



École Polytechnique de l'Université de Tours
64, Avenue Jean Portalis
37200 TOURS, FRANCE
Tél. +33 (0)2 47 36 14 14
www.polytech.univ-tours.fr

Département Informatique
4^e année
2011 - 2012

Rapport de stage



**Portage d'une application de GMAO
Windows Mobile sous Android**

Encadrant
Thomas TATU
thomas.tatu@atos.net

Étudiant
Pierre Fourreau
pierre.fourreau@etu.univ-tours.fr

Université François-Rabelais, Tours

DI4 2011 - 2012

Version du 22 août 2012

Table des matières

1	Remerciements	7
2	Introduction	8
3	Partie 1 : Présentation de l'entreprise	9
3.1	Atos origin	9
3.1.1	Présentation générale	9
3.1.2	Atos en quelques chiffres	10
3.1.3	Secteur d'activité	10
3.2	Atos Worldline	11
3.2.1	Présentation générale	11
3.2.2	Atos Worldline en quelques chiffres	13
3.2.3	Les Business Unit (BU)	14
3.2.4	La BU PST	14
4	Partie 2 : Cahier des charges	17
4.1	Présentation du projet	17
4.2	La Gestion de maintenance assistée par ordinateur	17
4.2.1	Présentation générale d'une GMAO	17
4.2.2	Bénéfices attendus	17
4.2.3	Secteurs d'activité concernés	18
4.2.4	Intégration de la GMAO dans le système d'information de l'entreprise	18
4.3	Description de l'environnement	18
4.4	Contraintes	19
4.4.1	Contraintes de délais	19
4.4.2	Contraintes techniques	19
4.4.3	Contraintes sur l'avancée du projet	19
4.5	Planning	19
4.6	Système de gestion de version	20

5	Partie 3 : Spécifications du système	21
5.1	Introduction	21
5.2	Fonctions générales du système	21
5.3	Spécifications du système	22
5.3.1	Processus métiers et fonctionnalités	22
5.3.2	Identification	22
5.3.3	Visualisation d'une tournée	23
5.3.4	Visualisation des interventions	23
5.3.5	Générer un compte rendu d'intervention	24
5.3.6	Obtenir l'historique d'un emplacement ou d'un équipement	25
5.3.7	Signalement de dysfonctionnement	26
5.4	Description des interfaces de l'application	26
6	Partie 4 : Conception de l'application	27
6.1	Solution adoptée	27
6.2	Authentification de l'utilisateur	27
6.2.1	Principe de l'authentification	27
6.2.2	Définition d'un web service	27
6.2.3	Outil simulant l'appel à un web service	29
6.2.4	Authentification dans l'application	29
6.3	Consultation de l'historique	31
7	Partie 5 : Implémentation de l'application	32
7.1	Eclipse	32
7.2	Visual Studio	32
7.3	Problèmes rencontrés	33
8	Partie 6 : Missions annexes	34
8.1	Modification de la charte graphique d'une application mobile	34
8.2	Déploiement d'une mise à jour sur les PDA Windows Mobile	37
8.2.1	Objectif	37
8.2.2	Fonctionnement	37
8.2.3	Utilitaire wceload	38
8.2.4	Manipulation	38
8.2.5	Conclusion	42
9	Conclusion	43
	Annexes	45

Table des figures

3.1	Evolution du chiffre d'affaires d'Atos Origin de 1991 à 2006	9
3.2	Carte de répartition mondiale	10
3.3	Les différentes activités d'Atos	11
3.4	Une présence mondiale pour Atos Worldline	12
3.5	Clients d'Atos Worldline	13
3.6	Chiffres d'Atos Worldline	14
3.7	Une journée avec Atos Worldline	15
3.8	Equipe PST A	16
4.1	Mercurial logo	20
5.1	Processus métier de l'application mobile	22
5.2	Accès réseau	22
5.3	Code couleur des interventions	23
6.1	Soap vs Rest	28
6.2	Simulation d'un appel à un web service avec Poster	29
6.3	Historique des actions	31
7.1	Mercurial logo	32
7.2	Visual Studio 2008	32
8.1	Charte graphique avant modification	35
8.2	Charte graphique après modification	36
8.3	Appareils mobiles pour tester l'application	37
8.4	Déploiement lien documentation officielle	38
8.5	Déploiement code	38
8.6	Déploiement chemin	38
8.7	Déploiement modification version 1	39
8.8	Déploiement modification version 1 code	40
8.9	Déploiement création projet	40
8.10	Déploiement sélection du certificat	41
8.11	Déploiement exécution du test	42

1	Interface login	45
2	Interface menu	45
3	Interface permettant de visualiser la liste des interventions	46
4	Interface permettant de visualiser une intervention choisie	46
5	Interface recherche de l'historique	47
6	Interface compte rendu et ajout d'une action	47
7	Interface compte rendu de la liste des actions	48

1. Remerciements

Je voudrai remercier Monsieur Jamy Rondeau pour m'avoir accueilli dans son équipe PST ainsi que toutes les personnes qui m'ont accueilli et intégré durant ce stage et qui ont fait en sorte que celui-ci se déroule le mieux possible.

Tout particulièrement, je tiens à remercier Monsieur Thomas Tatu pour ses conseils et sa patience à mon égard qui m'ont permis d'aller au bout de mon sujet de stage. Je remercie également mes collègues de l'équipe PST A-1A : Amine Saidane, Maxime Rafaillac, Kévin Dupont, Yohann Jouanneau, Julien Magri, et Giovanni Castellana. Egalement Antoine Lochet, Fabien Martineau, Emeric Pinon, et Jérémy Moulins de l'équipe PST A-1B.

Enfin je remercie Madame Marie-Françoise Decarnin et Madame Annie Morlet, travaillant à l'accueil de l'entreprise pour leur sympathie.

2. Introduction

Dans le cadre de ma formation au sein du Département Informatique de l'école Polytechnique de l'Université de Tours, un stage est effectué afin de mettre en pratique nos connaissances acquises au cours de notre formation.

Ce stage se déroule à ATOS Worldline, une société de services en ingénierie informatique (SSII) et dure 12 semaines (11 Juin au 31 Aout 2012). Pour ce stage, j'ai travaillé en binôme et ai été encadré par Thomas Tatu.

Le but de mon projet est de concevoir une application Android déjà existante sous Windows Mobile. Cette application Windows Mobile permet de gérer les interventions des agents sur les horodateurs de la mairie de Paris. Mais mon objectif était de concevoir une application plus générale permettant de gérer des équipements.

Ce rapport est constitué essentiellement de six parties. La première partie présentera l'entreprise où est effectué ce stage. La deuxième partie définit le cahier des charges. La troisième partie est consacrée aux spécifications du système. La quatrième partie décrit les spécifications et la conception du logiciel. Puis, la cinquième partie décrit l'implémentation du logiciel. Enfin, la sixième et dernière partie présentera les missions annexes qui m'ont été confiées durant ce stage.

3. Partie 1 : Présentation de l'entreprise

3.1 Atos origin

3.1.1 Présentation générale

Le groupe Atos est l'un des principaux acteurs internationaux dans les services informatiques.

Sa mission est de proposer des solutions de conseil, d'intégration de systèmes, d'infogérance et de services en ligne afin d'améliorer les résultats de ses clients. Implanté dans 40 pays à travers le monde, Atos Origin réalise un chiffre d'affaire annuel de 5,1 milliards d'euros en 2010 (avec un effectif de 50 000 personnes), dont 879 millions pour la filiale Atos Worldline. Atos Origin compte parmi ses clients de grands comptes internationaux dans tous les secteurs d'activités.

Issu de la fusion des groupes Atos et Origin en 2000, les acquisitions de KPMG Consulting en 2002 puis du groupe Sema en 2004 ont donné naissance à Atos Origin. Le groupe Atos Origin s'est construit par un certain nombre de fusions, ou d'acquisitions, l'amenant en une quinzaine d'années du statut d'acteur local à celui de premier fournisseur de services informatiques en France, et quatrième en Europe (derrière Sopra Group, Axway, filiale de Sopra Group et Capgemini).

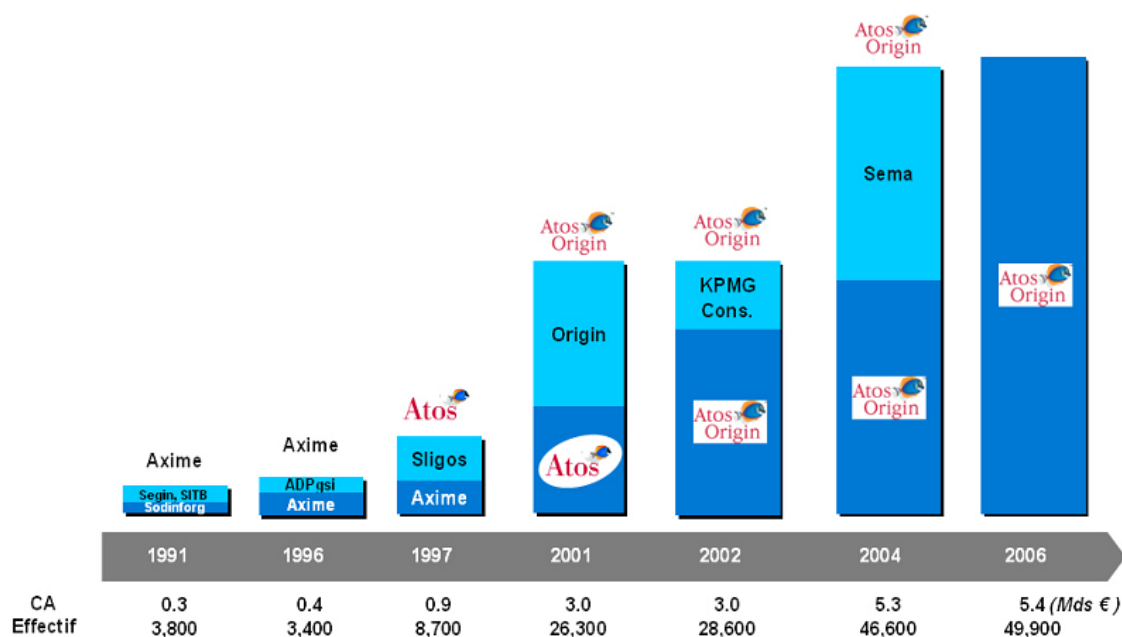


FIGURE 3.1 – Evolution du chiffre d'affaires d'Atos Origin de 1991 à 2006

Les revenus d'Atos Origin se répartissent entre le conseil (8% du CA), l'intégration de systèmes

(41%), et l'infogérance (50%). La forte proportion de l'infogérance assure un revenu récurrent important à l'entreprise. Les secteurs d'activités touchés par Atos Origin sont notamment la distribution et les biens de consommation de haute technologie (27% du CA), la finance (21%), les télécommunications (18%), ainsi que le secteur public (18%). Atos Origin dispose d'une forte présence européenne, avec 84% de son effectif basé en Europe, notamment en France (31%), au Benelux (21%) et en Grande-Bretagne (19%).

En France, le groupe Atos Origin est coté au marché d'Euronext Paris. Il est présent au travers de cinq entités, couvrant toute l'étendue du spectre des activités informatiques, de la conception (design) à la gestion de la production (run), en passant par la réalisation (build).

En 2010, Atos Origin annonce l'acquisition de Siemens IT Solutions and Services, c'est à la suite de cette acquisition que le groupe Atos Origin se renomme Atos.

3.1.2 Atos en quelques chiffres

Aujourd'hui, Atos comprend 78500 "Business Technologists" dans plus de 42 pays.



FIGURE 3.2 – Carte de répartition mondiale

3.1.3 Secteur d'activité

Atos est présent dans trois domaines d'activités : le conseil, l'intégration et l'infogérance. Ces trois domaines sont répartis dans les trois entités du groupe : Atos Consulting (conseil), Atos Intégration (infogérance et intégration) et Atos Worldline (infogérance).

Le groupe intervient dans des domaines variés tels que la finance, le secteur public et les services, les télécommunications et les médias, l'industrie des composants, la distribution ou encore l'industrie de transformation entre autres . On retrouve ainsi parmi ses clients, des grands comptes, des entreprises comme Shell, Alstom, BNP Paribas, SNCF ou les Jeux Olympiques.

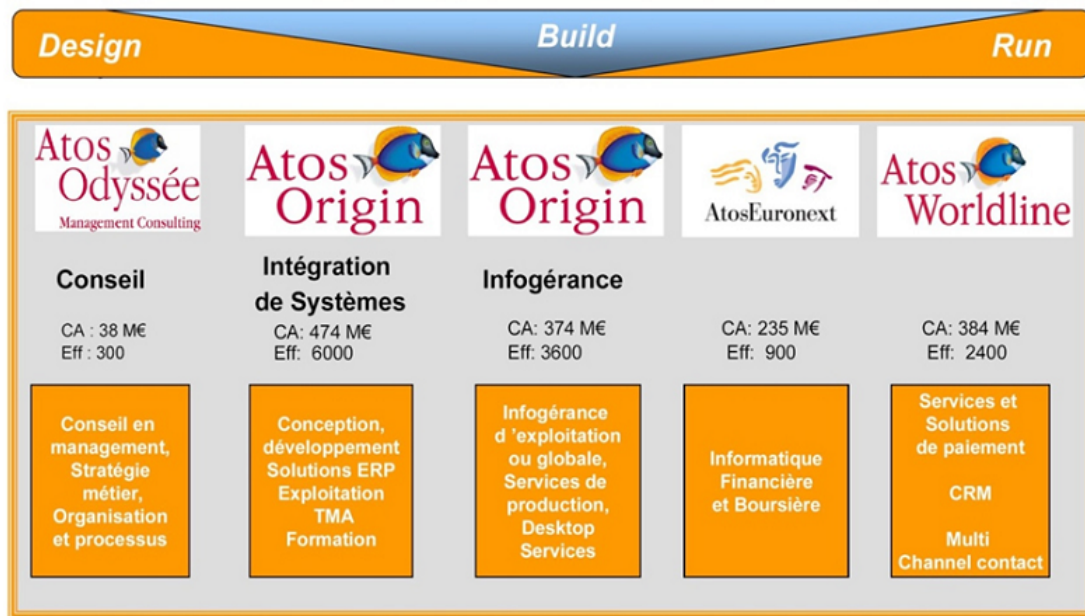


FIGURE 3.3 – Les différentes activités d'Atos

3.2 Atos Worldline

3.2.1 Présentation générale

Créée en 2004, Atos Worldline (AWL) est une des filiales d'Atos. Elle est depuis devenue l'un des leaders européens dans l'externalisation du processus métier (Business Process Outsourcing, BPO) et s'est spécialisée dans le traitement des échanges électroniques à grand volume.

Atos Worldline est née suite à la volonté de réunir vingt ans d'expériences internationales dans les domaines du paiement électronique, de la gestion de la relation clientèle et du MultiChannel Contact auparavant dispersés dans six divisions d'Atos Origin :

- Services
- Multimédia
- Atos Origin Processing Services (Allemagne)
- Atos Euronext Payment Solutions (France)
- Atos Origin Payment Solutions (Allemagne et Autriche)
- Atos Origin Transaction Processing (Grande-Bretagne)

Principalement basée en France, avec cinq sites majeurs et la plupart de ses collaborateurs, Atos Worldline est également présente en Allemagne, en Belgique, en Grande-Bretagne, en Hollande et au Luxembourg.

Atos Worldline propose à ses clients des solutions Build to Run, qui couvrent à la fois le développement de l'application, son hébergement dans ses data centers et sa maintenance, permettant ainsi de maîtriser le projet de bout en bout.

Atos Worldline se positionne également, mais plus rarement, sur des projets de Build pur (développement) ou de Run pur (hébergement et maintenance).



FIGURE 3.4 – Une présence mondiale pour Atos Worldline

Le 7 décembre 2006, Atos Origin a annoncé l'acquisition de Banksys et Bank Card Company (BCC), créant ainsi un leader européen des services de paiements, avec un chiffre d'affaires de 800 millions d'euros et 4800 employés dont 3000 en Europe. Atos Worldline génère son chiffre d'affaires au travers de ses trois domaines d'expertise :

- Des services de paiement et de traitement des cartes : ceux-ci couvrent l'ensemble de la chaîne de valeur dans les domaines de la monétique et des services en ligne
- Des services de CRM : les services proposés visent à aider les entreprises à valoriser leur capital client. Ces solutions s'appuient notamment sur une maîtrise des bases de données clients, des programmes de fidélité...
- Des services de Contacts Multi-Canaux : ceux-ci offrent un ensemble d'outils dédiés aux différentes technologies utilisées

La majorité de ces revenus provient de l'exploitation et de la maintenance des applications, bien que le développement tende à devenir de plus en plus important. Les principaux clients d'Atos Worldline sont évidemment des acteurs financiers tels que les banques européennes et sites utilisant le paiement en ligne. Mais Atos Worldline s'est diversifiée au cours des années et travaille maintenant avec des clients des cinq continents et dans des secteurs d'activités très variés.

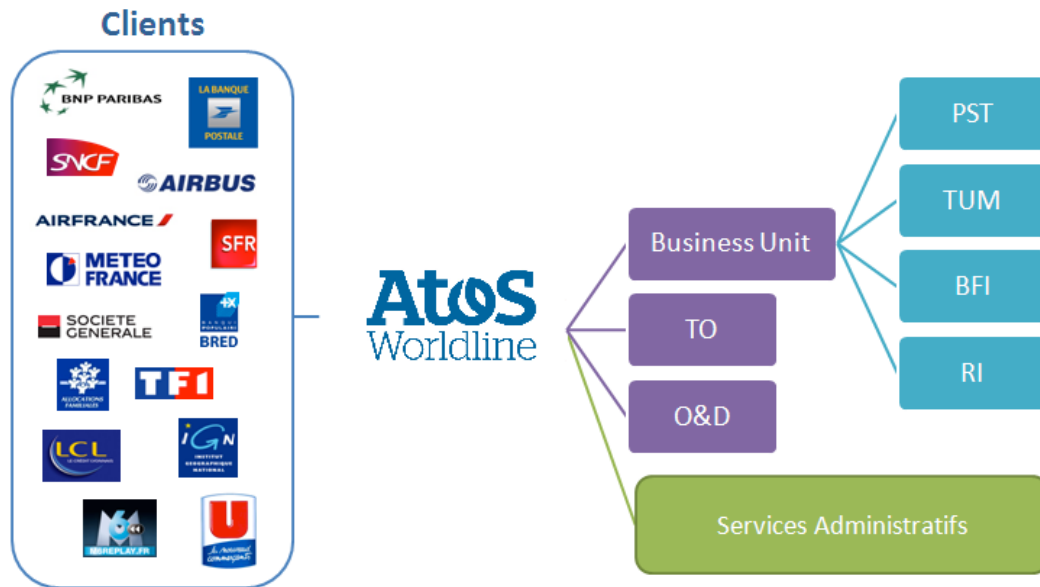


FIGURE 3.5 – Clients d'Atos Worldline

3.2.2 Atos Worldline en quelques chiffres

L'entreprise Atos Worldline étant une entreprise travaillant dans le domaine bancaire, elle se doit d'être certifiée PCI (Payment Card Industry), un programme de sécurisation des données mis en oeuvre par les réseaux internationaux (Visa, Mastercard, American Express). PCI impose le respect de règles strictes dans toute la chaîne de traitement : réseau, données cartes, contrôle des données, traçabilité des accès aux données, politique de sécurité. Cette certification garantit aux marchands confiance, sécurité et confidentialité.



FIGURE 3.6 – Chiffres d'Atos Worldline

Récemment, Atos a acquis Venture Infotek le 27 août 2010, leader du paiement sur le marché Indien, ce qui vient renforcer le cœur de métier d'Atos Worldline. Dans cette même démarche, le 4 juillet 2011, Siemens IT Services s'est vu intégré dans le groupe Atos Origin, d'où le nouveau logo, symbolisant évidemment Atos mais aussi cette nouvelle union : *Atos to Siemens*.

3.2.3 Les Business Unit (BU)

Afin de mieux répondre aux besoins de ses clients, Atos Worldline est divisé en Business Unit (BU) spécialisées sur des secteurs de marché et des processus métier particuliers. On trouve ainsi quatre principales BU :

- Banques et Finances (BFI)
- Télécommunications et Médias (TUM)
- Retail et Industrie (RI)
- Public, Santé, Transports (PST)

Ce sont ces BU qui ont en charge le développement et l'exploitation des applications des clients. Elles peuvent s'appuyer sur deux unités transversales pour mener à bien leur mission : Techniques Opérationnelles (TO) en charge de la mise en place, de la configuration et de la maintenance des serveurs. Offres et Développement (O&D) en charge de la création et de la maintenance des outils et des technologies internes.

3.2.4 La BU PST

La BU PST s'occupe des développements et des hébergements des grands comptes du secteur public (Ministères Français, IGN) et des transports (Air France, SNCF, Keolis). Elle bénéficie d'une forte

croissance, avec une augmentation régulière de son chiffre d'affaire et du nombre de ses collaborateurs depuis 2002. Ces derniers sont répartis sur les sites de Seclin, Blois, Lyon et Tours.

Chez PST, comme dans les autres BU, les responsables d'applications travaillent par équipes de cinq ou six, sous la responsabilité d'un chef d'équipe. Cette notion d'équipe ne correspond pas forcément à une équipe de projet. On peut trouver plusieurs projets au sein d'une même équipe ou plusieurs équipes sur un même projet, selon son importance.

Exemples de projets d'AWL : Afin de représenter au mieux leurs domaines de compétences, Atos Worldline met à disposition un document représentant une journée type d'un individu en exposant le nombre de fois où cette personne est confrontée aux services mis en place par AWL :



FIGURE 3.7 – Une journée avec Atos Worldline

J'ai personnellement fait partie de la BU PST durant mes 12 semaines de stage.

L'organigramme de mon équipe est le suivant :

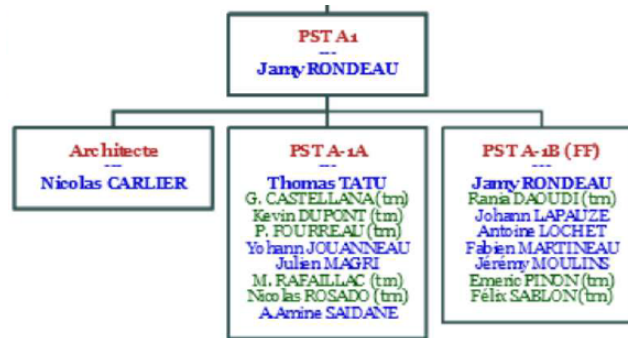


FIGURE 3.8 – Equipe PST A

4. Partie 2 : Cahier des charges

4.1 Présentation du projet

Notre projet consiste à concevoir une application Android qui existe déjà sous Windows Mobile. Cette application permet à un agent de gérer ses interventions sur des équipements.

4.2 La Gestion de maintenance assistée par ordinateur

4.2.1 Présentation générale d'une GMAO

La Gestion de maintenance assistée par ordinateur(GMAO) est une méthode de gestion assistée d'un logiciel destiné aux services de maintenance d'une entreprise afin de l'aider dans ses activités. L'équivalent en anglais de GMAO est CMMS (computerized maintenance management system).

L'objectif premier d'une GMAO est d'assister les services maintenance des entreprises dans leurs missions. Un service de maintenance a pour but de maintenir ou de rétablir un bien (comme un équipement) dans un état spécifié afin que celui-ci soit en mesure d'assurer un service déterminé (selon l'AFNOR).

Mais une GMAO peut aussi être profitable dans d'autres services de l'entreprise. On peut citer par exemple le service production ou le service exploitation. Pour ce dernier, cela pourrait être utile afin de fournir des informations sur l'état des équipements, mais aussi sur la direction financière ou générale de l'entreprise, en fournissant, par exemple, des indicateurs facilitant les prises de décisions en matière de renouvellement de parc.

On peut citer différentes fonctions que peuvent remplir une GMAO :

- Gestion des équipements : inventaire des équipements, localisation, gestion d'information dédiée par type d'équipement (production, bâtiments, véhicules, réseaux, ordinateurs...)
- Gestion de la maintenance : corrective, préventive...
- Gestion de la mise en sécurité des installations pour les travaux. de maintenance : consignation, centralisation, autorisation de sécurité, déconsignation...
- Gestion des stocks : magasins, quantités minimum ou maximum de réapprovisionnement.
- Gestion des achats : de pièces détachées ou de services.
- Gestion du personnel et planning : activités, métiers, planning de charge, prévisionnel, pointage des heures...
- Gestion des coûts et budget : de main d'œuvre, de stocks, d'achat, de location de matériel...

Dans notre situation, le but est de gérer des équipements.

4.2.2 Bénéfices attendus

Lorsqu'une GMAO est mise en place, des bénéfices sont attendus. On peut par exemple citer :

- Meilleure gestion et réduction des coûts (main-d'œuvre, pièces détachées, traitement administratif).
- Amélioration de la fiabilité et de la disponibilité des équipements.
- Optimisation des achats (aide aux appels d'offres, gestion des contrats de prestataires externes).
- Amélioration du retour d'expérience grâce à l'historique des travaux de maintenance.
- Amélioration de la planification des interventions.
- Recherche du ratio préventif/correctif optimal en fonction des équipements gérés et des objectifs de disponibilité.
- Meilleur contrôle de l'activité des sous-traitants et prestataires externes.
- Amélioration de la gestion des stocks (meilleur contrôle des sorties, aide aux inventaires).
- Traçabilité des équipements.

4.2.3 Secteurs d'activité concernés

A partir du moment où un secteur d'activité a des équipements à maintenir, celui-ci est alors potentiellement concerné par l'exploitation d'un outil de GMAO. On peut citer les secteurs :

- de l'industrie (automobile, pharmaceutique),
- de la production,
- de l'énergie (gaz, pétrole, électricité),
- des transports (routier, ferroviaire, aérien),
- médicaux (hôpitaux, cliniques),
- de l'immobilier (HLM, locatif),
- de la grande distribution,
- des collectivités locales (communauté urbaine, agglomération, aéroport),
- des travaux publics,
- des télécoms (gestion des équipements réseau),

Suivant chaque secteur d'activité, des fonctionnalités spécifiques peuvent exister.

4.2.4 Intégration de la GMAO dans le système d'information de l'entreprise

Une GMAO peut avoir intérêt à s'interfacer avec d'autres logiciels existants de l'entreprise :

- Une gestion des stocks : afin de gérer les stocks de pièce de rechange.
- Une gestion d'achat : afin de gérer les achats de pièce de rechange ou de service.
- Une gestion budgétaire ou financière.
- Une gestion électronique de document (GED) : cela permet d'accéder à des archives de plans ou de documents techniques.
- Une gestion de patrimoine (domaine immobilier).
- Un système temps réel de GTC/GTB : afin de générer automatiquement dans la GMAO des interventions en cas de survenance de certains événements détectés par capteurs, ou afin de mettre à jour les valeurs de compteurs d'utilisation concernant certains équipements.

Source : Wikipédia

4.3 Description de l'environnement

Cette application sera dédiée à un PDA endurci. Ce dernier tourne sous Android et fonctionne donc de la même manière qu'un Smartphone classique fonctionnant sous Android.

Nous n'avions pas encore reçu ce PDA lors du stage, il a donc fallu tester l'application sur différents téléphones Android.

4.4 Contraintes

4.4.1 Contraintes de délais

- Début du projet : 18 Juin 2012
- Fin du projet : 31 Août 2012

La durée totale du projet est donc de 11 semaines. En effet, durant la première semaine, nous avons été chargé de travailler sur le design d'une application mobile avec l'utilisation du framework JQuery Mobile.

4.4.2 Contraintes techniques

Optimisation de l'application pour un PDA fonctionnant sous Android.

4.4.3 Contraintes sur l'avancée du projet

L'application Windows Mobile sur laquelle repose notre projet n'est pas terminée et est encore en cours de développement. De ce fait, des modifications sur l'application Windows Mobile engendrent inévitablement des modifications sur notre application. Ce point est très important car nous avons dû, à plusieurs reprises, modifier notre application car l'application de base Windows Mobile avait changé.

4.5 Planning

Chaque semaine nous avons fait une réunion avec notre encadrant afin de faire un point sur l'avancement du projet. De plus, nous avons réalisé un chiffrage en jour/homme et un planning que nous mettions à jour suivant le déroulement du projet.

Ce planning a plutôt été bien respecté puisque nous avons livré notre application dans les délais impartis.

Enfin, nous avons quotidiennement tenu un journal de bord afin d'y indiquer les tâches en cours de réalisations, à effectuer, ou terminées. Pour cela nous utilisons l'interface web Trello.

4.6 Système de gestion de version

Nous utilisons un système de gestion de versions pour travailler plus efficacement durant ce projet.



FIGURE 4.1 – Mercurial logo

Étant donné que tous nos collègues utilisent Mercurial, nous avons logiquement choisi d'utiliser cet outil.

Cela nous permet durant tout le projet de : Pouvoir revenir en arrière Voir qui a modifié ce fichier en dernier

Ses principales caractéristiques sont, entre autres : sa rapidité et sa capacité à gérer les gros projets ; son utilisation sans nécessiter un serveur ; son fonctionnement complètement distribué ; sa robustesse dans la gestion des fichiers ASCII et binaires ; sa gestion avancée des branches et des fusions ; son interface web intégrée.

5. Partie 3 : Spécifications du système

5.1 Introduction

Ce projet a pour but de concevoir et développer une application Android permettant de gérer les interventions des agents sur des équipements.

5.2 Fonctions générales du système

Lorsqu'un agent s'est authentifié sur son PDA, il peut alors interagir avec la totalité de l'application.

Gestion des interventions d'un agent. Chaque intervention est associée à un équipement, et sur chaque équipement, un ensemble d'actions peut être effectué.

Lorsque l'agent consulte la liste des interventions du jour, il peut localiser celles-ci sur une carte directement sur son PDA.

De plus, une fois que l'agent a sélectionné une intervention concernant un équipement, il peut lancer une application GoogleMap intégrée au PDA (une connexion internet est exigée) afin de tracer un itinéraire sur une carte.

Les fonctions générales du système sont les suivantes :

- Visualiser les interventions
- Rédiger un compte rendu
- Déclarer un signalement
- Consulter l'historique des interventions

5.3 Spécifications du système

5.3.1 Processus métiers et fonctionnalités

Les processus métiers ainsi que les fonctionnalités sous-jacentes associées sont les suivants :

Processus métiers	Fonctionnalités
Connexion à l'application	Renseigner son identifiant d'agent <i>Récupération de l'ensemble des interventions attribuées à l'agent</i>
Gérer la tournée	Visualiser sa tournée sous forme de liste <i>Les interventions sont classées dans l'ordre d'optimisation du trajet</i>
	Visualiser sa tournée sous forme de carte <i>Les interventions sont identifiées par des points sur une carte</i>
	Visualiser une intervention de la tournée <i>Permet d'avoir le détail d'une intervention</i>
	Commencer une intervention
	Rédiger le compte rendu d'une intervention <i>Déclaration de l'état de l'équipement en partant (OK ou en panne)</i>
Obtenir l'historique d'un emplacement ou horodateur	Obtenir l'historique d'un emplacement ou d'un équipement <i>Permet de savoir si une anomalie constatée est normale ou non</i>
Effectuer un signalement	Déclarer un dysfonctionnement sur un équipement
Aider à la navigation	Appeler un programme externe pour l'aide à la navigation (Google Navigation) <i>Permet de se rendre sur les équipements facilement</i>

FIGURE 5.1 – Processus métier de l'application mobile

Certaines fonctionnalités ont besoin d'avoir accès au réseau pour fonctionner correctement. Le tableau ci-dessous indique dans quel cas l'accès au réseau est nécessaire et s'il doit l'être en temps réel :

Fonctionnalités	Accès réseau	Temps réel
Identification	Oui	Oui
Récupération de la tournée (synchronisation avec le système central)	Oui	Oui
Visualiser la tournée	Oui	Oui
Traiter les interventions	Non	
Aide au déplacement	Oui	Oui
Générer un compte rendu	Oui	Non
Historique des équipements et emplacements	Oui	Oui
Effectuer un signalement	Oui	Non

FIGURE 5.2 – Accès réseau

5.3.2 Identification

Lorsqu'un mainteneur essaye d'accéder à l'application, une page d'identification apparaît. Celui-ci doit alors saisir son identifiant de mainteneur pour rentrer dans l'application. Aucun mot de passe

n'est demandé.

Remarque : l'identifiant est demandé à chaque ouverture de l'application. Si le mainteneur ferme l'application en cours de journée, son identifiant sera redemandé à l'ouverture de l'application.

5.3.3 Visualisation d'une tournée

Consultation cartographique de la tournée

Sur l'application mobile, il est possible de visualiser l'ensemble des interventions de sa tournée sur une carte. La tournée est matérialisée sur une carte avec des points au niveau de chaque intervention. Cela permet d'avoir une représentation de la tournée.

Aide au déplacement des agents

L'application va intégrer Google Maps pour permettre aux agents de maintenance de circuler et surtout de trouver les emplacements des équipements.

Le système d'aide à la navigation est disponible en cliquant sur l'adresse de l'intervention. Le système récupère alors l'adresse et calcule automatiquement l'itinéraire à suivre pour se rendre sur le lieu de l'intervention.

5.3.4 Visualisation des interventions

L'application permet de visualiser la liste des interventions qui ont été attribuées au mainteneur porteur du PDA.

Les interventions sont ordonnées pour avoir le trajet optimisé. Si l'agent de maintenance les traite dans l'ordre indiqué dans le PDA, son trajet sera optimisé. Mais cela n'est pas obligation. Il n'y a pas de contrôle et d'obligation dans l'ordre d'exécution des interventions : l'agent de maintenance reste maître de son trajet et peut donc le modifier en fonction de sa connaissance du terrain et de son expérience.

Sur la liste des interventions, un code couleur est utilisé afin de visualiser l'état des interventions :

Vert	Intervention terminée
Rouge	Intervention non réalisable
Bleu	Intervention en cours
Sans couleur	Intervention à réaliser

FIGURE 5.3 – Code couleur des interventions

La liste des interventions permet de visualiser les éléments essentiels à l'intervention, à savoir :

- L'état de l'intervention
- L'identifiant de l'équipement
- L'adresse avec l'arrondissement
- La criticité de l'intervention
- L'état de l'intervention

En cliquant sur une intervention, il est alors possible d'avoir un plus de détail :

- L'adresse menant vers l'aide à la navigation
- La liste des actions à effectuer
- Le commentaire de l'intervention si présent

A partir d'une intervention non commencée, il est possible d'effectuer 3 actions :

- Accéder à l'aide à la navigation en cliquant sur l'adresse de l'intervention
- Commencer l'intervention
- Déclarer l'intervention comme étant impossible à réaliser

A partir d'une intervention commencée, il est possible de saisir le compte rendu d'intervention.

Le fait de commencer l'intervention signifie que le mainteneur est devant l'équipement et qu'il va commencer les manipulations pour mener à bien l'intervention demandée. Un compte rendu de l'intervention sera alors demandé à la fin de l'intervention pour connaître les actions qui ont été réalisées.

Le fait de déclarer l'intervention comme étant impossible à réaliser signifie que le mainteneur ne peut pas accéder à l'équipement ou que l'équipement est dans un état tel qu'il n'est pas possible d'intervenir dessus. Le détail de la raison de l'impossibilité est alors demandé au mainteneur par l'intermédiaire d'une liste déroulante. Les motifs suivants ont été identifiés : A partir d'une intervention non commencée, il est possible d'effectuer 3 actions :

- Accès impossible à l'emplacement
- Ouverture impossible
- Sous-ensemble manquant
- Autre

La saisie d'un commentaire est obligatoire dans le cas "Autre". Elle est facultative dans les autres cas.

5.3.5 Générer un compte rendu d'intervention

Lorsqu'une intervention a eu lieu, le mainteneur doit rédiger en fin d'intervention un compte-rendu de l'intervention. Il est alors demandé au mainteneur : les actions effectuées lors de l'intervention. Par défaut, les actions sont renseignées avec les actions prévues sur l'intervention. Il est alors possible de rajouter des actions unitaires ou d'en supprimer. La seule action unitaire qu'il est possible de rajouter est "Remplacement de l'élément" avec la qualification du sous-ensemble. Seuls les agents de maintenance peuvent rajouter cette action unitaire, les contrôleurs n'ont pas cette possibilité :

- Le fait de rajouter une action crée une nouvelle action dans le système. L'action passe alors directement à l'état " Terminée ".
- Le fait de supprimer une action entraîne une demande de précision au mainteneur.

En fonction des actions qui ont été sélectionnées précédemment, l'équipement ou l'emplacement peuvent changer d'état :

- soit automatiquement si une action terminée entraîne un changement de statut. Dans ce cas, le changement sera fait automatiquement par le système, sans intervention de l'agent de maintenance.
- soit l'agent de maintenance peut déclarer le statut de l'équipement à "En service" ou "Hors service" dans certains cas. Les cas où cela n'est pas possible sont définis ci-dessous. Dans le cas d'une déclaration de l'équipement "Hors service", l'agent de maintenance doit définir le problème constaté grâce à une liste finie.

La liste des motifs de déclaration d'un équipement HS sont : Afficheur, Batterie, Bouton, Carte principale, Clavier maintenance, Clavier usager, Imprimante, Lecteur, Modem, Paiement, Pile, PinPad, Sécurité, Autre.

Si le mainteneur choisit le motif "Autre", un commentaire est obligatoire. Dans les autres cas, le commentaire est facultatif.

Si une des actions effectuées a été de changer un matériel géré unitairement, l'agent de maintenance doit saisir les différentes informations nécessaires pour mettre à jour le système. Les informations nécessaires sont :

- Dans le cas du changement du SAM Monéo : le nouvel identifiant du SAM et sa date d'expiration.
- Dans le cas du changement de la SIM, le nouveau numéro de la SIM
- Dans le cas du changement de la serrure, le nouvel identifiant de serrure

Évènements liés à la génération d'un compte rendu

Lors de la saisie du compte rendu, on enregistre la validation des actions terminées dans l'historique des actions. Les informations enregistrées sont :

- État initial = planifiée
- État final = terminée
- Identifiant de l'agent = identifiant de l'agent ayant effectué la validation
- Date de création = date et heure de l'évènement

Dans le cas où il y a la création d'une nouvelle action sur un équipement ou un emplacement, on enregistre la création de l'action dans le journal des événements de l'équipement ou de l'emplacement afin de l'historiser. Les informations enregistrées sont :

- Date et heure de l'évènement = Date et heure de création de l'action
- Le type de l'évènement = Création d'une action
- Le descriptif de l'évènement = Création d'une action de type X par l'utilisateur Y

On enregistre également la création de l'action dans l'historique des actions. Les informations enregistrées sont :

- État initial = vide
- État final = terminée
- Identifiant de l'agent = identifiant de l'agent l'ayant effectué
- Date de création = date et heure de l'évènement

Dans le cas où il y a la suppression d'une action programmée sur un équipement ou un emplacement, on enregistre le changement d'état de l'action dans l'historique des actions. Dans le cas d'une action à reprogrammer, les informations enregistrées sont :

- État initial = planifiée
- État final = non terminée
- Identifiant de l'agent = identifiant de l'agent l'ayant effectué
- Date de création = date et heure de l'évènement

Dans le cas d'une action inutile, les informations enregistrées sont :

- État initial = planifiée
- État final = supprimé
- Identifiant de l'agent = identifiant de l'agent ayant demandé la suppression
- Date de création = date et heure de l'évènement

5.3.6 Obtenir l'historique d'un emplacement ou d'un équipement

Une fonctionnalité pour obtenir l'historique d'un emplacement ou d'un équipement est présente. Cette fonctionnalité permet à un mainteneur de connaître les 10 derniers événements (alarmes, signalements ou actions) qui ont eu lieu sur cet équipement. Cela est utile avant d'effectuer un signalement pour éviter de signaler un problème alors que tout est normal et que l'état constaté de l'équipement

est dû à une action normale.

Remarque : pour utiliser cette fonctionnalité, l'agent doit obligatoirement avoir du réseau car les informations sont récupérées en temps réel.

Pour obtenir cet historique, l'agent doit saisir :

- Soit le numéro de l'équipement.
- Soit l'adresse de l'équipement. Dans ce cas, une recherche de l'équipement est faite côté système. L'agent sélectionne alors l'équipement qui l'intéresse pour obtenir l'historique.

5.3.7 Signalement de dysfonctionnement

Une fonctionnalité de signalement de dysfonctionnement est également présente. Cette fonctionnalité permet à un mainteneur de déclarer un dysfonctionnement sur un équipement. L'équipement doit être identifié de la façon la plus précise possible :

- Soit le mainteneur est à proximité de l'équipement et il peut rentrer l'identifiant unique indiqué sur l'équipement ;
- Soit le mainteneur passe devant et il rentre le numéro de voie et l'adresse de l'emplacement où il y a un problème ;
- Soit il ne dispose pas des informations précédentes et il rentre alors des informations permettant de l'identifier une fois rentré à l'atelier.

Dans les 2 derniers cas, le signalement sera attribué par le système à un équipement virtuel. Ce sera alors au responsable d'interventions de définir précisément de quel équipement il s'agit à partir des informations fournies par le mainteneur et de déclarer une action sur cet équipement. Si les informations ne sont pas suffisantes pour identifier l'équipement, aucune action ne pourra être déclarée.

En plus d'identifier l'équipement, le mainteneur doit également indiquer le problème constaté, parmi une liste finie de problème.

Les éléments de la liste déroulante sont présentés pour avoir les types de signalement les plus fréquents en début de liste. Le choix "Autre" entraîne la saisie obligatoire d'un commentaire pour indiquer le problème rencontré.

Les champs composant un signalement sont donc :

- L'identification de l'équipement
- Le type de dysfonctionnement correspondant au problème constaté
- Un commentaire si nécessaire (obligatoire dans le cas du choix Autre)

Le fait de saisir un signalement de dysfonctionnement n'entraîne pas de changement d'état de l'équipement.

Évènements liés à la création d'un signalement

La réception d'un signalement sur un équipement est historisée dans le journal des évènements de l'équipement en question. Lors de la réception du signalement, les informations enregistrées sont :

- Date et heure de l'évènement = date et heure de réception du signalement
- Le type d'évènement = Réception d'un signalement
- Le descriptif de l'évènement = Création d'un signalement de type X par l'utilisateur Y

5.4 Description des interfaces de l'application

Pour visualiser les différentes interfaces de l'application mobile, voir en annexe de ce rapport.

6. Partie 4 : Conception de l'application

6.1 Solution adoptée

Nous développons une application avec une architecture classique de type Android. Cependant, nous nous sommes interrogés sur l'utilisation d'une architecture Maven. Après concertation avec notre encadrant et d'autres collègues, il s'est avéré que l'utilisation d'une architecture Maven n'était pas conseillée pour notre application Android.

De plus, nous nous sommes demandés si l'utilisation d'un framework comme Android Annotations pour développer plus rapidement pouvait être utile. Suite à des nouvelles discussions avec notre encadrant, nous avons jugé qu'il était plus sûr de développer notre application Android avec du Java dit classique, sans utiliser de framework spécifique. Cela était justifié car nous n'avions jamais développé d'application Android. Il est donc préférable de commencer par coder sans framework.

6.2 Authentification de l'utilisateur

6.2.1 Principe de l'authentification

Pour accéder à l'application mobile, l'utilisateur tombe sur une page d'authentification. Ce dernier doit indiquer son SOI (identifiant unique). L'application se charge alors d'utiliser un Web Service afin de réaliser des requêtes qui permettront à l'utilisateur de s'authentifier.

Nous allons décrire le principe de fonctionnement de l'authentification ainsi que la communication avec le Web Service .

Tout d'abord, le domaine des services web était quelque chose que nous n'avions jamais étudié ni utilisé. Il a donc fallu se documenter sur les services web notamment avec les services de type REST(Representational State Transfer) et SOAP(Simple Object Access Protocol).

6.2.2 Définition d'un web service

Un Web Service est un mécanisme de communication entre applications distantes à travers le réseau internet et qui est indépendant de tout langage de programmation ainsi que de toute plate-forme d'exécution.

Le Web Service dispose d'une architecture REST. L'architecture REST (Representational State Transfer) ou RESTful est un style d'architecture permettant de construire des applications tel que des applications Web, des Intranets ou encore des Web Services. Il faut savoir que l'architecture REST n'est pas une technologie à part entière, c'est en réalité un ensemble de conventions et de bonnes pratiques à respecter. L'architecture REST utilise les spécifications originelles du protocole HTTP, à l'inverse

des architectures SOAP ou XML-RPC qui réinvente une surcouche. Avant de rentrer plus dans les spécifications techniques de l'architecture REST, voici la répartition des architectures utilisées :

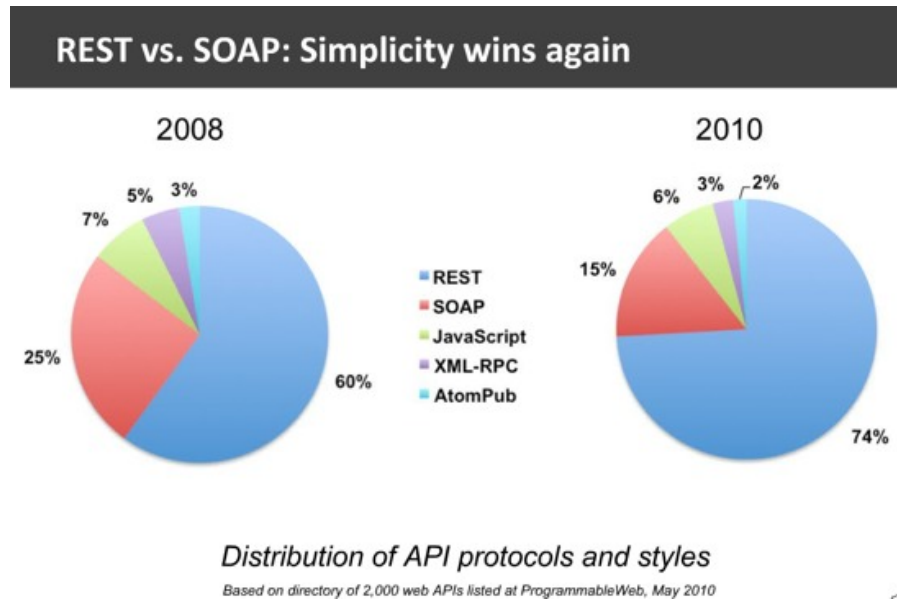


FIGURE 6.1 – Soap vs Rest

Nous pouvons constater que l'architecture REST est la plus répandue. Nous nous sommes aussi documenté sur l'architecture SOAP mais nous détaillerons seulement l'architecture REST, car c'est cette dernière qui est utilisée dans notre Web Service. L'architecture REST se base sur les URI (Uniform Resource Identifier) afin d'identifier une ressource. Ainsi une application se doit de construire ses URI (et donc ses URL) de manière précise, en tenant compte des contraintes REST. Il est nécessaire de prendre en compte la hiérarchie des ressources et la sémantique des URL pour les éditer.

Comme nous l'avons dit plus haut l'architecture REST utilise les spécifications originelles du protocole HTTP. Ainsi les verbes HTTP existants POST, GET, PUT, DELETE sont utilisés et correspondent respectivement aux verbes Créer(Create), Afficher(Read), Mettre à jour(Update) et Supprimer(Delete).

Un point important en architecture REST est que les réponses HTTP sont la représentation des ressources et non une ressource. Cela garantit de multiples représentations des ressources et laisse à l'auteur de la requête le choix de la représentation de la ressource demandée dans divers format tels que HTML, XML, CSV, JSON, etc.

Pour authentifier une requête en architecture REST, le principe est simple : Chaque requête est envoyée avec un jeton (token) passé en paramètre de la requête (requête GET/afficher). Ce jeton d'authentification est obtenu en envoyant une première requête d'authentification qui sera ensuite combinée avec nos requêtes. Ce token sera ensuite utilisé pour générer un hash de la requête, et ce même hash est passé comme jeton d'authentification afin de valider l'authentification pour une requête.

Maintenant que nous avons décrit le principe de fonctionnement de l'authentification en architecture

REST, nous allons voir plus concrètement comment notre application interagit avec le Web Service d'authentification.

6.2.3 Outil simulant l'appel à un web service

De nombreux appels à différents web service sont présents dans notre application. Afin de faciliter l'utilisation de ces derniers, nous testons l'appel à ces web service grâce à une extension présente sous Mozilla Firefox appelée POSTER.

Cette extension donne la possibilité de simuler l'appel à un web service. Pour simuler l'appel au web service permettant de récupérer un équipement suivant un identifiant donné, il faut procéder de cette façon :

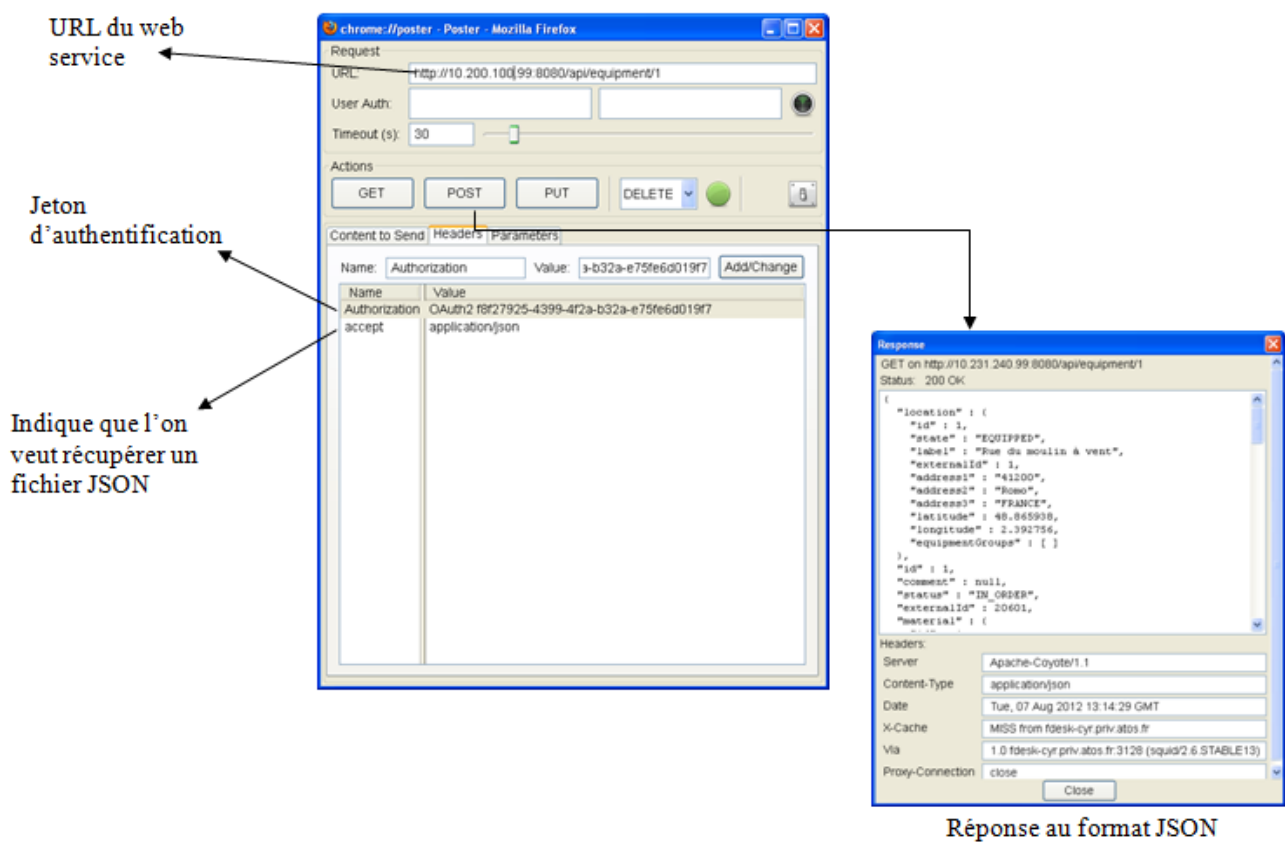


FIGURE 6.2 – Simulation d'un appel à un web service avec Poster

Grâce à cet outil, il est alors plus facile de récupérer les informations que l'on souhaite dans la réponse qui est au format JSON. Pour cela, il est possible de parser cette réponse avec le langage JAVA.

6.2.4 Authentification dans l'application

Dans notre situation, la communication avec le Web service se présente de la manière suivante :

- une requête POST pour l'identification de l'utilisateur
- une requête GET pour finaliser l'authentification dans le but d'accéder à la ressource demandée

La requête POST

L'utilisateur saisit son SOI (identifiant unique) dans l'interface d'accueil. Ce SOI est combiné avec un mot de passe ainsi qu'avec les autres informations d'identifications (non saisie par l'utilisateur mais transmises avec la requête). Afin de ne pas faire apparaître en clair, le mot de passe pour se logger à l'application nous avons dû crypter celui-ci. Nous nous sommes donc documentés sur la méthode de cryptage en AES - Rijndael en mode CBC qui fonctionne avec une clé de déchiffrement, un vecteur d'initialisation ainsi qu'un mot de passe chiffré. Une fois le mot de passe déchiffré et le SOI renseigné, la requête POST est désormais prête à être exécutée avec ces paramètres.

Une fois exécutée, le Web Service nous retourne un jeton (token) valable pour une certaine durée (qui est précisée dans la réponse). Nous devons ensuite combiner ce jeton d'authentification avec notre requête GET.

La requête GET

Le jeton d'authentification est passé dans le Header de la requête GET. Une fois la requête GET exécutée, le Web service renvoie un document XML (à ce stade l'authentification est complète au niveau du serveur) que l'on récupère dans une chaîne de caractères. Il a donc fallu transformer cette chaîne en document DOM afin de parser le document, traiter chaque balise et récupérer les informations contenues dans celui-ci. Une fois les informations récupérées nous pouvons les stocker afin de les réutiliser et finaliser l'authentification au niveau du client (l'utilisateur est redirigé vers le menu principal de l'application).

Finalement, lorsque l'utilisateur est bien authentifié, ce dernier peut accéder à l'application permettant de gérer les horodateurs.

Un mécanisme d'alerte (gestion des erreurs HTTP et des erreurs de saisies) est évidemment mis en place afin de prévenir l'utilisateur du succès ou de l'échec de son authentification.

6.3 Consultation de l'historique

L'application permet de récupérer l'historique des actions pour un équipement donné. L'utilisateur a la possibilité de saisir l'identifiant de l'équipement ou l'adresse d'un emplacement. Cette dernière va récupérer le ou les équipements présents à l'adresse donnée puis les afficher. L'utilisateur pourra alors choisir l'équipement qu'il souhaite.

Afin de réaliser ceci, il faut utiliser différents web service :

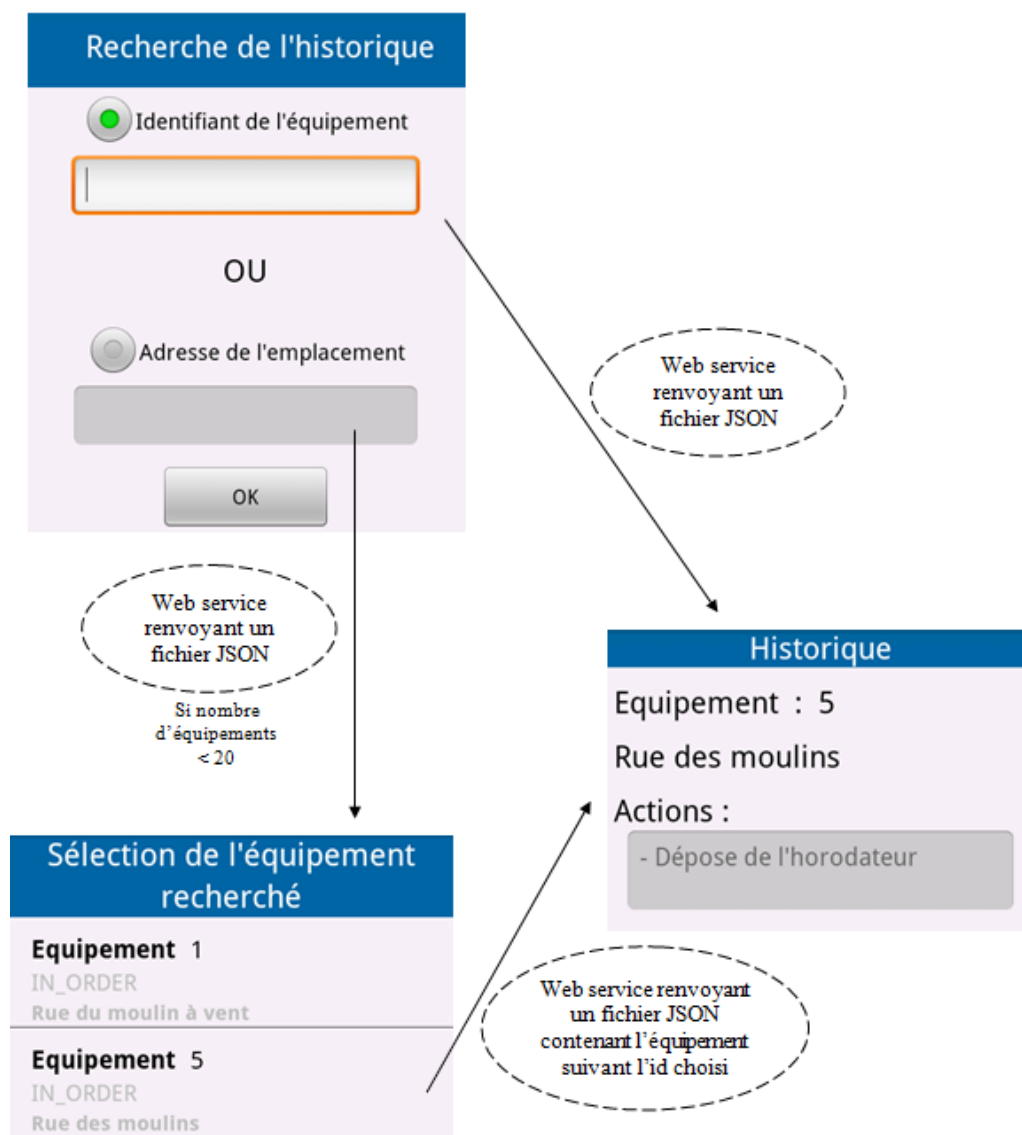


FIGURE 6.3 – Historique des actions

7. Partie 5 : Implémentation de l'application

7.1 Eclipse

Nous avons développé notre application Android avec l'environnement de développement open-source Eclipse. Ce dernier permet de développer avec le langage de programmation Java, créé par la firme Sun Microsystems. Eclipse donne la possibilité d'émuler une application Android en installant un plugin spécifique. Nous avons ensuite installé une AVD (Périphérique Android Virtuel/Android Virtual Device). Nous avons choisi la version de l'API Android 2.2 car cette dernière permet de rester compatible avec la majorité des mobiles Android sur le marché.

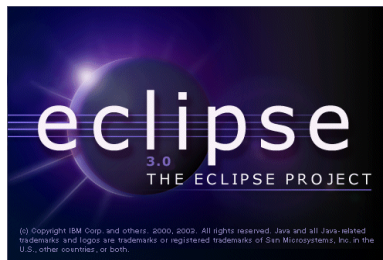


FIGURE 7.1 – Mercurial logo

7.2 Visual Studio

Nous avons utilisé Visual Studio afin de manipuler l'application Windows Mobile. Visual Studio est un ensemble complet d'outils de développement permettant de générer différents types d'applications.



FIGURE 7.2 – Visual Studio 2008

7.3 Problèmes rencontrés

Durant mon stage, j'ai évidemment rencontré plusieurs problèmes concernant le développement, mais ces derniers étaient, en général, résolus assez rapidement.

Le soucis majeur de ce stage a été que l'application que je devais développer était basée sur une autre application existante. Or, celle-ci était encore en cours de développement durant mon stage. De ce fait, il fallait se mettre à jour régulièrement du travail des personnes travaillant sur l'application Windows Mobile. Ces mises à jour entraînaient beaucoup de modifications et donc beaucoup de jours de développement à ajouter par rapport au planning initialement défini.

Étant donné que je n'avais jamais développé d'application mobile Android, j'ai dû me documenter et découvrir, durant une certaine période, les principes et le fonctionnement du développement Android. Mais j'ai déjà été confronté à la découverte d'un langage ou d'une nouvelle technologie durant mes anciens stages.

8. Partie 6 : Missions annexes

Durant mon stage, il y a eu deux missions annexes relativement importantes. Je considère celles-ci comme annexes car elles n'ont soit pas de lien direct avec mon sujet de stage, soit car elles concernent un autre sujet.

8.1 Modification de la charte graphique d'une application mobile

Durant la première de stage, nous avons travaillé sur le style graphique d'une application web pour mobile. En effet, notre tuteur a préféré attendre une semaine avant de nous attribuer notre sujet de stage. Cela est dû au fait que l'entreprise ATOS Worldline était en attente de la réponse d'une société qui aurait conduit à la création d'un nouveau projet auquel nous aurions pu être affectés.

Notre objectif était de modifier et d'améliorer le style graphique d'une application web pour mobile car l'équipe à laquelle nous appartenons devait livrer cette application web pour la fin de la semaine.

Nous avons d'abord commencé par nous documenter sur le développement d'applications web mobiles jQuery Mobile. Il fallait aussi découvrir le code source de l'application afin de pouvoir modifier ce code.

Cette application mobile permet à un utilisateur d'acheter une e-vignette directement sur son téléphone. Cette vignette a pour but de réduire le prix de la taxe payée par le conducteur du véhicule. De plus, l'utilisateur peut aussi consulter les tarifs.

Au départ, l'application mobile ressemblait à ceci :

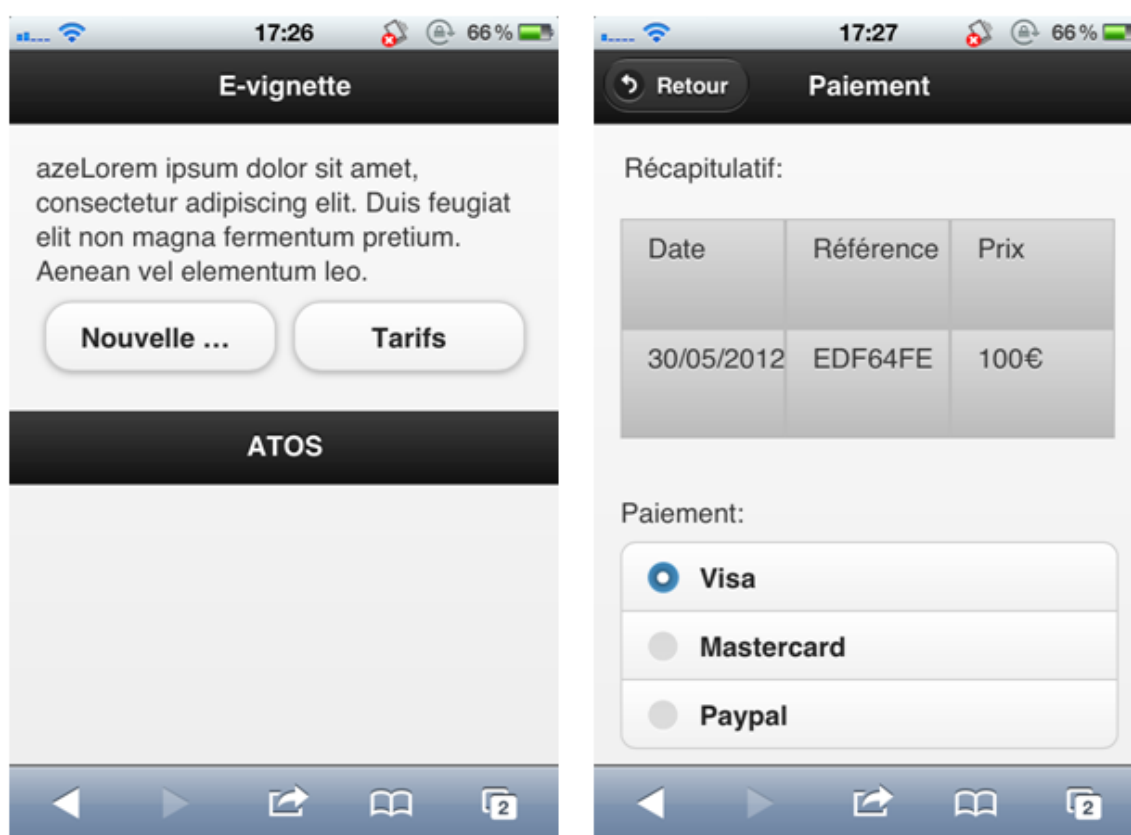


FIGURE 8.1 – Charte graphique avant modification

Une fois les sources de l'application récupérées, nous avons alors modifié directement le code grâce aux outils appropriés. Nous avons aussi utilisé un outil présent sur le web qui nous permettait de créer un thème avec jQuery Mobile.

A la fin de la semaine, l'application ressemblait à ceci :

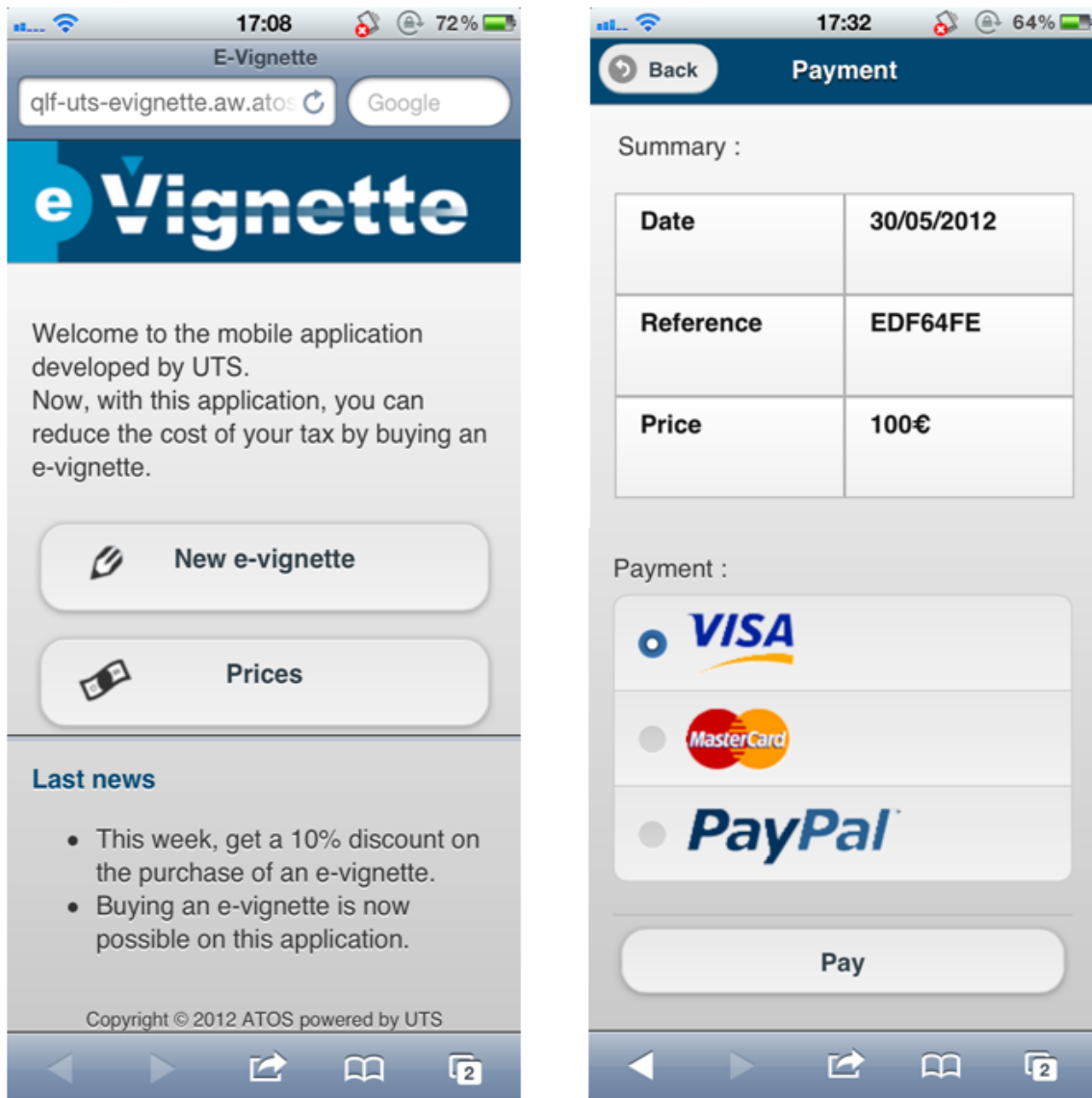


FIGURE 8.2 – Charte graphique après modification

Nous testons l'application mobile directement grâce à notre téléphone personnel mais aussi avec d'autres appareils que l'entreprise nous a prêtés. Nous avons donc testé sur :

- Téléphone Iphone 4 sous IOS
- Téléphone Google Nexus sous Android
- Téléphone Motorola Android
- Tablette Motorola sous Android

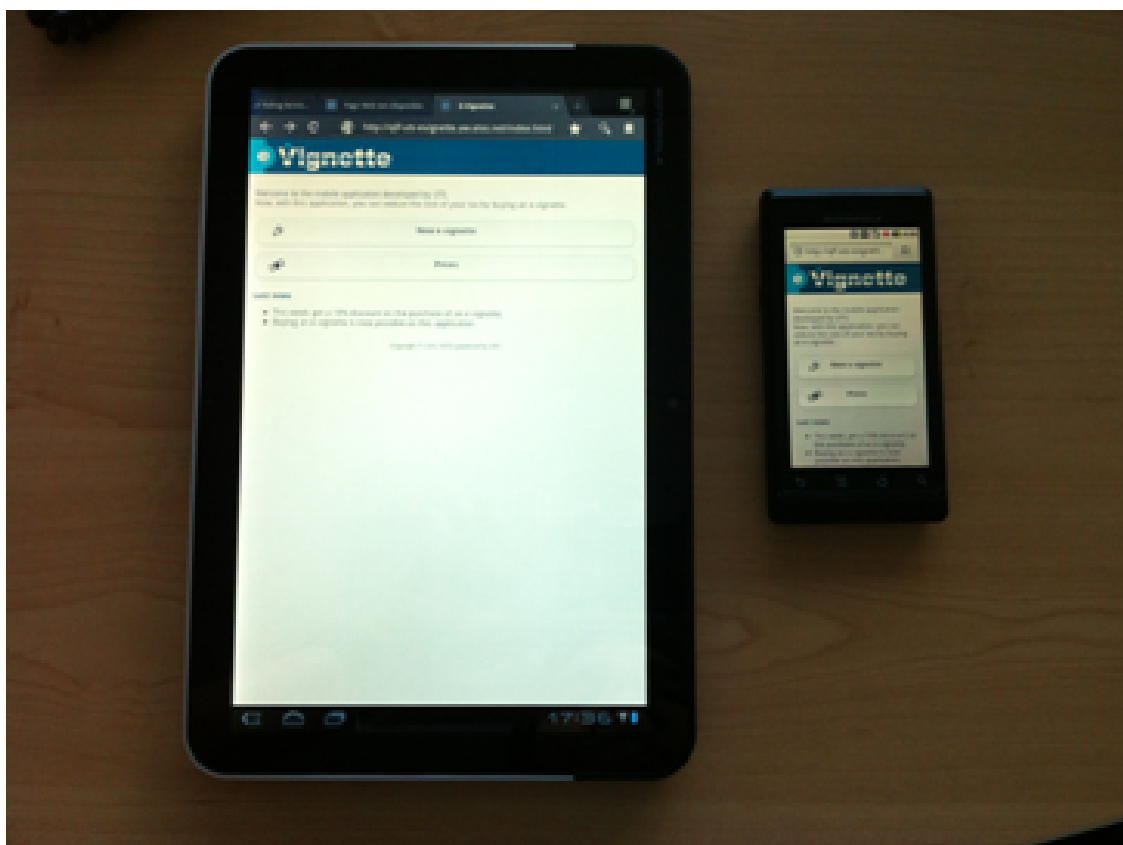


FIGURE 8.3 – Appareils mobiles pour tester l'application

Cela nous permettait de visualiser le rendu de l'application sur différents appareils. De ce fait, le style graphique mis en place devait être optimisé pour tout type d'appareil.

8.2 Déploiement d'une mise à jour sur les PDA Windows Mobile

La personne qui développait l'application Windows Mobile a fait appel à mon aide afin de savoir comment il était possible de déployer une mise à jour sur les PDA de l'entreprise. Cette mission étant prioritaire, j'ai laissé de côté mon projet et j'ai étudié cette question durant quatre jours.

8.2.1 Objectif

L'objectif de cette mission est de rechercher une solution pour déployer les mises à jour de l'application Windows Mobile sur les 20 PDA de l'entreprise à partir d'un serveur qui centralise les mises à jour.

8.2.2 Fonctionnement

L'application présente sur les PDA devra se connecter à distance à un Web Service afin de vérifier s'il y a eu une MAJ et dans le cas où il y en a une, il faudra déployer cette dernière (fichier .CAB contenant toute l'application) sur les PDA.

Deux Web Services devront donc être mis en place :

- Le premier Web Service se chargera de renvoyer un simple booléen (vrai si une MAJ est à faire, faux sinon).
- Le second Web Service permettra de récupérer la version à jour de l'application mobile (.CAB). Ce Web Service ne sera utilisé que dans le cas où une MAJ est à faire évidemment.

Une fois le fichier .CAB téléchargé il suffit de l'installer depuis l'application grâce à l'appel de wceload.exe.

8.2.3 Utilitaire wceload

La documentation officielle de Microsoft sur cet outil :

<http://msdn.microsoft.com/en-us/library/ms926281.aspx>

FIGURE 8.4 – Déploiement lien documentation officielle

L'utilisation de cet outil pourrait ressembler à ceci :

```
Process proc = Process.Start(\\windows\\wceload.exe, "\"" + cheminVersCAB + "\"");  
proc.WaitForExit();
```

FIGURE 8.5 – Déploiement code

Où la variable cheminVersCAB pourrait être :

```
\\APPEXported\\PatchInstaller\\AppPDAVer0.1.CAB
```

FIGURE 8.6 – Déploiement chemin

A ce moment là, la question était de savoir si l'exécutable wceload était bien présent sur le PDA.

Ce fichier n'est pas visible lorsque l'on parcourt le dossier Windows de l'Émulateur (fourni avec Visual Studio) et du PDA. Cependant, lorsque j'ai tenté de télécharger manuellement cet exécutable puis lorsque j'ai tenté de le déplacer dans le dossier Windows de l'Émulateur, ce dernier m'indiquait que ce fichier était déjà présent. Cela indique donc que l'on ne peut pas consulter tous les types de fichiers sur l'Émulateur et le PDA.

On peut donc supposer que la situation est semblable sur le PDA et que le fichier exécutable est bien existant. Ceci s'avère être le cas puisque l'exécution de wceload fonctionne sur le PDA.

8.2.4 Manipulation

Il a fallu se documenter afin de comprendre comment procéder pour générer le fichier .CAB. Le but est donc de créer 2 fichiers .CAB différents. Il y aura donc une version 1 de l'application et une version 2.

Voici comment j'ai testé l'utilitaire wceload :

- Ajout de la ligne de code permettant d'utiliser wceload dans le projet grâce à un bouton de test
- Déploiement de ce projet sur le PDA

- Modification de la source du projet pour distinguer la version 1 de la version 2 puis déploiement et copie du fichier .CAB dans un dossier de test (ici nous choisirons le dossier TEMP du PDA)
- Test de l'utilitaire wceload sur la version 1 du projet afin de lancer l'installation de la version 2

Avant tout manipulation, j'ai installé un certificat sur le PDA. Ce certificat permet au PDA de reconnaître les applications SGCHMobile. Pour ce faire, il faut déposer le fichier AWL-CodeSigning-Cert.cab sur le PDA puis l'exécuter. Ce dernier indique alors que le certificat a bien été installé.

Modification du code de la version 1

J'ai d'abord créé un bouton de test sur le menu principal de l'application et ai modifié les intitulés du menu afin de bien distinguer la version 1 et la version 2.



FIGURE 8.7 – Déploiement modification version 1

Sur ce bouton de test, j'ai ajouté les lignes de code permettant d'exécuter un fichier .CAB

```
private void button1_Click(object sender, EventArgs e)
{
    Process proc = Process.Start("wceload.exe", "/silent /noui \"%\\Temp\\testMAJ.CAB\"");
    proc.WaitForExit();
}
```

Mode transparent
pour l'utilisateur

Chemin du fichier .CAB
de la version 2

FIGURE 8.8 – Déploiement modification version 1 code

Ici j'ai choisi que le fichier testMAJ.CAB de la version 2 se trouvera dans le dossier TEMP du PDA. Cet emplacement correspond à l'endroit où se trouvera le fichier téléchargé par le futur Web Service.

Déploiement de la version 1 sur le PDA

Pour déployer la première version de l'application il faut créer un nouveau projet sous Visual Studio. Il faut alors créer un projet de type "Projet CAB Smart Device".

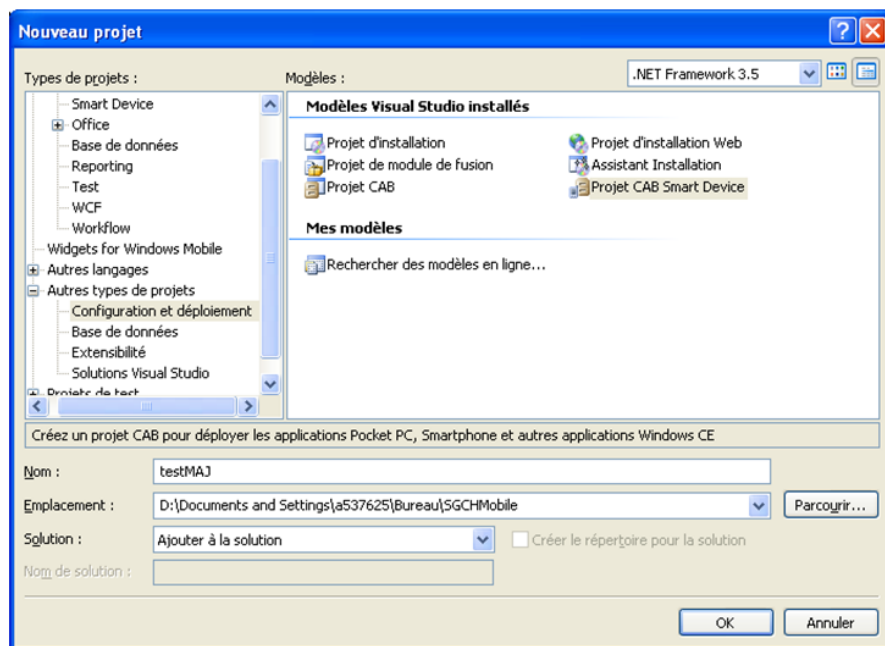


FIGURE 8.9 – Déploiement création projet

Une fois le projet crée, il faut choisir la sortie du projet via le dossier nommé "Dossier d'application".

Certification et installation de l'application

Pour certifier l'application, il faut aller dans les propriétés du nouveau projet. J'ai alors dû cocher la signature Authenticode puis "à partir du magasin". On peut aller gérer les certificats afin d'importer le fichier AWL-CodeSigning-key.pfx. Une fois l'importation réussie, ce dernier devrait apparaître dans la liste des certificats. Il faut alors le sélectionner puis l'appliquer.

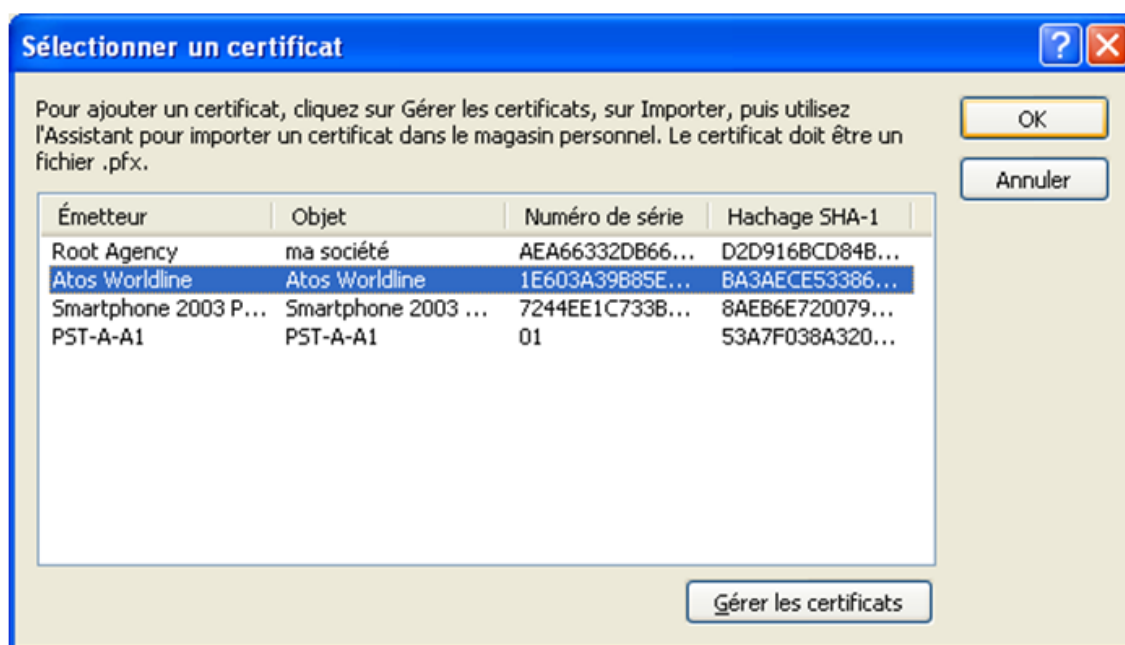


FIGURE 8.10 – Déploiement sélection du certificat

Une fois que le projet est généré puis que le fichier .CAB du projet est copié sur le PDA (fait grâce à ActiveSync), il faut alors installer l'application grâce au fichier .CAB.

Modification du code de la version 2 et déploiement

Une fois l'installation de la version 1 terminée, j'ai supprimé le projet testMAJ que je viens de créer puis ai modifié le menu de l'application afin de bien distinguer la version 1 et la version 2.

La phase de déploiement est identique à celle précédemment effectuée.

Attention à ne pas oublier de signer l'application avec le certificat approprié.

J'ai alors déposé le fichier .CAB de la version 2 dans le dossier TEMP du PDA.

Exécution du test sur le PDA

Le test peut maintenant être effectué. Il faut alors lancer la version 1 de l'application déjà installée puis cliquer sur le bouton de test.

Après une attente de 10 secondes environ, l'utilisateur retombe à la racine de son application. Il lui suffit de relancer celle-ci et on remarque bien que la version 2 a bien remplacé la version 1.

A noter que le fichier .CAB de la version 2 présent dans le répertoire TEMP du PDA a été effacé automatiquement après l'installation.



FIGURE 8.11 – Déploiement exécution du test

8.2.5 Conclusion

L'installation d'un fichier .CAB depuis une application est donc désormais possible grâce à l'utilitaire wceload.

J'ai eu beaucoup de problèmes concernant la certification afin de ne plus avoir de pop-up signalant que l'application, lors de l'installation, provenait d'un éditeur inconnu. Grâce à Antoine Locket ce problème a été résolu.

D'un point de vue personnel, cette mission m'aura confronté à un problème concernant un domaine où je n'avais aucune connaissance technique. J'ai donc dû m'adapter et laisser de côté mon projet de stage afin de répondre à une demande urgente de mon tuteur.

9. Conclusion

Ce stage a été très bénéfique et très enrichissant tant sur le plan professionnel que personnel. Cette nouvelle expérience professionnelle dans une SSII de taille importante m'a appris beaucoup de choses et permis de découvrir le monde des SSII, qui m'était inconnu.

Réaliser une application Android m'aura permis de découvrir le développement mobile qui m'étais inconnu. Je souhaitais monter en compétence sur l'univers du développement Android, je suis donc pleinement satisfait de mon stage.

De plus, j'ai eu la chance d'intégrer une jeune équipe dynamique et à l'écoute. En effet, s'il s'avérait que je bloquais sur un problème de développement par exemple, je pouvais aisément demander à un collègue de mon équipe afin de lui demander de l'aide.

Annexes

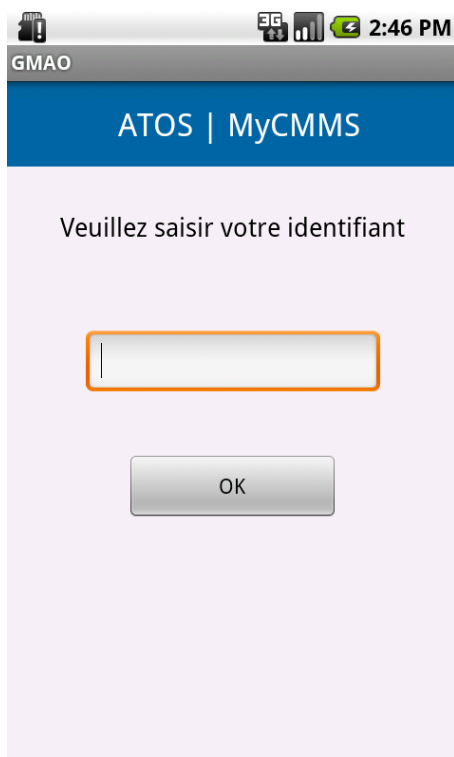


FIGURE 1 – Interface login

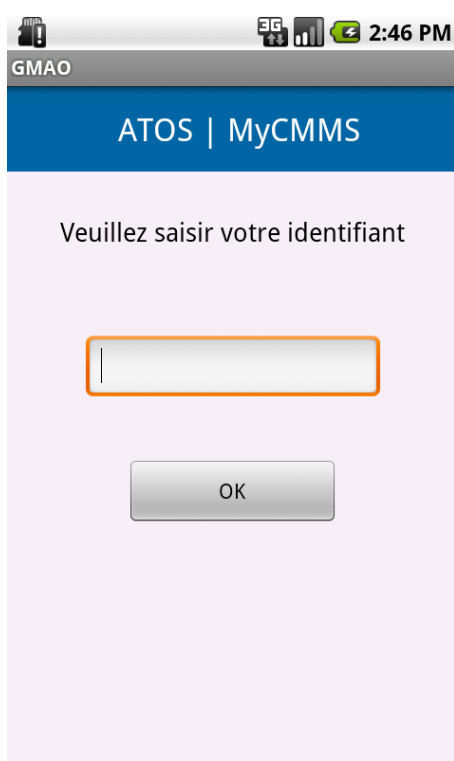


FIGURE 2 – Interface menu

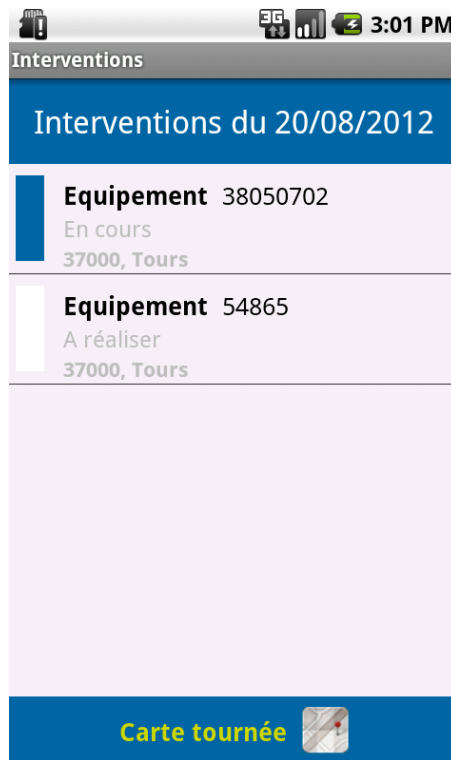


FIGURE 3 – Interface permettant de visualiser la liste des interventions



FIGURE 4 – Interface permettant de visualiser une intervention choisie



FIGURE 5 – Interface recherche de l'historique

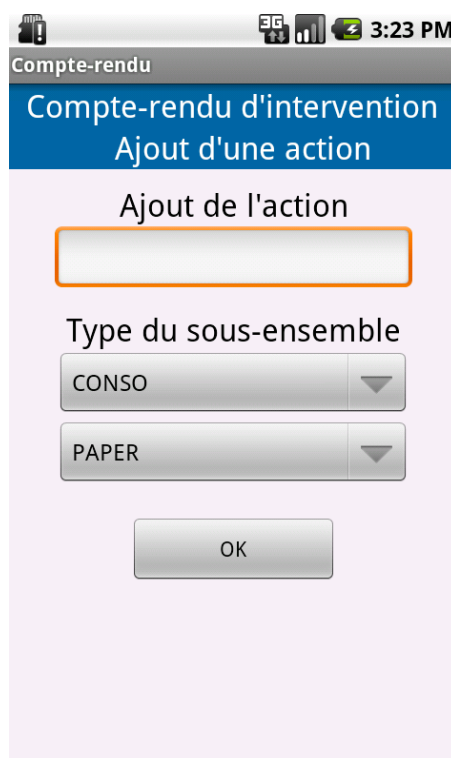


FIGURE 6 – Interface compte rendu et ajout d'une action



FIGURE 7 – Interface compte rendu de la liste des actions

Portage d'une application de GMAO Windows Mobile sous Android

Département Informatique
4^e année
2011 - 2012

Rapport de stage

Résumé : Ce stage a consisté à porter une application existante Windows Mobile sous Android. L'application existante permet de gérer les horodateurs de la Mairie de Paris. Cela permet à un agent de consulter sur un PDA les interventions qu'il doit effectuer un jour donné. L'application Android devait répondre à une offre plus globale où les horodateurs sont remplacés par des équipements. Il s'agit alors d'une GMAO (Gestion de Maintenance Assistée par Ordinateur) fonctionnant directement sur un PDA Android.

Mots clefs : GMAO - Windows Mobile - Android - développement - gestion de projet - gestion d'équipements

Abstract: The goal of this course was to bring an existing application Windows Mobile to Android operating system. The existing application permits to manage timestamps of the City of Paris. This offers the possibility to an agent to consult on a PDA which interventions should be performed a given day. The Android application should respond to a wider offer, where timestamps are replaced by equipment. Then it is a CMMS (Computerized Maintenance Management System) working directly on a PDA Android.

Keywords: CMMS - Windows Mobile - Android - development - project management - equipment management

Encadrant
Thomas TATU
thomas.tatu@atos.net

Étudiant
Pierre Fourreau
pierre.fourreau@etu.univ-tours.fr