



INSA RENNES

DÉPARTEMENT INFORMATIQUE, 4^{ÈME} ANNÉE

Rapport de Pré-étude

Numérisation de documents en mobilité sur iPhone/iPad pour Mac OS X

Auteurs

Joanna CARTRON
Claire COHAN
Anne-Lyse GUENVER
Ewen LE GOURRIER
Yerima MALIKI
Nicolas MALOEUVRE
Bastien VANNIER

Encadrants

Bertrand COUASNON
Ivan LEPLUMEY

21 octobre 2011

Table des matières

Introduction	3
1 Contexte du projet	4
1.1 Attente	4
1.2 Lacunes et faiblesses des applications existantes	5
1.2.1 Les applications iPhone/iPad	5
1.2.2 De l'iPhone/iPad vers le Mac	7
2 Cahier des charges	9
2.1 Sur Ipad et Iphone	9
2.1.1 La prise de vue	9
2.1.2 Le transfert	9
2.1.3 Les modes d'utilisation	10
2.2 Sur Mac OS X	10
2.2.1 Réception de document	10
2.2.2 Partage des documents et intégration à Mac OS X	10
2.2.3 Traitement des objets reçus	10
3 L'environnement de développement	12
3.1 Un panel de technologies riches et innovantes	12
3.1.1 Des systèmes d'exploitations différents mais liés	12
3.1.2 La quête de la mobilité : iPhone/iPad	13
3.1.3 L'utilisation annexe de Linux	14
3.2 Outils et langages	15
3.2.1 Objective-C	15
3.2.2 L'environnement de développement : XCode	16
4 Gestion de projet	18
4.1 Outils de gestion	18
4.1.1 Un gestionnaire de documents	18
4.1.2 Le Calendrier	18
4.1.3 Un gestionnaire de version	18
4.2 Planification	19
4.2.1 Atos et l'IT Challenge	19
4.2.2 Élaboration de la planification	19
Conclusion	22
Bibliographie	24

Introduction

Aujourd'hui, la dématérialisation de documents ainsi que leur gestion en mobilité sont devenus nécessaires. Notre projet, intitulé « Numérisation de documents en mobilité sur iPhone/iPad pour Mac OS X » s'inscrit dans un objectif d'amélioration de l'usage de ces deux composantes pour l'utilisateur. Pour cela nous allons créer une première application destinée à la numérisation sur les supports iPhone/iPad. La seconde, développée pour Mac, aura pour but la réutilisation future de ces documents numérisés. Le tout dans un souci d'interopérabilité la plus poussée possible. Par l'intermédiaire de ce projet nous participons également au concours IT¹ Challenge organisé par l'entreprise Atos.

Dans un premier temps, nous confronterons les attentes de l'application à ce qui existe déjà en termes d'applications de numérisation. Par la suite nous établirons un cahier des charges plus précis que nous découperons en deux parties : l'application sur Mac OS X et celle sur iOS. Puis, nous aborderons des aspects plus techniques, à savoir les périphériques utilisés (iPhone, iPad) ainsi que notre environnement de développement. Enfin, nous exposerons nos outils de gestion de projet, notre première planification et les conséquences de notre participation à l'IT Challenge sur le calendrier initial.

Afin de mener à bien ce projet, nous sommes assistés au sein de l'INSA² par Bertrand Couasnon et Ivan Leplumey, et en contact auprès d'Atos avec Anne Benoit-Tissier de l'agence de Rennes.

1. Information Technology

2. Institut National aux Sciences Appliquées

Chapitre 1

Contexte du projet

1.1 Attente

Nos façons de communiquer ont été complètement transformées avec l'apparition d'internet, des réseaux sociaux, ou encore des Smartphones. L'informatique nomade s'invite désormais partout. Ce concept nouveau, dont l'essor est lié à l'avènement des Smartphones et Tablet, se caractérise par des équipements matériels et des infrastructures qui permettent les échanges d'information numérique. Désormais, un appareil n'est plus dédié à une fonction précise, mais cumule différentes actions possibles. La tendance est, en effet, à la mobilité, rapidité et polyvalence, comme on peut le voir pour les derniers Smartphones qui disposent aussi d'appareil photo, caméra, navigateur internet, etc. De plus, les applications pour mobiles offrent une multitude de possibilités et accentuent ce phénomène d'objet multitâches.

C'est dans ce contexte que Bertrand Couasnon et Ivan Leplumey, membres de l'équipe Imadoc [11] à l'IRISA¹, qui travaillent sur la numérisation de documents et la reconnaissance d'écriture manuscrite, nous ont proposé de travailler sur une solution de numérisation par téléphone. Cette dernière permettrait de transférer une photo vers un ordinateur, ou une tablette, sans intermédiaire. Il est à l'heure actuelle déjà possible de transférer des fichiers entre ces appareils, mais la procédure n'est pas automatique. Le but de notre application serait d'automatiser cette procédure afin qu'à la suite d'une prise de vue, le fichier soit disponible directement sur l'ordinateur ou la tablette, mais puisse aussi être personnalisé (annotations, OCR², édition, etc).

Les supports papiers sont de moins en moins utilisés, il faut les stocker, les ranger alors qu'un document numérique prend moins de place et se retrouve simplement. Par exemple, en réunion, une personne qui prend des notes avec un ordinateur peut numériser un papier distribué ou un schéma dessiné sur un tableau avec son téléphone et le recevoir instantanément sur son Mac. La rapidité de l'action devrait lui permettre de ne pas perdre le fil de la présentation. Le cliché pourra être modifié, recadré, annoté, envoyé à un traitement de reconnaissance de caractères.

Apple [14], marque à l'image novatrice, dynamique, reconnue pour l'ergonomie et l'esthétique de ses produits, ne propose pas de processus d'échange de documents entre différents équipements sans passer par un intermédiaire, de manière totalement confidentielle. Nous allons développer notre

1. Institut de Recherche en Informatique et Systèmes Aléatoires

2. Optical Character Recognition

solution sur cet environnement afin de numériser avec transparence des documents avec l'iPhone ou l'iPad pour les envoyer vers un iPad ou un Mac. Dans cet environnement, il existe désormais *iCloud* [3] qui permet de synchroniser différents appareils Apple entre eux. Cependant cette solution, nous le verrons plus en détails, n'est pas suffisante pour aboutir à un transfert transparent.

Notre solution permettra donc de numériser simplement des documents dans un objectif principal de simplicité, en effet, l'utilisation, la connexion, la synchronisation et l'intégration des documents côté Mac se devront d'être le plus simple possible. D'autres fonctionnalités optimiseront le processus, ainsi, il sera possible d'annoter un cliché, et ce à partir de n'importe lequel des trois supports. La photo doit être de bonne qualité, de ce fait une fonction permettant de recadrer l'image semble presque indispensable. Il faudrait aussi gérer les documents numérisés comme par exemple grouper des photos dans un PDF, insérer l'image dans un document texte etc. Une reconnaissance automatique de document sera aussi mise en place via un service web, afin de transformer un document manuscrit en fichier texte éditable par la suite. Comme vous le constatez, il est possible de personnaliser et d'ajouter de nombreuses fonctionnalités à une application dont le but premier est de permettre la numérisation rapide et efficace de documents, mais il faut toujours garder à l'esprit l'importance de l'ergonomie et de la facilité d'utilisation.

1.2 Lacunes et faiblesses des applications existantes

Nous avons cherché à savoir si les applications que nous avons à réaliser lors de notre projet n'existaient pas déjà. Nous avons donc fait des recherches sur les applications de numérisation existantes et ensuite nous avons procédé au test des applications se rapprochant le plus de nos besoins. Nous avons surtout testé des applications sur iPhone. En effet, les applications sont quasiment identiques entre leur version iPhone et iPad. Sous iPad elles exploitent surtout la plus grande largeur de l'écran.

1.2.1 Les applications iPhone/iPad

Nous avons trouvé un certains nombres d'applications de numérisation, certaines plus fonctionnelles que d'autres ou bien plus ergonomiques mais aucune ne correspondait parfaitement aux attentes du projet.

Nous avons notamment distingué plusieurs catégories d'applications.

Des applications centrées sur la numérisation de documents et peu sur le partage

L'application *TurboScan* permet de numériser des documents variés et de créer des fichiers PDF multipages à partir d'un ensemble de documents numérisés. Elle permet également le traitement des documents comme le recadrage, la modification de la luminosité, etc.

Cependant, si vous ne désirez pas envoyer votre document par email ou par connexion directe de votre périphérique, la seule solution qui vous est proposée est le partage via un serveur externe (*Evernote*, *iCloud*, etc). Cette solution n'est pas rassurante pour des documents sensibles : nous ne savons pas si le service est assez sécurisé et nous n'avons pas la maîtrise sur ce dernier.

Des applications complètes mais peu ergonomiques

Des applications connues comme *DocScanner*, *JotNot* et *Genius Scan* 1.1 sont très performantes mais peu ergonomiques à l'usage. En effet, dans un contexte de mobilité la numérisation doit être rapide et le nombre de paramétrages, effectués après la prise de vue, limité. Ces applications sont pourtant très proches des besoins en fonctionnalités de notre projet mais elles sont peu pratiques et lentes.



FIGURE 1.1 – La numérisation d'un tableau : L'image est recadrée et remise droite

La numérisation dans l'application *JotNot* 1.2 prend beaucoup de temps car il y a un ajustement automatique des ombres, des contrastes et de la balance des blancs. De plus l'application détecte automatiquement le document dans l'image, ce qui supprime les informations inutiles (présence de stylo, tasse de café, etc). Ce traitement est trop long et cela ne convient pas au contexte de mobilité.

Au cours de nos tests, nous nous sommes rendu compte qu'il fallait passer par de nombreuses étapes successives à partir de la prise de vue pour aboutir au partage du document numérisé. Notre projet se situe donc principalement dans un axe d'amélioration de l'ergonomie et de la rapidité de la numérisation. Notre application devra être simple à utiliser, que ce soit lors de la connexion avec d'autres périphériques, lors de la synchronisation ou lors de l'intégration du côté Mac.

Nous avons repéré entre autres deux fonctionnalités intéressantes pour notre projet. Avec certaines applications (*JotNot*, *Genius Scan*, etc), il est possible d'accéder à nos fichiers grâce au Wi-Fi et par un serveur interne (comme un lecteur réseau partagé sur réseau Wi-Fi). Il faut alors mettre en marche le serveur Wi-Fi de *JotNot*, ouvrir un navigateur internet sur l'ordinateur et se rendre sur l'adresse indiquée dans l'application. Cependant l'ordinateur et le périphérique doivent être connectés sur le même réseau Wi-Fi. Dans notre projet nous allons justement exploiter le partage en Wi-Fi et peut être réutiliserons nous ce mode de fonctionnement.

On peut aussi parler de la présence dans l'application *DocScanner* d'une fonction de reconnaissance optique de caractères (ROC ou OCR en anglais) embarquée qui permet d'extraire le texte



FIGURE 1.2 – Le mode de partage Wi-Fi de JotNot

d'un document. Dans notre sujet il nous est justement demandé de prévoir un traitement de reconnaissance à distance. Cependant nous aimerions qu'il ne soit pas inclus dans l'application et qu'il puisse reconnaître le contenu de documents manuscrits. L'utilisateur pourra donc sélectionner le web-service de son choix selon le type de documents à traiter.

Des applications trop spécialisées

Certaines applications de numérisation sont uniquement centrées sur un contexte d'utilisation. On peut citer *Card Scanner* qui permet de scanner des cartes de visites et d'importer les informations de contact directement dans le carnet d'adresses de l'iPhone. On peut aussi parler de *iNoteDeFrais* qui permet aux professionnels de classer leurs justificatifs de frais rapidement et efficacement. Cette application est très fonctionnelle mais la numérisation des justificatifs de frais reste sommaire et aucun traitement d'images n'est proposé. De plus le partage se limite aux emails ou bien à l'utilisation de *iCloud* à côté. Ces applications sont très performantes dans leur usage mais elles ont chacune leur mode de fonctionnement propre. De même, elles ne sont pas, dans le cas de *Card Scanner* par exemple, conçues pour faire autre chose que de la numérisation de cartes de visites. Nous voudrions donc que notre application soit fonctionnelle et généraliste, le tout regroupé dans une interface unique et ergonomique.

1.2.2 De l'iPhone/iPad vers le Mac

Au niveau du Mac, on peut distinguer deux catégories d'applications, celles de traitement d'images et celles de transfert avec l'iPhone/iPad.

Applications de traitement de documents numérisés

Il existe également une application *DocScanner* pour Mac qui permet le traitement d'images numérisées. Elle dispose de fonctions de conversion d'images en document PDF ainsi que de fonctions de correction automatique de la perspective, de l'orientation, du bruit... Il est possible de glisser-déposer des images dans *DocScanner*, mais aussi de l'application vers le bureau ou vers d'autres logiciels. Les documents sont optimisés pour l'impression et peuvent être traités par une reconnaissance de caractères. Cependant il existe deux points négatifs à cette application, pouvant amener à une insatisfaction des utilisateurs : l'application est peu intuitive et aucune communication avec l'iPhone n'est prise en charge dans ce procédé.

Il existe beaucoup d'autres applications liées au traitement d'images, mais aucune autre ne semble spécialiste des documents numérisés ni présenter une liaison pour échanger facilement avec un périphérique iOS.

Applications de transfert de fichiers

Bien sûr, nous ne traiterons pas des applications nécessitant une liaison câblée, puisqu'un de nos objectifs primordiaux est la rapidité. Les applications existantes passent toutes par un système d'hébergement de documents.

Dans la première catégorie, on peut évoquer *Dropbox* [9], qui est un service de stockage et de partage de fichiers en ligne. *Dropbox* est accessible depuis n'importe quel navigateur et les fichiers partagés sont copiés sur le serveur *Dropbox*. On peut aussi parler de *Showtime*, qui permet de créer une bibliothèque en ligne avec ses images, à laquelle on peut ensuite accéder à partir de n'importe quel équipement disposant d'une connexion. Cependant, ces deux services passent par un intermédiaire et le transfert de documents nécessite quelques manipulations : connexion sur l'iPhone puis sur le Mac etc. Par conséquent ils ne correspondent pas bien à l'ergonomie que l'on souhaite et ils ne permettent pas d'effectuer des transferts en toute transparence. Depuis peu, *iCloud*, disponible avec iOS 5, permet le partage de fichiers entre plusieurs appareils de l'environnement Apple. *iCloud* stocke automatiquement vos contenus de façon à ce qu'ils soient toujours accessibles depuis votre iPhone, iPad, iPod Touch, Mac et PC sans la moindre synchronisation, ni la moindre gestion. Ce type de transfert pourra donc être proposé dans le cadre de notre application, cependant la transmission n'est pas instantanée et la confidentialité n'est pas garantie. De plus, il faut toujours trouver une solution dans le cas où l'on ne dispose pas de connexion.

En conclusion, aucune des applications existantes, que ce soient pour iPhone ou Mac, ne dispose d'un service instantanée et confidentiel de transfert, d'un système d'annotations et n'offre le choix de choisir son propre système de traitement d'images.

Chapitre 2

Cahier des charges

Le projet a pour objectif d'établir une communication transparente entre un Mac, un iPhone et un iPad pour le partage de documents. Dans la suite nous parlerons principalement de l'iPhone mais l'iPad est également concerné.

2.1 Sur Ipad et Iphone

Le noyau de l'application sera le même sur les deux périphériques, seule l'interface graphique sera différente du fait de la différence de dimension. Le principal atout de cette application est d'être dotée d'une ergonomie des plus intuitives. Dans cette application, il se dégage trois grandes fonctionnalités : la prise de vue, le transfert vers le mac et les modes d'utilisation que nous développerons ci-après.

2.1.1 La prise de vue

Elle consiste à prendre en photo un document papier, un tableau, une photo ou tout autre support. L'utilisateur pourra spécifier un mode d'utilisation auquel cas différents traitement prédéfinis seront effectués en fonction de ce mode. On pourra prendre plusieurs photos à la suite si le support est trop grand afin d'en faire un document de plusieurs pages. Un aperçu sera disponible avant sa sauvegarde, puis si le mode « utilisateur avancé » est spécifié, l'application proposera différents traitements. L'application aura son propre système de fichiers, où l'utilisateur pourra consulter les documents numérisés.

2.1.2 Le transfert

Une fois le cliché réalisé et traité, il est visible depuis l'ensemble des périphériques (Mac, iPhone ou iPad). La synchronisation se fera par Wi-Fi ou Bluetooth, suivant la possibilité de ces réalisations. Dans le cas du Wi-Fi, une connexion ad-hoc pourrait être la solution. Ce transfert devra être transparent, confidentiel et instantané. Le transfert pourra être envisagé avec *iCloud*, tout en garantissant la sécurité des données. L'utilisateur pourra choisir de supprimer automatiquement le document sur l'appareil lors du transfert.

2.1.3 Les modes d'utilisation

L'utilisateur pourra choisir entre plusieurs modes, en fonction de ses besoins. Un mode rapide permettant une simple numérisation, un mode utilisateur avancé pour lequel des options de traitements tels que le recadrage automatique, l'annotation, l'utilisation d'un web-service spécifique tel que l'OCR seront disponibles. Il choisira également avec quel(s) périphérique(s) il souhaite se synchroniser, et par quels moyens technologiques. L'utilisateur pourra aussi définir ses propres modes afin de prédéfinir tous ces traitements et les rendre automatiques. On peut aussi penser à synchroniser un mode avec une application tierce, par exemple le mode « note » pourrait permettre de prendre en photo un texte afin de le rentrer directement dans l'application *Notes* de l'iPhone.

2.2 Sur Mac OS X

L'application Mac doit, au travers d'une interface intuitive, permettre l'utilisation des objets créés par l'iPhone.

2.2.1 Réception de document

Après leur création sur l'iPhone, les données sont transférées sur le Mac. Ce transfert se fait de façon transparente et sans intervention de l'utilisateur. Ce comportement est permis par la reconnaissance de l'iPhone par l'application via une synchronisation WiFi ou Bluetooth. Lors de la réception de nouveaux fichiers, l'application alertera l'utilisateur de la disponibilité de ceux-ci.

Un accès rapide aux derniers fichiers numérisés ainsi que des informations sur l'avancement du chargement ou l'état de la synchronisation pourront être fournis.

2.2.2 Partage des documents et intégration à Mac OS X

La réception d'un document entraîne sa copie dans le système de fichier de l'application. Les fichiers disponibles sont affichés dans l'interface de l'application de façon à être disponibles pour leur utilisation future. Les fichiers sont stockés dans un dossier spécifique à l'application ce qui permet de les manipuler sans lancer l'application. Les annotations liées aux fichiers permettent leur classement à l'intérieur de l'application. De cette façon, il sera facile d'effectuer une recherche pour trouver les documents appropriés.

Un point primordial de l'application est l'intégration au système d'exploitation et la collaboration avec d'autres applications. Ainsi, à partir d'applications tierces telles que Notes, on pourra créer des liens vers les documents numérisés. Cette fonctionnalité pourra d'ailleurs tirer parti du fonctionnement des liens dans Mac OS X qui restent valides lorsque le fichier est déplacé. De plus, des modes de prises de vue existant sous iPhone pourront mener à des traitements spécifiques. Ainsi l'application pourra par exemple créer automatiquement un rendez-vous dans *iCal* à partir d'un document numérisé dans le mode « A faire ».

2.2.3 Traitement des objets reçus

L'application en plus de rendre disponible les fichiers numérisés par l'iPhone permet également de les traiter. Ainsi, on pourra recadrer un document ou modifier les annotations qui lui sont liées.

Ces annotations seront rendues disponibles et utilisables par d'autres applications.

Les web-services disponibles sur l'iPhone seront également accessibles sur le Mac. Parmi ceux-ci, on pourra trouver la reconnaissance optique de caractère (OCR).

Chapitre 3

L'environnement de développement

Dans le cadre du projet, nous avons eu à nous intéresser de plus près à l'environnement dans lequel nous allions devoir évoluer. Notre projet se basant principalement sur des technologies Apple, vous trouverez ci-dessous un descriptif du matériel qui sera utilisé ainsi que des langages et outils mis à notre disposition.

3.1 Un panel de technologies riches et innovantes

Apple, fondé en 1976 par Steve Jobs et Steve Wozniak, est aujourd'hui la société possédant le plus grand chiffre d'affaires au monde (49,2 milliards). Concurrent historique de Microsoft, Apple a finalement réussi à se hisser et à se tailler une place conséquente dans le monde des nouvelles technologies. En mai 2010, ce dernier a même dépassé son rival historique en termes de valeur boursière (259 milliards contre 171 pour Microsoft) [1]. De part ses aspects créatifs, Apple a su développer et diversifier son panel d'activités. D'une part avec la création de technologies innovantes en avance sur leur temps telles que l'iPhone ou l'iPad, et d'une autre part avec la consolidation et l'amélioration constante apportées à leurs systèmes d'exploitation dédiés.

3.1.1 Des systèmes d'exploitations différents mais liés

Nous pouvons distinguer l'existence de deux systèmes d'exploitation dans l'environnement Apple équipant des standards matériels différents.

Mac OS X

Mac OS X [20] est le système d'exploitation propriétaire équipant l'ensemble des ordinateurs produits par Apple à ce jour. Il fait partie de la famille des systèmes d'exploitation UNIX [23], famille née en 1969 lors de la création d'un système d'exploitation multi-tâches et multi-utilisateurs conceptuellement ouvert. Aujourd'hui Mac OS X se présente dans sa version 10.7.2 dit « Lion » disponible sur le marché depuis le 20 juillet 2011. Cette dernière version apporte de nombreuses nouveautés, dont certaines inspirées directement du système d'exploitation présent sur l'iPad : iOS.

iOS

Quant à lui, iOS [16] est le système d'exploitation propriétaire développé pour l'iPhone, l'iPad et l'iPod Touch entre autres. Il est dérivé de Mac OS X dont il partage les fondations. iOS comporte quatre couches d'abstraction, similaires à celles de Mac OS X : une couche « Core OS », une couche « Core Services », une couche « Media » et une couche « Cocoa ». Il se décline depuis le 12 octobre 2011 dans sa dernière version dite « iOS 5 ». Cette dernière apporte, en plus d'une puissance accrue, un certain nombre de nouveautés par rapport à son aînée parmi lesquelles nous pouvons trouver Siri, un composant lié à la reconnaissance vocale, ou bien encore *iCloud*, basé sur le système de « Cloud Computing » pour le partage instantané et transparent de données diverses et variées entre divers matériels Apple. Cette technologie offre des fonctionnalités se rapprochant fortement de la solution technologique que nous aurons à développer, il conviendra donc de l'étudier en profondeur par la suite.

L'intégralité du développement que nous aurons à fournir sur Apple s'effectuera sous Mac OS X. L'une des deux applications devra être exécutable sur Mac OS X tandis que l'autre aura vocation à être utilisable sur iOS.

3.1.2 La quête de la mobilité : iPhone/iPad

Depuis novembre 2007, Apple s'est lancé sur le marché de la mobilité avec en première ligne le développement du désormais célèbre iPhone. Vient ensuite l'iPad en avril 2010 avec la volonté de conquérir le marché des tablettes tactiles destinées au grand public.

iPhone

L'iPhone [18] est une famille de smartphones dont la mise à jour et l'évolution sont constantes et régulières. De ce fait l'iPhone se décline depuis le 14 octobre 2011 dans sa version « iPhone 4S » (4^e génération) embarquant un système d'exploitation dédié : iOS 5. Ce smartphone possède une interface utilisateur axée principalement sur le multi-touch [21]. Il s'agit d'une technique d'interaction tactile entre l'humain et la machine. Ceci permet à l'utilisateur d'interagir sur son écran avec plusieurs doigts et de créer une interaction riche via des gestes intuitifs tels que l'écartement de deux doigts pour le zoom par exemple. En plus des fonctionnalités de base, l'iPhone est également doté d'une caméra permettant de prendre des photos (8 Méga pixels pour l'iPhone 4S), fonctionnalité indispensable dans le cadre de la numérisation de documents en mobilité. De plus, la présence d'un client Internet (pour naviguer sur le Web ou consulter son courrier électronique) et de technologies telles que le Wi-Fi (802.11n) et le Bluetooth se révéleront être des solutions techniques de choix dans le cadre de l'interopérabilité entre les divers périphériques Apple.

iPad

Quant à lui, l'iPad [17], tablette orientée vers les médias tels que les livres, films, musiques, jeux, mais aussi vers l'Internet, vient se positionner à mi-chemin entre un smartphone de type iPhone et un ordinateur portable. Depuis mars 2011, l'iPad 2^e génération est venu remplacer l'iPad premier du nom sorti un an auparavant dont la principale innovation réside dans l'ajout d'une caméra s'avérant

être de qualité discutable. D'ailleurs ce manque de qualité et de performance concernant cette dernière pourra s'avérer être un frein et sources de problèmes, notamment en matière de traitements d'images, dans le cadre de notre projet. Mis à part ce détail, l'iPad est doté de technologies similaires à l'iPhone telles que le Wi-Fi ou le Bluetooth.

Du point de vue des marchés, nous pouvons voir dans un premier temps que l'iPhone se trouve être en concurrence directe avec les smartphones équipés d'Android (voir figure 3.1), tandis que l'iPad domine très largement le marché des tablettes tactiles (voir figure 3.2). Apple est donc l'un des acteurs ayant une place prépondérante sur le marché des nouvelles technologies.

Smartphone market share

March '11, Nielsen Mobile Insights, National

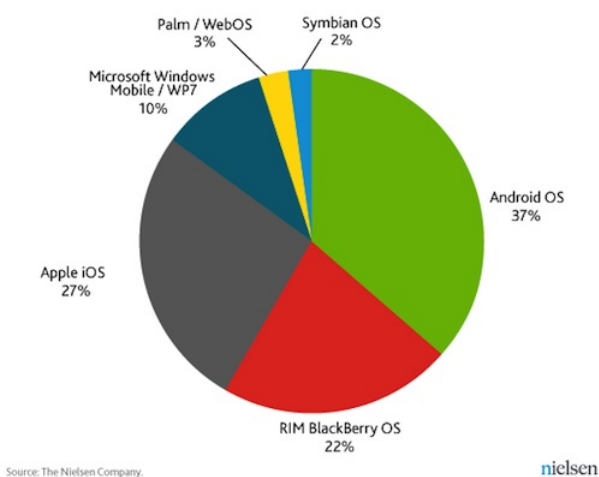


FIGURE 3.1 – Part de marché iPhone

Tablet ownership and usage

Tablet Brands Owned - Primary

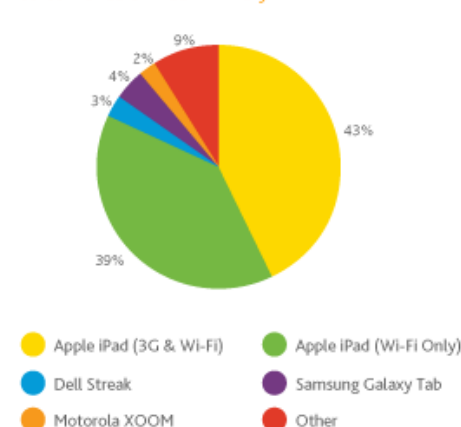


FIGURE 3.2 – Part de marché iPad

3.1.3 L'utilisation annexe de Linux

La mise en œuvre du projet demande également de manipuler un service Web sous Linux [19] afin d'envoyer des documents vers ce dernier à des fins de traitements. Pour rappel, Linux est un système d'exploitation faisant partie de la famille des systèmes d'exploitation UNIX, tout comme Mac OS X. Ce dernier offre une alternative libre aux systèmes propriétaires tels que Microsoft Windows ou Mac OS X. Il permet le fonctionnement de tous types d'applications (navigateurs web, clients de messagerie, etc). Il nous faudra donc utiliser les technologies relatives à cette plate-forme afin d'utiliser un service Web nous permettant d'accéder à différentes méthodes de traitement de documents développées par l'équipe Imadoc de l'IRISA ou par d'autres laboratoires.

3.2 Outils et langages

3.2.1 Objective-C

Pourquoi utiliser Objective-C ?

Objective-C [22] n'est bien sûr pas le seul langage utilisé pour programmer sous Mac OS mais c'est le langage que préconise Apple, il évolue d'ailleurs grâce à Apple. En utilisant ce langage, nous avons plus de chances que notre application puisse se maintenir facilement et qu'elle puisse utiliser les API compatibles.

L'Objective-C est principalement utilisé par deux systèmes d'exploitation : Mac OS X et GNU mais c'est surtout grâce à Apple qu'il s'est fait connaître. Depuis 2009, sa popularité s'est accrue de façon impressionnante ; en septembre 2011, il est arrivé à la sixième place dans le classement Tiobe (classant les langages en fonction du nombre de recherches effectuées sur le web). Réaliser ce projet nous permettra donc également de nous former à un langage dont les perspectives d'avenir semble prometteuses.

Qu'est ce que Objective-C ?

L'objective-C est un langage de programmation objet, c'est une extension du C ANSI nées dans les années 1980, tout comme le C++. Il est basé sur la bibliothèque de classes Cocoa pour Apple. Il a choisit la voie du dynamisme de Smalltalk dont il emprunte aussi une partie de la syntaxe. L'organisation du code source est semblable au C avec un fichier header « .h » et le code dans un fichier « .m ». Tous les objets dérivent d'une classe racine, comme beaucoup de langage objet, il s'agit ici de « NSObject » pour l'utilisation avec Cocoa. Le langage permet également de pointer tout objet grâce à un même type id qui ne nécessite pas de *cast*.

Pour programmer en Objective-C, il est important de connaître un minimum de vocabulaire et principalement de faire la différence entre les fonctions et les méthodes :

- une donnée d'instance correspond à un attribut,
- une méthode correspond à une fonction membre,
- une fonction correspond elle à une fonction de type C n'ayant pas accès aux données d'instance, ce qui ne l'empêche pas d'être liée à l'objet.

La syntaxe des méthodes peut sembler assez déroutante au premier abord puisque les appels de méthodes se font par envoi de messages en plaçant entre crochets l'objet puis le message ([objet methode :argument]). La syntaxe peut également se rapprocher du langage humain grâce à la présence d'étiquettes comme présenté ci-dessous :

```
/* Declaration d'une methode */
- (void)eatWith:(NSString)name andSpeakAbout:(NSString)subject ;
/* Exemple d'appel de la methode sur un objet */
[Paul eatWith:Pauline andSpeakAbout:Subject];
```

L'API Cocoa

Cocoa [15] est l'API¹ créée par Apple pour le développement orienté objet sur Mac OS X et donc notamment pour Objective-C. C'est un ensemble de classes permettant de programmer de façon native sous Mac OS X (et iOS avec Cocoa Touch). Elle évolue au fur et à mesure en fonction des besoins d'Apple pour ses OS, faisant ainsi d'elle une des plus active du marché.

3.2.2 L'environnement de développement : XCode

Présentation

XCode [24] est un environnement de développement pour Mac OS X offrant une suite logicielle (graphiques, audio, etc). Il permet de programmer sous différents langages tel que C, C++, Objective-C. Fourni en standard avec chaque Mac, il permet de développer des applications pour Mac, mais aussi pour iPhone et iPad. Il intègre Cocoa et représente ainsi un environnement de développement productif et facile à utiliser. C'est d'ailleurs avec cet outil qu'Apple a développé Mac OS X et iOS.

Son contenu

L'environnement de développement XCode contient :

- **un IDE** permettant une autocomplétion de code avancée
- **Interface Builder**, permettant de créer des interfaces utilisant les palettes, collections, widgets instanciés par Apple et également d'en faire dériver nos propres composants. L'interface créée est enregistrée dans un fichier (.nib ou .xib) où les objets sont archivés prêts à être exécutés. Ainsi, contrairement à la majorité des outils de création d'interface graphique, aucune génération de code n'est nécessaire pour construire l'interface.
- **Apple LLVM compilateur** : plus que compiler notre code, ce compilateur évalue aussi constamment le code et nous indique ainsi les erreurs et propose des solutions. Il compile le code deux fois plus vite que GCC, et produit également des applications s'exécutant plus rapidement.
- **Instruments** est un analyseur de performance et de comportement. Il permet, une fois l'application lancée sur un iPhone ou l'iPod Touch branché à l'ordinateur, d'observer en temps réel ses performances au niveau du processeur, mais également, par exemple, du moteur graphique ou de l'accéléromètre. Il traque ainsi les goulets d'étranglement des applications pour Mac OS X et iOS.
- **iOS simulator**, permet de tester son application virtuellement sur son Mac. On peut également simuler le multi-touch si l'ordinateur est pourvu de trackpad.

Améliorations ajoutées par la version 4

Depuis juin 2010, une version 4 est sortie. Elle a cassé la compatibilité avec de nombreux anciens systèmes, incluant toutes les architectures PowerPC, les SDKs pour Mac OS X Tiger (10.4) et

1. Application Programming Interface

Leopard (10.5), ainsi que tous les SDKs iOS antérieurs au 4.3. Il est toujours possible de produire des versions binaires d'applications pour des plateformes plus anciennes mais de façon limitée.

La version 4 consolide les outils d'édition de l'IDE et de Interface Builder dans une seule et même application, elle apporte surtout une amélioration au niveau de l'ergonomie. Elle offre entre autre une nouvelle zone de travail regroupant tous les dossiers de travail régulier, un nouveau débogueur LLDB qui permet d'afficher une trace de la pile lors de l'exécution et de zoomer sur la partie en cours de débogage.

Accessibilité

Pour pouvoir développer son application pour Mac OS X ou iOS, il faut avant tout posséder un Mac. Pour pouvoir mettre son application sur l'appStore ou même tester son application sur un iPhone ou sur iOS Simulator, il faut posséder un compte développeur. Dans le cadre d'une licence universitaire, l'inscription est gratuite, sinon elle coûte 99,00€ par an. Une fois le compte obtenu, on peut développer autant d'applications que l'on souhaite. XCode est en téléchargement gratuit depuis le passage sous Mac OS X Lion mais avant son coût s'élevait à 3,99€ pour le grand public.

Les avis sur XCode 4 sont assez variables, son architecture est très différente des anciennes versions et déroute souvent les utilisateurs des versions précédentes. Tout le monde semble apprécier les nouvelles fonctionnalités et l'ergonomie, notamment l'intégration de Interface Builder à l'IDE. Cependant, il semble souffrir de bugs et manquer de performance dans certains cas. XCode 4 étant l'avenir de XCode, il est quand même conseillé de l'utiliser et puisque les projets restent compatibles avec XCode 3, nous pourrons changer de version en cours de projet si nous rencontrons trop de difficultés.

Chapitre 4

Gestion de projet

4.1 Outils de gestion

En plus des outils de gestion tel que MS Project, nous avons jugé utile de créer des espaces de travail commun.

4.1.1 Un gestionnaire de documents

Pour travailler sur le projet, nous avons besoin d'un espace sur lequel nous pouvons partager des documents tel que des synthèses de réunions ou de recherches. *Google docs* nous permet de le faire facilement puisqu'il se gère comme une arborescence de fichiers et il nous permet également de pouvoir écrire simultanément, ce qui est parfois pratique lors de réunions.

4.1.2 Le Calendrier

Nous avons également créé un calendrier commun qui nous permet d'inscrire les réunions prévues ainsi que les heures pendant lesquelles chacun de nous a travaillé sur le projet. Il est ainsi facile de comptabiliser le nombre d'heures passées sur chaque type de tâches tout au long du projet. Nous avons préféré utiliser le Calendrier Google à celui proposé avec notre ENT¹ INSA car nous utilisions déjà *Google docs* pour gérer nos documents, il nous semblait donc plus logique de rester sur la plateforme Google.

4.1.3 Un gestionnaire de version

Pour le développement des applications nous aurons besoin d'un gestionnaire de version, celui-ci nous permettra de développer chacun de notre côté une partie du code, l'outil se chargeant de gérer la mise en commun. Il permet également de sauvegarder les différentes versions du projet, nous permettant ainsi de restaurer une ancienne version si nécessaire. Afin de pouvoir gérer nos fichiers à l'INSA, nous avons choisis d'utiliser le SVN proposé par l'INSA avec la *Forge*. On pourra également l'intégrer à XCode 4 car il permet d'y intégrer *Subversion*. La *Forge* de l'INSA propose également

1. Environnement Numérique de Travail

un calendrier, un diagramme de Grantt, un wiki, un suivi de demandes et un suivi du temps qui passe. N'ayant pas encore accès à notre compte, nous n'avons pas pu tester ces différents outils. Mais si ceux-ci s'avèrent efficace, nous les ajouterons à nos outils de gestion et nous basculerons peut-être également notre Calendrier Google vers celui de la forge.

4.2 Planification

4.2.1 Atos et l'IT Challenge

Une nouvelle dimension s'est ajoutée à notre projet, puisque nous avons décidé de participer au concours IT Challenge de Atos. Trente-six écoles dans le monde concourront autour du thème de l'innovation et de la mobilité. Autrement dit, une participation est recevable s'il s'agit d'une application mobile inédite et utile au quotidien. Notre projet s'inscrit dans ce cadre et c'est d'un commun accord qu'on se lance dans ce challenge. Cependant, le concours nous impose des dates limites, présentations et dossiers qui nous impose donc de prendre en compte notre participation dans notre planification. En effet, au premier semestre, seul un dossier de participation (business plan, description de l'application, son utilisation, ses fonctionnalités) est à rendre avant décembre, avant de savoir si notre candidature fait partie des quinze sélectionnées. Si nous sommes élus, cela signifie qu'à la mi-avril nous devons présenter un prototype de notre application à un jury, soit avoir une application qui fonctionne, c'est à dire plus d'un mois avant la date de livraison de l'INSA. Cette contrainte n'est donc pas négligeable, puisqu'elle risque d'influencer nos façons de travailler et divise donc notre phase de développement en deux : une pour l'IT challenge, puis une autre pour les finitions et améliorations avant la livraison à l'INSA.

4.2.2 Élaboration de la planification

Un projet, dès qu'il est suffisamment complexe et a des enjeux importants, consiste en un ensemble d'actions visant un résultat défini, connu et mesurable. Notre projet, aux multiples objectifs (IT Challenge et INSA), nécessite que l'on suive une démarche visant à organiser de bout en bout le bon déroulement de celui-ci. C'est dans cette optique que nous envisageons l'utilisation d'une méthode agile de gestion de projet afin d'avoir un suivi parfait de la réalisation du projet.

Notre première planification se structure en quatre temps qui s'articulent de la façon suivante. La phase initiale est l'analyse, elle consiste en la rédaction des rapports de pré-étude, de spécification et de planification. Durant cette période, nous allons aussi nous familiariser avec notre projet, prendre en main les outils de développement, se former aux langages de programmation dont nous allons avoir besoin etc. Une soutenance permettra de présenter une ébauche de nos idées et de comment nous allons gérer notre temps pour mener à bien ce projet. C'est aussi durant cette phase que nous rédigerons le dossier de participation au concours Atos.

Vient ensuite la phase de conception, qui est la plus importante de notre projet, c'est en effet à ce moment que nous construisons les fondations de nos applications, en spécifiant les besoins fonctionnels, et en élaborant différents diagrammes de modélisation. L'objectif de cette phase est d'explicitier le fonctionnement et l'agencement des différents éléments. Cette période s'achève par la rédaction d'un rapport de conception.

Suit la phase de construction, qui elle tournera autour du développement, c'est à dire le codage de l'application. Des tests réguliers permettront aussi de jauger la stabilité et les fonctionnalités de notre projet. Cette phase se clôture par le déploiement de l'application créée.

Pour terminer, la phase de livraison consiste à remettre un produit fini et fonctionnel. Dans notre cas, il y en aura deux, une pour le concours Atos et une autre, plus tard, pour l'INSA. Elle comprend aussi l'élaboration des livrables finaux c'est à dire rapport final, page HTML et documentation en ligne.

Le diagramme de Gantt, voir figure 4.1 page 21, illustre au mieux notre planification initiale.

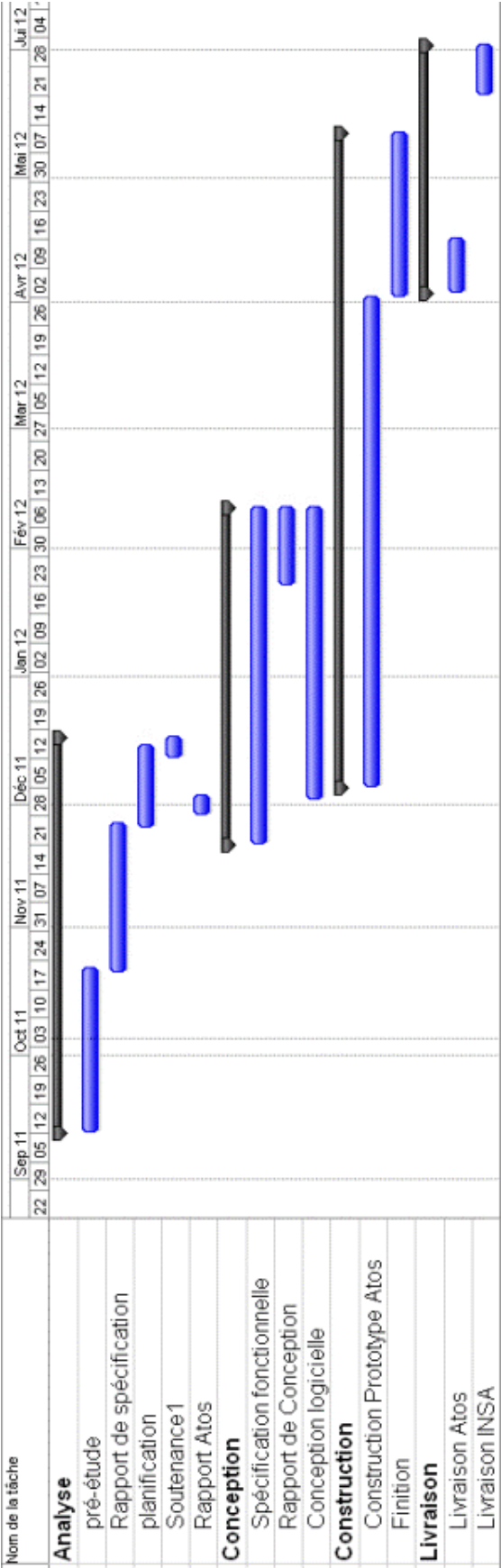


FIGURE 4.1: Première planification

Conclusion

A travers les informations recueillies durant cette première phase d'analyse, nous avons acquis une première vision du cadre d'évolution de notre projet : « Numérisation de documents en mobilité sur iPhone/iPad pour Mac OS X ». De grands objectifs s'en sont dégagés, nous permettant d'identifier clairement les besoins réels liés à ce dernier. Il s'agit ici de concevoir un système d'applications simple, ergonomique, répondant à un fort contexte de mobilité et pouvant s'adapter à de nombreuses situations, le tout en offrant un confort d'utilisation accru, notamment grâce à une synchronisation transparente et efficace entre divers périphériques de type Apple.

De plus, la réalisation de ce projet nous permettra de prendre en main et d'allier différentes technologies. A travers lui, nous allons découvrir et utiliser des ressources spécifiques (Objective-C, Cocoa, etc) ainsi que des outils informatiques. Notamment, nous verrons plus en détails le développement mobile ainsi que les modes de connexion entre différents supports matériels.

Nous allons maintenant poursuivre notre projet en étudiant et définissant les spécifications générales et fonctionnelles. Pour cela, nous nous intéresserons au fonctionnement interne et à l'architecture des deux applications nécessaires à la réalisation du cahier des charges.

Plus tard, nous verrons en détails l'impact en terme d'échéances de notre participation à l'IT Challenge. Nous aurons à formaliser dans notre dossier d'inscription différents aspects n'étant pas évoqués dans le projet (modèle économique, bénéfices pour les marques partenaires, etc) ce qui nous apportera un point de vue différent sur les attentes que peut avoir un client vis à vis de son produit.

Bibliographie

- [1] Rue 89. *La guerre des géants du web*. Disponible sur <<http://www.rue89.com/2011/09/20/la-guerre-des-geants-du-web-dans-une-application-rue89-221996>>. (Consulté en septembre 2011).
- [2] Mac Aficionados. *Les meilleures applications de mobilité pour Mac, iPhone et iPad 2*. Disponible sur <<http://macaficionados.wordpress.com/2011/04/04/les-meilleures-applications-de-mobilite-pour-mac-iphone-et-ipad-2/>>. (Consulté en septembre 2011).
- [3] Apple. *iCloud*. Disponible sur <<https://www.icloud.com>>. (Consulté en octobre 2011).
- [4] Apple. *Site des développeurs Apple*. Disponible sur <<http://developer.apple.com/>>. (Consulté en septembre 2011).
- [5] AppStore. *DocScanner*. Disponible sur <<http://itunes.apple.com/fr/app/docscanner/id412705688?mt=12>>. (Consulté en septembre 2011).
- [6] Pierre Y. Chatelier. *Objective-C pour le développeur avancé*. Eyrolles, 2010.
- [7] Robert. Clair. *Objective-C 2.0 : Guide pratique pour les développeurs Mac et iOS*. Pearson, 2011.
- [8] Creaceed. *Prizmo, OCR sur Mac*. Disponible sur <<http://www.creaceed.com/fr/prizmo/mac/>>. (Consulté en septembre 2011).
- [9] Dropbox. *Site officiel*. Disponible sur <<http://www.dropbox.com/features>>. (Consulté en septembre 2011).
- [10] GénérationIPHONE.fr. *Mover pour le partage de fichiers entre iPhone et Mac*. Disponible sur <<http://www.generationiphone.fr/mover-permet-le-partage-de-fichier-entre-iphone-et-mac/>>. (Consulté en septembre 2011).
- [11] Irida. *Imadoc*. Disponible sur <<http://www.irisa.fr/imadoc/web>>. (Consulté en septembre 2011).
- [12] iTunes. *Profil de Xcode sur l'iTunes Store*. Disponible sur <<http://itunes.apple.com/fr/app/xcode/id448457090?mt=12>>. (Consulté en septembre 2011).
- [13] OsxDaily. *Article à propos de Xcode 4*. Disponible sur <<http://osxdaily.com/2011/02/03/xcode-4-gm-released-for-developers/>>. (Consulté en septembre 2011).
- [14] Wikipedia. *Apple Inc.* Disponible sur <http://en.wikipedia.org/wiki/Apple_Inc.>. (Consulté en septembre 2011).

-
- [15] Wikipedia. *Cocoa*. Disponible sur <http://fr.wikipedia.org/wiki/Cocoa_%28Apple%29>. (Consulté en septembre 2011).
 - [16] Wikipedia. *iOS*. Disponible sur <[http://fr.wikipedia.org/wiki/IOS_\(Apple\)](http://fr.wikipedia.org/wiki/IOS_(Apple))>. (Consulté en septembre 2011).
 - [17] Wikipedia. *iPad*. Disponible sur <<http://fr.wikipedia.org/wiki/Ipad>>. (Consulté en septembre 2011).
 - [18] Wikipedia. *iPhone*. Disponible sur <<http://fr.wikipedia.org/wiki/IPhone>>. (Consulté en septembre 2011).
 - [19] Wikipedia. *Linux*. Disponible sur <<http://fr.wikipedia.org/wiki/Linux>>. (Consulté en septembre 2011).
 - [20] Wikipedia. *Mac OS X*. Disponible sur <http://fr.wikipedia.org/wiki/Mac_os_x>. (Consulté en septembre 2011).
 - [21] Wikipedia. *Multi-touch*. Disponible sur <<http://fr.wikipedia.org/wiki/Multi-touch>>. (Consulté en septembre 2011).
 - [22] Wikipedia. *Objective-C*. Disponible sur <<http://fr.wikipedia.org/wiki/Objective-C>>. (Consulté en septembre 2011).
 - [23] Wikipedia. *Unix*. Disponible sur <<http://fr.wikipedia.org/wiki/Unix>>. (Consulté en septembre 2011).
 - [24] Wikipedia. *Xcode*. Disponible sur <<http://fr.wikipedia.org/wiki/Xcode>>. (Consulté en septembre 2011).