



Université du Québec  
**École de technologie supérieure**

RAPPORT TECHNIQUE  
PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE  
DANS LE CADRE DU COURS GTI792 PROJET DE FIN D'ÉTUDES EN GÉNIE DES TI

**SERVICES ÉTS MOBILE**

ALEXANDRE GIRARD  
GIRA15098706

DÉPARTEMENT DE GÉNIE LOGICIEL ET DES TI

**Professeur superviseur**

**ALAIN APRIL**

MONTRÉAL, 11 AVRIL 2011  
HIVER 2011

## REMERCIEMENTS<sup>1</sup>

Pour arriver à accomplir les objectifs établis par le projet de fin d'études, il l'a fallu l'aide de plusieurs personnes. Cette section est donc dédiée à souligner et remercier ses personnes qui sans leur appuies se projet de fin d'études n'aurait pu être possible.

Premièrement, je tiens à remercier mon collègue de travail, Mathieu Lemay, qui en premier lieu a eu l'idée du projet et qui a été tout au long un bon partenaire avec de bonnes aptitudes de travaux et ayant une bonne compréhension de la méthodologie à suivre pour mener à terme le projet ce qui m'a grandement aidé à atteindre les objectifs ce celui-ci.

De plus, j'aimerais remercier mon superviseur de projet, Alain April, pour son implication dans le projet, que ce soit pour la bonne orientation donnée en début de projet, sa disponibilité pour les rencontres et pour avoir facilité la mise en contact avec les personnes ressources de Cheminot.

Finalement, je tiens à remercier le personnel de soutien de l'école et plus encore les professeurs côtoyés au cours de mon baccalauréat en génie des technologies de l'information. Ils m'ont premièrement permis d'obtenir un cheminement de qualité universitaire en génie des TI et également de m'avoir appris les connaissances nécessaires à ma future carrière et qui ont été appliquées dans le cadre de ce projet de fin d'études.

---

<sup>1</sup> La section Remerciements est basée sur le rapport d'équipe, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté.

# **RAPPORT FINAL INDIVIDUEL SERVICES ÉTS MOBILE**

**ALEXANDRE GIRARD  
GIRA15098706**

## **RÉSUMÉ<sup>23</sup>**

Les services en ligne offerts aux étudiants, dans une université, sont multiples et essentiels pour la réussite de tous. Il est donc très important d'y retrouver le maximum d'informations nécessaires et qu'on puisse y accéder de n'importe quel endroit. À l'École de technologie supérieure (ÉTS), plusieurs services sont disponibles via l'application en ligne *Cheminot*. On y retrouve principalement l'inscription aux cours, la visualisation de l'horaire et du dossier étudiant ainsi que le cheminement universitaire de l'étudiant avec les différents cours suivis. Il n'est actuellement pas possible d'accéder à ces services lorsqu'on utilise un appareil mobile, et ces appareils mobiles sont de plus en plus populaires auprès des étudiants.

Ce projet, Services ÉTS Mobile, consiste à concevoir une application mobile, pouvant être exécutée sur les appareils iPhone et iPod touch, qui regroupera les services les plus utilisés par les étudiants afin qu'ils puissent accéder aux informations qu'ils veulent, peu importe leur emplacement. Les services qui seront disponibles sont l'inscription aux cours, la visualisation de l'horaire, la visualisation du dossier étudiant et l'inscription aux cours. Tous ces services seront donc accessibles par les étudiants lorsqu'une connexion Internet sans fil ou 3G sera à leur portée via leur appareil mobile.

Ce rapport vous présentera les différentes phases du projet, soit l'analyse, la conception et l'implémentation de Services ÉTS Mobile. Par contre, tout n'est pas présenté dans ce rapport, car les tâches ont été séparées en deux (2) pour les membres de l'équipe. On y retrouve seulement les éléments qui sont liés au choix de cours que ce soit pour l'ajout d'un cours à l'horaire, la suppression d'un cours à l'horaire et la confirmation d'un choix de cours.

Les éléments de ce rapport qui vous seront présentés sont, en premier lieu, l'exposé du problème, dans lequel on retrouve la problématique, le contexte, la description du projet ainsi que les objectifs. Ensuite, il sera question de la valeur technique et de conception, c'est-

---

<sup>2</sup> La section Résumé est basée sur le rapport d'équipe, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté.

<sup>3</sup> La section Résumé est basée sur le rapport individuel, par Mathieu Lemay. Par contre, le contenu a été adapté.

à-dire, de la méthodologie utilisée, les hypothèses et les analyses. Les résultats seront également présentés dans une description complète.

## TABLE DES MATIÈRES

INTRODUCTION .....	1
EXPOSÉ DU PROBLÈME .....	3
1.1 Problématique .....	3
1.2 Description du projet.....	5
1.3 Objectifs.....	6
MÉTHODOLOGIE ET ÉLABORATION DE LA SOLUTION.....	8
1.4 Méthodologie .....	8
1.4.1 Approche.....	8
1.4.2 Outils utilisés .....	10
1.5 Livrables et planification .....	12
1.5.1 Description des artéfacts .....	12
1.5.2 Planification .....	13
RÉSULTATS 14	
1.6 Résultats du projet.....	14
1.6.1 Analyse .....	14
1.6.2 Conception .....	18
1.6.3 Élaboration de la solution .....	23
1.7 Discussion des résultats .....	30
1.8 Difficultés rencontrées .....	31
1.9 Améliorations possibles .....	31
CONCLUSION.....	33
RECOMMANDATIONS .....	35
RÉFÉRENCES BIBLIOGRAPHIQUES.....	36

## **LISTE DES TABLEAUX**

Tableau 1: Liste des artéfacts produits .....	12
Tableau 2: Portée du projet .....	15
Tableau 3: Risques.....	16
Tableau 4: Description des tables de la BD .....	22
Tableau 5 : Correspondance pour l'horaire .....	28

**LISTE DES FIGURES**

Figure 1: Architecture matérielle .....	19
Figure 2 : Schéma de la base de données.....	21
Figure 3: Écran de l'horaire pour le choix de cours .....	25
Figure 4 : Écran d'ajout de cours à l'horaire .....	26

## LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES<sup>4</sup>

<b>App Store</b>	Boutique en ligne pour se procurer des applications pour un appareil mobile Apple.
<b>Applet</b>	Un applet est un logiciel s'exécutant dans la fenêtre d'un navigateur web, Cheminot est un applet.
<b>BD</b>	Acronyme utilisé pour Base de données. C'est là où sont emmagasinées les données sur les dossiers étudiants, offre de cours, etc.
<b>Cheminot</b>	Système actuel utilisé par l'École de technologie supérieure pour offrir les services d'inscription et de consultation du dossier de l'étudiant en ligne.
<b>Connexion 3G</b>	Permet l'accès haute vitesse à Internet via un réseau cellulaire.
<b>Diagramme UML</b>	Langage universel qui permet de modéliser afin d'avoir une vue plus claire du problème.
<b>Dossier étudiant</b>	Dossier rattaché à un étudiant unique qui permet de suivre les différentes informations reliées à celui-ci.
<b>ÉTS</b>	Sigle pour École de technologie supérieure.
<b>Étudiant</b>	Un étudiant, dans notre contexte, représente une personne ayant un profil étudiant dans le système de l'ÉTS, qui est inscrite à au moins 1 cours.
<b>iOS</b>	Système d'exploitation installé sur les appareils mobiles suivants : iPhone, iPod touch, iPad.
<b>iPhone / iPod touch</b>	Appareils mobiles offerts par la compagnie Apple pour lesquels la solution s'intéresse.

---

<sup>4</sup> Cette liste est reprise du document SRS de Services ÉTS Mobile, par Alexandre Girard et Mathieu Lemay



<b>Java</b>	Technologie présentement utilisée par le système Cheminot, qui nécessite une machine virtuelle Java installée sur le poste.
<b>Objective-C</b>	Langage de programmation utilisé par tous les appareils et ordinateurs de la compagnie Apple.
<b>Services ÉTS Mobile</b>	Application mobile permettant aux étudiants de l'École de technologie supérieure l'inscription aux cours et de consulter leur dossier.
<b>SRS</b>	Document qui présente les spécifications des exigences logicielles.
<b>Transaction</b>	Une opération effectuée par un étudiant nécessitant un intrant et produisant un extrant qui est attaché avec un marqueur temporel et qui affecte un changement sur le dossier étudiant sur la BD.
<b>WiFi</b>	Technologie de communication sans fil utilisée par un appareil mobile afin d'avoir accès à Internet en passant par un routeur sans fil.

## INTRODUCTION<sup>5</sup>

La popularité des appareils mobiles intelligents utilisant les réseaux sans fil 3G ou WIFI prouve que la mobilité est un aspect fort important de l'informatique en cette nouvelle décennie. Tout le monde, aime accéder à l'information que ce soit au niveau des nouvelles, de leur travail, de leur école en peu de temps et partout. Cette nouvelle tendance porte l'informatique a un autre niveau, soit celui de l'intégration des solutions actuelles vers les appareils mobiles populaires.

Dans le cadre du projet de fin d'études, le choix de sujet effectué en collaboration avec mon partenaire Mathieu Lemay a été grandement influencé par la popularité des téléphones intelligents. De plus, nombreux sont les étudiants qui ont éprouvé des problèmes ou des insatisfactions face à la solution actuelle de l'école *Cheminot* qui utilise une technologie vieillissante nommée *Applet*. Ainsi, l'objectif de ce projet est donc de concevoir une application mobile, pouvant être exécutée sur un appareil intelligent populaire, dans ce cas il a été convenu dans la phase d'analyse qu'il s'agissait des appareils de la plateforme Apple soit le iPhone et iPod touch, qui permettrait d'intégrer des services du Cheminot actuel sous une version mobile. Il faut par contre noter que le but de cette application n'est pas de faire concurrence ou de remplacer la solution actuelle de Cheminot mais plutôt d'offrir une version mobile allégée de différents services de cheminot pour les utilisateurs des appareils iPhone et iPod Touch. Cette application permettra ainsi aux étudiants de l'école d'effectuer son choix de cours sur la route, peu importe où qu'ils sont. Du même coup, ils pourront visionner leur horaire tel qu'il est pour la nouvelle session à venir d'ajouter et de supprimer des cours à celle-ci et finalement de confirmer leur choix de cours pour appliquer les changements effectués.

---

<sup>5</sup> L'introduction est inspirée du rapport d'équipe, par Alexandre Girard et Mathieu Lemay, mais elle a été adaptée au contenu.

Ce rapport permettra donc de résumer le projet en abordant les principaux échelons. Le rapport est divisé en différentes sections soient la problématique, la description du projet, les objectifs de celui-ci, la méthodologie suivie, les livrables qui ont été produits et une discussion sur les résultats, les problèmes encourus et les possibles améliorations. Finalement la dernière partie du rapport portera sur les résultats et recommandations afin de mettre en opération une telle application étant basé sur la preuve de concept finale quant au service de choix de cours.

## EXPOSÉ DU PROBLÈME

### 1.1 Problématique<sup>6</sup>

Jusqu'à ce jour, les étudiants qui veulent consulter leurs informations personnelles, soit leur dossier étudiant, leur horaire, l'évolution de leur moyenne, le cheminement dans leur programme où lorsqu'ils veulent s'inscrire aux cours, doivent nécessairement utiliser l'application *Cheminot* via le site web de l'université. *Cheminot* est un système central regroupant l'ensemble des services automatisés offert aux étudiants par l'université. Il représente donc un point d'accès critique et essentiel lors des périodes de pointe où les étudiants ont besoin de ces différents services. Pendant ces périodes, l'achalandage du site est très élevé. Par exemple, lors de la période d'inscription aux cours pour chaque session, lors de la modification de ce choix de cours.

Le problème étant que l'utilisation de *Cheminot* pose plusieurs contraintes quant à l'accessibilité et l'interopérabilité. Premièrement, l'étudiant doit avoir accès à un ordinateur muni d'une connexion internet et d'un navigateur Internet compatible avec la technologie Java SUN qui permet de lancer un module Applet sur la machine client. Les facteurs de qualité du logiciel mentionné ci-dessus sont donc mis en cause, car beaucoup d'étudiants se servent de leur téléphone cellulaire pour leur navigation Internet, par exemple d'un iPhone de Apple, et celui-ci ne supporte pas *Cheminot* puisque l'appareil ne peut supporter l'affichage d'un applet Java.

---

<sup>6</sup> La problématique est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté à la version finale du projet.

Le client *Applet* est également une technologie vieillissante et de moins en moins utilisée sur le marché. Un tel type de client oblige le client à s'occuper de la majeure partie du traitement ce qui rentre en conflit avec les différents aspects de qualité logicielle dans une architecture client-serveur. Ainsi, le lancement de *Cheminot* est précédé d'une série d'opérations qui affecte le critère de performance de l'application. À chaque ouverture, les étudiants doivent patienter de nombreuses secondes pour charger l'ensemble des données nécessaires au bon fonctionnement de l'application. Cette solution est donc l'antithèse du principe de donner le moins d'information au client possible pour éviter les pertes de performances et de sécurité reliée à la gestion du client. Ce principe est encore plus fort lorsqu'on parle d'une application mobile.

De plus, l'interface graphique proposée par la solution est peu flexible. Aucun redimensionnement de la fenêtre n'est possible. Ainsi, l'utilisateur est contraint d'utiliser une fenêtre de petite taille et de faible résolution étant donné que celle-ci ne peut s'agrandir. De plus cette limitation peut donc rentrer en conflit la résolution et du système d'exploitation étant donné la grandeur prédéterminée en pixels absolus de la fenêtre *Cheminot*.

Le client mobile devra donc éviter de reproduire les mêmes contraintes que ceux imposés par la solution actuelle en offrant plus de flexibilité, une interface simple d'utilisation étant donné les contraintes de la grandeur d'écran, et finalement en offrant une bonne performance et sécurité en allégeant le traitement du côté du client mobile.

## 1.2 Description du projet<sup>7</sup>

Le projet de Services ÉTS Mobile a pour objectif de concevoir une application mobile permettant aux étudiants d'utiliser différents services de l'ÉTS sur la route. Ces services incluent l'accès aux différentes informations du dossier étudiant et un module permettant de visionner l'horaire et d'effectuer un choix de cours directement sur l'appareil mobile.

La plateforme mobile a été choisie avec selon plusieurs critères. Le premier et le plus important des critères est la popularité chez les étudiants de l'ÉTS de cette plateforme. En deuxième lieu, il était question d'évaluer la portabilité des appareils supportant cette plateforme soit la possibilité d'utiliser celle-ci peu importe la location géographique, de l'étudiant, avec l'utilisation d'un réseau 3G de cellulaire ou d'un réseau WIFI. Pour répondre à ces critères de popularité et de portabilité, le choix était contraint aux différents appareils cellulaires intelligents populaires sur le marché. Dans ce cas, la plateforme BlackBerry OS, Android et le iOS de Apple devenaient les choix logiques.

Afin de confirmer le choix, un sondage<sup>8</sup> parmi un échantillon de 40 d'étudiants a été effectué, permettant ainsi de confirmer le choix de la plateforme. Celui-ci a permis de choisir la plateforme iOS de Apple supporté par les appareils iPhone / iPod touch comme étant l'appareil intelligent le plus utilisé à l'ÉTS. Ce résultat n'était pas une surprise étant donné que le iPhone est un des téléphones les plus vendus dans le monde, son interface est très facile d'utilisation et quelques milliards de téléchargements d'applications ont été effectués, démontrant la popularité du iPhone.

---

<sup>7</sup> La problématique est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté à la version finale du projet.

<sup>8</sup> Le sondage se trouve en annexe 2 du document de vision, par Alexandre Girard et Mathieu Lemay

Cependant, l'utilisation d'une telle plateforme mobile impose certaines contraintes de développement. L'application se doit d'être très bien optimisée et facile d'utilisation, car tous les services seront réunis sur un écran d'une dimension de 3,5 pouces (environ 9 cm) de diagonale. Pour ce faire, une barre de navigation dans la partie supérieure de l'écran sera utilisée afin de pouvoir facilement naviguer entre les différentes fenêtres et chaque fenêtre sera disposée de la même façon, pour garder une certaine homogénéité.

Services ÉTS Mobile pourra fournir plusieurs services auxquels les étudiants de l'ÉTS ont accès, actuellement, via le site web de l'université (à l'aide de l'application *Cheminot*). L'application sera limitée aux fonctions de consultation d'horaire, de consultation du dossier étudiant et d'un service de choix de cours, incluant l'ajout de cours à l'horaire ainsi que la suppression de cours. Tel que mentionné dans l'introduction au projet ce rapport individuel sera centrer sur les services de choix de cours, soit pour le visionnement de l'horaire actuel pour la prochaine session, l'ajout et la suppression de cours et finalement l'affectation des changements via une confirmation de choix de cours.

### **1.3 Objectifs<sup>9</sup>**

Cette section du rapport permettra de présenter l'ensemble des objectifs qui ont été retenus dans la dernière itération du projet. À noter que ce rapport individuel présente seulement une portion des objectifs complets du projet d'équipe. De plus, les objectifs sont tirés du rapport d'étape, mais avec les derniers changements qu'ils ont connus dans la dernière phase du projet.

---

<sup>9</sup> La problématique est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté à la version finale du projet.

*OBJ-01 - Faciliter l'accès aux différents services en ligne offerts par l'université :*

Cet objectif impose la conception d'une application mobile pour la plateforme iPhone/ iPod touch qui sera accessible de partout (dans le cas d'un iPhone muni d'une connexion 3G de données cellulaires) ou avec la présence d'un réseau sans-fil « WiFi » (pour le iPod touch ou un iPhone sans connexion 3G).

*OBJ-04 - Offrir un outil d'inscription (choix de cours)*

Cet objectif définit qu'un service de choix de cours sera disponible dans l'application mobile. Celui-ci offrira les mêmes services offerts par Cheminot pour l'inscription sous une version mobile permettant ainsi à l'étudiant d'effectuer son inscription de partout. Ce service permettra donc de modifier son choix de cours actuel et de confirmer celui-ci en affectant les changements dans les systèmes de l'ÉTS.

*OBJ-05 - Offrir une multitude de services accessibles via une même application mobile simple et facilement accessible*

L'application devra offrir une interface simple permettant de naviguer dans les différents outils facilement. L'utilisateur pourra donc ainsi changer de service facilement via des outils de navigation intuitifs sans la nécessité de changer d'application.

Il est à noter que l'étendue à changer à plusieurs reprise depuis le début du projet. Plusieurs objectifs ont été laissés de côté afin d'assurer le bon déroulement du projet et le respect de l'échéancier. Les objectifs qui ont été laissés de côté ont été sélectionnés puisqu'il apportait une moins grande valeur ajoutée pour une version mobile. L'emphase a donc été mise sur le module de choix de cours qui bénéficierait grandement d'une version mobile pour les périodes critiques de l'inscription à l'ÉTS.



## MÉTHODOLOGIE ET ÉLABORATION DE LA SOLUTION

### 1.4 Méthodologie

Le choix de la méthodologie s'est effectué en début de projet. Il a été convenu d'utiliser une méthodologie définissant les artefacts clairs pour la conception d'un logiciel dans toutes ces phases. Puisque l'atteinte des objectifs du projet était la priorité, le choix du processus de développement logiciel devenait primordial. Ainsi, cette étape de sélection ne devait pas être négligée puisqu'elle impactait directement la bonne continuité du projet.

#### 1.4.1 Approche<sup>10</sup>

Suite à l'analyse de quelques processus populaires, celui qui a été retenu est OpenUP (anciennement OpenUP/Basic). Il convient très bien au projet courant, car il conserve les caractéristiques essentielles de RUP / Unified Process en utilisant une approche itérative, comprenant des cas d'utilisation, la gestion des risques et une approche centrée sur l'architecture. Il est conçu pour des petits projets, ayant des équipes jusqu'à 6 personnes et comportant 3 à 6 mois d'efforts de développement. Il était important d'adopter un processus itératif, car cela nous permet un retour en arrière en tout temps, permettant d'avoir un produit fonctionnel plus rapidement.

---

<sup>10</sup> La problématique est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté à la version finale du projet.

Cette méthodologie est basée sur les quatre grandes phases d'un projet de conception d'un logiciel soient :

- Départ
- Élaboration/Analyse
- Construction/Implémentation
- Transition

Dans le cadre de notre projet, la méthodologie OpenUP a permis d'identifier les artefacts essentiels, en voici la liste :

ARTF01- Document de vision

« *Vision document* »

ARTF 02- Document de spécification des exigences logicielles

« *Software requirements specifications* »

ARTF -03\* Diagramme des cas d'utilisation « *Use case diagram* »

ARTF -04\* Cas d'utilisation « *Use case* »

ARTF -04\* Modèle du domaine « *Domain class diagram* »

ARTF -05\* Diagramme de classes logicielles « *Class diagram* »

ARTF 06- Document d'architecture du logiciel

« *Software architecture document* »

ARTF 07- Document assurance qualité

« *Software Quality Document* »

*\*Artefact inclus en annexe dans le document de spécification des exigences logicielles*

Certains artefacts seront présentés plus en détail dans la section *1.5.1-Description des artefacts*.

### 1.4.2 Outils utilisés<sup>11</sup>

Les outils présentés dans cette section sont ceux qui auront été utilisés durant tout le projet, autant au début que dans le futur. Tout d'abord, Microsoft Word et le logiciel de correction Antidote ont été grandement utilisés pour la rédaction des artefacts de RUP. Microsoft Powerpoint a été utilisé pour concevoir la présentation du projet. L'outil Eclipse a été utilisé pour la création du code du serveur en langage de programmation Java. Il sert également pour effectuer les tests unitaires de celui-ci. Xcode et Interface Builder sont utilisés pour écrire le code de l'application mobile sur iPhone / iPod touch, en utilisant le langage de programmation Objective-C. Toutes les maquettes et les interfaces utilisateurs ont été réalisées à l'aide de l'outil Balsamiq Mockups. Une base de données MySQL a été utilisée pour permettre de fournir les données à l'application mobile. Le logiciel MySQL Workbench a été utilisé pour créer les tables, et exécuter des commandes SQL sur la base de données. Finalement, l'outil Visual Paradigm a servi pour produire les diagrammes UML pour plusieurs artefacts.

Pour résumer, les outils qui ont été utilisés afin de générer les différents livrables du projet sont les suivants :

- *Microsoft Word*
- *Microsoft Powerpoint*
- *Druide Antidote*
- *Eclipse*
- *Xcode*
- *Interface Builder*

---

<sup>11</sup> La problématique est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté à la version finale du projet.

- *Balsamiq Mockups*
- *MySQL*
- *MySQL Workbench*
- *Visual Paradigm*

Plusieurs outils ont été nécessaires pour effectuer la gestion de l'ensemble du projet. Le principal est Dropbox, qui a servi à la fois de gestion des versions du code source et pour partager les artefacts entre les membres de l'équipe. Évidemment, Microsoft Word a été utilisée pour faire l'échéancier du projet ainsi que le plan de travail. Finalement, la méthodologie de travail est basée sur OpenUP pour la plupart du projet, soit pour les livrables et pour le cycle de vie de Services ÉTS Mobile.

En résumé, les outils utilisés pour la gestion du projet sont les suivants :

- *Dropbox*
- *Microsoft Word*
- *OpenUP*

Les outils de collaboration entre les membres de l'équipe sont les suivants :

- *Dropbox*
- *Gmail email*
- *Gmail chat*

## 1.5 Livrables et planification<sup>12</sup>

Cette section permet d'identifier les artefacts du projet qui seront réalisés pendant toutes les phases de celui-ci. On retrouve également la description de chaque artefact et la planification de leur réalisation.

### 1.5.1 Description des artefacts

**Tableau 1: Liste des artefacts produits<sup>13</sup>**

Nom de l'artefact	Description
ARTF01- Document de vision « <i>Vision document</i> »	Ce document permet de positionner le problème, d'identifier les intervenants et de décrire, de façon globale, le projet et ses fonctionnalités requises. L'objectif d'un tel document est donc de collecter, d'analyser et de définir les besoins et les caractéristiques du projet.
ARTF 02- Document de spécification des exigences logicielles « <i>Software requirements specifications</i> »	On y retrouve l'ensemble des exigences fonctionnelles et non fonctionnelles ainsi que des contraintes spécifiques afin de bien définir l'application. Il comporte aussi un glossaire, la liste des acteurs, les cas d'utilisation, et les diagrammes associés. Avec ce document, il est alors possible de débiter l'implémentation de l'application.

<sup>12</sup> La problématique est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay. Par contre, le contenu a été adapté à la version finale du projet.

<sup>13</sup> La liste des artefacts est inspirée du rapport d'étape, par Alexandre Girard et Mathieu Lemay

<p>ARTF 06- Document d'architecture du logiciel</p> <p><i>«Software architecture document »</i></p>	<p>Il présente le choix retenu au niveau de l'architecture logicielle et réseau, donc de la technologie utilisée. On y retrouve également une explication des choix vis-à-vis les classes logicielles créées.</p>
<p>ARTF 07- Document assurance qualité</p> <p><i>«Software Quality Document»</i></p>	<p>Il décrit tous les objectifs liés à la qualité du logiciel, les différentes activités prévues et une planification des tests.</p>

### 1.5.2 Planification

Tel que prévu dans les itérations l'ensemble des livrables a été livré selon l'échéancier fixé. Cependant, un livrable qui avait été sélectionné en début de projet a été négligé par faute de temps soit le document de test. Pour plus de détail sur la planification du projet, veuillez vous référer à la section VII du rapport d'équipe.

## RÉSULTATS

### 1.6 Résultats du projet

Dans cette section il sera question d'élaborer sur les différents résultats obtenus au cours du projet. Ainsi, l'ensemble des phases principal de la méthodologie seront couvertes soient l'analyse, la conception et l'élaboration de la solution (implémentation).

#### 1.6.1 Analyse

Cette section permettra de mettre en valeur les différents éléments clés qui ont été abordés durant la phase d'analyse du projet. Pour de plus amples détails sur la phase d'analyse, veuillez vous rapporter au document de vision, disponible en annexe II du rapport d'équipe.

##### 1.6.1.1 Portée

La portée est un élément clé de tout projet informatique. Celle-ci permet donc en début de projet de définir clairement l'étendue du projet à être livré. Il est important de se tenir le plus fidèlement possible à la portée initiale du projet afin d'assurer le respect de l'échéancier. Il est à noter que la portée de ce document représente seulement la portion du projet assigné dans ce rapport individuelle due à la séparation des tâches. Donc, comme on peut le voir dans le tableau ci-dessous, seulement les modules permettant l'ajout de cours, la suppression de cours et la confirmation du choix de cours seront traités.

**Tableau 2: Portée du projet**

Traités	Non traités
1-Ajouter un cours à l'horaire	4-Authentification
2-Supprimer un cours à l'horaire	5-Consultation du dossier étudiant
3-Confirmer un choix de cours	6-Consultation d'horaire
	7-Obtenir de l'aide

- 1- Le module d'ajout de cours à l'horaire permet à l'étudiant de retrouver dans la liste de cours de son programme les différents cours auxquelles il peut s'inscrire pour la prochaine session. Cette liste permet d'afficher plusieurs informations pertinentes à l'ajout du cours soit le sigle, son numéro de groupe, le nom du cours, la plage horaire du cours et la plage horaire de la période de TP. Ainsi, l'étudiant peut ajouter un cours à son horaire en affectant directement un horaire en mode de travail. Ce module permet également de détecter les conflits d'horaire et ainsi d'avertir l'utilisateur que le cours ne peut être ajouté.
- 2- L'outil de suppression de cours est un outil simple qui permet en sélectionnant un cours de supprimer celui-ci de l'horaire de travail de l'étudiant. L'horaire de travail est défini comme l'horaire qui est en cours de modification sur la session actuelle de l'étudiant et qui n'a pas encore été confirmé par celui-ci.
- 3- Finalement, il est possible avec l'outil de confirmation de choix d'envoyer les changements sur le serveur qui affecteront la base de données.



### 1.6.1.2 Risques

Parmi l'ensemble des risques qui ont été identifiés lors du rapport d'étape, voici donc ceux qui s'appliquent aux modules qui seront traités, voir section *1.6.1-Portée*.

**Tableau 3: Risques<sup>14</sup>**

Risque	Impact	Probabilité	Mitigation / atténuation
RI-01 : Leadership	Moyen	Bas	Délégation d'un membre de l'équipe comme étant celui qui dirigera le projet.
RI-02 : Conflit d'équipe	Moyen	Bas	Rencontres régulières entre les membres de l'équipe et discussion des idées de chacun.
RI-03 : Disponibilité des membres de l'équipe	Moyen	Bas	Prévoir d'avance les rencontres et planifier les plages horaires disponibles pour les membres de l'équipe.
RI-04 : Difficulté d'implémentations	Élevé	Élevé	Le iPhone/iPod touch utilise un système d'exploitation indépendant des autres langages connus du marché avec ses propres librairies. De plus, la programmation pour une plateforme mobile implique de l'optimisation de code étant donné les limitations de l'appareil (mémoire, processeur, réseau, etc.). L'utilisation de ressource en ligne

<sup>14</sup> Le tableau de risques provient du rapport d'étape, par Alexandre Girard et Mathieu Lemay

			permettra de limiter les embûches. De plus, la liaison entre le serveur Java et l'application client iPhone représente un fort risque.
RI-05 : Perception des utilisateurs	Moyen	Moyen	Aller chercher les réactions d'un ou plusieurs utilisateurs types durant les différentes phases de développement de l'application.
RI-07 : Retard dans l'échéancier	Élevé	Élevé	Découper le projet en plusieurs projets et livrables afin d'obtenir des étapes de remises fréquentes et réalisables.
RI-08 : Temps de réponse de l'application/Problème de performance	Moyen	<b>Moyen</b>	L'utilisation des meilleurs standards de développement sur la plateforme iPhone devrait limiter ce problème. Étant donné que l'ensemble des commandes clients sont dépendantes d'une communication avec le serveur, le temps de réponse devient un facteur à ne pas négliger.

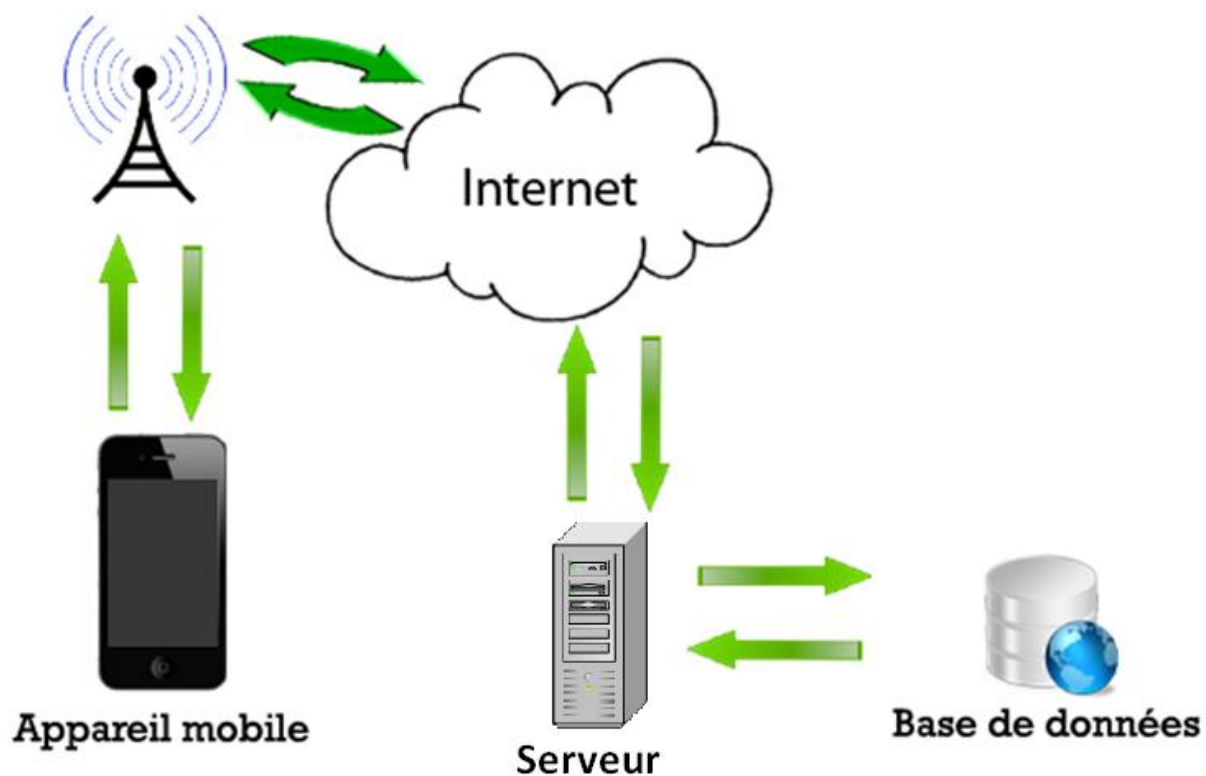
La plupart des risques ont été identifiés correctement en début de projet. Ainsi, il était plus simple pour l'équipe de prévoir et de planifier des mesures pour atténuer ceux-ci. Il n'y pas eu de conflit d'équipe, ni de problème de disponibilité étant donné que la division des tâches était juste et équitable et que les rencontres étaient planifiées d'avance selon la disponibilité des membres. Le risque de difficulté d'implémentation ayant été ciblée comme critique dans le projet s'est reflété sur un temps de développement un peu plus long, mais qui avait été prévu étant donné la nature de l'architecture et le langage Objective-C. Ce langage propriétaire pour le iPhone/iPod touch était peu connu des membres de l'équipe. Finalement, la communication entre les différents composants a représenté la plus grosse embuche durant l'implémentation.

### **1.6.2 Contraintes**

Tel qu'identifié en début de projet les plus grandes contraintes pour ce projet se situaient au niveau de l'aspect mobile de l'application. Premièrement, des tests de connexions devaient être effectués en lançant l'application pour valider la disponibilité d'une connexion internet. De plus, la grandeur de l'écran des appareils iPhone/iPod touche devait être prise en considération dès la phase d'analyse afin d'offrir une solution adéquate pour de tels types d'appareils. Il faut considérer que l'aspect vendeur des applications iPhone/iPod Touch se situe dans la qualité des interfaces utilisateurs. Finalement, ne faut pas oublier que les utilisateurs n'auront pas accès à aucun périphérique d'entrée tels une souris ou un clavier, tout doit donc être pensé pour supporter le multi tactiles intégrer dans l'écran des appareils supportés.

### **1.6.3 Conception**

Suite à la phase d'analyse du projet de Services ÉTS Mobile, la phase de conception a pu être entamée. Étant donné la nature du projet, soit la mobilité, il était important de considérer premièrement l'architecture matérielle afin d'assurer le bon fonctionnement du client mobile (iPhone/iPod Touch). La figure suivante vous présente les liens existants entre les différents composants de l'architecture.



**Figure 1: Architecture matérielle<sup>15</sup>**

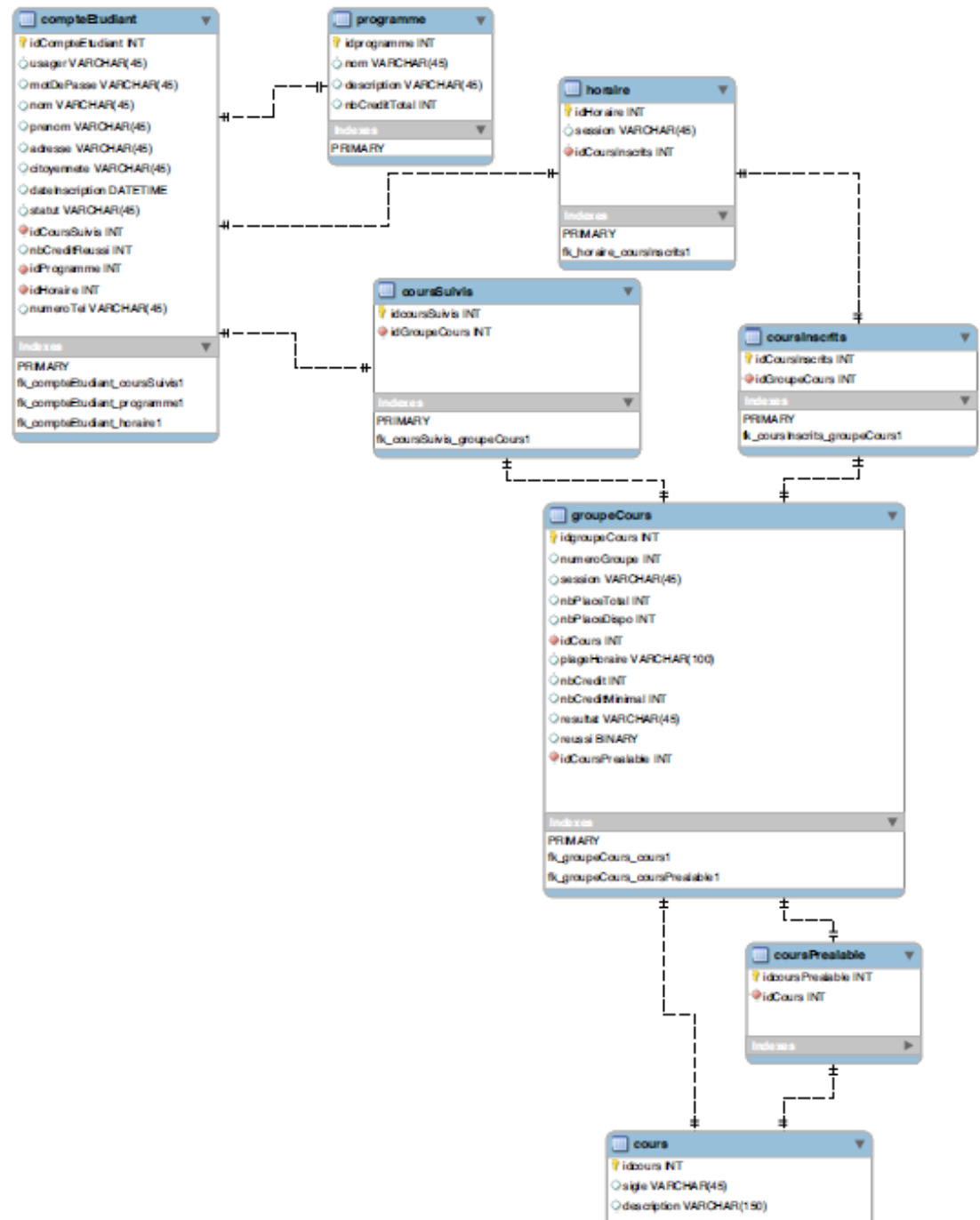
La figure ci-dessus présente en premier lieu le client mobile. Celui-ci communique directement via une la connexion 3G ou par un accès WiFi sur internet pour envoyer les demandes sur le serveur. Le serveur traitera la demande et enverra une requête sur la base de données afin d’obtenir les informations nécessaires pour traiter celle-ci. Une fois le traitement effectué, le serveur enverra la réponse sur internet en destination de l’appareil mobile. Celui-ci n’aura ainsi qu’a mettre à jour sa vue pour corresponde aux données envoyer par le serveur.

---

<sup>15</sup> La figure est prise du document de vision, par Alexandre Girard et Mathieu Lemay

Afin d'assurer la liaison entre l'appareil mobile et le serveur une adresse statique a été utilisée pour identifier le serveur. Cela représente une contrainte pour le serveur, mais elle peut être facilement contournée en utilisant un serveur DNS qui permettrait de définir un nom de domaine qui serait lié en tout temps à l'adresse IP actuelle du serveur.

Ensuite, il était important de concevoir la base de données avant de concevoir le reste de l'application puisqu'elle représente le point d'entrée pour récupérer les données des étudiants. Les données récupérées sur celle-ci sont des données fictives qui ont été créées pour les besoins de l'application. Elles contiennent plusieurs données de consultation sur l'étudiant. De plus plusieurs tables ont été créées afin d'assurer le fonctionnement du module de choix de cours. Voici donc un aperçu des tables qui ont été identifiées comme nécessaires pour la conception de cette base de données.



**Figure 2 : Schéma de la base de données**

Le choix de conception de la BD ne sera pas détaillé puisqu'il ne s'agit pas de la composante principale du projet. Toutefois le tableau ci-dessous vous présente une brève description de l'ensemble des tables composant la BD.

**Tableau 4: Description des tables de la BD**

Nom de la table	Description
<b>T01- compteEtudiant</b>	Contient différentes informations sur l'étudiant ainsi que ces informations d'authentification.
<b>T02- programme</b>	Contient l'ensemble des programmes offert à l'ÉTS.
<b>T03-horaire</b>	Un horaire particulier pour une session associé à un étudiant.
<b>T04-coursSuivis</b>	Contient la liste des cours suivis par l'ensemble des étudiants.
<b>T05-coursInscrits</b>	Contient la liste des cours inscrits pour un étudiant associé à l'horaire pour le choix de cours.
<b>T06-groupeCours</b>	Contient l'ensemble des données correspondant à un groupe pour un cours. À noter que beaucoup d'informations du cours sont contenues dans cette table afin d'éviter les problèmes si les informations d'un cours ont à changer dans le temps.
<b>T07-coursPrealable</b>	Liste qui contient l'ensemble des cours préalables pour tous les cours.
<b>T08-cours</b>	Contient les informations générales pour un cours.

Pour le système de choix de cours, les tables qui sont principalement impliquées sont *coursInscrits* qui présentent la liste des cours auquel un étudiant est inscrit pour la prochaine session, *groupeCours* qui présentent l'ensemble des groupes pour tous les cours ouverts pour l'inscription. Finalement, la table horaire fait un lien entre l'étudiant authentifié, *compteEtudiant*, et la liste des cours inscrits, *coursInscrits*.

Pour la conception du serveur et du client iPhone, il était important de garder en tête que le serveur agirait comme la composante effectuant la majeure partie du traitement. La raison étant que dans une architecture client-serveur, le serveur doit être la composante qui s'occupe principalement du traitement pour libérer les ressources du client et ainsi améliorer l'expérience de l'utilisateur. Dans ce cas ce principe est encore plus justifiable étant donné la nature « mobile » du client. Pour connaître les détails de toutes les classes logicielles avec le diagramme de classes, il faut se référer au document de spécifications des exigences logicielles (SRS), qu'on trouve en annexe III du rapport d'équipe.

#### **1.6.4 Élaboration de la solution**

Cette section permettra de définir les principaux éléments de la phase d'implémentation de OpenUP soit dans le cas présent la conception d'un prototype fonctionnelle proposant l'ensemble des objectifs préétablis. L'architecture tel que présenté à dans la phase de conception propose que l'application mobile iPhone/iPod Touch joue un rôle de client qui servira à refléter la vue selon les requêtes effectuées par l'utilisateur. Dans ce cas cette section mettra l'accent sur les interfaces utilisateurs du client mobile ainsi que sur les échanges de message qui auront lieu entre le client et le serveur.

##### **1.6.4.1 Interfaces utilisateurs**

Afin de respecter le critère de qualité logiciel d'utilisabilité, il est important de porter une attention particulière à la réalisation des interfaces utilisateurs. Pour ce faire, le moins de pages différentes doivent être utilisées afin de faciliter l'utilisation des services et ainsi de



réduire le temps de navigation. Il a donc été convenu d'utiliser un menu simple d'utilisation regroupant les différents services. De plus pour la création des interfaces, les éléments graphiques standard offerts par les outils de développement de Apple ont été utilisés pour faciliter la conception et du même coup en rendant les interfaces plus standard et flexible pour les différents appareils (iPhone/iPod Touch, iPad, etc.). Il faut cependant noter que les interfaces ont été créées avec l'aide des prototypes statiques « sketchboard » qui ont été créés dans la phase de conception. Veuillez vous référer au document SRS qu'on trouve en annexe III du rapport d'équipe.

Le service de choix de cours a été modélisé avec seulement 2 écrans. Cette approche tel que discuté ci-dessus permet de simplifier l'utilisation du prototype et diminue donc le temps nécessaire pour accomplir les différents cas d'utilisation du choix de cours (ajout de cours, suppression de cours et confirmation de choix de cours).

Un premier écran permet de voir l'horaire en mode de travail actuel de l'étudiant. Il est donc possible de voir les cours pour lesquelles l'étudiant est actuellement inscrit. On y retrouve également les boutons permettant d'accéder à la page pour ajouter un cours à l'horaire et un bouton pour confirmer le choix de cours actuel. De plus il est possible de supprimer un cours à l'horaire en appuyant directement sur la plage horaire du cours ou de la période de TP correspondant au cours voulant être supprimé. De plus un écran d'aide peut être demandé via l'ensemble des services offerts par le prototype fonctionnel. Lorsque l'utilisateur appuie sur un cours pour être supprimé, une fenêtre de type « pop-up » est lancée permettant donc à l'utilisateur de confirmer la suppression du cours. La confirmation est également validée avec l'utilisation d'un tel type de fenêtre. Afin de concevoir cette vue, il était important de simplifier la présentation de l'horaire. Pour ce faire, il a été convenu d'utiliser 15 boutons représentant chacune une plage horaire de la semaine soit le matin, l'après-midi ou le soir pour les journées du lundi au vendredi. Chacun d'eux porte l'étiquette du cours présentement ajouté pour cette plage horaire. Une barre de navigation au haut de l'écran permet d'utiliser les différentes options offertes pour l'utilisateur (dans ce cas l'ajout d'un cours, la confirmation du choix de cours et le retour au menu principal). Ainsi, la

présentation de l'horaire est simple et épurée et s'adapte bien sur l'écran du iPhone/iPod Touch avec une orientation sous format « paysage ». Cet écran est présenté à la figure 3.



**Figure 3: Écran de l'horaire pour le choix de cours**

Le deuxième écran permet de lister l'ensemble des cours pouvant être ajouté à l'horaire pour cet usager. Cette liste affiche le sigle du cours, le numéro de groupe, le nom du cours, la plage horaire du cours et la plage horaire du TP. Cette page est affichée lorsque l'étudiant appuie sur le bouton « Ajouter un cours » sur la page d'horaire en mode de choix de cours. Il est possible d'ajouter le cours voulu en appuyant sur l'élément de la liste correspondant à celui-ci. Cet écran utilise l'élément de liste graphique standard pour le iPhone soit un de type « UITableView ». Ainsi, l'utilisation de cet écran est simple puisqu'elle présente seulement la liste des cours et un (1) bouton permettant de revenir à l'écran d'horaire. Cet écran est présenté à la figure 4.



Figure 4 : Écran d'ajout de cours à l'horaire

### 1.6.4.2 Échanges avec le serveur

Les échanges avec le serveur représentent un élément crucial de l'architecture. Les formats de messages ont été standardisés pour faciliter la communication entre le client iPhone et le serveur multi-utilisateurs. Cette section vous présentera les messages standardisés utilisés par le prototype fonctionnel pour le service de choix de cours.

Lors de l'affichage de l'horaire en mode de travail, le client mobile envoie la requête suivante :

```
GETHORAIRE-NuméroSession
```

```
Ex. : GETHORAIRE-182
```

Le iPhone est en mode réception et il attend la réponse du serveur. Le serveur peut donc retrouver la session de l'utilisateur et lui retourner son horaire de travail tel que présente sur la base de données. Le serveur doit donc faire du traitement pour afficher correctement les cours dans l'ordre nécessaire pour l'affichage de l'horaire. Lorsque le serveur répond, il répond sous cette forme :

```
SETHORAIRE- NuméroSession, L1, L2, L3, Ma1, Ma2, Ma3, Me1, Me2, Me3, J1, J2, J3, V1, V2, V3
```

```
Ex. : SETHORAIRE-182, LOG120-01 Cours 18h00 à 21h00, , CHM131-02 TP, ...
```

Les paramètres « Cours1,Cours2,... » représentent la liste des cours à l'horaire en ordre de plage horaire soit du lundi matin ou vendredi après-midi. Si une plage horaire n'est pas occupée, le serveur laisse vide le paramètre pour cette plage horaire. Le client n'a qu'à mettre à jour les étiquettes des boutons avec le contenu du message reçu. Le tableau suivant vous présente l'ordre des paramètres envoyés au client.

**Tableau 5 : Correspondance pour l'horaire<sup>16</sup>**

LUNDI	MARDI	MERCREDI	JEUDI	VENDREDI
L1	Ma1	Me1	J1	V1
L2	Ma2	Me2	J2	V2
L3	Ma3	Me3	J3	V3

Ensuite, l'affichage de la liste de cours pouvant être ajouté à l'horaire est effectué en envoyant une requête sous le format suivant au serveur :

```
GETLISTECOURS-NuméroSession
```

Ex. : GETLISTECOURS-182

Le serveur répond en renvoyant l'ensemble des cours offerts pour cette session auxquelles l'étudiant possède les prérequis et n'étant pas dans sa liste de cours inscrits actuellement. La réponse est envoyée sous ce format :

```
SETLISTECOURS-NuméroSession,Info cours1,Info cours2,...
```

Ex. : SETLISTECOURS-NuméroSession,LOG120-01 Conception de logiciels  
Horaire Cours : Mardi 18h00-21h30 Horaire TP : Jeudi 18h00-21h30, ...

Chaque paramètre reçu est donc une chaîne de caractères préformatée par le serveur contenant l'ensemble des informations pertinentes du cours. Le client n'a donc qu'à affecter l'ensemble des éléments de la liste de cours possible à ajouter avec chacune des chaînes de caractères.

Pour l'ajout de cours, le client envoie le message sous la forme suivante :

```
ADDCOURS-NuméroSession, Sigle-NumeroGroupe
```

Ex. : ADDCOURS-182,LOG120-01

---

<sup>16</sup> La figure est prise du rapport final individuel de Mathieu Lemay

Une fois la requête traitée par le serveur il renvoie la réponse sous la forme suivante :

ADDCOURS-TRUE/FALSE, L1, L2, L3, Ma1, Ma2, Ma3, Me1, Me2, Me3, J1,  
J2, J3, V1, V2, V3

Ex. : ADDCOURS-182, TRUE, LOG120-01 Cours 18h00 à 21h00, , CHM131-  
02 TP, ...

Le premier argument retourné permet de valider si l'ajout du cours est possible et ne cause pas de conflit d'horaire avec les cours actuellement inscrits. Dans ce cas l'argument prend la valeur « TRUE » sinon il prend la valeur « FALSE ». Dans le cas où l'argument est « TRUE » les paramètres suivants envoyés correspondent à l'horaire formaté comprenant le nouveau cours ajouter.

La demande de suppression d'un cours s'effectue en utilisant le format de message suivant sur le client :

DELETECOURS-NuméroSession, Sigle

Ex. : DELETECOURS-182, LOG120

Vous noterez qu'aucun numéro de groupe n'est envoyé au serveur puisqu'il est impossible qu'un horaire contienne plus d'un groupe pour le même cours. La suppression est affectée sur le serveur, mais aucune réponse n'est retourné au client étant donné que la gestion de la suppression est très simple et peu donc être également effectué sur le client pour éviter la sur communication.

Finalement, la demande de confirmation du choix de cours est envoyée sous le format suivant :

CONFIRMER-NuméroSession

Ex. : CONFIRMER-182

Le serveur affectera ainsi les changements effectués sur l'horaire de travail sur la base de données dans la table des cours inscrits pour la session à venir. Une fois de plus, aucun message de retour n'est envoyé.

## 1.7 Discussion des résultats<sup>17</sup>

La priorité du projet était de concevoir une preuve de concept offrant différents services de l'école ÉTS aux étudiants sous un format mobile. Celle-ci devait être réalisée en suivant les différentes phases de développement logiciel tel qu'établi dans la méthodologie choisie. Le projet a donc répondu aux attentes initiales puisqu'il a été possible malgré le court délai disponible d'offrir une preuve de concept fonctionnelle. Celle-ci reflète différents services de Cheminot en version mobile pouvant être exploités sur un appareil iPhone/iPod Touch.

Malgré plusieurs embûches, il a été possible de couvrir différents services pouvant profiter d'une version mobile telle que le choix de cours et la visualisation de l'horaire. Cette preuve de concept n'est cependant pas comptable avec les éléments d'architecture de l'école actuel et demanderait donc plusieurs changements afin d'être supportée. Elle représente donc une idée, ayant été conçue selon les meilleures pratiques logicielles actuelles.

La méthodologie suivie, OpenUp, a permis d'identifier les livrables dès le début du projet. De plus l'approche itérative proposée par celle-ci a été utilisée tout au cours du projet. Il a donc été possible de revoir plusieurs éléments telles les exigences au fur que le projet évoluait à travers les itérations. L'ensemble des artefacts produits ont été utilisés au cours de la phase d'implémentation. Également, la cédule initiale quant au livrable a été assez bien suivie, ce qui a permis le bon déroulement du projet.

---

<sup>17</sup> La discussion des résultats est inspirée du rapport d'équipe, par Alexandre Girard et Mathieu Lemay

## **1.8 Difficultés rencontrées**

La principale difficulté rencontrée a été la mise en communication des différents éléments de l'architecture. Puisqu'aucun membre de l'équipe n'avait déjà utilisé des bibliothèques permettant aux appareils de type iPhone/iPod Touch de communiquer avec un serveur Java sous la forme de message, beaucoup de temps a dû être accordé à la compréhension des outils en place. Plusieurs embûches dans la phase d'implémentation ont donc été étroitement liées avec la gestion de la connexion et le formatage de la requête pour l'envoi sur le réseau.

De plus, les traitements en parallèle ou plus communément appelés « thread » sur le iPhone/iPod Touch ont également causé des problèmes. Plusieurs actions de la communication demandaient de la gestion par « thread ». Il a été possible de résoudre une partie du problème en utilisant une bibliothèque dénommée « asyncSocket » implémentant le concept de « socket » sur l'iPhone et simplifiant l'utilisation de celui-ci en gérant le traitement continu nécessaire pour la gestion des messages reçus sur le « socket ».

Finalement, le temps a été une contrainte en fin de projet. L'implémentation a demandé plus de temps de prévu, ainsi la création des artefacts finaux a été condensée afin d'assurer d'atteindre les dates prévues.

## **1.9 Améliorations possibles**

Bien que le projet c'est avéré un succès plusieurs facteurs auraient pu être amélioré celui-ci. Premièrement, l'application est loin d'être complète et n'implémente pas l'ensemble des services de l'ÉTS afin de rendre cette solution plus intéressante dans un contexte où celle-ci serait réellement utilisée par l'ÉTS. Par exemple, la solution aurait pu inclure l'ensemble des fonctions financières de l'école soit le visionnement de facture, le paiement de facture, l'inscription aux bourses de l'école, etc.

De plus, les communications n'ont pas été optimisées afin d'assurer la limitation des échanges et ainsi de diminuer le temps de traitement des requêtes clients sur le mobile. Pour



ce faire, il aurait été possible de limiter la structure des messages en utilisant des acronymes ou des lettres prédéfinies pour l'ensemble des actions.

La sécurité a été traitée au niveau de la validation de champ, cependant un tel type d'application bénéficierait d'un protocole sécurisant les messages envoyés entre le client et le serveur. Par exemple, le « socket » pourrait utiliser le protocole *Secure Socket Layer (SSL)* afin d'assurer la confidentialité des échanges de messages.

Finalement, une meilleure planification en début de projet aurait pu être faite pour les derniers artéfacts afin de limiter davantage les périodes de pointes de travaux en fin de projet.

## CONCLUSION<sup>18</sup>

L'objectif initial du projet était concevoir une application mobile regroupant plusieurs services de l'École de technologie supérieure accessible par les étudiants. Cette idée de projet a été grandement influencée par le besoin manifesté par plusieurs étudiants de l'école. Ainsi, ce projet de fin d'études a permis d'analyser, de concevoir et d'implémenter une telle solution pouvant répondre à un tel besoin.

Le prototype fonctionnel produit n'est pas parfait, mais il a été conçu selon les meilleures pratiques logicielles vues au cours du baccalauréat en génie des technologies de l'information. Du même coup, l'emphase a été mise sur les concepts principaux d'une telle application, soit la simplicité d'utilisation, la performance du client et l'accessibilité. Il est donc possible d'affirmer que la solution proposée répond à ces différents critères de qualité logicielle.

L'utilisation d'une méthodologie formelle telle OpenUP a permis de mettre en pratique l'écriture d'artéfacts pour un projet de conception d'un logiciel. Il a donc été possible de produire différents artéfacts regrouper dans les principales phases de la conception d'un logiciel. La nature de projet a permis également d'utiliser un échancier pour mieux structurer le projet. Celui-ci a été mise à jour à chaque itération, permettant ainsi de garder un œil sur l'avancement du projet et du retard de certains artéfacts.

---

<sup>18</sup> La discussion des résultats est inspirée du rapport d'équipe, par Alexandre Girard et Mathieu Lemay

Finally, the key factors of project success can be summarized by the use of a formal methodology, the respect of pre-established objectives, the follow-up of the project by iteration and a good communication and distribution of tasks among team members.

## RECOMMANDATIONS

Afin de suivre l'idée du projet, plusieurs recommandations peuvent être proposées.

En premier lieu, il serait intéressant que le département informatique de l'ÉTS se base sur l'idée du projet afin de regarder les options possibles afin de répondre à ce besoin des étudiants. Pour ce faire, il pourrait utiliser le projet comme point de départ pour l'intégrer à l'architecture en place à l'École de technologie supérieure.

Afin de se conformer à la politique de l'école certains éléments devraient être revus soit la gestion des informations sensibles des étudiants. De plus, l'utilisation d'un protocole sécurisé dans le cadre d'une utilisation réelle deviendrait un élément essentiel pour assurer la confidentialité des données.

Également, il serait intéressant de poursuivre le projet en offrant une telle offre de services mobiles sous différentes plateformes mobiles afin de couvrir l'ensemble des besoins des étudiants (android, BlackBerry, etc.).

Finalement, l'optimisation du client Cheminot en mode applet pourrait également bénéficier de plusieurs notions analysées au cours de ce projet.

## RÉFÉRENCES BIBLIOGRAPHIQUES

LARMAN, Craig. 2006. Applying UML and Patterns: An Introduction to Object-Oriented Analysis And Design And Iterative Development, third edition, USA: Prentice Hall PTR, 703 pages.

Introduction to OpenUP, 2008. OpenUP. En ligne.

<<http://epf.eclipse.org/wikis/openup/>>. Consulté le 20 janvier 2011.

École de technologie supérieure (ÉTS), 2011. ÉTS. En ligne.

<<http://www.etsmtl.ca/>>. Consulté le 20 janvier 2011.

ChemiNot, 2011. ÉTS. En ligne.

<<http://lancelot.etsmtl.ca/cheminot/go.html>>. Consulté le 20 janvier 2011.

The Objective-C Programming Language, 2011. Apple. En ligne.

<<http://developer.apple.com/library/mac/#documentation/Cocoa/Conceptual/ObjectiveC/>>

Consulté le 20 février 2011.

LEMAY, Mathieu et GIRARD, Alexandre, Rapport final d'équipe, GTI792-Projet de fin d'études en génie des technologies de l'information

LEMAY, Mathieu et GIRARD, Alexandre, Rapport d'étape, GTI792-Projet de fin d'études en génie des technologies de l'information

LEMAY, Mathieu et GIRARD, Alexandre, Document des spécifications des exigences logicielles, GTI792-Projet de fin d'études en génie des technologies de l'information

LEMAY, Mathieu et GIRARD, Alexandre, Document de vision, GTI792-Projet de fin d'études en génie des technologies de l'information

SRS document template, 2011. Villanova University Department of Computing Sciences. En ligne. <[http://www.csc.villanova.edu/~tway/courses/csc4181/srs\\_template-1.doc](http://www.csc.villanova.edu/~tway/courses/csc4181/srs_template-1.doc)>. Consulté le 30 janvier 2011.

Vision document template, 2011. School of Science and Computer Engineering. En ligne. <<http://sce.uhcl.edu/whiteta/sdp/visionDocumentTemplate.doc>>. Consulté le 16 janvier 2011.

Cocoa Asynchronous Socket, 2011. Google code project. En ligne. <<http://code.google.com/p/cocoaasyncsocket/>>. Consulté le 5 mars 2011.

Alphanumeric characters validation in Objective-C, 2009. Stack Overflow. En ligne. <<http://stackoverflow.com/questions/1671605/how-to-check-if-a-string-only-contains-alphanumeric-characters-in-objective-c>>. Consulté le 10 mars 2011.

Classes Reachability, 2011. Apple. En ligne. <[http://developer.apple.com/library/ios/#samplecode/Reachability/Listings/Classes\\_Reachability\\_m.html](http://developer.apple.com/library/ios/#samplecode/Reachability/Listings/Classes_Reachability_m.html)>. Consulté le 8 mars 2011.

Icône de choix de cours, 2011. Spanishmedellin. En ligne. <<http://spanishmedellin.com/images/icon-courses.png>>. Consulté le 30 février 2011.

Icône du dossier étudiant, 2011. Imagemania. En ligne. <<http://www.imagemania.net/data/media/30/Dossier%203D.png>>. Consulté le 30 février 2011.

Icône de l'horaire , 2011. Geekpedia. En ligne. <[http://www.geekpedia.com/Pictures/Icons/schedule\\_1736\\_128.png](http://www.geekpedia.com/Pictures/Icons/schedule_1736_128.png)>. Consulté le 30 février 2011.