traitillés représentent une compatibilité partielle.

GCCalendar : est un composant externe open source qui permet d'afficher un calendrier pour une journée.[5] La version v 1.1.1 est une adaptation de la version officielle faite par moi-même pour mieux répondre aux besoins du projet et qui permet l'affichage du composant dans une partie de l'écran pour l'iPAD et non uniquement en plein écran sur iPhone. Cette adaptation prend aussi en charge le changement de journée grâce au mouvement "glisser" du doigt sur l'écran. Cette version a été transmise à l'auteur pour ainsi faire évoluer ce composant.

7 Gestion de la documentation

La documentation sera conçue selon les différentes normes IEEE sur la documentation de software. Elle contiendra notamment les documents suivants :

1. Suite à des problèmes technique rencontré pour communiquer via SVN, le protocole Git est utilisé à partir de la release 2

1. SPMP - Software Project Management Plan (IEEE 1058). Ce document contient toutes les informations concernant l'organisation d'un projet de développement de software . Il a pour but de rendre transparente l'organisation du projet et aide les chefs de projets à avoir un aperçu global de l'état d'avancement.
2. SRS - Software Requirements Specification(IEEE 830). Ce document contient la documentation concernant la spécification et l'analyse.
3. SDD - Software Design Description(IEEE 1016). Ce document contient la documentation concernant la conception et l'implémentation.
4. STD - Software Test Documentation(IEEE 1008). Ce document contient la documentation concernant les tests effectués.

Il est important d'indiquer que la documentation ne sera pas complètement conforme à la norme, car cela représenterait une trop grande charge de travail. En effet, les différentes normes sont très complètes et plusieurs chapitres ne sont pas adaptés à notre projet. La variante de documentation qu'on utilise est inspirée de celle utilisé par la "Hochschule Luzern" (école d'ingénieurs Suisse alémanique dans laquelle j'ai eu l'occasion d'étudier durant 6 mois).

Les documents seront regroupés en chapitre pour former le rapport final du projet.

8 Gestion des finances

Ce travail fait partie du processus de formation et ne traite pas en détail de l'aspect financier. Cependant voici tout de même quelques informations que l'on peut citer :

8.1 Ressources humaines

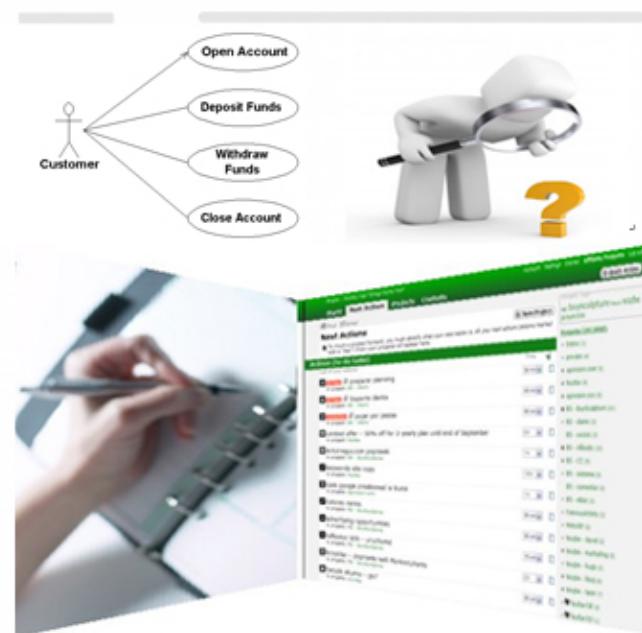
- 1 futur ingénieur HES à 100% soit 40 heures par semaine durant 12 semaines. Une bourse est versée par l'EIA-FR à l'étudiant pour le transport jusqu'au Liban ainsi que le logement sur place.

8.2 Ressources matérielles

- 1 iPad2 et 1 iPhone 4 mis à disposition par l'EIA-FR
- local de travail mis à disposition par l'ESIB

Chapitre 4

Software Requirement Specification



1 Introduction

Ce chapitre contient les spécifications détaillées des exigences du projet ainsi que l'analyse faite pour chaque tâche. Ces deux étapes sont essentielles pour partir dans la bonne direction dès le début de chaque itération. La section analyse décrit en général les cas d'utilisation que l'on identifie ainsi que la faisabilité de ces derniers. Le détail des cas d'utilisation et de leurs besoins graphique est documenté dans la section exigences fonctionnelles de ce chapitre.

2 Analyse

2.1 Aperçu global

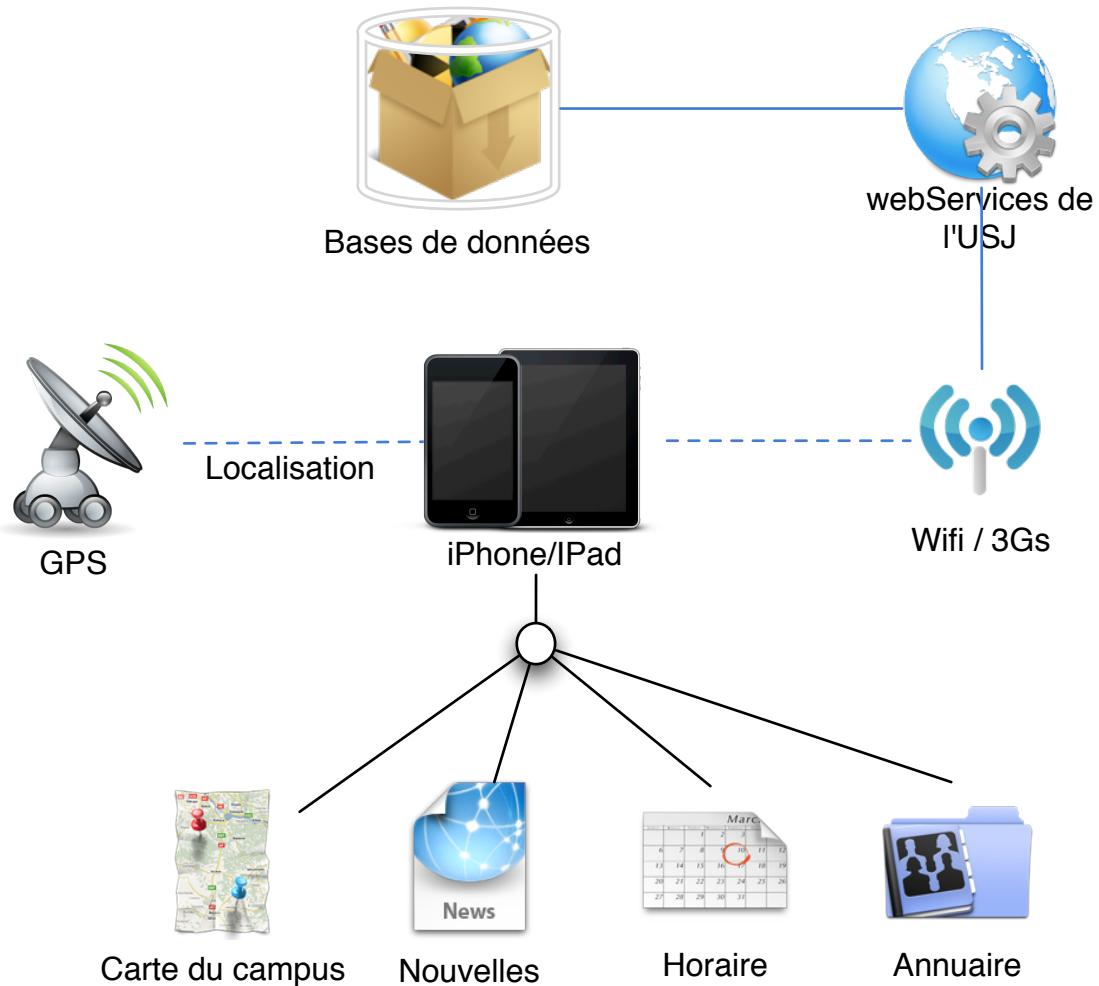


FIGURE 4.1: Vue globale des fonctionnalités du produit

2.2 Use case

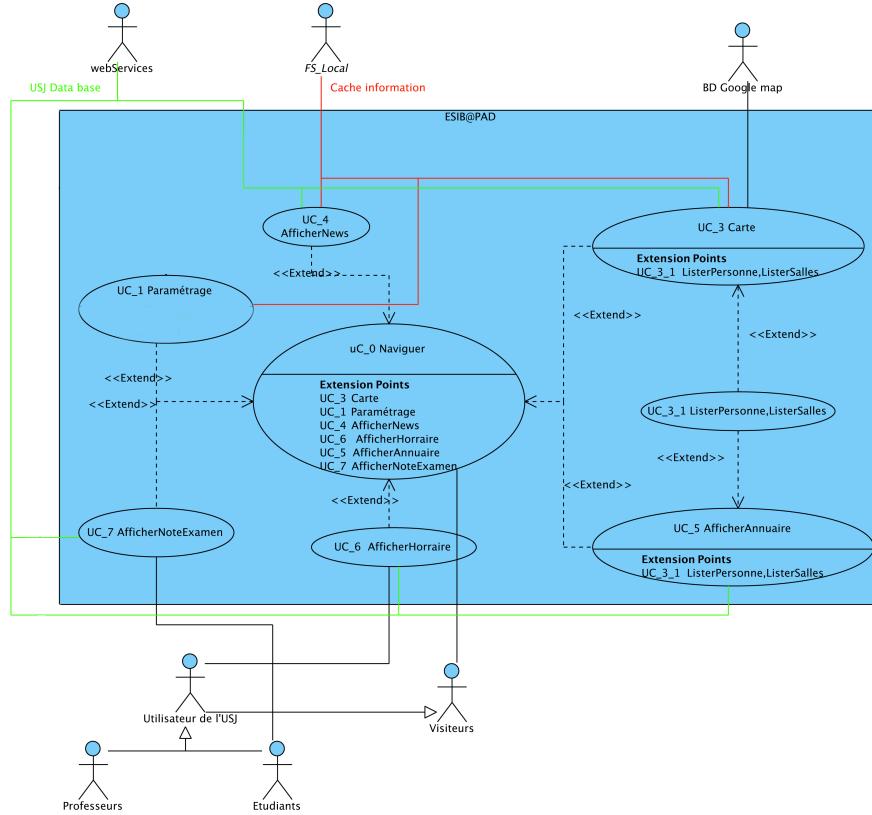


FIGURE 4.2: Diagramme des cas d'utilisation de l'application

Les fiches descriptives du chapitre 3.2 détaillent les cas d'utilisation.

2.3 Description des utilisateurs

Professeurs : représente les professeurs de l'USJ.

Étudiants : représente les étudiants de l'USJ.

Visiteurs : représente les visiteurs et les personnes non enregistrées de l'USJ.

WebServices : représente les web services de l'USJ qui nous permettent d'accéder aux informations de la base de données.

FS_Local : représente les données stockées en local sur l'appareil.

BD Google map : représente la base de données Google pour les images de la carte.

2.4 Faisabilité

UC_0 Naviguer L'iOS permet d'afficher différentes vues et naviguer d'une façon simple entre elles. Ce cas d'utilisation est **100 % faisable**.

UC_1 Paramétrage L'iOS permet de stocker des paramètres d'application d'une façon simple. Ce cas d'utilisation est **100 % faisable**.

UC_3 Carte La librairie MapKit livrée avec l'iOS permet d'afficher des cartes en se basant sur la base de données de google map. Il est aussi possible d'ajouter des annotations à des emplacement précis de la carte. Cette librairie est utilisable gratuitement et librement, elle répond à tous les besoins de notre application. Il est aussi possible, grâce au gps des appareils, de détecter la position de l'utilisateur. Ce cas d'utilisation est **100 % faisable**.

UC_4 AfficherNews Les outils à disposition permettent d'effectuer aisément ce genre de tâches . Pour le détail des news, il est possible d'intégrer un navigateur web dans l'application. Ce cas d'utilisation est **100 % faisable**.

UC_5 AfficherAnnuaire L'annuaire est disponible via les web services, le but ici est de le présenter d'une façon pratique à l'utilisateur. Ce cas d'utilisation est **100 % faisable**.

UC_6 AfficherHoraire L'annuaire n'est pas disponible via les web services, les données seront simulées à l'aide du web service local sur la machine du développeur. **80 % faisable**.

UC_7 AfficherNoteExamen Les notes d'examens ne sont pas disponibles via les web services, les données seront simulées à l'aide du web service local sur la machine du développeur. **80 % faisable**.

3 Spécification des exigences

3.1 Spécification des interfaces

Interfaces utilisateur Aucune ligne graphique n'est imposée, la seule contrainte est d'utiliser les logos originaux de l'école.

Interfaces Hardware L'iPhone et l'iPad possèdent un écran tactile qui sera utilisé pour interagir avec l'utilisateur. D'autres capteurs comme le gyroscope, caméra ou accéléromètre sont disponibles sur l'appareil mais ne seront pas utilisés pour ce projet.

L'iPhone et l'iPad se connectent à internet via le 3Gs et le WIFI pour récupérer les données des web services.

Interfaces Software L'application utilise les web services de l'USJ pour accéder aux bases de données. Les web services n'existent pas avant la création de l'application, il faut définir la manière de communiquer avec les web services. Pour ce faire, M.Medawar fournit un fichier XML ainsi que le XML Schema correspondant au résultat attendu d'un web ser-

vice et le service informatique de l'USJ fournira ce service. L'annexe E(/Documentation/Annexes/E) contient tout les fichiers XML et XML Schema fournis au service informatique.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2   <!-- Résultat de l'appel https://www.url.com/webService.php
3     avec les paramètres en POST suivant:
4       usr = 'elias.medawar'
5       pwd = '1234'
6       op   = 'testCommunication'
7 -->
8   <response xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
9     xsi:noNamespaceSchemaLocation="schemaTest.xsd">
10    <!-- Une réponse valide pour tester la communication -->
11    <status>0</status>
12    <usrGroup>2</usrGroup>
13    <commentaire><![CDATA[Communication possible]]></commentaire>
14  </response>

```

Listing 4.1: Exemple de code XML fournit au service informatique de l'USJ

```

<?xml version="1.0" encoding="UTF-8"?>
<xss:schema xmlns:xss="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xss:element name="response">
    <xss:complexType>
      <xss:sequence>
        <xss:element ref="status"/>
        <xss:element ref="usrGroup"/>
        <xss:element ref="commentaire" minOccurs="0" maxOccurs="10"/>
      </xss:sequence>
    </xss:complexType>
  </xss:element>
  <xss:element name="status">
    <xss:simpleType>
      <xss:restriction base="xs:int">
        <xss:enumeration value="-2"></xss:enumeration><!-- Internal error in the webservices
          .KO -->
        <xss:enumeration value="-1"></xss:enumeration><!-- Wrong password or login.KO -->
        <xss:enumeration value="0"></xss:enumeration><!-- Successful execution.OK -->
      </xss:restriction>
    </xss:simpleType>
  </xss:element>
  <xss:element name="usrGroup">
    <xss:simpleType>
      <xss:restriction base="xs:int">
        <xss:enumeration value="0"></xss:enumeration><!-- User login is not registered,
          assume that it's a visitor -->
        <xss:enumeration value="1"></xss:enumeration><!-- User login corresponds to a
          professor -->
        <xss:enumeration value="2"></xss:enumeration><!-- User login corresponds to a student
          -->
      </xss:restriction>
    </xss:simpleType>
  </xss:element>
  <xss:element name="commentaire" type="xs:string"/>
</xss:schema>

```

Listing 4.2: Exemple de XML Schema fournit au service informatique de l'USJ

Les fichiers XML ainsi que leur format n'ont pas été pris en compte lors du développement des webservices. Le service informatique a décidé d'utiliser son propre format. Donc les fichiers XML fournis ont simplement été ignorés et ne sont pas à exploiter.

Protocoles de communications : L'application communique officiellement via HTTPS avec les web services, mais pour l'instant les services web sont configurés pour communiquer via HTTP.

3.2 Exigences fonctionnelles

Pour définir correctement les exigences, la description des cas d'utilisation est effectuée. Des prototypes de l'interface graphique sont aussi réalisés. Ces prototypes sont utilisés comme documents de travail lors la prise de décision concernant l'emplacement des éléments et leurs fonctionnements. Ce concept et cette manière de travailler sont inspirés de diverses références bibliographiques à ce sujet[6]. Certains prototypes ne représentent pas toujours la version finale implémentée car la version finale est implémentée en fonction des possibilités du système et du temps à disposition .

Les prototypes ont été faits au départ en version informatique mais par la suite ils ont été fait à la main pour une question de gain de temps. Les prototypes faits à la main ont été remplacés dans la documentation par des captures d'écran qui correspondent.

Naviguer

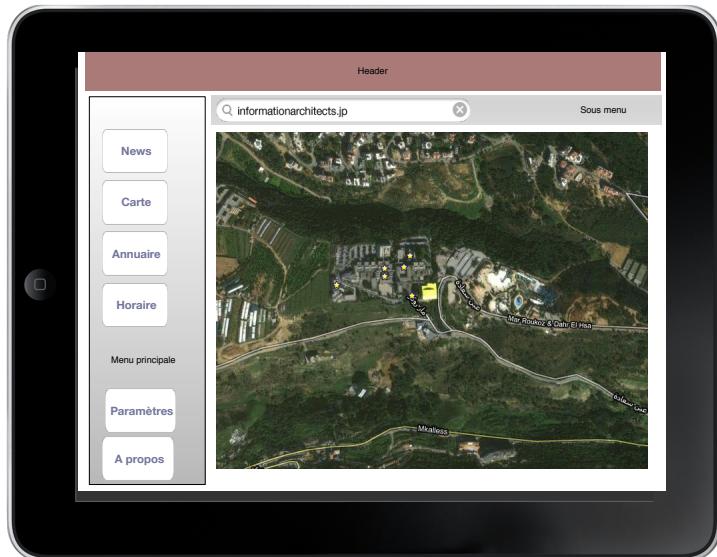
L'utilisateur doit pouvoir naviguer dans les différents menus de l'application.

Nom du Use Case	Naviguer
Ref	UC_0
Déclencheur	L'utilisateur démarre l'application
Précondition	
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche le menu de l'application. 2. L'utilisateur clique sur un élément du menu. 3. Le système affiche la vue correspondante au bouton cliqué . 4. L'utilisateur effectue la tâche dont il a besoin à l'aide de la vue affichée. 5. L'utilisateur revient sur la page du menu de l'application. 6. Recommencement au point 2 du UC.
Enchaînements alternatifs	Commence au point 4 du scénario nominal(sur IPad). <ol style="list-style-type: none"> 1. L'utilisateur clique sur un autre élément du menu. 2. Continue au point 3 du scénario nominal.
Status actuel	Planifié : <input checked="" type="checkbox"/> , Implémenté : <input checked="" type="checkbox"/> , Testé : <input checked="" type="checkbox"/> , Validé : <input checked="" type="checkbox"/>

Besoin graphique



(a) Navigation sur iPhone



(b) Navigation sur IPad

Paramétrer l'application

L'utilisateur doit pouvoir choisir les paramètres de l'application et les sauvegarder.

Nom du Use Case	Paramétrage
Ref	UC_1
Déclencheur	L'utilisateur clique sur le bouton paramètre de la navigation
Précondition	
Scénario nominal	<ol style="list-style-type: none"> Le système restaure les valeurs des paramètres depuis le fichier de configuration et les affiche. L'utilisateur choisit une option qu'il désire modifier. L'utilisateur modifie la valeur. L'utilisateur confirme qu'il a finit de modifier la valeur Le système sauvegarde la valeur.

Enchaînements alternatifs	
Status actuel	Planifié : <input checked="" type="checkbox"/> , Implémenté : <input checked="" type="checkbox"/> , Testé : <input checked="" type="checkbox"/> , Validé : <input checked="" type="checkbox"/>

Besoin graphique



FIGURE 4.3: Wireframe illustrant les modifications des paramètres sur l'iPhone



FIGURE 4.4: Wireframe illustrant les modifications des paramètres sur l'iPad

Visualiser la carte

L'utilisateur doit pouvoir visualiser la carte du campus avec les différentes informations utiles pour se retrouver dans le campus.

Nom du Use Case	Carte
Ref	UC_3
Déclencheur	L'utilisateur presse sur le bouton carte de l'application
Précondition	
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche la carte de tous les campus ainsi que la position actuelle de l'utilisateur. 2. Le système permet de choisir un campus pour en voir le détail. 3. Le système affiche les bâtiments principaux du campus. 4. Le système permet de naviguer, sélectionner des bâtiments ou une personne pour les afficher sur la carte.
Enchaînements alternatifs	
Status actuel	Planifié : <input checked="" type="checkbox"/> , Implémenté : <input checked="" type="checkbox"/> , Testé : <input checked="" type="checkbox"/> , Validé : <input checked="" type="checkbox"/>

Besoin graphique



FIGURE 4.5: Wireframe illustrant les fenêtres de l'affichage des cartes sur l'iPone



FIGURE 4.6: Wireframe illustrant les fenêtres de l'affichage des cartes sur l'iPad

Sur la Figure 4.6 on peut voir, que les menus des deux versions sont les mêmes mais sur l'iPad, au lieu d'ouvrir chaque élément du menu dans une nouvelle fenêtre, les éléments sont ajouté comme des "Pop-up" dans la page principale.

Afficher les nouvelles

L'utilisateur doit pouvoir visualiser les nouvelles du campus .

Nom du Use Case	AfficherNews
Ref	UC_4
Déclencheur	L'utilisateur presse sur le bouton news de l'application
Précondition	
Scénario nominal	<ol style="list-style-type: none"> Le système affiche les news du campus. L'utilisateur peut cliquer sur une news pour voir le détail de cette dernière. Depuis le détail de la news, l'utilisateur peut, à l'aide d'un bouton retour, revenir à l'aperçu de l'ensemble des news.
Enchaînements alternatifs	
Status actuel	Planifié : <input checked="" type="checkbox"/> , Implémenté : <input checked="" type="checkbox"/> , Testé : <input checked="" type="checkbox"/> , Validé : <input checked="" type="checkbox"/>

Besoin graphique



FIGURE 4.7: Wireframe illustrant les fenêtres de l'affichage des news sur l'iPone

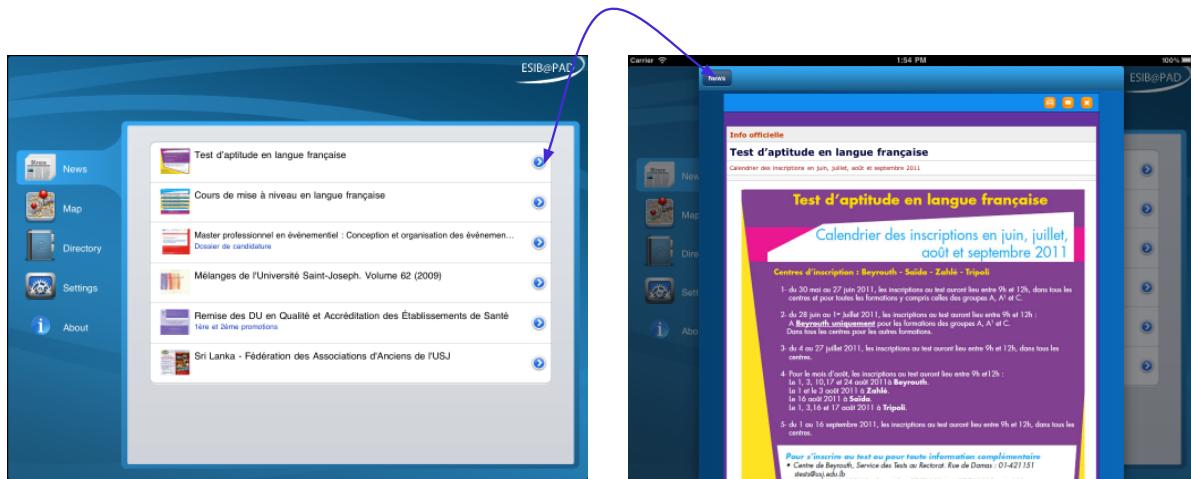


FIGURE 4.8: Wireframe illustrant les fenêtres de l'affichage des news sur l'iPad

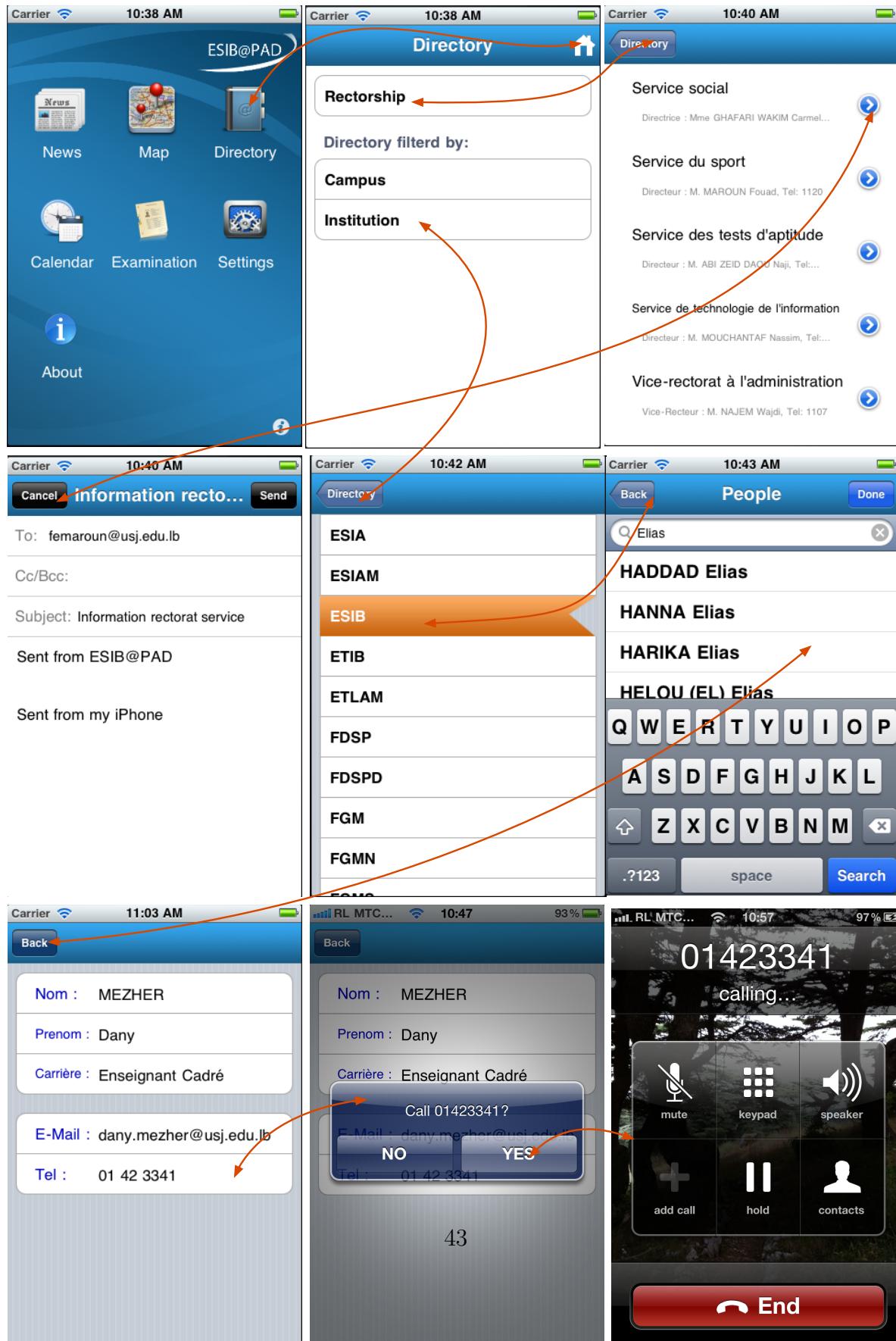
Afficher l'annuaire

L'utilisateur doit pouvoir visualiser l'annuaire de l'USJ.

Nom du Use Case	AfficherAnnuaire
Ref	UC_5
Déclencheur	L'utilisateur presse sur le bouton annuaire de l'application
Précondition	

Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche la liste des possibilités de regroupement : <ol style="list-style-type: none"> (a) Par campus (b) Par institution (c) Services 2. L'utilisateur choisit un regroupement qu'il veut 3. L'utilisateur choisit le sous-groupe désiré. 4. Le système cherche les données en cache si elles s'y trouvent sinon depuis les services web. 5. Le système affiche la liste des personnes trouvées. 6. L'utilisateur presse sur le bouton pour obtenir plus de détail d'une personne. 7. Le système affiche le détail de la personne. 8. Le système permet le démarrage d'appel sur un clique sur le numéro de téléphone ou l'envoi d'un e-mail suite à un clique sur l'adresse mail.
Enchaînements alternatifs	
Status actuel	Planifié : <input type="checkbox"/> , Implémenté : <input type="checkbox"/> , Testé : <input type="checkbox"/> , Validé : <input checked="" type="checkbox"/>

Besoin graphique



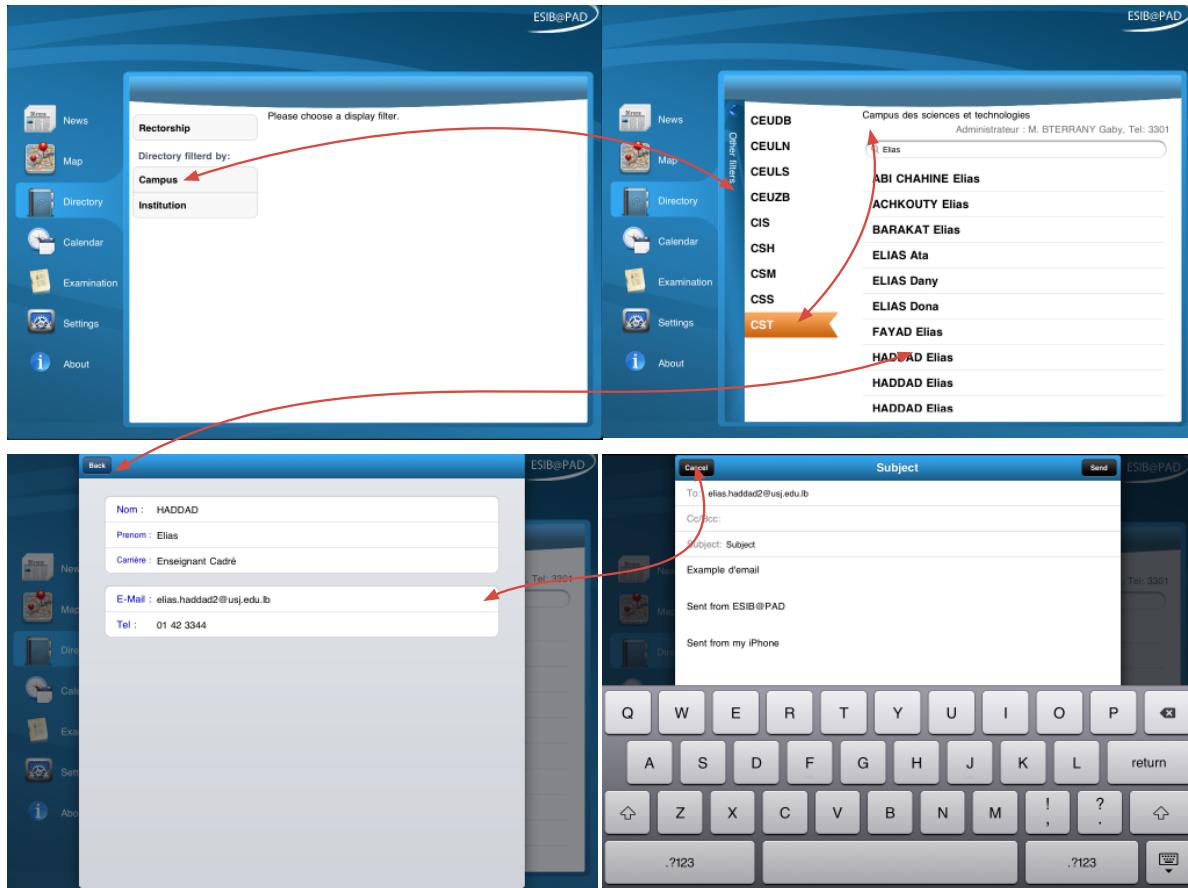


FIGURE 4.10: Wireframe illustrant les fenêtres de l'affichage de l'annuaire sur l'iPad

Afficher l'horaire

Les utilisateurs de l'USJ doivent pouvoir visualiser leurs horaires.

Nom du Use Case	AfficherHoraire
Ref	UC_6
Déclencheur	L'utilisateur presse sur le bouton horaire de l'application
Précondition	
Scénario nominal	<ol style="list-style-type: none"> Le système affiche un calendrier contenant le jour courant. L'utilisateur peut naviguer facilement au jours suivants et précédents. L'utilisateur peut, en cliquant sur un cours, afficher son emplacement.

Enchaînements alternatifs	
Status actuel	Planifié : <input type="checkbox"/> , Implémenté : <input type="checkbox"/> , Testé : <input type="checkbox"/> , Validé : <input type="checkbox"/>

Besoin graphique

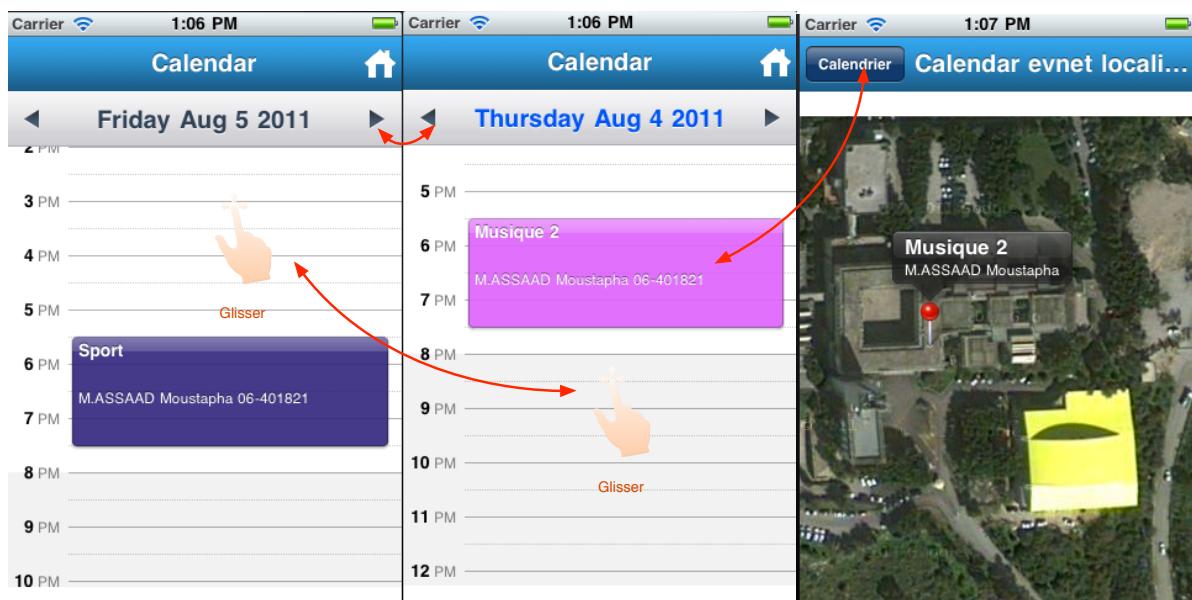


FIGURE 4.11: Wireframe illustrant les fenêtres de l'horaire sur l'iPone

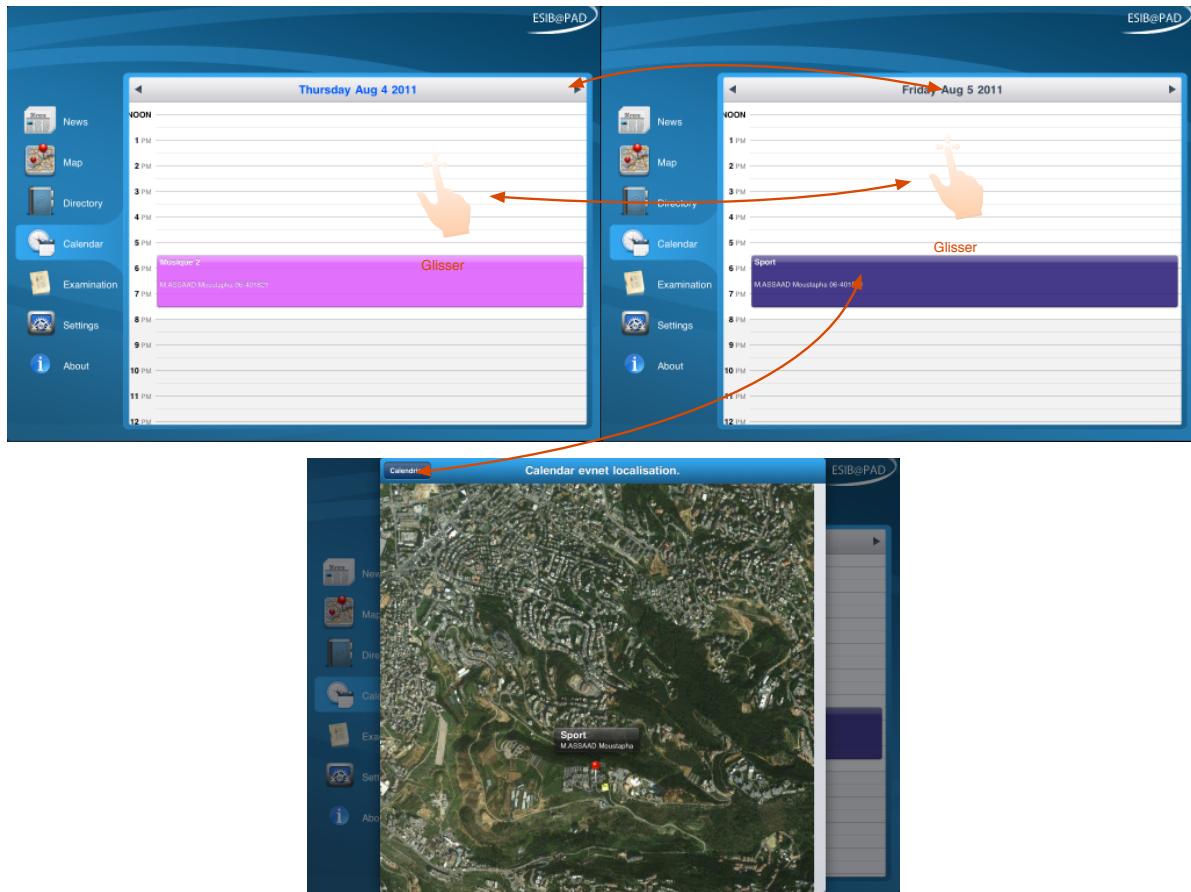


FIGURE 4.12: Wireframe illustrant les fenêtres de l'horaire sur l'iPad

Afficher le résultat des examens

Les étudiants de l'USJ doivent pouvoir visualiser leurs résultats d'examen.

Nom du Use Case	AfficherNoteExamen
Ref	UC_7
Déclencheur	L'utilisateur presse sur le bouton résultat d'examen de l'application
Précondition	
Scénario nominal	<ol style="list-style-type: none"> Le système affiche la liste des examens et la note obtenue.

Enchaînements alternatifs	Commence au point 0 du scénario nominal quand l'utilisateur n'est pas un étudiant. 1. Le système affiche un message d'erreur. 2. Le système redirige l'utilisateur vers la fenêtre de paramétrage de l'application.
Status actuel	Planifié : <input checked="" type="checkbox"/> , Implémenté : <input checked="" type="checkbox"/> , Testé : <input checked="" type="checkbox"/> , Validé : <input checked="" type="checkbox"/>

Besoin graphique



FIGURE 4.13: Wireframe illustrant les fenêtres des résultats d'examen sur l'iPone

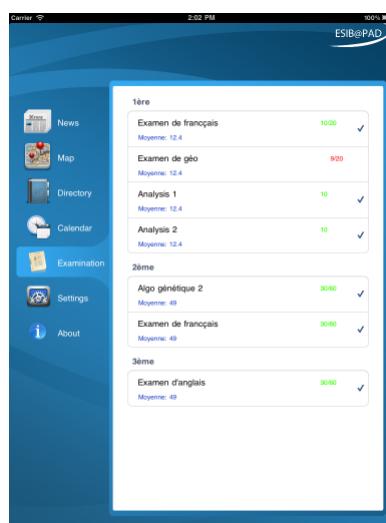
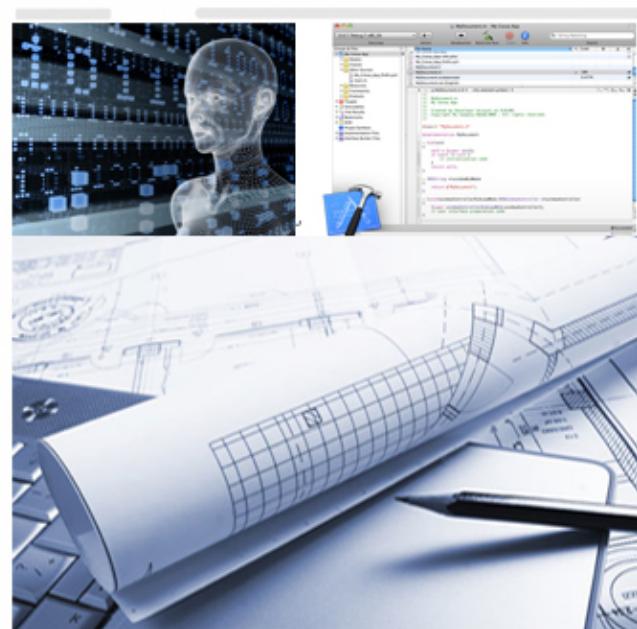


FIGURE 4.14: Wireframe illustrant les fenêtres des résultats d'examen sur l'iPad

Chapitre 5

Software Design Description



1 Introduction

1.1 But du chapitre

Ce chapitre décrit les variantes d'architectures étudiées pour le projet ESIP@PAD, l'architecture finale choisie ainsi que le détail du design final du projet. A l'aide de ce document, il est possible de comprendre le fonctionnement technique de l'ensemble du projet.

1.2 Aperçu du chapitre

La section 2 contient l'architecture de notre système ainsi que les différentes alternatives possibles.

La section 3 contient la description détaillée de l'implémentation de chaque composant du système ainsi que le détail concernant les algorithmes et techniques utilisés pour réaliser les parties majeures du système.

2 Architecture du système

2.1 Architecture choisie

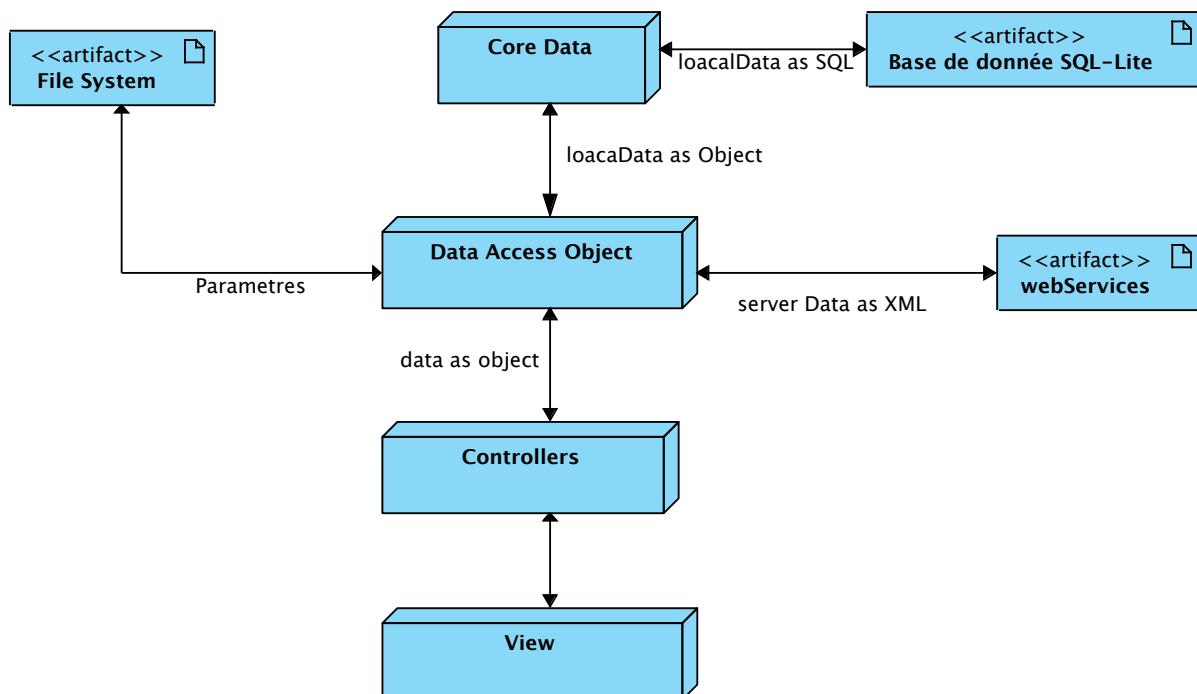


FIGURE 5.1: Vue global de l'architecture du système

Ce diagramme(Figure 6.1) nous donne un aperçu des différentes couches qui forment l'architecture du système.

View : Cette couche du système sert à représenter graphiquement l'information. Elle est étroitement liée à la couche Controllers.

Controllers : Cette couche du système s'occupe de charger les données dans la vue. Et de réagir correctement aux événements reçus de l'utilisateur.

Data Access Object : Cette couche fait référence au pattern de DAO [2] qui nous permet d'accéder aux données de notre application sans se soucier d'où elles proviennent, où elles seront stockées, ni comment. Elle contient aussi la logique métier de l'application. A l'aide de cette couche, il nous est facile de changer le support de stockage des données car toute la logique et l'accès aux données y sont centralisées.

Core Data est le framework de persistance de l'iOS qui permet d'accéder d'une manière simplifiée aux données stockées dans la base de données SQL-Lite ou sous format XML. Ce framework nous permet de décrire la base de données et ses relations et de générer des objets Objective-c qui correspondent aux entités de la base de données. Les principes du fonctionnement sont les mêmes que ceux du fameux framework JPA[3] de java.

2.2 Discussion des alternatives d'architectures

Différentes alternatives d'architectures se sont offertes à nous en début du projet et voici les principales.

Alternative 1 : Sans base de données

Cette alternative supprime la couche core data et la remplace par un stockage des données xml reçus des web-services sur le support de données de l'appareil. Cette alternative requiert moins de temps de développement mais en contrepartie, il faut parser les fichiers xml à chaque utilisation des données. L'iOS n'est pas encore optimisée pour le traitement des fichiers xml et permet uniquement de parser un fichier mais sans faire des requêtes du type XPath sur les fichiers xml. Des frameworks ont été développés par diverses entreprises pour permettre le requêtage de fichiers XML, mais leurs performances restent tout de même moins bonnes que celles d'une base de données.

Alternative 2 : Objet pour la communication en C++

Cette alternative propose de décrire les objets pour la communication(Core data \Leftrightarrow DAO \Leftrightarrow Controllers) en C++. Avec cette alternative, on augmente la portabilité de notre application et on offre la possibilité de réutiliser ces mêmes objets sur d'autres plateformes (telle que Android). Après une brève recherche, il s'est avéré que Core data n'est pas

capable d'utiliser les objet C++. Comme nous utilisons une base de données SQL-Lite et que Core data génère automatiquement les objets correspondants au contenu de la base de données nous avons mis de côté cette alternative. Par contre la base de données étant décrite en SQL-Lite, elle peut être exportée vers d'autres appareils et à partir de cette base de données il est facile de générer les objets de communication à l'aide des outils propres à chaque plateforme.

2.3 Composants du système

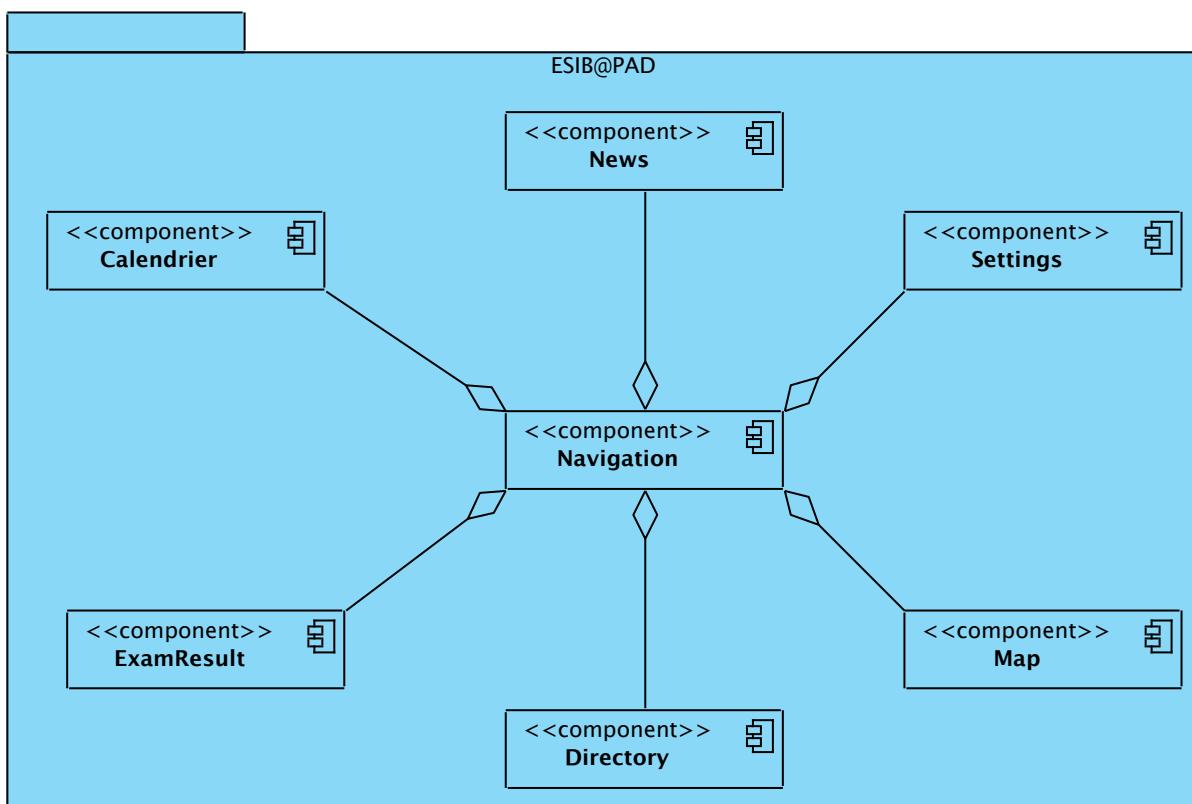


FIGURE 5.2: Diagrammes de composant du système

L'application est découpée en composants pour ainsi permettre de bien séparer les tâches que l'on est en train d'effectuer, faciliter la réutilisation de parties de l'application et rendre les tests plus efficaces vu que l'on se concentre sur une partie et non pas un tout.

Les composants n'ont pas été développés complètement dans les règles de l'art vu qu'on n'a pas pour chaque composant un fichier compiler qui permet sa réutilisation ainsi qu'une interface pour y accéder (système de black-box). Cette entrave à la règle est dûe au manque d'intérêt pour l'obtention de composants si sophistiqués vu qu'il ne seront pas réutilisés en dehors de notre application et pour des raisons d'économie de temps. Mais cependant

le code est organisé et conçu de manière à faciliter sa réutilisation dans un autre cadre et à pouvoir être transformé en composant complet(black-box) si besoin.

Description des composants

Voici une brève description des composants, le détail concernant leurs implémentations se trouve dans la section suivante.

Navigation :

Ce composant se charge de présenter un menu avec des boutons pour accéder aux composants de l'application. Lors d'un clic sur un des ces boutons, il est présenté à l'endroit approprié. Chaque composant appelé fera appel au composant "Navigation" pour être déchargé de la vue principale.

Map :

Ce composant affiche une carte avec des indications sur les divers lieux de l'université. Il permet aussi de chercher l'emplacement d'une personnes ou d'une salle.

Settings :

Ce composant permet de configurer les différents paramètres de l'application.

News :

Ce composant permet d'afficher les news de l'université. Le détail des news est aussi affichable sous forme de page web intégrée dans l'application et contenant le détail tel qu'il se trouve sur le site internet de l'USJ.

Calendrier :

Ce composant permet d'afficher l'emploi du temps de chaque membre de l'université.

ExamResult :

Ce composant permet aux étudiants d'afficher les résultats des examens.

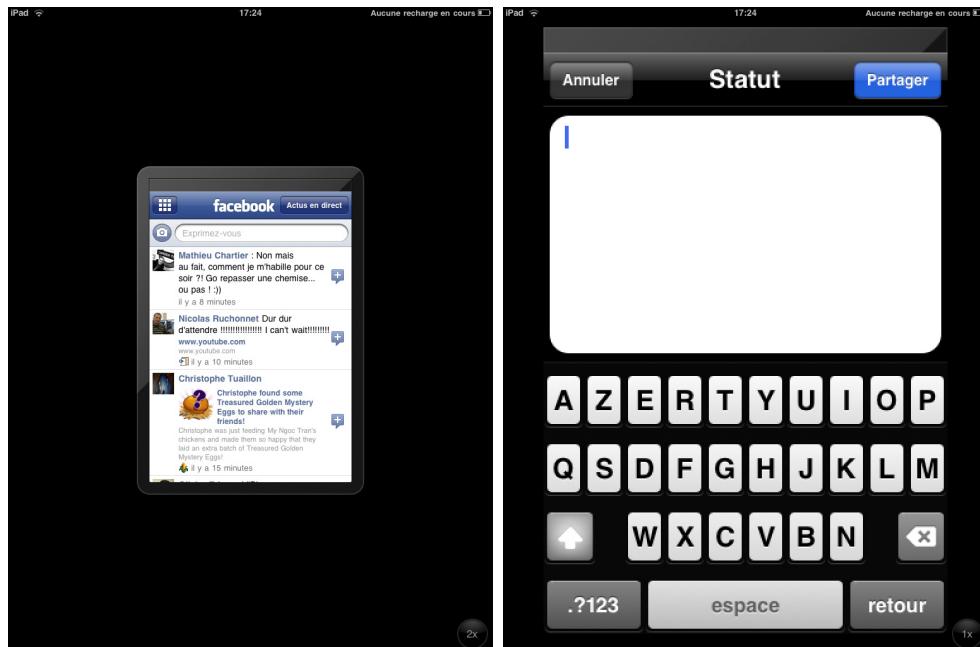
Directory

Ce composant permet d'afficher l'annuaire de l'université. Il offre la possibilité d'envoyer des e-mails ou de lancer des appels à partir de l'application.

3 Conception et Implémentation des composants

3.1 Compatibilité graphique iPhone,iPad

Dès le lancement de l'iPad, Apple y a intégré un simulateur d'iPhone qui permet à toute application iPhone de s'exécuter sur l'iPad.



(a) Aperçu de l'application FaceBook version iPhone exécutée sur iPad (zoom 1x) (b) Aperçu de l'édition d'un texte d'une application iPhone sur l'iPad (zoom 2x)

Cette façon de faire peut être considérée comme une compatibilité, mais elle est minime vu qu'on étire simplement l'application iPhone pour la rendre plus grande mais on n'exploite nullement les capacités de l'écran de l'iPad.

A la création d'un projet à l'aide de XCode, ce dernier nous demande si l'on veut créer une application iPhone, iPad ou universel. En choisissant universel, on s'attend à pouvoir faire une application pour un appareil et qu'elle soit compatible avec les deux. Mais l'illusion de cette compatibilité apparaît assez vite. En effet pour la partie graphique il y a deux dossiers, un premier nommé iPhone et l'autre iPad. Et là on comprend que la partie graphique doit être faite en grande partie à double. Ceci ne signifie pas que tout doit

être fait à double, la plupart des composants ont leurs équivalents sur les deux appareils. Les deux composants uniquement disponibles sur iPad sont le composant Split views et Popovers. Dans une publication concernant[1] ce sujet Apple dit :

Conditional Coding

In order to achieve your design goals for a Universal application, you will need to use conditional coding to determine the availability of features when your app is running. Conditional coding allows you to make sure you're loading the right resources, using functionality that's supported by the device and properly leveraging hardware that's available.

Et en effet tout au long du développement et pour les éléments que l'on désire utiliser sur les deux appareils sans duplication du code, il a fallu tester pour savoir quel appareil exécute l'application et redimensionner les vues pour qu'elles soient à la bonne taille.

Cette incompatibilité peut s'expliquer par la différence de taille des écrans, iPad : 241.2 mm x 185.7 mm et iPhone : 115.2 mm x 58.6 mm. Et de plus l'information ne doit pas être organisée de la même façon sur les 2 appareils ; sur iPad on pourra facilement présenter un plus grand nombres d'informations sur un seul écran. Tandis que sur iPhone on doit essayer de minimiser l'information à afficher pour garantir qu'elle reste lisible.

3.2 Système de cache

Un des points importants pour notre application est la mise en cache des données. Cette importance est dûe à la vitesse de la connexion internet au Liban et de manière générale sur tous les appareils mobiles. Ce manque de capacité des appareils mobiles augmente le temps nécessaire pour accéder aux données. De plus, même de nos jours, il n'est pas possible d'accéder à Internet de partout.

Pour le stockage en cache, un mécanisme de sauvegarde des dates d'exécution des requêtes a été mis en place. Pour chaque requête que l'on fait au serveur, 3 variables sont envoyées : le nom et password de l'utilisateur ainsi que le contenu que l'on désire obtenir. Dès l'exécution d'une requête, la date de son exécution est enregistrée, à la prochaine exécution si le temps écoulé entre la dernière exécution et aujourd'hui est supérieur à X(actuellement défini à 30 jours¹) on exécute à nouveau la requête, sinon on récupère les valeurs du cache.

1. Valeurs modifiable dans le fichier Settings.plist du code source

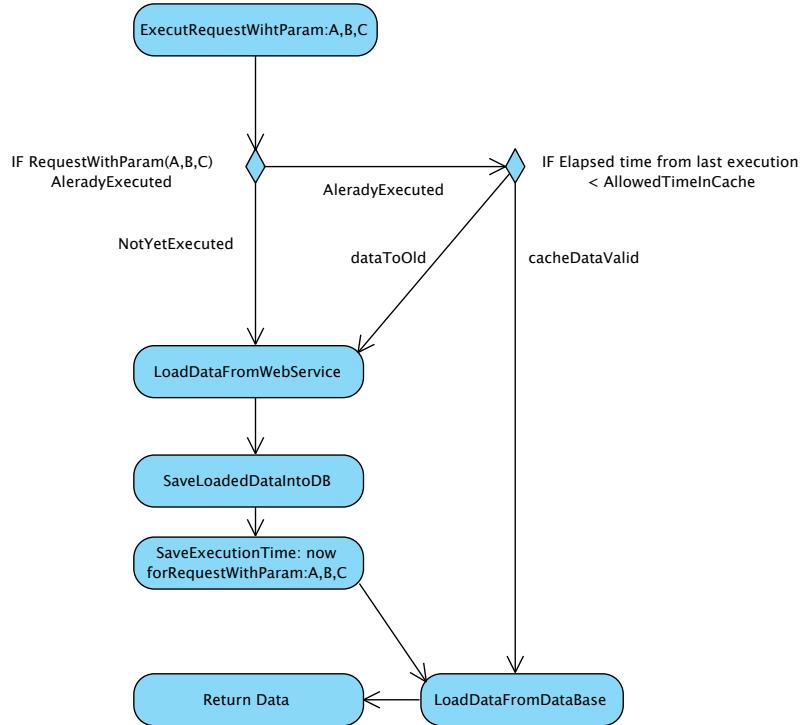


FIGURE 5.3: Illustration du fonctionnement du système de cache.

Pour plus de détails à ce sujet, voir le code source de la méthode `areDataUpToDate` de la classe `GenericDAO` ainsi que `setLastUpdateTimeForKey` et `getLastUpdateTimeForKey` de la classe `SettingsDAO`.

3.3 Accès aux données des services web

Le service informatique de l'université a mis en place un service web qui permet l'accès aux données via le protocole HTTP et il retourne essentiellement le contenu de la base de données sous format XML. Le service web se trouve actuellement à l'adresse : `http://www.usj.edu.lb/web-services/web-service.php`. Le paramétrage des services web se fait à l'aide de la méthode Post du protocol HTTP et notamment en transmettant les paramètres suivants :

1. **usr** le nom de l'utilisateur. guest est le nom par défaut et est valable pour les utilisateurs n'ayant pas de compte dans la base de données de l'USJ. Les autres utilisateurs doivent saisir leur id habituel pour les logins dans l'école.
2. **pwd** le mot de passe de l'utilisateur. guest est le mot de passe pour les invités
3. **op** est le nom de l'opération que l'on veut exécuter. Voici la liste des opérations possibles :

Nom de l'opération	Description	paramètres
listeServRec	Renvoie la liste des services de l'USJ.	aucun
listeCampus	Renvoie la liste des campus de l'USJ.	aucun
listeInst	Renvoie la liste des institutions de l'USJ.	param0 = Code campus (optionnel)
listeEmpNom	Recherche la liste des employés de l'USJ d'après le prénom et/ou le nom.	param0 = Nom(optionnel) param1= Prénom(optionnel). Au moins un paramètre est obligatoire
listeEmpInst	Renvoie la liste des employés d'une institution de l'USJ.	param0 = Code Institution(obligatoire)
listeEmpCampus	Renvoie la liste des employés d'un campus de l'USJ.	param0 = Code campus(obligatoire)
listeBatiments	Renvoie la liste des bâtiments d'un campus de l'USJ.	param0 = Code campus(obligatoire)
listeActualites	Renvoie les news de l'USJ.	aucun
listeHoraires	Renvoie l'horaire d'une personne selon les données de login.	Fonctionnalité uniquement implémentée sur les webServices en local.
listeNotes	Renvoie les résultats d'examen d'une personne selon les données de login.	Fonctionnalité uniquement implémentée sur les webServices en local.

TABLE 5.1: Liste des opérations possibles via les services web

Une page internet <http://www.usj.edu.lb/web-services/send.php> permet de saisir les paramètres et de les exécuter pour tester les services web.

Pour plus d'informations, concernant les web services, contacter M.Pascal TUFENKJ , Tel : +961 1 421 132 , Email : ptufenkji@usj.edu.lb

Services web locaux

Afin de minimiser la dépendance vis-à-vis des services web de l'USJ et de leur état d'avancement, des services web locaux ont été créés. Leur fonctionnement est très simple. Il s'agit

d'une page PHP hébergée sur la machine du développeur qui renvoie le fichier xml correspondant au nom de l'opération op.

Exemple : on envoie une requête http avec les paramètres en poste suivant : usr=Elias, pwd=1234,op=listeNotes. Le fichier xml listeNotes.xml qui est stocké sur la machine du développeur est retourné à l'utilisateur.

Pour plus d'informations, concernant les web services locaux voir le code source de la page webServices.php .

Téléchargement de fichier XML en Objective-c

Le téléchargement de données de grande taille se fait de manière asynchrone à l'aide de la classe NSURLConnection.

```
-(void)loadDataFromInternet{

    // Initialise the request with the server url.
    NSMutableURLRequest *request = [NSMutableURLRequest requestWithURL:[NSURL
    URLWithString:@“http://serveurAdress.com/webServices.php”] cachePolicy:
    NSURLRequestReloadIgnoringCacheData timeoutInterval:120.0];

    [request setHTTPMethod:@“POST”]; // Define the method of the request to Post

    NSString * postParam = [NSString stringWithFormat:@“usr=%@&pwd=%@&op=%@&param0=
    CST”, set.login,set.password,@“listEmpCampus”];
    // Post parm for getting the list of person of the CST campus

    [request setHTTPBody:[webServicePostHeader dataUsingEncoding:NSUTF8StringEncoding]];
    // The delegate is self and self implement the
    NSURLConnection *connection = [[[NSURLConnection alloc] initWithRequest:request delegate:
    self] autorelease];
    if (connection) {
        receivedData = [[NSMutableData data] retain];
    }
}

// this method is called for receiving every 256 byte of data
-(void)connection:(NSURLConnection *)connection didReceiveData:(NSData *)data {
    [receivedData appendData:data];
}
// this method will be called at the end of the connection (all data received)
-(void)connectionDidFinishLoading:(NSURLConnection *)connection{
    // Parse received XML file.
}
```

Listing 5.1: Téléchargement d'un fichier XML depuis internet de manière asynchrone et en transmettant les paramètres de la requête par POST .

Exploiter des XML en Objective-c

L'exploitation de fichiers XML se fait à l'aide du parser de l'iOS qui nous donne l'accès à l'information par événement (SAX).

```

// receivedData is an NSDATA object with row data of an XML file.
NSXMLParser *parseur=[[NSXMLParser alloc] initWithData:receivedData];
// Self is realising the NSXMLParserDelegate protocol
[parseur setDelegate: self];
// For the example the xml file is :
// <root>
//   <row>
//     <variable1>the value of variable 1 </variable1>
//     <variable2>the value of variable 2 </variable2>
//   </row>
//</root>
if ([parseur parse] == NO){
    //Parsing error, mange it hier.
}
[parseur release];

//This method is called each time that an XML balise is opened
- (void)parser:(NSXMLParser *)parser didStartElement:(NSString *)elementName namespaceURI:(
    NSString *)namespaceURI qualifiedName:(NSString *)qName attributes:(NSDictionary *)
    attributeDict{
    if ([elementName isEqualToString:@"row"]){
        // Create new object for getting data
        crntObject = [[NSObject alloc] init]
    }
    self .crntElementName =elementName;
}
//This method is called each time that a string ( balise content ) is found
- (void)parser:(NSXMLParser *)parser foundCharacters:(NSString *)string{

    NSString *newString = [[NSString alloc] initWithFormat:@"%@%@", crntCharacters,string];
    self .crntCharacters = newString;
    [crntObject setValue:newString forKey:self .crntElementName ];
    // We set the string value to the variable with the name of the current element
}
//This method is called each time that an XML balise is closed
-(void)parser:(NSXMLParser *)parser didEndElement:(NSString *)elementName namespaceURI:(
    NSString *)namespaceURI qualifiedName:(NSString *)qName{
    if ([elementName isEqualToString:@"row"]){
        // Close of the root element
        if (![ crntObject save]){
            NSLog(@"Whoops, couldn't save");
        }
    }
}

```

Listing 5.2: Parsing d'un fichier XML en Objective-c .

3.4 Base de données

Pour stocker les données reçues depuis les services web, une base de données SQL-Lite est utilisée. La base de données est complètement gérée par Core Data. La création des entités se fait dans XCode en éditant le fichier ESIB_PAD_SOURCES.xcdatamodeld

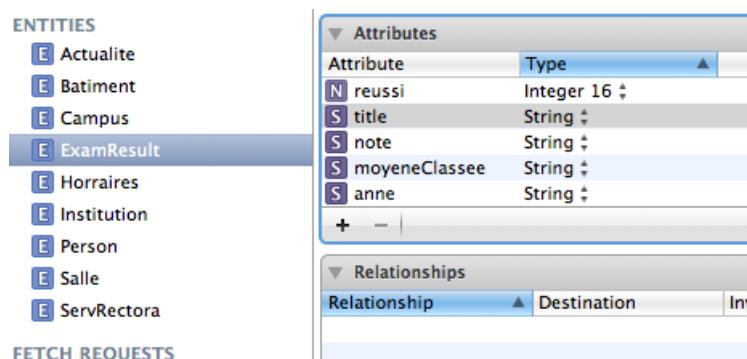


FIGURE 5.4: Interface dans X-Code permettant l'édition de la base de données

Une fois les entités créées et éditées, il est possible de générer les objets correspondants aux entités à l'aide de la manipulation suivante : Sélectionner les entités ⇒ Dans le menu : Editor ⇒ choisir Create NSMangedObject Subclass...

Une fois les objets correspondants aux entités générés, il est possible de les manipuler directement depuis le code. Dans notre application, la manipulation des données se fait dans la couche DAO du système.

```
//Getting the managedObjectContext
// It's important at the creation of the project to choose application with support of Core Data to
// get access to the managedObjectContext
NSManagedObjectContext * managedObjectContext = [(ESIB\_PAD\_SOURCESAppDelegate *)[
    UIApplication sharedApplication] delegate].managedObjectContext;

// Example: creating a new Campus in the DB.
NSManagedObject newObject = [NSManagedObject alloc];
newObject = [NSEntityDescription insertNewObjectForEntityForName:@"Campus"
    inManagedObjectContext:managedObjectContext];
[newObject setValue:@35.000 forKey:@"latitude"];
[newObject setValue:@38.000 forKey:@"longitude"];
[newObject setValue:@"CST" forKey:@"code"];
[newObject setValue:@"Rue de la paix 12" forKey:@"adresse"];
// ...
// ...
[managedObjectContext save:&error];// Persisting the new data.

//Example: Reading list of person from campus CST
// The sql equivalent: SELECT * FROM Person where campus ='CST'
```

```

NSFetchRequest * crntRequest = [[NSFetchRequest alloc] init];
NSEntityDescription *entity = [NSEntityDescription entityForName:self.entityDescription
    inManagedObjectContext:@'Person'];
[crntRequest setPredicate:[NSPredicate predicateWithFormat:@'campus = CST' ]];// Filtering
[crntRequest setEntity:entity ];
NSError *error;
NSArray *items = [managedObjectContext executeFetchRequest:crntRequest error:&error];

for (Person *p in items) {
    NSLog('User name is: %@", p.name);
}
[crntRequest release];

//Example: Deleting the campus CST from the DB
// The sql equivalent: Delete * from campus where code ='CST';

NSFetchRequest * crntRequest = [[NSFetchRequest alloc] init];
NSEntityDescription *entity = [NSEntityDescription entityForName:self.entityDescription
    inManagedObjectContext:@'Campus'];
[crntRequest setPredicate:[NSPredicate predicateWithFormat:@'code = CST' ]];// Filtering
[crntRequest setEntity:entity ];
NSError *error;
NSArray *items = [self.managedObjectContext executeFetchRequest:crntRequest error:&error];

for (Person *p in items) {
    [managedObjectContext deleteObject:managedObject];
}
[managedObjectContext save:&error];// Persisting the changement
[crntRequest release];
    
```

Listing 5.3: Exemple de création de lecture et de suppression de données dans la base à l'aide de Core Data .

3.5 Navigation

Diagramme de séquence

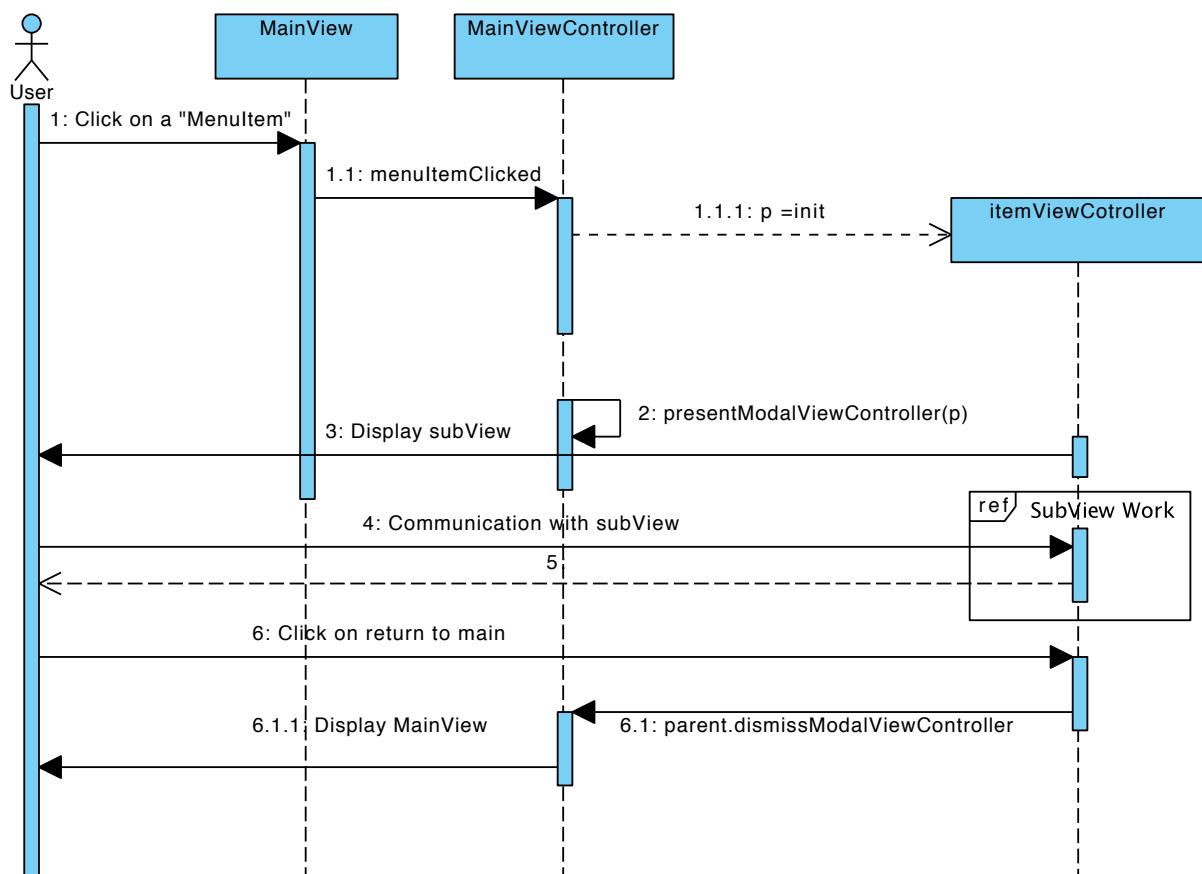


FIGURE 5.5: Diagramme de séquence du principe de la navigation

Le diagramme de séquence est valable pour les deux appareils. La seule différence est que sur l'IPad la vue chargée ne cachera pas l'écran entier mais rien qu'une partie de l'écran.

Diagramme de classe

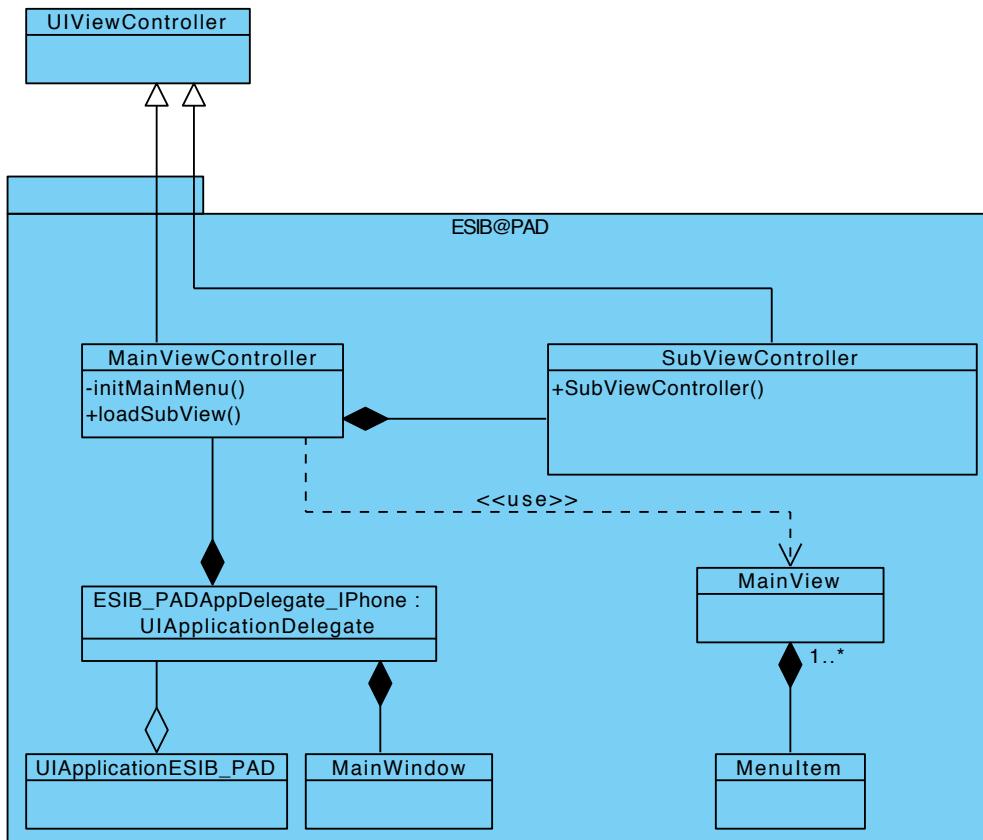


FIGURE 5.6: Diagramme de classe du composant MainView

Discussion

Pour faciliter l'ajout ou la suppression d'éléments dans le menu, ce dernier est créé automatiquement à partir d'un fichier plist(Fichier xml de configuration pour l'iOS).

HeightElement	Number	120
HeightElementIPad	Number	70
▼ ItemLogo	Diction...	(7 items)
About	String	Information.png
Calendar	String	plan.png
Directory	String	Directory.png
Examination	String	exam.png
Map	String	Maps.png
News	String	news.png
Settings	String	SystemPreferences.png
▼ ListItem	Array	(7 items)
Item 0	String	News
Item 1	String	Map
Item 2	String	Directory
Item 3	String	Calendar
Item 4	String	Examination
Item 5	String	Settings
Item 6	String	About
WidthElement	Number	100

FIGURE 5.7: Contenu du fichier MenuItemsParam.plist de configuration du menu pour la navigation

L'accès en lecture et écriture aux données des fichier plist se fait très facilement comme ceci :

```
// Writing value in Plist file
-(void)setValueForKey:(NSString *) theKey value:(NSString *) value {
    NSArray *paths = NSSearchPathForDirectoriesInDomains( NSDocumentDirectory,
                                                          NSUserDomainMask, YES);
    NSString *path = [[paths objectAtIndex: 0] stringByAppendingPathComponent: @"PlistFile.plist"];
    NSMutableDictionary * plistDict = [[NSMutableDictionary alloc] initWithContentsOfFile:path];
    if (!plistDict){
        plistDict = [[NSMutableDictionary alloc] init];
    }
    [ plistDict setValue:value forKey:theKey];
    [ plistDict writeToFile:path atomically: YES];
    [ plistDict release ];
}

// Reading value form Plist file
-(NSString *)getValueForKey:(NSString *) theKey {
    NSString *docsDir = [NSSearchPathForDirectoriesInDomains(NSDocumentDirectory,
                                                          NSUserDomainMask, YES) objectAtIndex:0];
    NSString *path = [docsDir stringByAppendingPathComponent: @"PlistFile.plist"];
    NSMutableDictionary* plistDict = [[NSMutableDictionary alloc] initWithContentsOfFile:path];
    NSString * d = [plistDict valueForKey:theKey];
    [plistDict autorelease];
    return d;
}
```

Listing 5.4: Code d'écriture et de lecture dans un fichier plist.

Suite à des problèmes d'affichage rencontrés lors de la rotation des appareils, il a été décidé de centraliser la gestion de rotation des appareils dans cette partie de l'application. Pour ce faire, on va s'enregistrer pour recevoir les notifications de rotation et après chaque rotation, on va redessiner l'interface en fonction de l'orientation. Une fois l'orientation de l'appareil détectée, on va forcer le système à redessiner la vue comme on le désire.

```
//Registering for notification
[[NSNotificationCenter defaultCenter] addObserver:self selector:@selector(didRotate:) name:@""
                                         UIDeviceOrientationDidChangeNotification" object:nil];

- (void) didRotate:(NSNotification *) notification
{
    UIInterfaceOrientation currentOrientation = [[UIDevice currentDevice] orientation];

    // Important: Somme times, the current device orientation is Unknown and then the only other
    // way to know the orientation is the variable self.interfaceOrientation
    if (currentOrientation == UIDeviceOrientationUnknown ||
        currentOrientation == UIDeviceOrientationFaceUp ||
        currentOrientation == UIDeviceOrientationFaceDown){
        currentOrientation = self.interfaceOrientation;
    }
    if( UIDeviceOrientationIsLandscape(currentOrientation)){
        // Device is in landscape redraw view for this orientation
    }else{
        // Device is in portrait redraw view for this orientation
    }
}
```

Listing 5.5: Code d'enregistrement pour la notification de rotation des appareils.

3.6 Settings

Diagramme de séquence

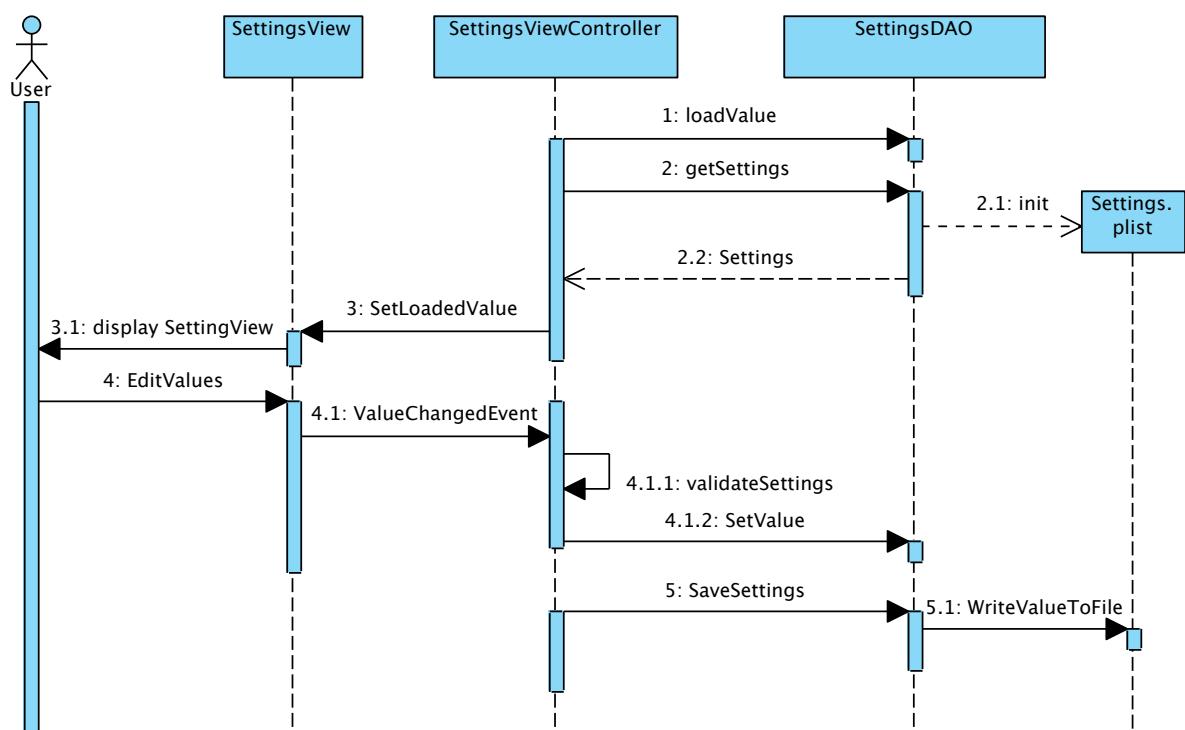


FIGURE 5.8: Diagramme de séquence concernant la lecture et la modification des paramètres

Diagramme de classe

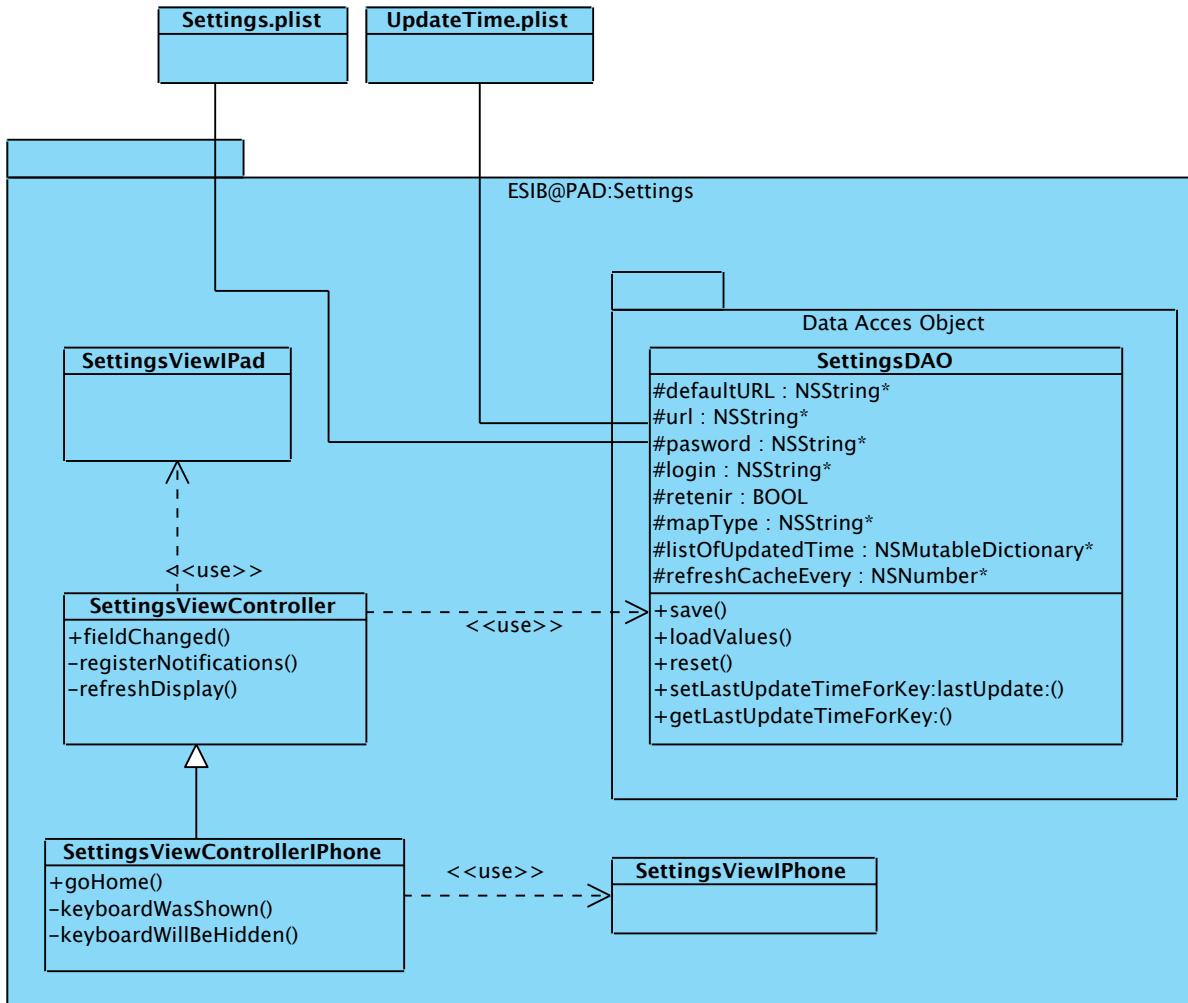


FIGURE 5.9: Diagramme de classe du composant Settings

Discussion

Appel propose un système relativement simple pour modifier les paramètres d'une application. Ce système est capable, à partir d'un fichier XML, de créer l'interface graphique pour modifier son contenu. Le système d'appel **n'a pas été utilisé car il oblige l'utilisateur à sortir de l'application** et d'aller dans la fenêtre de paramétrage du système d'exploitation pour modifier les paramètres de l'application.

Le détail ainsi que l'utilité concernant les fonctions `setLastUpdateTimeForKey` et `getLastUpdateTimeForKey` est expliqué dans le chapitre 3.2.

3.7 Map

Diagramme de séquence

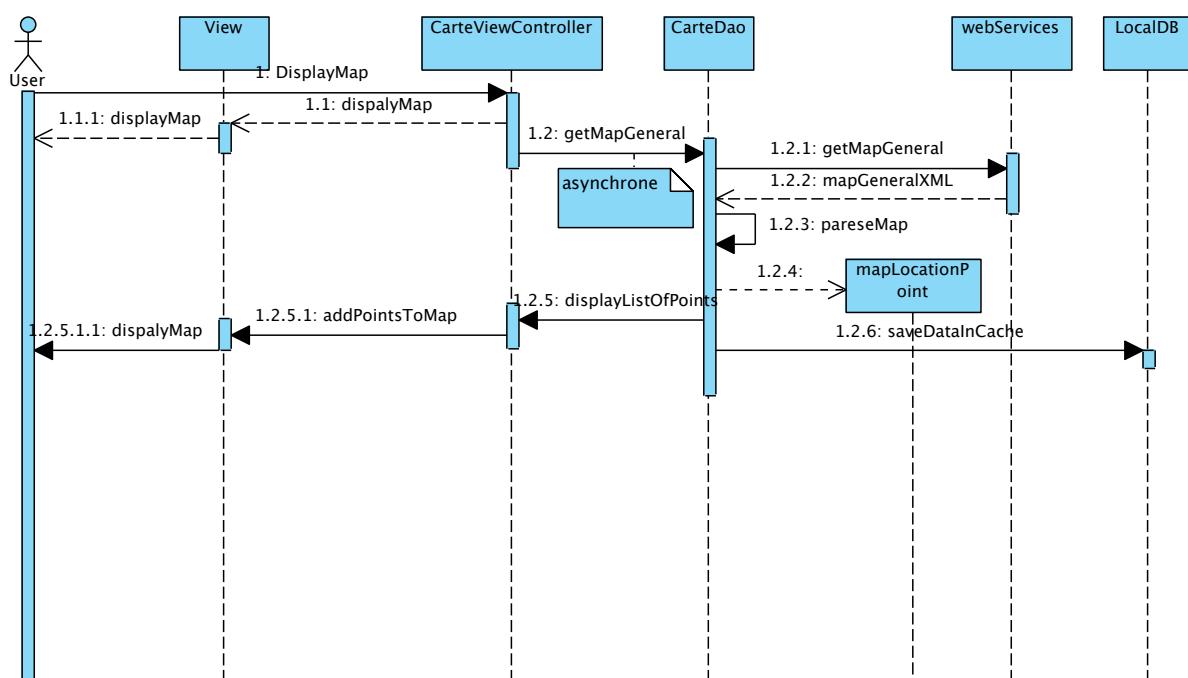


FIGURE 5.10: Exemple de séquence concernant l'affichage de la carte

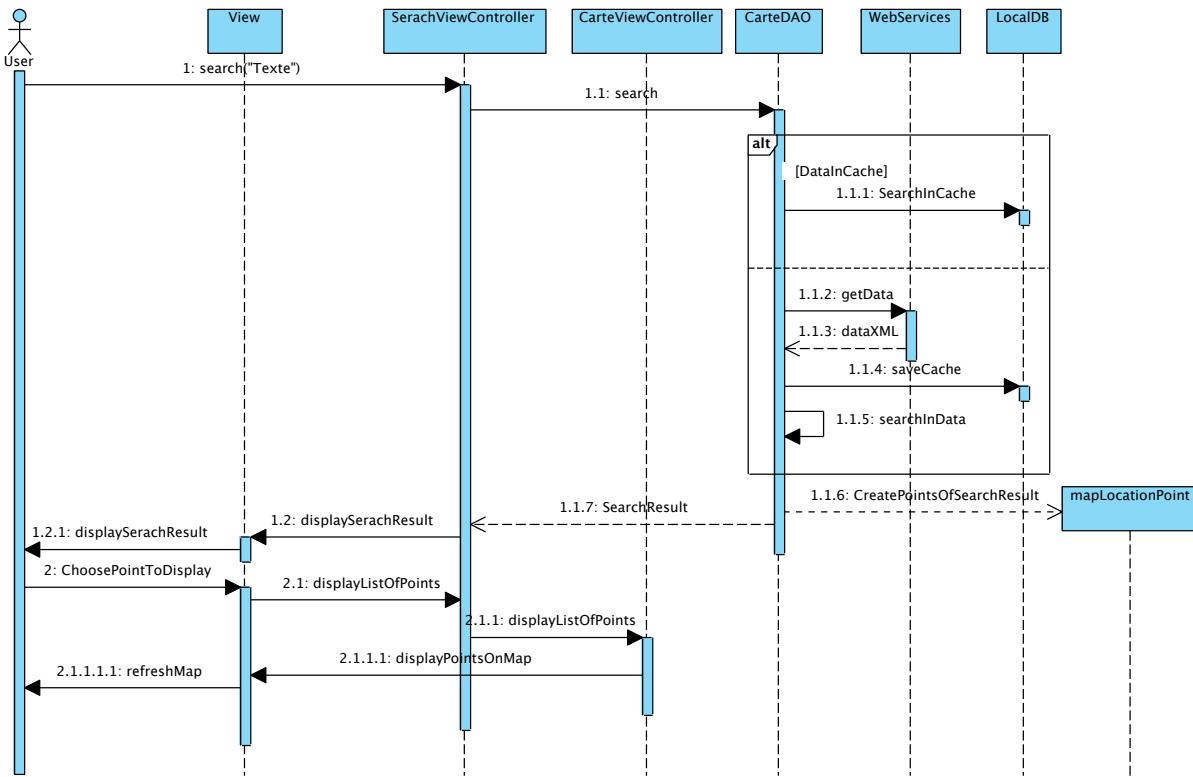


FIGURE 5.11: Exemple de séquence concernant la recherche d'un élément sur la carte

Le diagramme de séquence est valable pour les deux appareils. La seule différence est que sur l'IPad la vue chargée ne cachera pas l'écran entier mais rien qu'une partie de l'écran.

Diagramme de classe

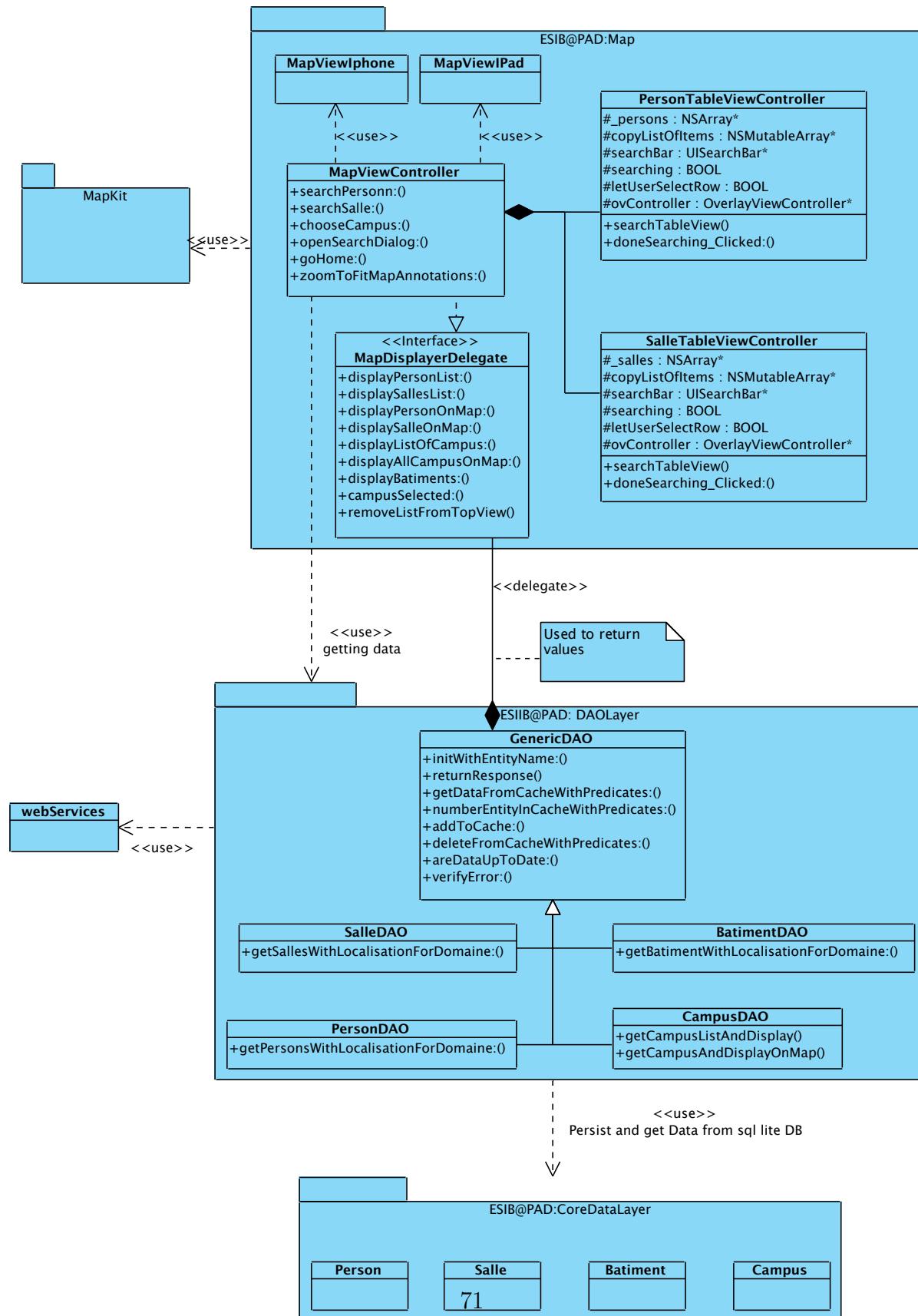


FIGURE 5.12: Diagramme de classe du composant Map

Discussion

Le framework MapKit qui exploite les images de googleMap est utilisé pour afficher la carte. Son utilisation est simple et possède un excellent tutoriel sur internet [7]. Il est très facile de démarrer avec cette librairie. Ce framework nous permet d'ajouter des indicateurs sur la carte pour signaler les emplacements intéressants.

```

MKMapView * map = [[MKMapView alloc] initWithFrame:self.view.frame];
CLLocationCoordinate2D coordinate;
coordinate.latitude = 35.000;
coordinate.longitude = 33.000;
MapLocations *annotation = [[[MapLocations alloc] initWithName:@"Un exemple d'annotation"
    description:@"Voici une description" coordinate:coordinate] autorelease];
[map addAnnotation:annotation];
    
```

Listing 5.6: Code de création d'un objet MKMapView et l'ajout d'une annotation.

Il est tout à fait pensable d'intégrer par la suite des cartes des campus plus détaillées aux cartes existantes.

L'appel asynchrone nous permet de télécharger des données comme la liste de personnes ou l'emplacement des bâtiments depuis internet d'une manière transparente. Avec les appels asynchrones on évite que toute l'interface graphique soit gelée.

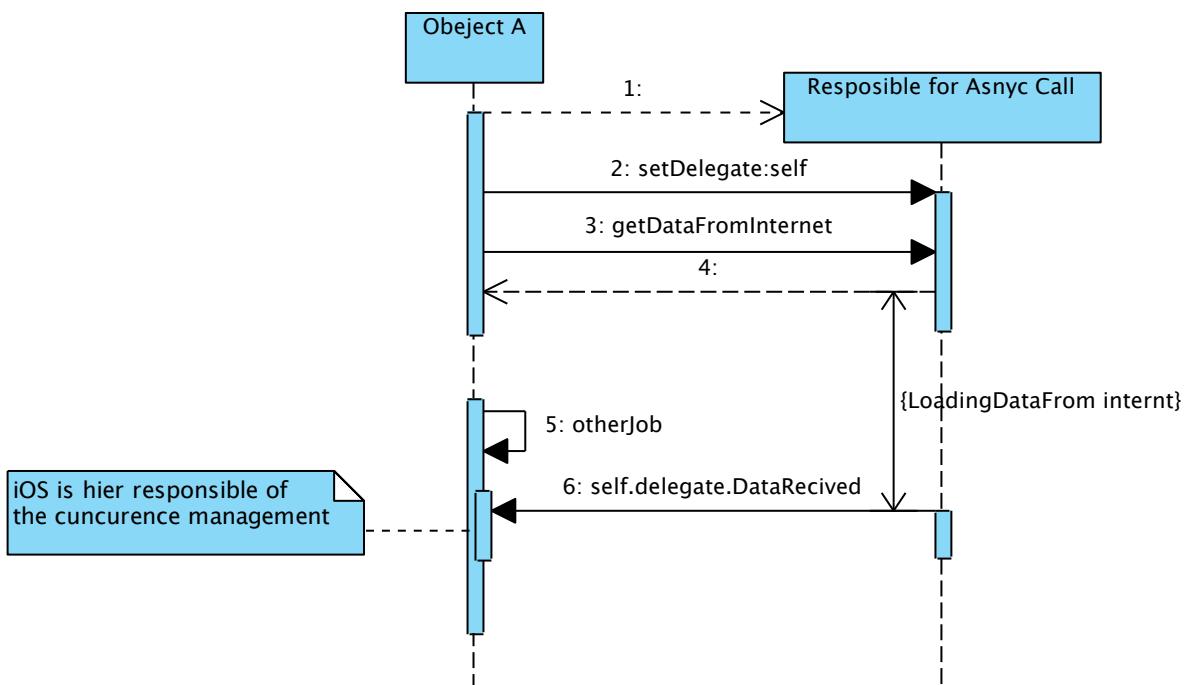


FIGURE 5.13: Diagramme de séquence illustrant l'appel asynchrone pour télécharger des données depuis internet

3.8 News

Diagramme de séquence

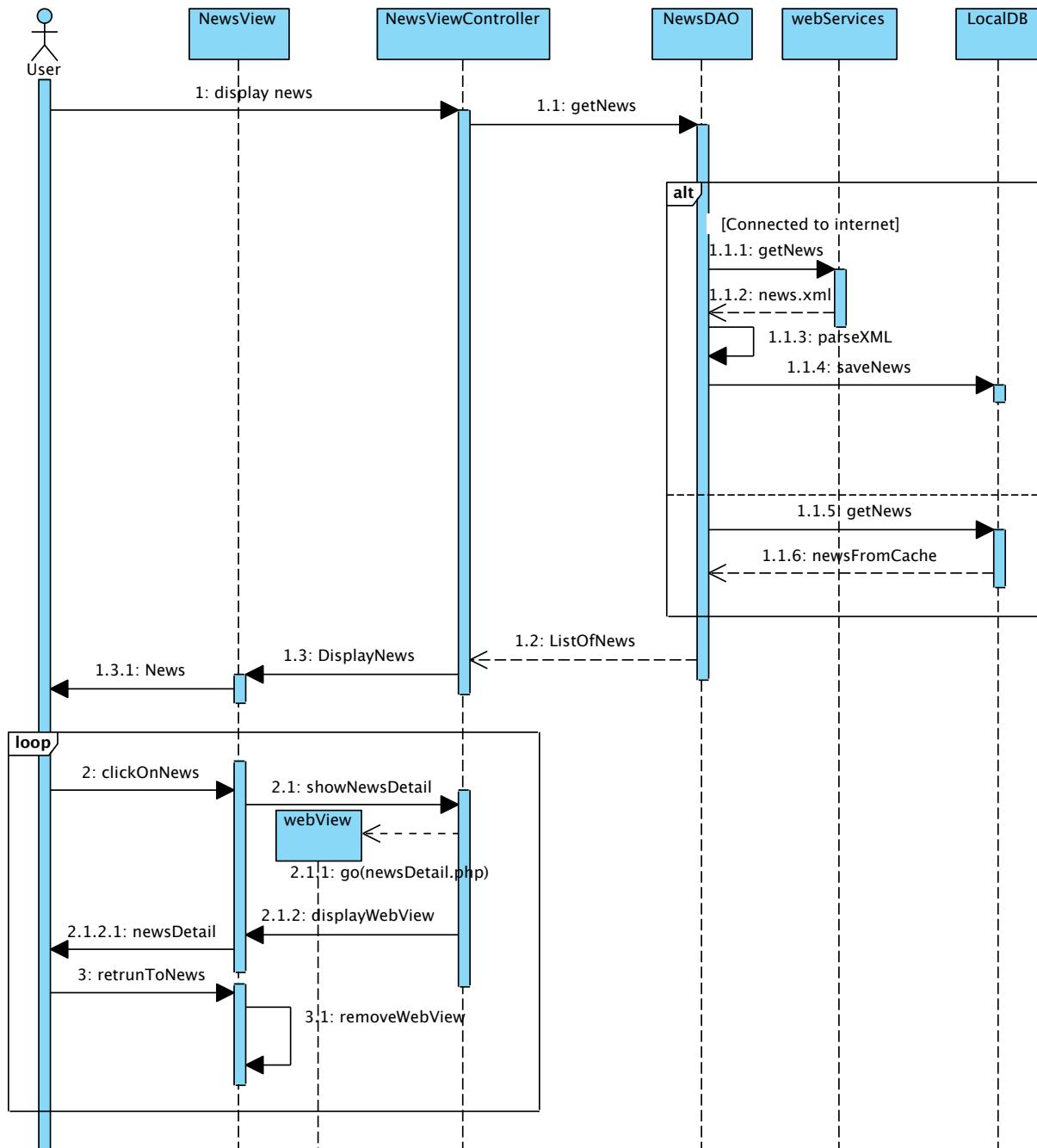
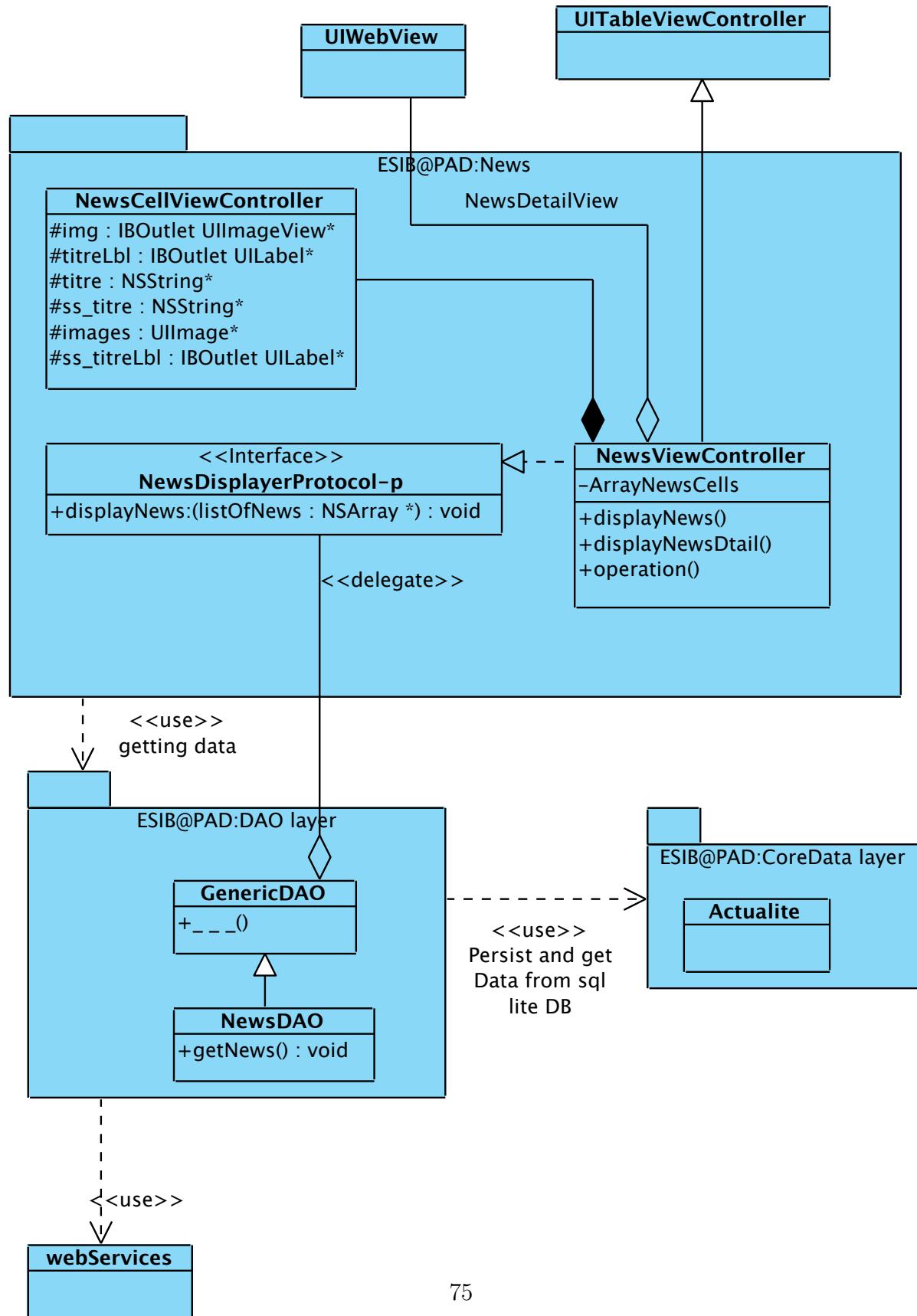


FIGURE 5.14: Exemple de séquence concernant l'affichage des news

Ce diagramme de séquence nous montre que les news sont de tout façon téléchargées depuis internet même si elles sont déjà en cache. Ce choix est dû à la nature des données

qui doivent être toujours à jour. Cependant si il n'y a pas de connexion internet, les informations en cache seront tout de même affichées.

Diagramme de classe



Discussion

Personnalisation des cellules d'un tableau : Afin de rendre le design graphique plus attrayant, les cellules du tableau ont été personnalisées. Il existe 2 principales façons pour modifier l'apparence des cellules :

1. La première consiste à modifier dans le code l'apparence avec des méthodes telles que `setBackground`, `setColor`. Cette méthode a un désavantage qui est de devoir exécuter la modification après chaque modification pour voir le résultat. De plus, pour chaque cellule, les mêmes opérations seront refaites à chaque création.
2. La deuxième solution est de créer un fichier NIB² à l'aide de l'outil graphique (Interface builder) inclus dans X-Code. Les avantages ici sont que l'on peut visuellement voir le résultat et on a une très grande liberté d'expression. De plus, les objets sont directement stockés sous format binaire dans l'application et on ne doit pas, pour chaque cellule, les dessiner à nouveau. L'inconvénient est que cette manière de faire nécessite plus de connaissances techniques.

La deuxième variante a été utilisée et il est ainsi possible de personnaliser rapidement l'apparence des news.

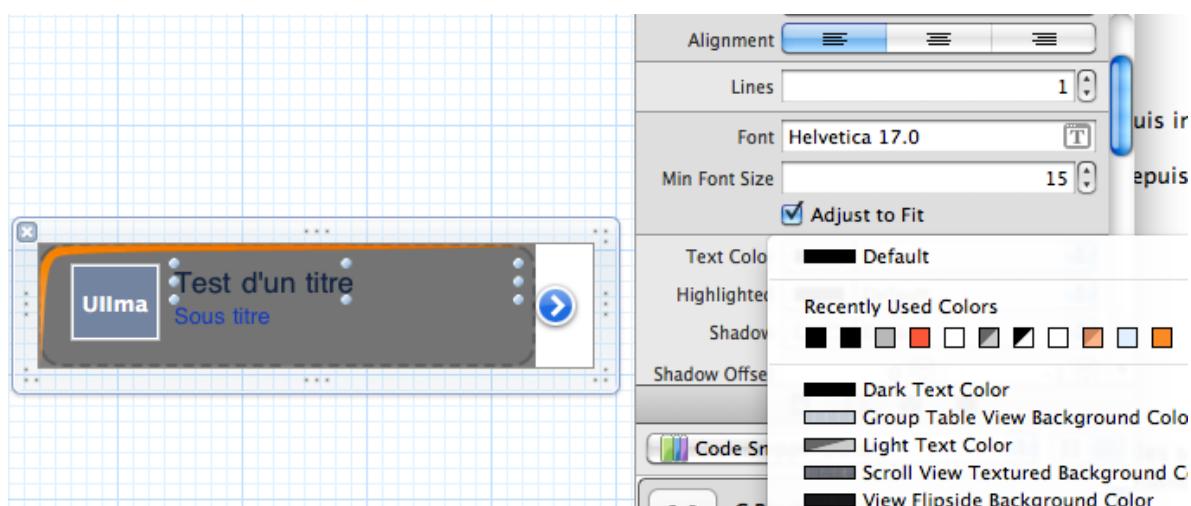


FIGURE 5.16: Illustration de la modification de l'apparence des cellules et plus précisément de la couleur du texte du titre.

Téléchargement d'image à partir d'Internet l'iOS offre la possibilité de charger des images depuis internet d'une manière simplifiée. Mais le téléchargement est fait d'une manière synchrone. Pour palier à ce problème nous pouvons utiliser la classe `NSURL-Reqes` pour télécharger les données brutes et les traiter en tant qu'image une fois toutes les données reçues. Pour plus de détails, voir le code source de la classe `AsyncImageView` dans le dossier Utility.

2. http://fr.wikipedia.org/wiki/Interface_Builder

3.9 Directory

Diagramme de séquence

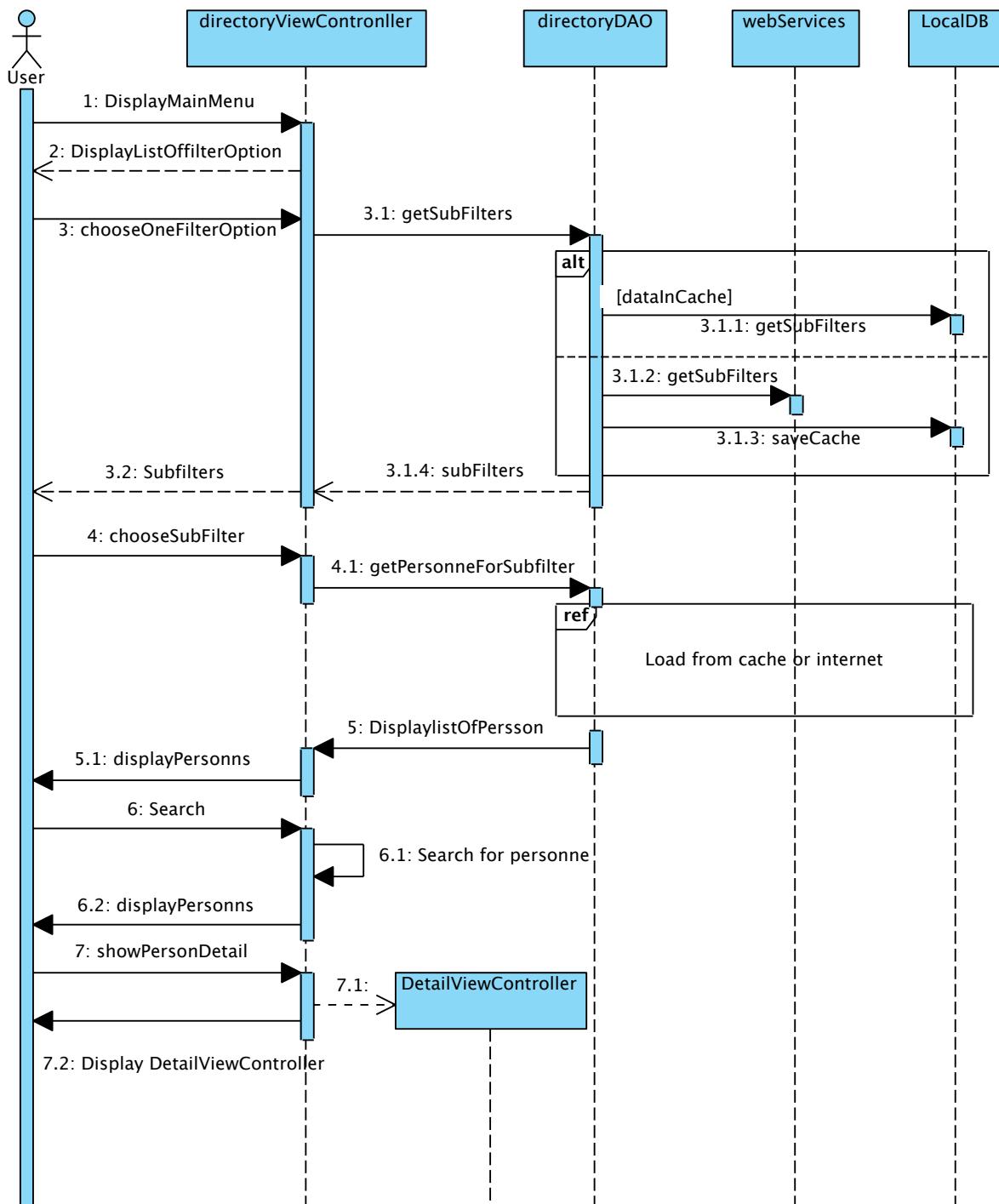


FIGURE 5.17: Exemple de séquence concernant choix d'un filtre d'affichage et ensuite l'affichage d'une personne de l'annuaire

Diagramme de classe

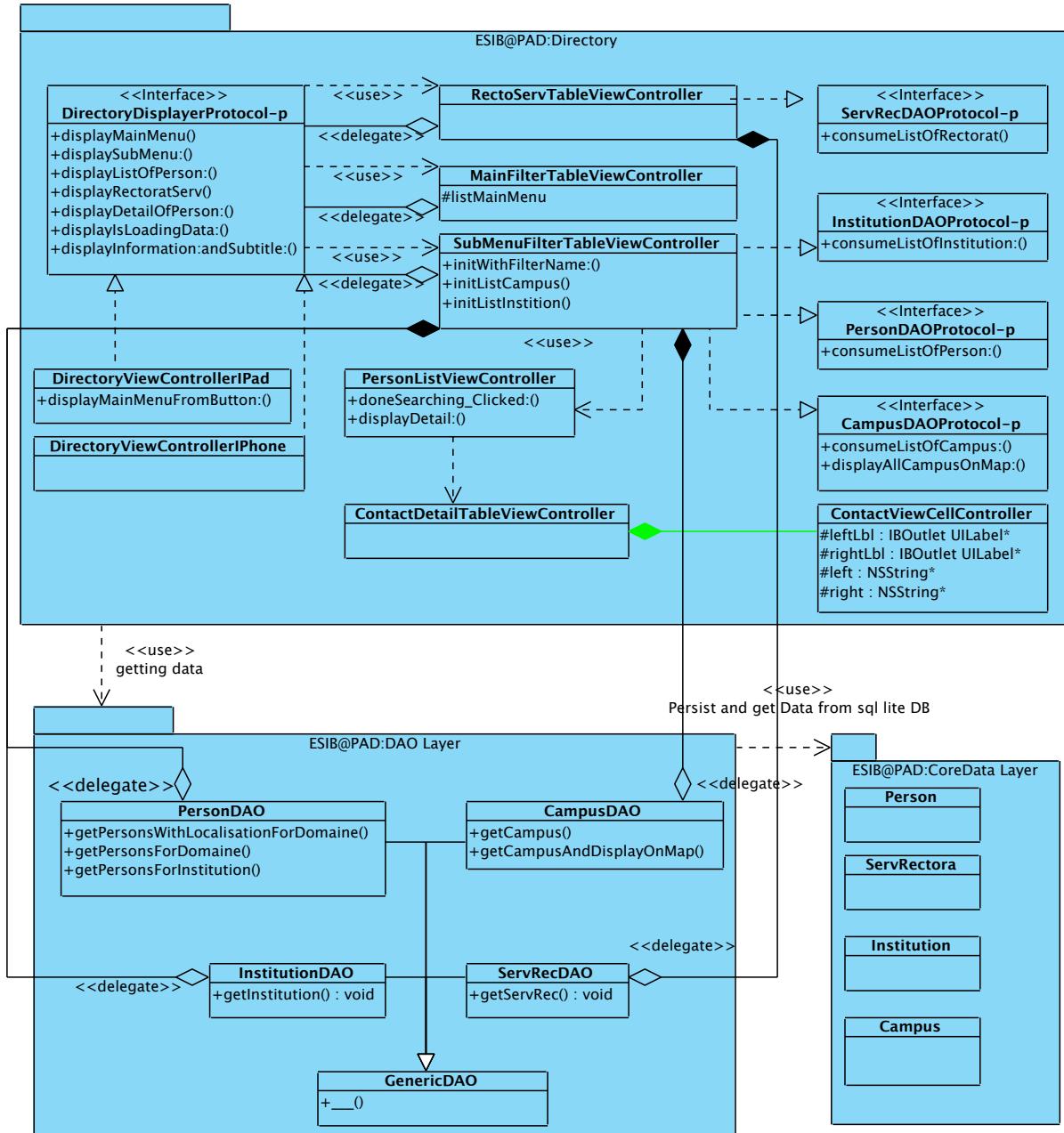


FIGURE 5.18: Diagramme de classe du composant News

Discussion

Interface graphique sur iPad : Il était prévu d'utiliser au départ un UISplitViewController qui permet d'avoir deux parties, une pour naviguer et l'autre pour afficher le contenu. Mais après de longues heures de recherches et d'essais, il s'avère que l'utilisation

de ce composant dans une partie de l'écran (et non en plein écran) n'est pas possible. De ce fait, une interface qui répond à nos besoins a été conçue. Cette interface nous permet d'avoir un élément à droite pour l'affichage de contenu et à gauche une zone réservée pour la navigation. Les effets de transition ont été faits à l'aide des fonctions d'animation de la classe UIView.

```

-(void) animatteView:(UIView *) toAnim{
    // The first anim of the size to 50 x 50 and his position at x = 100 y =200;
    [UIView animateWithDuration:0.5 delay:0 options: UIViewAnimationCurveEaseOut
        animations:^{
            CGRect rect = CGRectMake(100, 200, 50, 50);
            toAnim.frame= rect;
        }
        completion:^(BOOL finished){
            NSLog(@"First anim finish")
        }];
    // The second anim of the alpha value to 0 (transparent)
    // The delay propriete help us to sync the animations
    [UIView animateWithDuration:0.5 delay:0.5 options: UIViewAnimationCurveEaseOut
        animations:^{
            toAnim.alpha= 0;
        }
        completion:^(BOOL finished){
            NSLog(@"Second anim finish")
        }];
    [UIView commitAnimations];// Starting the animation.
}

```

Listing 5.7: Exemple de 2 animations à l'aide de la classe UIView. La première change la taille et l'emplacement d'un élément graphique et la deuxième change sa transparence.

Il aurait été intéressant d'avoir le temps pour rendre ce composant plus générique et de le publier sur internet pour ainsi éviter à d'autres utilisateurs de devoir refaire le même travail.

Recherche : Pour offrir à l'utilisateur la fonction chercher, deux méthode s'offrent à nous :

1. La première consiste à utiliser les requêtes SQL-Lite pour faire la recherche dans la base de données et d'afficher le résultat.
2. La deuxième est de faire la recherche directement dans la liste d'objets actuellement affichés et de masquer les éléments qui ne répondent pas au texte de recherche.

La deuxième façon est celle conseillée par Apple. La première obligerait à chaque requête de réinitialiser chaque objet et serait trop coûteuse en matière de ressources système.

Voici le code qui nous permet de filtrer les éléments dans la liste :

```

- (void) searchTableView {
    NSString *searchText = searchBar.text;

```

```

NSMutableArray *searchArray = [[NSMutableArray alloc] init];// Strings for searching

// _persons is an array with the current displayed list
for (Person * p in _persons)
{
    NSString * s = [NSString stringWithFormat:@"%@ %@", p.nom , p.prenom ,p.carriere];
    // Whe want to search in the files : nom , prenom ,carriere
    [searchArray addObject:s];
}
int i=0;
for (NSString *sTemp in searchArray)// We check each row
{
    NSArray* separatedWord = [searchText componentsSeparatedByString: @" "];
    for (NSString *word in separatedWord) {
        NSRange titleResultsRange = [sTemp rangeOfString:word options:
NSCaseInsensitiveSearch];
        if (titleResultsRange.length > 0){
            [copyListOfItems addObject:[_persons objectAtIndex:i]];
            break;
        }
        i++;
    }
    [searchArray release];
    searchArray = nil;
    [self display:copyListOfItems];
}
    
```

Listing 5.8: Methode de recherche dans une UITableView.

Lancer un appel téléphonique : La philosophie d'Apple veut garder chaque application dans son propre cadre et limiter la communication avec d'autres applications pour des raisons de sécurité. Cependant, quelques tâches de base sont tout de même permises et l'une de celles-là est de lancer un appel téléphonique depuis d'autres applications. Mais une fois l'appel lancé, l'application est mise en background et elle ne sera pas remise au premier plan à la fin de l'appel. Cette contrainte est connue et elle est impossible à contourner.

```

- (void)calltoNum:(NSString *)telNumber
{
    NSString *s = [[NSString alloc] initWithFormat:@"tel://%@",telNumber];
    [[UIApplication sharedApplication]openURL:[NSURL URLWithString:s]];
    [s release];
}
    
```

Listing 5.9: Lancement d'un appel téléphonique sur l'iPhone.

Écrire un e-mail : On peut bien sûr utiliser la même façon que pour l'appel téléphonique (remplacer tel :// par mailto), mais il existe une autre variante plus élégante. Cette va-

riante nous permet de rester dans l'application et d'éviter qu'à la fin de l'écriture de l'e-mail l'utilisateur doive réouvrir l'application pour continuer son travail. Le composant MFMailComposeViewController fournit avec l'iOS nous permet de faire ceci.

```

- (void)sendEmailTo:(NSString *)destination
{
    if ([MFMailComposeViewController canSendMail]) {

        MFMailComposeViewController *mailComposer = [[MFMailComposeViewController alloc]
        init];
        [[mailComposer navigationBar] setTintColor:[UIColor colorWithRed:0.03f green:0.03f blue
        :0.03f alpha:1.0f]];
        mailComposer.mailComposeDelegate = self;
        [mailComposer setSubject:@"Subject"];
        [mailComposer setMessageBody:@"Sent from ESIB@PAD" isHTML:NO];
        [mailComposer setToRecipients:[NSArray arrayWithObject:destination]];
        [self presentViewController:mailComposer animated:YES];
        [mailComposer release];
    } else {
        UIApplication *app = [UIApplication sharedApplication];
        [app openURL:[NSURL URLWithString:
                    [NSString stringWithFormat:@"mailto:%@?subject=%@&body=%@",
                    personInformation.email, @"Subject", @"Sent from ESIB@PAD"]]];
    }
}
    
```

Listing 5.10: Ouverture de la fenêtre d'écriture d'e-mail

3.10 Calendrier

Diagramme de séquence

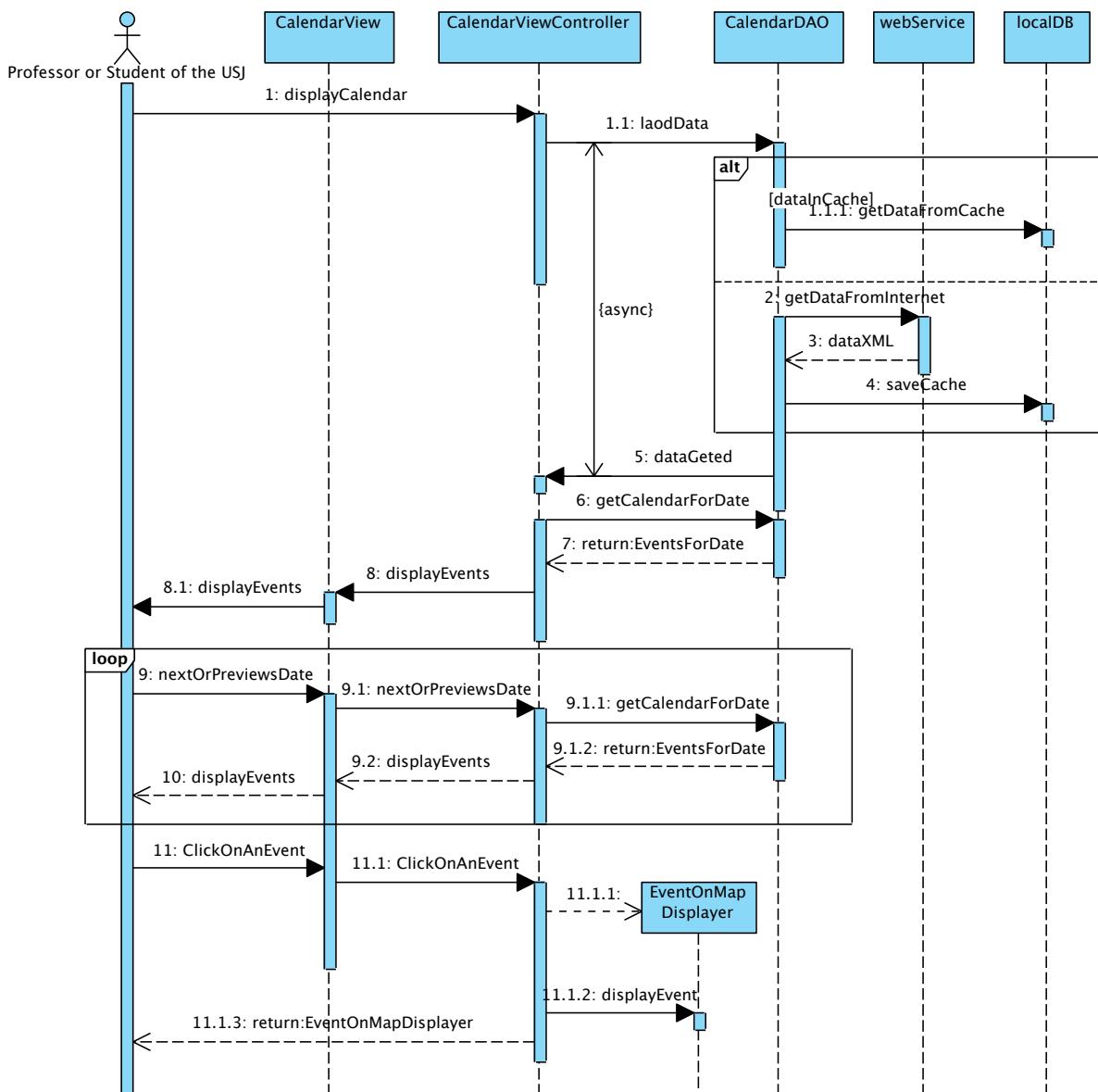


FIGURE 5.19: Exemple de séquence concernant l'affichage de l'horaire pour une journée et l'affichage du détail de l'évènement sur la carte

Diagramme de classe

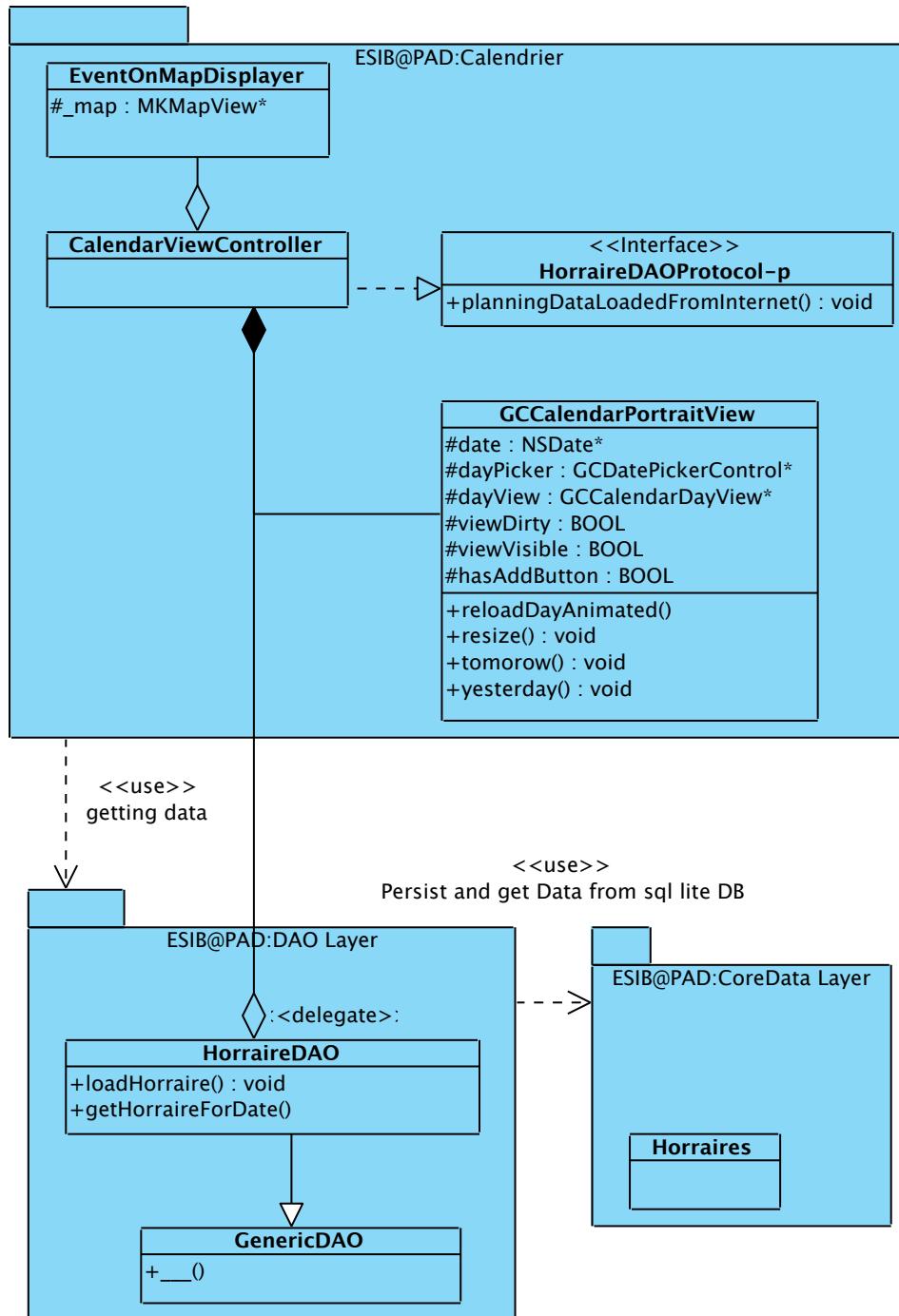


FIGURE 5.20: Diagramme de classe du composant calendrier

Discussion

GCCalendarPortraitView est un composant Opensource téléchargé depuis internet et qui permet d'afficher une journée d'un calendrier avec des événements. Ce composant était de base uniquement compatible en mode plein écran sur iPhone et ne supportait pas la rotation de l'écran. Les modifications nécessaires ont été faites pour pouvoir l'utiliser dans une partie spécifique de l'écran(plus grand pour l'iPad). L'ajout de la possibilité de passer au jour suivant, précédent grâce au mouvement de glissement du doigt a aussi été ajouté.

```

-(void) viewDidLoad{
    [super viewDidLoad];
    // Swipe Right notification
    UISwipeGestureRecognizer *swipeGesture = [[UISwipeGestureRecognizer alloc] initWithTarget:self
        action:@selector(swipe:)];
    swipeGesture.direction = UISwipeGestureRecognizerDirectionRight;
    [dayView addGestureRecognizer:swipeGesture];
    [swipeGesture release];

    // Swipe Left notification
    UISwipeGestureRecognizer *swipeGestureLeft = [[UISwipeGestureRecognizer alloc] initWithTarget:
        self action:@selector(swipe:)];
    swipeGestureLeft.direction = UISwipeGestureRecognizerDirectionLeft;
    [dayView addGestureRecognizer:swipeGestureLeft];
    [swipeGestureLeft release];
}
// Event sent when the swipe mouvement is recognized.—
-(void)swipe:(UISwipeGestureRecognizer *)swipe{

    if (swipe.direction == UISwipeGestureRecognizerDirectionLeft){
        [self tomorrow];
    } else if (swipe.direction == UISwipeGestureRecognizerDirectionRight){
        [self yesterday];
    }
}

```

Listing 5.11: Enregistrement pour les notifications du mouvement glissement du doigt et réception de l'événement

Affichage de l'événement sur la carte Le même framework MapKit utilisé dans le composant Map est réutilisé ici pour afficher l'emplacement de l'événement sur la carte.

3.11 ExamResult

Diagramme de séquence

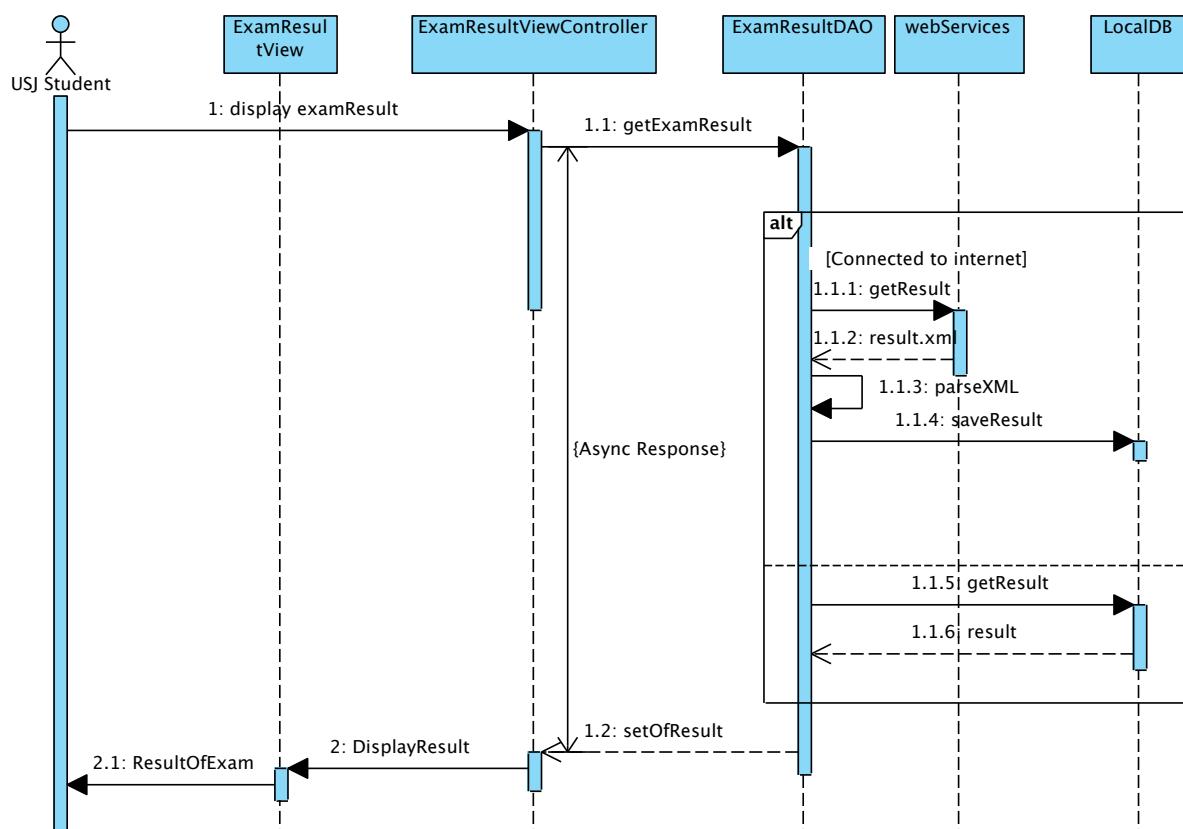


FIGURE 5.21: Exemple de séquence concernant l'affichage de l'horaire pour une journée et l'affichage du détail de l'évènement sur la carte

Diagramme de classe

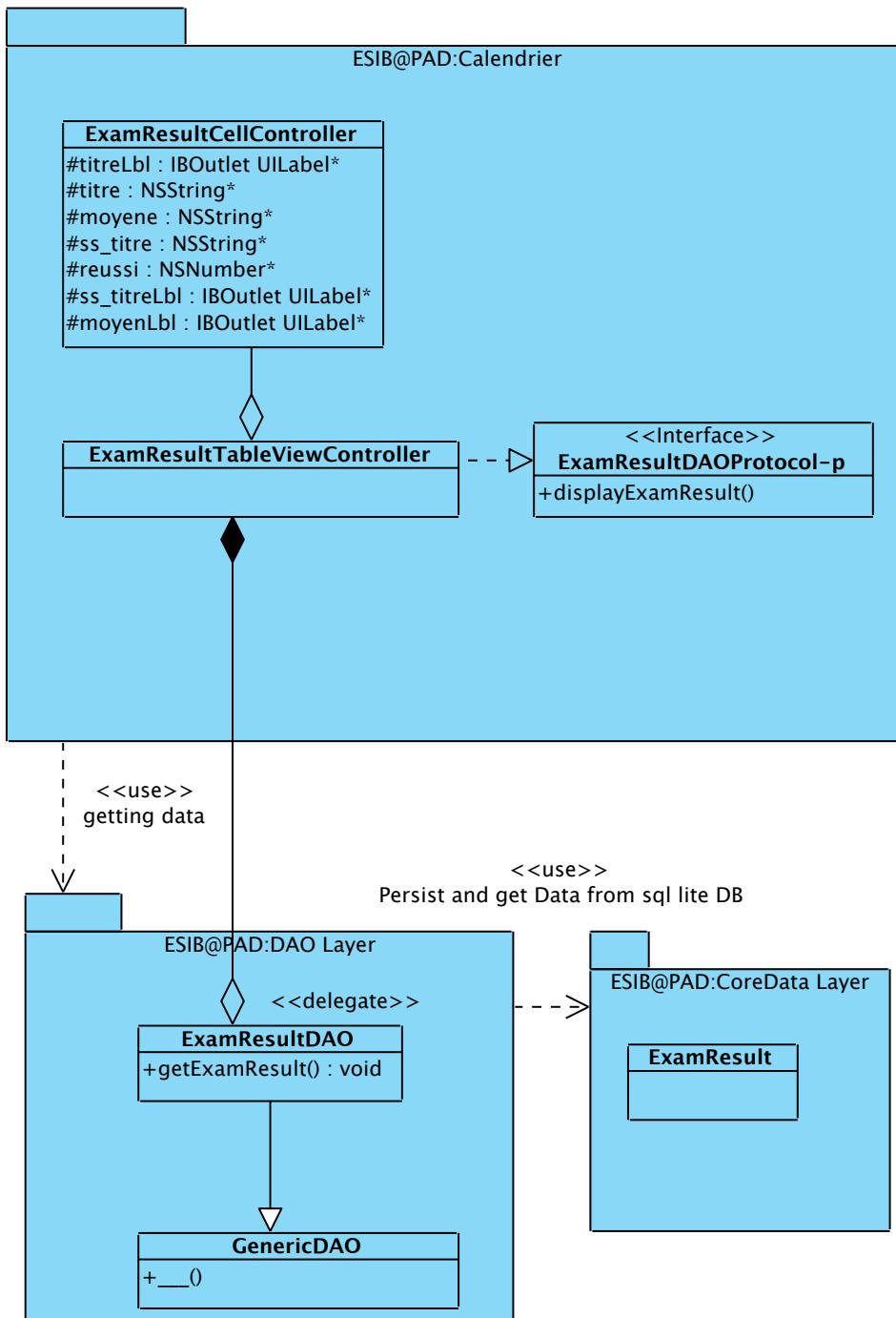


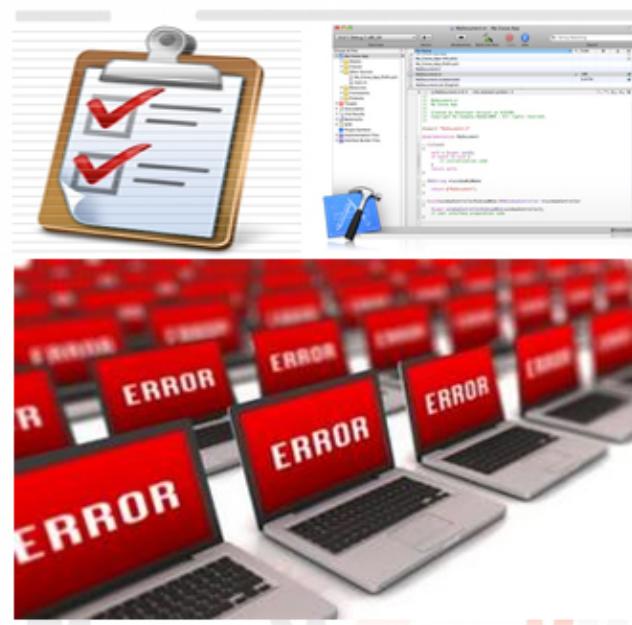
FIGURE 5.22: Diagramme de classe du composant calendrier

Discussion

Tout comme les news , si il y a une connexion internet, les données seront directement téléchargées depuis internet et non pas prises depuis le cache.

Chapitre 6

Software Test Description



1 Introduction

Ce chapitre explique une partie importante d'une application qui est celle de test. La section cas de test décrit les cas de test pour les tests de fonctionnalités. La partie protocole de test regroupe les protocoles de test qui ont été effectués après chaque release.

1.1 Philosophie de test

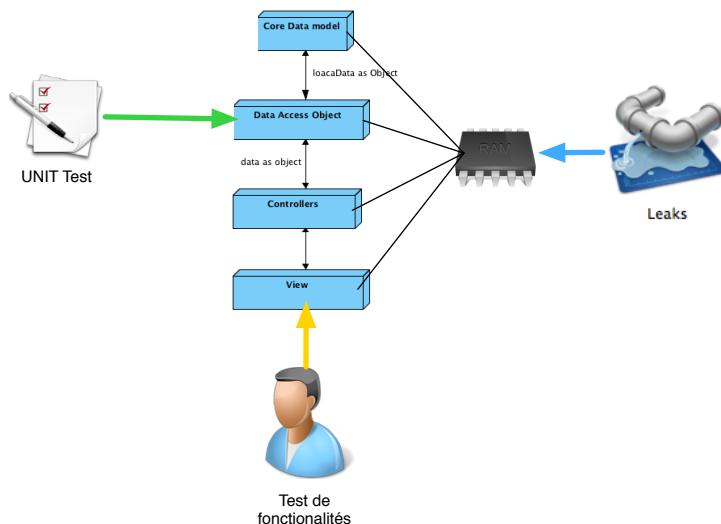


FIGURE 6.1: Vue globale de l'architecture du système

Les tests se feront sur trois niveaux, le premier niveau est celui des tests de fonctionnalités qui sont des tests faits par l'humain selon un procédé décrit dans la section 3. Le deuxième niveau est celui fait dans le code et qui est reproductible automatiquement (Unit test). Le troisième niveau est celui des tests de fuite dans la mémoire(Memory Leaks), qui consiste à observer la mémoire lors de l'utilisation de l'application(cas d'utilisation chapitre 3) et de vérifier qu'aucune variable n'est stockée en mémoire indéfiniment. Il est important d'indiquer que les cas de tests sont imaginés en même temps que la spécification, ce qui nous permet d'avoir un point de référence concernant les objectifs à atteindre. Voir le SPMP chapitre "Processus technique" pour plus d'information.

A la fin de chaque itération, un protocole de test est rédigé après avoir testé les nouvelles fonctionnalités et retesté les anciennes. Grâce à cette stratégie, on est sûr que les nouvelles fonctionnalités n'empêche pas le fonctionnement des anciennes et que le tout reste compatible.

2 Organisation des tests

2.1 Éléments à tester

- Le bon fonctionnement des différents cas d'utilisations.
- Des tests unitaires pour la partie logique métier.
- Des analyses de fuites dans la mémoire (Leaks) doivent être faites, car pour être visible sur l'appstore, une application ne doit pas contenir de Leaks.

2.2 Éléments à ne pas tester

- La sécurité des web services.
- La cohérence des résultats retournés par les web services.

2.3 Outils de test et environnement

Le framework OCUnit¹ nous permet de faire écrire des tests unitaires. Son fonctionnement est assez proche de celui JUnit pour Java. Les fonctions tel que STAssertEqualObjects ou STAssertTrue permettent de faire échouer ou réussir un test.

Pour le test des Memory Leaks l'outil Instruments² qui est prévu à cet effet est utilisé. Depuis XCode, choisir 'Product ⇒ Profile' pour exécuter à l'aide de Instruments. Une fois le code lancé, choisir Leaks comme template de base. Ensuite, utiliser l'application selon un cas de test. L'outil vous indiquera si des leaks existent et leurs emplacements.

3 Cas de test

3.1 Navigation

ID	1
Description	Teste que l'on peut bien naviguer d'une vue à l'autre sans erreurs.

1. <http://developer.apple.com/tools/unittest.html>

2. <http://developer.apple.com/library/mac/#documentation/DeveloperTools/Conceptual/InstrumentsUserGuide>

Déroulement	<ul style="list-style-type: none"> – Fermer complètement l'application si elle était ouverte. – Ouvrir l'application. – Pour chaque vue, cliquer sur le logo pour ouvrir, voir si le résultat obtenu est cohérent, revenir au menu principal. – Pour chaque vue, cliquer sur le logo pour ouvrir, modifier le contenu dans la vue, fermer l'application à l'aide du bouton menu de l'appareil, réouvrir l'application, vérifier que c'est toujours cette vue qui est visible et qu'aucune information n'a été perdue après la manœuvre. – Pour chaque vue, cliquer sur le logo pour ouvrir. faire 4 x une rotation de 90 degrés à l'appareil. Vérifier qu'après chaque rotation la vue est dans le bon sens et que les éléments sont affichés correctement.
--------------------	--

3.2 Paramétrage

ID	2
Description	Teste que l'on peut modifier les paramètres de l'application
Déroulement	<ul style="list-style-type: none"> – Fermer complètement l'application si elle était ouverte. – Ouvrir l'application. – Ouvrir la fenêtre de paramètres. – Pour chaque champ : <ol style="list-style-type: none"> 1. Éditer la valeur. 2. fermer la fenêtre de paramètres. 3. Réouvrir la fenêtre de paramètres. 4. Vérifier que les valeurs sont bien celles saisies auparavant. – Modifier tout les paramètres. – Fermer complètement l'application(À l'aide du gestionnaire d'applications et non seulement à l'aide du bouton menu.) – Réouvrir l'application et être sûr que les modifications ont bien été enregistrées.

3.3 Carte

ID	3
Description	Test du bon fonctionnement de la carte.
Déroulement	<ul style="list-style-type: none"> - Se rendre au campus de l'ESIB - Se connecter à internet - Fermer complètement l'application si elle était ouverte. - Ouvrir l'application. - Ouvrir la fenêtre de la carte. - Presser le bouton Localisez-moi et vérifier que l'endroit retourné est au bon emplacement. - Vérifier que l'application affiche des indicateurs sur les principaux immeubles du campus. - Se déplacer dans le campus et vérifier que l'indicateur de position actuelle suit le déplacement. - Saisir le nom d'une personne dans la barre de recherche, vérifier qu'on obtient un indicateur concernant l'emplacement du bureau de cette personne en suivants les écrans. - Saisir le nom d'une classe dans la barre de recherche et faire de même que l'étape précédente. - Saisir le nom d'un bâtiment dans la barre de recherche et faire de même que l'étape précédente. - Presser sur le bouton de Navigation par élément. <ul style="list-style-type: none"> - Choisir : Bâtiments. - Choisir un bâtiment spécifique et vérifier que son emplacement est affiché sur la carte. - Se déconnecter d'Internet et recommencer les étapes précédentes. Les mêmes fonctionnalités doivent être visibles.

3.4 News

ID	4
Description	Test du bon fonctionnement de l'affichage des news.

Déroulement	<ul style="list-style-type: none"> – Installer à neuf l’application. – Se connecter à internet. – Fermer complètement l’application si elle était ouverte. – Ouvrir l’application. – Ouvrir la fenêtre des news. – Comparer le résultat avec celui de la page internet : http://www.usj.edu.lb/ – Visualiser le détail des news et vérifier que le détail correspond à la news. – Se déconnecter d’Internet et recommencer les étapes précédentes. Les mêmes fonctionnalités doivent être visibles.
--------------------	--

3.5 Directory

ID	5
Description	Test du bon fonctionnement de l’annuaire.
Déroulement	<ul style="list-style-type: none"> – Installer à neuf l’application. – Se connecter à internet. – Fermer complètement l’application si elle était ouverte. – Ouvrir l’application. – Ouvrir la fenêtre de l’annuaire. – Choisir au moins 3 filtres d’affichages différents et s’assurer que les résultats sont bien cohérents. – Choisir une personne et envoyer un e-mail via le lien mail. – Sur l’iPhone, cliquer sur le numéro de téléphone d’une personne et vérifier que l’appel est lancé. – Se déconnecter d’Internet et recommencer les étapes précédentes. Les mêmes fonctionnalités doivent être visibles.

3.6 Calendrier

ID	6
Description	Test du bon fonctionnement de l’affichage du calendrier.

Déroulement

- Installer à neuf l'application.
- Se connecter à internet.
- Fermer complètement l'application si elle était ouverte.
- Ouvrir l'application.
- Paramétrier l'application pour obtenir les données depuis le web service local.
- Ouvrir la fenêtre du calendrier.
- Passez au jour suivant ou précédent en faisant un mouvement de glissement sur le calendrier.
- En cliquant sur un cours, la carte s'affiche, avec l'emplacement de dernier.
- Se déconnecter d'Internet et recommencer les étapes précédentes. Les mêmes fonctionnalités doivent être visibles.

3.7 Résultats des examens

ID	7
Description	Test du bon fonctionnement de l'affichage des résultats des examens.
Déroulement	<ul style="list-style-type: none"> – Installer à neuf l'application. – Se connecter à internet. – Fermer complètement l'application si elle était ouverte. – Ouvrir l'application. – Paramétrier l'application pour obtenir les données depuis le web service local. – Ouvrir la fenêtre de résultats des examens. – Vérifier que les résultats sont affichés correctement et que les notes sont bien celles obtenues lors de l'examen. – Se déconnecter d'Internet et recommencer les étapes précédentes. Les mêmes fonctionnalités doivent être visibles.

4 Protocole de test

Ce qu'on entend par protocole de test, c'est l'exécution de l'application en suivant pas à pas les étapes décrites dans le cas de test. En parallèle à cette exécution, les tests de Leaks dans la mémoire sont effectués. Après chaque release, tout le protocole de test est répété pour garantir que les nouvelles fonctionnalités n'empêche pas le bon fonctionnement des anciennes. L'exécution des tests unitaire fait aussi partie du protocole de test et leurs logs sont mentionnés ci dessous.

4.1 Protocole de test 1

Version testé : 0.1 (<https://esibpad.googlecode.com/svn/tags/0.1>)

Date du test : 20/06/2011

Cas de test : Navigation

ID	1
Description	Tester si l'on peut bien naviguer d'une vue à l'autre sans erreurs.
Commentaires	Il n'existe pour le moment qu'une seule page.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Paramétrage

ID	2
Description	Tester si l'on peut modifier les paramètres de l'application
Commentaires	<ol style="list-style-type: none"> La fonction retenir n'est pas encore implémentée correctement, les valeurs sont de toutes façons enregistrées. La validité des champs n'est pas implémentée, les valeurs peuvent être incohérentes. Les valeurs des champs "Retenir et carte" ne sont enregistrées qu'en cas de modification d'un autre champ de type texte.
Objectif atteint	partiellement 75% <input checked="" type="checkbox"/>

Visa

Elias Medawar

Test unitaire

```
Test Suite 'ESIB_PADTests' started at 2011-06-20 06:33:02 +0000
Test Case '-[ESIB_PADTests testSettings]' started.
    Testing the settings DAO
Test Case '-[ESIB_PADTests testSettings]' passed (0.003 seconds).
Test Suite 'ESIB_PADTests' finished at 2011-06-20 06:33:02 +0000.
Executed 1 test, with 0 failures (0 unexpected) in 0.003 (0.003) seconds
```

Listing 6.1: Log des test unitaires

Objectif atteint : Complètement 100 % ✅

Test de fuite dans la mémoire

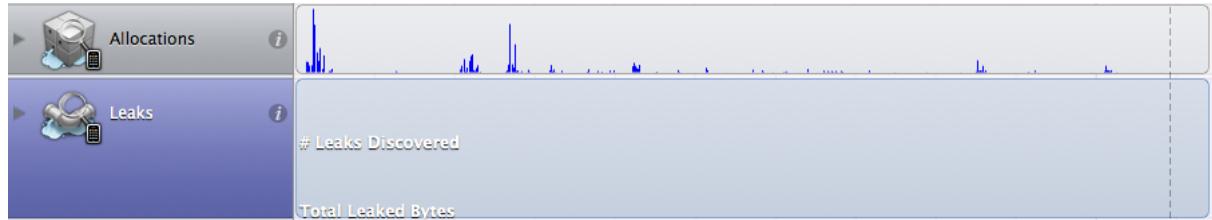


FIGURE 6.2: Résultat de l'analyse des Leaks à l'aide d'Xcode

Objectif atteint : Complètement 100 % ✅

On peut voir que le code ne contient aucune fuite de mémoire.

4.2 Protocole de test 2

Version testé : 0.2 (https://github.com/eia-fr/ESIB_PAD/tree/0.2)
Date du test : 05/07/2011

Cas de test : Navigation

ID	1
Description	Tester si l'on peut bien naviguer d'un vue à l'autre sans erreurs

Commentaires	Lors du chargement des informations depuis internet, le logo loading n'est pas centré quand l'iphone est en paysage.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Paramétrage

ID	2
Description	Tester si l'on peut modifier les paramètres de l'application
Commentaires	<ul style="list-style-type: none"> 1. La fonction retenir n'est pas encore implémentée correctement, les valeurs sont de toutes façon enregistrées. 2. La validité des champs est validée seulement au moment que les utiliser. 3. La carte est toujours en mode satellite.
Objectif atteint	partiellement 85% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Carte

ID	3
Description	Test du bon fonctionnement de la carte.
Commentaires	<ul style="list-style-type: none"> 1. La position de l'utilisateur est de toute façon affichée. 2. Les coordonnées latitude et longitude sont inversées pour les bâtiments du campus CTS 3. L'application ne fonctionne pas quand un campus n'a pas de bâtiment à afficher.
Objectif atteint	Partiellement 95% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Test unitaire

```

Test Suite 'ESIB_PAD_SOURCESTests' started at 2011-07-05 07:37:38 +0000
Test Case '-[ESIB_PAD_SOURCESTests testCarte]' started.
Testing the Campus DAO: You must uninstall the application before using this test
Loading async the campus data from internet
Waiting 30 sec for disabling the internet connection
Getting campus data from cache?
Comparing local and distant data
Test Case '-[ESIB_PAD_SOURCESTests testCarte]' passed (90.071 seconds).
Test Case '-[ESIB_PAD_SOURCESTests testSettings]' started.
Testing the settings DAO
Test Case '-[ESIB_PAD_SOURCESTests testSettings]' passed (0.004 seconds).
Test Suite 'ESIB_PAD_SOURCESTests' finished at 2011-07-05 07:39:08 +0000.
Executed 2 tests, with 0 failures (0 unexpected) in 90.075 (90.077) seconds

```

Listing 6.2: Log des test unitaires

Objectif atteint : Complètement 100 %

Tester des appels de méthodes asynchrones n'est pas une tâche évidente. Notre classe de test tourne dans le thread A et la classe testée dans le thread B . Pour parvenir à faire cette opération on utilise les méthodes de synchronisation mises à disposition de l'IOS pour bloquer le thread A durant le temps du téléchargement des données depuis internet du thread B. Pour recevoir les données, notre classe de test est définie comme déléguée de la classe testée. Ainsi quand le thread B reçoit et a traité les données il notifie le thread A du résultat obtenu.

Test de fuite dans la mémoire

Objectif atteint : partiellement 50 %

Un problème avec la classe NSPredicate crée des Leaks, selon la théorie le code n'en contient pas. Mais l'outil de mesure en détecte, des recherches plus approfondies pour trouver une solution seront réalisées.

4.3 Protocole de test 3

Version testé : 0.3 (https://github.com/eia-fr/ESIB_PAD/tree/0.3.1)

Date du test : 14/07/2011

Cas de test : Navigation

ID	1	
----	---	--

Description	Tester si l'on peut bien naviguer d'une vue à l'autre sans erreurs.
Commentaires	Lors du chargement des informations depuis internet, le logo loading n'est pas centré quand l'iPhone est en paysage.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Paramétrage

ID	2
Description	Tester si l'on peut modifier les paramètres de l'application.
Commentaires	<ul style="list-style-type: none"> 1. La fonction retenir n'est pas encore implémentée correctement, les valeurs sont de toutes façon enregistrées.
Objectif atteint	partiellement 95% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Carte

ID	3
Description	Test du bon fonctionnement de la carte.
Commentaires	<ul style="list-style-type: none"> 1. La position de l'utilisateur est de toute façon affichée.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : News

ID	4
Description	Test du bon fonctionnement de la carte.

Commentaires	1. Les news ne sont pas toujours les mêmes que celles sur le site, apparemment les webservices ne fournissent qu'une partie des news pour l'application.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Test unitaire

Test Suite '**ESIB_PAD_SOURCESTests**' started at 2011-07-12 14:40:32 +0000

Test Case '**-[ESIB_PAD_SOURCESTests testCarte]**' started.

Testing the Campus DAO: You must uninstall or reset cache of the application before testing

Loading async the campus data from internet

DATA received

Test Case '**-[ESIB_PAD_SOURCESTests testCarte]**' passed (60.564 seconds).

Test Case '**-[ESIB_PAD_SOURCESTests testNews]**' started.

Testing the News DAO: You must uninstall or reset cache of the application before testing

Loading async the news data from internet

DATA received

Test Case '**-[ESIB_PAD_SOURCESTests testNews]**' passed (61.191 seconds).

Test Case '**-[ESIB_PAD_SOURCESTests testSettings]**' started.

Testing the settings DAO

Test Case '**-[ESIB_PAD_SOURCESTests testSettings]**' passed (0.073 seconds).

Test Suite '**ESIB_PAD_SOURCESTests**' finished at 2011-07-12 14:42:34 +0000.

Executed 3 tests, with 0 failures (0 unexpected) in 121.828 (121.830) seconds

Listing 6.3: Log des test unitaires

Objectif atteint : Complètement 100 %

Le principe de test de classe asynchrone décrit plus haut est réutilisé pour le test : test-News.

Test de fuite dans la mémoire

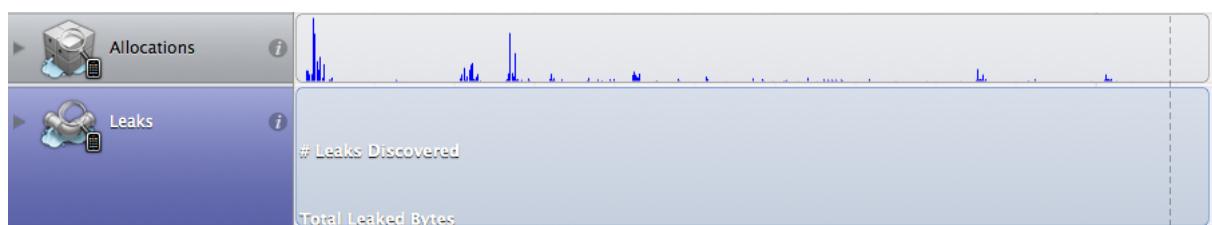


FIGURE 6.3: Résultat de l'analyse des Leaks à l'aide d'Xcode

Objectif atteint : complètement 100%

4.4 Protocole de test 4

Version testé : 0.5 (https://github.com/eia-fr/ESIB_PAD/tree/05)

Date du test : 05/08/2011

Cas de test : Navigation

ID	1
Description	Tester si l'on peut bien naviguer d'une vue à l'autre sans erreurs.
Commentaires	Si on est entrain de loader des données depuis internet et que l'on choisit de passer à une autre fenêtre, l'application plante.
Objectif atteint	Partiellement 95% <input type="checkbox"/>
Visa	Elias Medawar

Cas de test : Paramétrage

ID	2
Description	Tester si l'on peut modifier les paramètres de l'application
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Carte

ID	3
Description	Test du bon fonctionnement de la carte.
Commentaires	<ol style="list-style-type: none"> La position de l'utilisateur est de toute façon affichée.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : News

ID	4
Description	Test du bon fonctionnement de la carte.
Commentaires	<ul style="list-style-type: none"> 1. Les news ne sont pas toujours les mêmes que celles sur le site, apparemment les webservices ne fournissent qu'une partie des news pour l'application.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Directory

ID	5
Description	Test du bon fonctionnement de l'annuaire.
Commentaires	<ul style="list-style-type: none"> 1. Si on télécharge l'annuaire d'un Campus, et qu'ensuite on désire avoir l'annuaire d'une institution (qui est une sous-entité d'un campus) les informations de cette institution sont à nouveau téléchargées depuis internet. Ce qui signifie qu'on télécharge 2 fois les mêmes informations.
Objectif atteint	Partiellement 90% <input type="checkbox"/>
Visa	Elias Medawar

Cas de test : Calendrier

ID	6
Description	Test du bon fonctionnement de la carte.
Commentaires	<ul style="list-style-type: none"> 1. aucun bug détecté.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Test unitaire

Test Suite '**ESIB_PAD_SOURCESTests**' started at 2011-08-07 12:48:23 +0000
 Test Case '**-[ESIB_PAD_SOURCESTests testCarte]**' started.
 Testing the Campus DAO: You must uninstall or reset cache of the application before testing
 Loading async the campus data from internet
 Recievied async the Campus DATA
 Test Case '**-[ESIB_PAD_SOURCESTests testCarte]**' passed (2.025 seconds).
 Test Case '**-[ESIB_PAD_SOURCESTests testDirectory]**' started.
 Testing the Person DAO **for** the directory fonctionality: You must uninstall or reset cache of the application before testing
 Loading async the directory of the cmapus CST from internet
 List of person recievied
 Test Case '**-[ESIB_PAD_SOURCESTests testDirectory]**' passed (17.585 seconds).
 Test Case '**-[ESIB_PAD_SOURCESTests testNews]**' started.
 Testing the News DAO: You must uninstall or reset cache of the application before testing
 Loading async the news data from internet
 Recievied async the news DATA
 Test Case '**-[ESIB_PAD_SOURCESTests testNews]**' passed (4.727 seconds).
 Test Case '**-[ESIB_PAD_SOURCESTests testPlaning]**' started.
 Testing the Horraire DAO **for** the planning fonctionality: You must uninstall or reset cache of the application before testing
 Loading async the plannong from internet
 Recievied async the planning from internet
 Test Case '**-[ESIB_PAD_SOURCESTests testPlaning]**' passed (18.140 seconds).
 Test Case '**-[ESIB_PAD_SOURCESTests testSettings]**' started.
 Testing the settings DAO
 Test Case '**-[ESIB_PAD_SOURCESTests testSettings]**' passed (0.005 seconds).
 Test Suite '**ESIB_PAD_SOURCESTests**' finished at 2011-08-07 12:49:25 +0000.
 Executed 5 tests, with 0 failures (0 unexpected) in 55.508 (55.538) seconds

Listing 6.4: Log des test unitaires

Objectif atteint : Complètement 100 % ☐

Le principe de test de classe asynchrone décrit plus haut est réutilisé pour les tests : testPlaning et testDirectory.

Test de fuite dans la mémoire

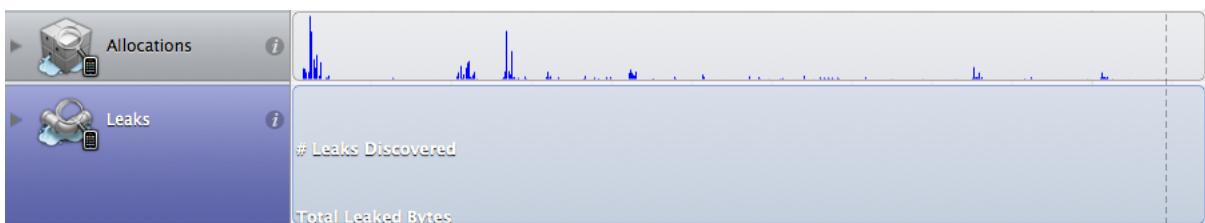


FIGURE 6.4: Résultat de l'analyse des Leaks à l'aide d'Xcode

Objectif atteint : complètement 100%

4.5 Protocole de test 5

Version testé : 0.6 (https://github.com/eia-fr/ESIB_PAD/tree/06)

Date du test : 15/08/2011

Cas de test : Navigation

ID	1
Description	Tester si l'on peut bien naviguer d'une vue à l'autre sans erreurs.
Commentaires	Dans la vue du calendrier, quand on ouvre la carte et après on fait une rotation de l'appareil, au retour au menu principal, l'application plante.
Objectif atteint	Partiellement 95% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Paramétrage

ID	2
Description	Tester si l'on peut modifier les paramètres de l'application.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Carte

ID	3
Description	Test du bon fonctionnement de la carte.
Commentaires	<ol style="list-style-type: none"> La position de l'utilisateur est de toute façon affichée.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : News

ID	4
Description	Test du bon fonctionnement de la carte.
Commentaires	<ul style="list-style-type: none"> 1. Les news ne sont pas toujours les mêmes que celles sur le site, apparemment les webservices ne fournissent qu'une partie des news pour l'application.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Directory

ID	5
Description	Test du bon fonctionnement de l'annuaire.
Commentaires	<ul style="list-style-type: none"> 1. Si on télécharge l'annuaire d'un Campus, et qu'ensuite on désire avoir l'annuaire d'une institution (qui est une sous-entité d'un campus) les informations de cette institution sont à nouveau téléchargées depuis internet. Ce qui signifie qu'on télécharge 2 fois les mêmes informations.
Objectif atteint	Partiellement 95% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Calendrier

ID	6
Description	Test du bon fonctionnement du calendrier.
Commentaires	<ul style="list-style-type: none"> 1. aucun bug détecté.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Cas de test : Résultats des examens

ID	7
Description	Test du bon fonctionnement de l'affichage des notes.
Commentaires	1. aucun bug détecté.
Objectif atteint	Complètement 100% <input checked="" type="checkbox"/>
Visa	Elias Medawar

Test unitaire

Test Suite '**ESIB_PAD_SOURCESTests**' started at 2011-08-07 12:48:23 +0000

Test Case '**-[ESIB_PAD_SOURCESTests testCarte]**' started.

Testing the Campus DAO: You must uninstall or reset cache of the application before testing

Loading async the campus data from internet

Recievied async the Campus DATA

Test Case '**-[ESIB_PAD_SOURCESTests testCarte]**' passed (2.025 seconds).

Test Case '**-[ESIB_PAD_SOURCESTests testDirectory]**' started.

Testing the Person DAO **for** the directory fonctionality: You must uninstall or reset cache of the application before testing

Loading async the directory of the cmapus CST from internet

List of person recievied

Test Case '**-[ESIB_PAD_SOURCESTests testDirectory]**' passed (17.585 seconds).

Test Case '**-[ESIB_PAD_SOURCESTests testNews]**' started.

Testing the News DAO: You must uninstall or reset cache of the application before testing

Loading async the news data from internet

Recievied async the news DATA

Test Case '**-[ESIB_PAD_SOURCESTests testNews]**' passed (4.727 seconds).

Test Case '**-[ESIB_PAD_SOURCESTests testPlaning]**' started.

Testing the Horraire DAO **for** the planning fonctionality: You must uninstall or reset cache of the application before testing

Loading async the plannong from internet

Recievied async the planning from internet

Test Case '**-[ESIB_PAD_SOURCESTests testPlaning]**' passed (18.140 seconds).

Test Case '**-[ESIB_PAD_SOURCESTests testSettings]**' started.

Testing the settings DAO

Test Case '**-[ESIB_PAD_SOURCESTests testSettings]**' passed (0.005 seconds).

Test Suite '**ESIB_PAD_SOURCESTests**' finished at 2011-08-07 12:49:25 +0000.

Executed 5 tests, with 0 failures (0 unexpected) in 55.508 (55.538) seconds

Listing 6.5: Log des test unitaires

Objectif atteint : Complètement 100 %

Test de fuite dans la mémoire

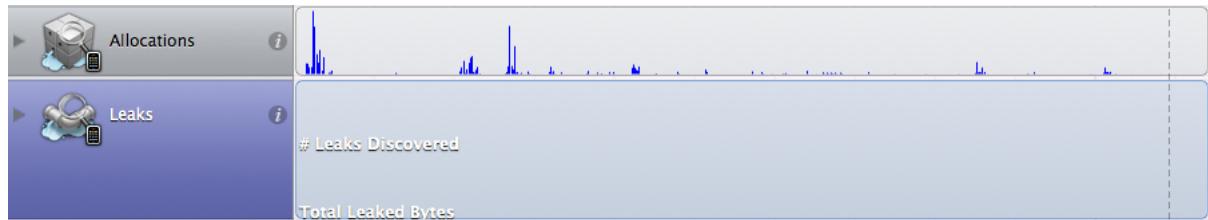


FIGURE 6.5: Résultat de l'analyse des Leaks à l'aide d'Xcode

Objectif atteint : complètement 100 %

Chapitre 7

Conclusion

1 Évolutions futures

Le résultat obtenu est un prototype fonctionnel et prêt à être déployé. Cependant plusieurs fonctions peuvent venir le compléter pour le faire évoluer. Voici les principales évolutions possibles

1.1 Accès à toutes les données via les services web

Pour l'instant la base de données contient moins d'information que ce que l'application supporte. Les coordonnées GPS des bureaux de personnes ainsi que ceux des salles de cours n'ont pas été répertoriées. Le planning de chaque personne ainsi que le résultat des examens ne sont pas encore accessibles. Cette évolution vise plus la partie "services web" de l'application mais elle est essentielle pour la suite de ce projet.

1.2 Nouvelles fonctionnalités

Les idées de fonctionnalités supplémentaires proposées dans le cahier des charges peuvent compléter ce projet. Les idées les plus intéressante seraient :

1. La fonction "Où es-tu ?" qui permet d'envoyer un message à une personne, et, si la personne le désire, ses coordonnées GPS seront envoyées en réponse.
2. Modification du calendrier. Permettre la modification(ajout/suppression/modification) d'événements dans le calendrier.

1.3 Version de l'application sur d'autres appareils

D'autres appareils mobiles ont les mêmes capacités que l'iPhone et l'iPad. Il serait aussi intéressant de rendre ce projet compatible avec les autres appareils.

1.4 Une version multilingue du projet.

1.5 Des cartes plus complètes des campus

L'idée est de fournir des cartes plus détaillées des campus avec des plans détaillés des étages. Ces plans complèteraient la carte actuelle de google map.

1.6 Accessible pour d'autres universités

Beaucoup d'autres universités ont besoin du même genres de fonctionnalités. L'idée ici est d'offrir la possibilité à d'autres universités d'utiliser cette application mais avec leurs propres services web pour la gestion des données.

2 Impressions personnelles

Pour moi ce projet ne représente pas uniquement le travail de 3 mois mais plutôt le résultat de 3 ans d'études passionnantes. Mon principal but était d'appliquer le plus grand nombre de concepts vu lors de mes études d'une façon exemplaire. Ce but est pour moi atteint et je suis satisfait de mon travail. Ce travail m'a permis de consolider la rampe qui me servira pour le départ dans la vie active et le monde du travail de l'ingénierie.

C'est avec grand plaisir que j'ai été confronté au mode de travail de l'USJ au Liban. J'ai trouvé que le mode de travail à l'USJ est plus orienté résultat que celui auquel j'étais confronté jusqu'à présent (à l'école ou lors d'autres expériences de travail). En ce sens je considère ce mode de travail plus proche de celui qui m'attend dans l'industrie.

Le seul point négatif est qu'il n'a pas toujours été évident de garder une motivation constante tout au long du projet. Ceci est dû à la durée(3 mois) de travail seul sur un projet.

J'ai pu traiter d'une manière assez profonde différents concepts de développement iOS et peut me considérer désormais comme jeune connaisseur dans ce domaine. C'est avec un petit chagrin au coeur que je termine ce projet passionnant car il reste des choses intéressantes à y développer.

3 Remerciements

Je tiens à remercier M. Dany Mezher en tant que responsable externe qui m'a suivi tout au long de ce travail. Il m'a soutenu autant sur le plan technique avec ses conseils d'expert dans le domaine du développement iOS que sur le plan social en m'accueillant généreusement et chaleureusement au Liban. J'espère que cette application pourra être utile à l'USJ.

Je remercie aussi mes deux responsables internes, Mme Elena Mugellini et M. Omar Abou Khaled qui eux aussi m'ont suivi tout au long de mon travail. Je remercie encore une fois M. Omar Abou Khaled qui a organisé et rendu possible cet échange au Liban.

Je remercie également mes deux experts, M. Marc Wuergle et M. Roland Marro.

Je remercie Mme Caroline Geissbühler pour la correction du rapport. Et plus particulièrement pour son soutien tout au long du projet.

Pour terminer, un merci particulier revient à ma famille au Liban qui m'a accueilli et soutenu et a ainsi rendu ces 3 mois de séjour au Liban plus agréable .

4 Conclusion

Je m'attendais à une plus grande compatibilité entre l'iPhone et l'iPad et en conséquence à passer moins de temps sur la partie implémentation de chaque itération.

La méthodologie de travail itérative a permis de garantir une bonne qualité du produit final. J'ai pu constater en découplant le temps à disposition en itérations l'utilité d'avoir des jalons et de fixer des objectifs intermédiaires.

Les objectifs fixés au début du projet ont été atteints. Le prototype rendu est fonctionnel, il peut être facilement finalisé et exploité. La documentation rendue avec le projet facilite la réutilisation et la continuation du projet par une autre personne. Le code rendu est commenté et peut être aisément retravaillé par des tiers.

Chapitre 8

Indexes

Références

- [1] Apple (auteur inconnu). Publication concernant la compatibilité iphone/ipad, 2011.
<http://devimages.apple.com/iphone/resources/introductiontouniversalapps.pdf>.
- [2] Java Sun (auteur inconnu). Le pattern de dao, 2011. <http://java.sun.com/blueprints/corej2eepatterns/Patterns/DataAccessObject.html>.
- [3] Wikipedia (auteur inconnu). Article décrivant le framework jpa, 2011.
http://en.wikipedia.org/wiki/Java_Persistence_API.
- [4] Wikipedia (auteur inconnu). Le modèle du cycle en v, 2011.
http://fr.wikipedia.org/wiki/Cycle_en_V.
- [5] fraca7 (pseudonyme auteur inconnu). Composant de calendrier compatible iphone, 2010. <https://github.com/fraca7/GCCalendar>.
- [6] Thomas Sarlandie. Programation iPhone OS3, volume 2. Eyrolles, 2010. Cahpitre 5 : Conception, ergonomie.
- [7] Ray Wenderlich. Introduction to mapkit on ios tutorial, 2011.
<http://www.raywenderlich.com/2847/introduction-to-mapkit-on-ios-tutorial>.

Glossary

Core Data Core Data is part of the Cocoa API in Mac OS X first introduced with Mac OS X 10.4 Tiger and for iOS with iPhone SDK 3.0.[2] It allows data organised by the relational entity-attribute model to be serialised into XML, binary, or SQLite stores. .(Source wikipedia)http://en.wikipedia.org/wiki/Core_Data . 60

EIA-FR École d'ingénieurs et d'architectes de Fribourg- Suisse(<http://eia-fr.ch>).
24, 25, 28

ESIB École Supérieure des Ingénieurs de Beyrouth- Faculté de l'USJ - Liban(<http://www.fi.usj.edu.lb/>). 19, 24, 25, 28

Git Git est un logiciel de gestion de versions décentralisée. C'est un logiciel libre créé par Linus Torvalds, le créateur du noyau Linux, et distribué sous la GNU GPL version 2.<http://fr.wikipedia.org/wiki/Git>. 26

GPS Le Global Positioning System (GPS) – que l'on peut traduire en français par « système de positionnement mondial » – est un système de géolocalisation fonctionnant au niveau mondial.Plus de détail sur wikipedia. 19

iOS iOS, anciennement iPhone OS, est le système d'exploitation mobile développé par Apple pour l'iPhone, l'iPod touch, et l'iPad..(Source wikipedia).[http://fr.wikipedia.org/wiki/IOS_\(Apple\)](http://fr.wikipedia.org/wiki/IOS_(Apple)) . 32

Objective-C L'Objective-C est un langage de programmation orienté objet réflexif. C'est une extension du C ANSI, comme le C++, mais qui se distingue de ce dernier par sa distribution dynamique des messages, son typage faible ou fort, son typage dynamique et son chargement dynamique.Aujourd'hui, il est principalement utilisé pour le développement d'application Mac OS X et son dérivé iOS pour le développement iPhone,iPad,iPod.(Source wikipedia). .Référence Apple sur l'objective-c :<http://developer.apple.com/documentation/Cocoa/Conceptual/ObjectiveC/ObjC.pdf> . 24, 25

SDD Software Design Description(IEEE 1016). Ce document contient la documentation concernant la conception et l'implémentation . 28

Skype Skype est un logiciel propriétaire qui permet aux utilisateurs de passer des appels téléphoniques via Internet. . .Site officiel :www.skype.com . 18

SPMP Software Project Management Plan est le document contenant toutes les informations concernant l'organisation d'un projet de développement de logiciel selon la norme IEEE 1058 . Norme disponible à cette adresse :<http://standards.ieee.org/findstds/standard/1058-1998.html> . 19, 27, 91

SRS Software Requirements Specification(IEEE 830). Ce document contient la documentation concernant la spécification et l'analyse. . 28

STD Software Test Documentation(IEEE 1016). Ce document contient la documentation concernant les tests effectués. . 28

SVN Subversion (en abrégé svn) est un système de gestion de versions, distribué sous licence Apache et BSD. Site officiel :<http://subversion.apache.org/> . 26

USJ Université Saint-Joseph à Beyrouth. 5 campus dont l'FI,1873 enseignants,500 membres du personnel et 12000 étudiants(<http://www.usj.edu.lb/>). 13, 24, 32, 33

XCode XCode est un environnement de développement pour Mac OS X.<http://fr.wikipedia.org/wiki/Xcode> . 60

Table des figures

3.1	Vue détaillée du planning du projet	19
3.2	Vue globale du planning du projet avec les milestones	22
3.3	Illustration du modèle de développement en V qui est appliqué à chaque itération.	23
3.4	Compatibilité entre composants et versions du logiciel. Les traits continus représentent une compatibilité à 100 %, les traits traitillés représentent une compatibilité partielle.	27
4.1	Vue globale des fonctionnalités du produit	31
4.2	Diagramme des cas d'utilisation de l'application	32
4.3	Wireframe illustrant les modifications des paramètres sur l'iPhone	37
4.4	Wireframe illustrant les modifications des paramètres sur l'iPad	37
4.5	Wireframe illustrant les fenêtres de l'affichage des cartes sur l'iPone	39
4.6	Wireframe illustrant les fenêtres de l'affichage des cartes sur l'iPad	39
4.7	Wireframe illustrant les fenêtres de l'affichage des news sur l'iPone	40
4.8	Wireframe illustrant les fenêtres de l'affichage des news sur l'iPad	41
4.9	Wireframe illustrant les fenêtres de l'affichage de l'annuaire sur l'iPone	43
4.10	Wireframe illustrant les fenêtres de l'affichage de l'annuaire sur l'iPad	44
4.11	Wireframe illustrant les fenêtres de l'horaire sur l'iPone	45
4.12	Wireframe illustrant les fenêtres de l'horaire sur l'iPad	46
4.13	Wireframe illustrant les fenêtres des résultats d'examen sur l'iPone	47
4.14	Wireframe illustrant les fenêtres des résultats d'examen sur l'iPad	47
5.1	Vue global de l'architecture du système	51
5.2	Diagrammes de composant du système	53
5.3	Illustration du fonctionnement du système de cache.	57
5.4	Interface dans X-Code permettant l'édition de la base de données	61
5.5	Diagramme de séquence du principe de la navigation	63
5.6	Diagramme de classe du composant MainView	64
5.7	Contenu du fichier MenuItemsParam.plist de configuration du menu pour la navigation	65
5.8	Diagramme de séquence concernant la lecture et la modification des paramètres	67
5.9	Diagramme de classe du composant Settings	68

5.10 Exemple de séquence concernant l'affichage de la carte	69
5.11 Exemple de séquence concernant la recherche d'un élément sur la carte	70
5.12 Diagramme de classe du composant Map	71
5.13 Diagramme de séquence illustrant l'appel asynchrone pour télécharger des données depuis internet	72
5.14 Exemple de séquence concernant l'affichage des news	73
5.15 Diagramme de classe du composant News	75
5.16 Illustration de la modification de l'apparence des cellules et plus précisément de la couleur du texte du titre.	76
5.17 Exemple de séquence concernant choix d'un filtre d'affichage et ensuite l'affichage d'une personne de l'annuaire	77
5.18 Diagramme de classe du composant News	78
5.19 Exemple de séquence concernant l'affichage de l'horaire pour une journée et l'affichage du détail de l'évènement sur la carte	82
5.20 Diagramme de classe du composant calendrier	83
5.21 Exemple de séquence concernant l'affichage de l'horaire pour une journée et l'affichage du détail de l'évènement sur la carte	85
5.22 Diagramme de classe du composant calendrier	86
 6.1 Vue globale de l'architecture du système	91
6.2 Résultat de l'analyse des Leeks à l'aide d'Xcode	98
6.3 Résultat de l'analyse des Leaks à l'aide d'Xcode	102
6.4 Résultat de l'analyse des Leaks à l'aide d'Xcode	105
6.5 Résultat de l'analyse des Leaks à l'aide d'Xcode	109

Listings

4.1	Exemple de code XML fournit au service informatique de l'USJ	34
4.2	Exemple de XML Schema fournit au service informatique de l'USJ	34
5.1	Téléchargement d'un fichier XML depuis internet de manière asynchrone et en transmettant les paramètres de la requête par POST	59
5.2	Parsing d'un fichier XML en Objective-c	60
5.3	Exemple de création de lecture et de suppression de données dans la base à l'aide de Core Data	61
5.4	Code d'écriture et de lecture dans un fichier plist.	65
5.5	Code d'enregistrement pour la notification de rotation des appareils.	66
5.6	Code de création d'un objet MKMapView et l'ajout d'une annotation.	72
5.7	Exemple de 2 animations à l'aide de la classe UIView. La première change la taille et l'emplacement d'un élément graphique et la deuxième change sa transparence.	79
5.8	Méthode de recherche dans une UITableView.	79
6.1	Log des tests unitaires	98
6.2	Log des tests unitaires	100
6.3	Log des tests unitaires	102
6.4	Log des tests unitaires	105
6.5	Log des tests unitaires	108

Chapitre 9

Annexes

1 Imprimées

1.1 Annexe A - directive du projet

Contient la directive distribuée au début du projet.

1.2 Annexe F Flyer

Le résumé du projet en allemand.

1.3 Annexe G Cahier des charges

Contient le cahier des charges établi au début du projet.

1.4 Annexe H Journal de bord

Le journal de bord.

2 le CD

Emplacement	Description
— CodeSource	Contient tout les ressources nécessaires pour le développement à l'aide de XCode.
	suite de la page suivante

	suite de la page précédente
Emplacement	Description
—AppDevCenterCertification	Contient les fichiers permettant d'accéder au compte développeur de l'EIF.
—ESIB_PAD_SOURCES	Code source de l'application.
—ESIB_PAD_SOURCES	
—Components	Contient les différents composants développés. La partie DAO de chaque composant est dans le dossier DAO.
—Calendrier	Composant Calendrier.
—Directory	Composant Directory.
—iPad	Partie graphique concernant l'iPAD.
—iPhone	Partie graphique concernant l'iPone.
—ExamResult	Composant qui affiche les résultats des examens.
—Map	Composant Map.
—Navigation	Composant Navigation avec le menu.
—News	Composant News.
—Settings	Composant qui permet le paramétrage de l'application.
—iPad	Partie graphique concernant l'iPAD.
—iPhone	Partie graphique concernant l'iPone.
—Utility	Dossier contenant des classes à but d'utilitaires utilisées et développées dans le projet.
—GCCalendar	version modifiée du composant https://github.com/fraca7/GCCalendar/.
—CoreDataEntities	Contient les entités de la base de données (Classe représentant une table dans la DB).
—DAO	Contient toutes les classes qui regroupent la logique métier et permet l'accès aux données.
—ESIB_PAD_SOURCES.xcdatamodeld	Contient le modèle de la base de données.
—Images	Contient toutes les images et logos utilisés par l'application.
—ESIB_PAD_SOURCESTests	Contient le code de test de l'application.
—desing	Contient les fichiers Photoshop CS5 de design de l'application.
—webServices	Contient le service web local écrit en php ainsi que les données.
—Documentation	Contient la documentation sous format PDF ainsi que le code Latex.
—makeRapportFinal, make	sont des fichiers bash permettant de générer la documentation (Compatible MAC OS X).
	suite de la page suivante

suite de la page précédente

Emplacement	Description
——Annexes	
——A_Directives	
——B_Plannigs	
——C_ProcesVerbaux	
——D_DiagrammesUML	
——E_XML_WEB_SERVICES	
——F_Flyer	
——G_Cahie_des_charges	
——H_Journal_de_bord	
——Source	Code latex pour la génération de la documentation
——CDC	Cahier des charges
——RapportFinal	Le rapport final.
——RapportRelease	Les différents rapports de release.
——SDD	
——SPMP	
——SRS	
——STD	
——comon	
——-figure	Contient toutes les figures et images utilisées dans la documentation.
——-logos	Les logos utilisés

Annexe A Directives du projet

Annexe F Flyer

Annexe G Cahier des charges

Annexe H Journal de bord