

**h e g**

Haute école de gestion  
Genève

## **SIRH Web**

# **Développement d'un client riche multiplateforme**



**Travail de Bachelor réalisé en vue de l'obtention du Bachelor HES**

par :

**Mathieu HERMANN**

Conseiller au travail de Bachelor :

**Peter DAEHNE, Professeur HES**

**Genève, le 16 juillet 2014**

**Haute École de Gestion de Genève (HEG-GE)**

**Filière Informatique de Gestion**

## Déclaration

Ce travail de Bachelor est réalisé dans le cadre de l'examen final de la Haute école de gestion de Genève, en vue de l'obtention du titre de Bachelor en Informatique de Gestion.

L'étudiant a envoyé ce document par email à l'adresse d'analyse remise par son conseiller au travail de Bachelor pour analyse par le logiciel de détection de plagiat URKUND. [http://www.orkund.fr/student\\_gorsahar.asp](http://www.orkund.fr/student_gorsahar.asp)

L'étudiant accepte, le cas échéant, la clause de confidentialité. L'utilisation des conclusions et recommandations formulées dans le travail de Bachelor, sans préjuger de leur valeur, n'engage ni la responsabilité de l'auteur, ni celle du conseiller au travail de Bachelor, du juré et de la HEG.

« J'atteste avoir réalisé seul le présent travail, sans avoir utilisé des sources autres que celles citées dans la bibliographie. »

Fait à Genève, le 16 juillet 2014

Mathieu HERMANN

## Remerciements

Je tiens bien évidemment à remercier la Haute École de Gestion et l'ensemble de ses professeurs que j'ai eu l'occasion de côtoyer durant mes 3 années de cette formation de bachelier au sein de cet établissement. Je tiens plus spécialement à remercier Monsieur Daehne, Professeur HES, qui m'a suivi dans notre travail de GREP mais aussi dans ce travail de Bachelor.

Je tiens également à remercier la société informatique genevoise Qim info qui m'a donné la chance de réaliser ce travail de Bachelor en situation réelle. Durant cette expérience j'ai eu la chance de côtoyer des personnes qui m'ont accordé leur confiance et qui ont toujours été très chaleureuses. Remerciement tout spécial à Loïc Malot, responsable de mon projet en interne, qui en plus de sa sympathie a toujours été de bon conseil dans les moments cruciaux de ce projet.

Finalement, je tiens à remercier également mes proches qui ont supporté avec brio mon indisponibilité pour des raisons professionnelles liées à la réalisation de ce travail.

## Résumé

L'adoption en masse des smartphones et des tablettes de ces dernières années commence à bouleverser les habitudes d'utilisation des ressources informatiques. Autrefois utilisés comme de simples gadgets de divertissement, les smartphones doivent aujourd'hui répondre à une demande de plus en plus forte d'utilisateurs désirant faire de cette petite machine, leur premier outil de travail.

Les limitations de puissance qui autrefois permettaient de très rapidement écarter l'idée d'utiliser un smartphone comme outil de travail ne sont plus d'actualité. En effet, les smartphones s'approchent des puissances de calcul des ordinateurs et par conséquent deviennent largement capables d'exécuter des programmes qui autrefois pouvaient sembler bien trop lourds.

La seconde limitation liée à la taille des écrans et plus spécifiquement la résolution de ces derniers devient aussi de moins en moins problématique, les technologies utilisées deviennent au fil des années moins coûteuses et de nos jours, il n'est pas rare de retrouver des définitions Full HD dans des écrans de seulement quatre ou cinq pouces à tarifs abordables. Les résolutions actuelles permettent d'afficher bien plus d'informations qu'autrefois, cependant un travail d'adaptation du contenu reste nécessaire pour garantir un certain confort d'utilisation.

Aujourd'hui il n'y a donc plus de barrière réelle à l'utilisation d'applications, autrefois destinées aux ordinateurs, sur un téléphone ou une tablette. Cependant l'ordinateur, bien que délicat à transporter, dispose encore d'un avantage de confort lié à sa grande taille d'écran et son clavier de saisie qui font de lui un outil de productivité duquel il est difficile de se passer entièrement.

L'idéal serait donc de pouvoir développer une application qui soit capable de fonctionner sur l'ensemble des plateformes susceptibles d'être employées par les utilisateurs. Il faudrait également que cette application devienne disponible depuis n'importe quel lieu géographique afin d'en faciliter encore l'accès.

La réponse à ce besoin passe souvent par le développement spécifique sur un nombre de plateformes choisies au préalable. En plus d'être très longue et coûteuse, cette méthode implique à chaque mise à jour un déploiement sur l'ensemble des versions de notre application.

C'est justement là qu'intervient le client riche multiplateforme. Développer une application unique, disponible et adaptée sur chaque système.

# Table des matières

Déclaration.....	i
Remerciements.....	ii
Résumé .....	iii
Liste des figures.....	vi
<b>1. Introduction.....</b>	<b>1</b>
<b>1.1 Qu'est-ce qu'un client riche ?.....</b>	<b>2</b>
1.1.1 Le client lourd .....	2
1.1.1.1 Ses forces.....	2
1.1.1.2 Ses faiblesses .....	2
1.1.2 Le client léger .....	2
1.1.2.1 Ses forces.....	2
1.1.2.2 Ses faiblesses .....	2
1.1.3 Le client riche .....	3
1.1.4 Technologies existante .....	4
<b>1.2 Une application multiplateforme ? .....</b>	<b>7</b>
<b>2. Contexte de mon projet.....</b>	<b>9</b>
<b>2.1 La société Qim info.....</b>	<b>9</b>
2.1.1 Organigramme.....	10
2.1.2 Les objets métiers .....	11
2.1.2.1 Collaborateur .....	11
2.1.2.2 Unité opérationnelle.....	11
2.1.2.3 Trigramme .....	11
2.1.2.4 Tâche.....	11
2.1.2.5 Mission.....	11
2.1.2.6 Revue (de mission).....	12
2.1.2.7 Formation.....	12
2.1.2.8 Certification.....	13
2.1.2.9 Contact .....	13
<b>2.2 SIRH v4.0 (l'ancienne) .....</b>	<b>13</b>
2.2.1 Architecture de l'application .....	13
2.2.1.1 Synchronisation des données .....	14
2.2.2 Fonctionnalités .....	15
2.2.2.1 Afficher le détail d'un collaborateur .....	16
2.2.2.2 Calendriers de revues, formations, certifications et missions .....	17
2.2.2.3 Afficher les contacts d'un responsable .....	18
<b>2.3 Objectifs à atteindre .....</b>	<b>18</b>
<b>2.4 Ressources à disposition .....</b>	<b>19</b>
<b>3. Développement de SIRH Web.....</b>	<b>20</b>
<b>3.1 Les besoins métiers .....</b>	<b>20</b>
<b>3.2 Base de données .....</b>	<b>21</b>
3.2.1.1 Système de gestion de base de données .....	21

3.2.1.2	Analyse et conception .....	21
<b>3.3</b>	<b>Conception des maquettes .....</b>	<b>22</b>
3.3.1	Technologie .....	22
3.3.2	Aperçu des maquettes .....	22
3.3.3	Validation.....	23
<b>3.4</b>	<b>Structure de l'application.....</b>	<b>23</b>
<b>3.5</b>	<b>Design de l'application.....</b>	<b>25</b>
<b>3.6</b>	<b>Edition d'objets.....</b>	<b>26</b>
<b>4.</b>	<b>Présentation de SIRH Web.....</b>	<b>29</b>
<b>4.1</b>	<b>Interface utilisateur.....</b>	<b>30</b>
4.1.1	Interactions possibles .....	31
<b>4.2</b>	<b>Fonctionnalités .....</b>	<b>31</b>
4.2.1	Le filtrage par unité opérationnelle.....	31
4.2.2	La recherche syntaxique.....	32
4.2.3	Les modes d'affichage.....	33
4.2.4	Page des collaborateurs .....	34
4.2.5	Page de détail d'un collaborateur.....	35
4.2.6	Page des missions.....	36
4.2.7	Page de détail d'une mission .....	37
4.2.8	Page des revues.....	38
4.2.9	Page des formations.....	39
4.2.10	Page des certifications .....	40
4.2.11	Page des contacts.....	41
<b>4.3</b>	<b>État de développement.....</b>	<b>41</b>
<b>5.</b>	<b>Conclusion .....</b>	<b>42</b>

## Liste des figures

Figure 1 : Découpage MVC de AngularJS .....	5
Figure 2 : AngularJS, synchronisation modèle-vue .....	6
Figure 3 : Exemple de responsive design .....	8
Figure 4 : Logo de la société Qim info.....	9
Figure 5 : Organigramme Qim info.....	10
Figure 6 : Menu principal de SIRH v4.0 .....	14
Figure 7 : Schéma de synchronisation des données .....	15
Figure 8 : Détails du collaborateur sur SIRH v4.0 .....	16
Figure 9 : Calendrier des certifications sur SIRH v4.0 .....	17
Figure 10 : Calendrier des contacts sur SIRH v4.0 .....	18
Figure 11 : Aperçu des tables de la base.....	21
Figure 12 : Maquette des collaborateurs sur SIRH Web .....	22
Figure 13 : Maquette de détail d'une mission sur SIRH Web .....	23
Figure 14 : Maquette de modification d'une revue sur SIRH Web .....	23
Figure 15 : Schéma d'interactions de l'architecture.....	25
Figure 16 : Exemple d'édition du plugin Angular-xeditable.....	26
Figure 17 : Exemple d'édition d'une ligne avec Angular-xeditable.....	27
Figure 18 : Ajout d'un objet à l'aide de la fenêtre modal.....	28
Figure 19 : Affichage des détails d'un objet.....	28
Figure 20 : Aperçu de l'interface de SIRH Web sur un écran d'ordinateur.....	30
Figure 21 : Aperçu de SIRH Web sur un écran de smartphone.....	30
Figure 22 : Interaction de l'interface SIRH Web .....	31
Figure 23 : Interaction de l'interface SIRH Web sur mobile .....	31
Figure 24 : Aperçu du mode « vignettes ».....	33
Figure 25 : Aperçu du mode « liste » .....	33
Figure 26 : Aperçu de la page des collaborateurs .....	34
Figure 27 : Aperçu de la page de détail d'un collaborateur .....	35
Figure 28 : Aperçu de la page des missions .....	36
Figure 29 : Aperçu de la page du détail d'une mission.....	37
Figure 30 : Aperçu de la page des revues.....	38
Figure 31 : Aperçu du détail de la revue.....	38
Figure 32 : Aperçu de la page formations .....	39
Figure 33 : Aperçu du détail de la formation.....	39
Figure 34 : Aperçu de la page des certifications.....	40
Figure 35 : Aperçu du détail d'une certification.....	40
Figure 36 : Aperçu de la page des contacts .....	41
Figure 37 : Aperçu du détail d'un contact.....	41

# 1. Introduction

Dans le cadre de mon travail de Bachelor réalisé durant ma formation d'informaticien de gestion à la Haute École de Gestion de Genève, j'ai fait le choix d'orienter mes recherches sur un sujet d'actualité, « les clients riches multiplateformes ». En effet, de nos jours nous remarquons que les habitudes des utilisateurs tendent de plus en plus vers une utilisation principale sur smartphones et tablettes rendant par conséquent les applications uniquement disponibles sur PC assez vite contraignantes à utiliser.

Ayant déjà un certain attrait pour le sujet avant de débiter ce travail, je savais qu'il allait m'être impossible d'aborder l'ensemble des solutions disponibles concernant le client riche multiplateforme. De plus l'évolution constante des technologies rendrait plus ou moins obsolète ce travail d'ici quelques mois.

C'est pourquoi j'ai fait le choix de réaliser cette recherche non pas seulement de manière théorique, mais également de manière pratique en l'appliquant à une situation réelle. Cela me permettra de proposer un état de l'art à ce jour mais également de montrer plus globalement les étapes de conception et développement de ce type de projet sans pour autant être focalisé sur une technologie particulière.

L'opportunité de cette mise en pratique m'a été offerte par une société de services informatiques de la région genevoise : Qim info.

Ce qui va suivre ne va donc pas être une solution à tous vos problèmes de développement, en revanche je vais tenter de vous présenter de manière concise les différents paramètres à prendre en compte, un aperçu des technologies existantes mais également les moments clés du développement d'un client riche multiplateforme.

Afin d'établir les bases nécessaires à la compréhension de cette recherche, je détaillerai, dans un premier temps, ce qu'est un client riche mais également quelle dimension prend le mot « multiplateforme » dans ce projet.



## 1.1 Qu'est-ce qu'un client riche ?

Pour expliquer ce qu'est un client riche, il me semble tout d'abord nécessaire de revenir sur les deux autres catégories majeures de clients (également nommées applications) que l'on distingue à ce jour. Je détaillerai les forces et les faiblesses de chacune.

### 1.1.1 Le client lourd

Le client lourd, souvent appelé logiciel par le grand public, est principalement caractérisé par une installation sur la machine de l'utilisateur. VLC, iTunes, Mozilla Firefox, Paint sont quelques exemples très connus de clients lourds.

#### 1.1.1.1 Ses forces

La mise en œuvre d'un client lourd comporte certains avantages et c'est pour cette raison qu'il reste aujourd'hui encore un choix privilégié lors du développement d'applications :

- Étant installé sur le système, le logiciel peut dialoguer facilement avec le reste du système.
- Le logiciel profite des performances de la machine ainsi que de sa rapidité d'accès à la mémoire.

#### 1.1.1.2 Ses faiblesses

Depuis le début des années 2000 cependant, on remarque une baisse significative dans l'adoption du client lourd. Cette baisse est sans doute due à ses différentes faiblesses :

- Les installations et les mises à jour sont à effectuer sur chaque poste et demandent donc une organisation importante pour le déploiement.
- Le client lourd est développé pour un système défini et par conséquent dispose d'une compatibilité quasi inexistante avec d'autres systèmes.

### 1.1.2 Le client léger

Basé sur les faiblesses du client lourd, le client léger (également appelé application internet) ne nécessite pas d'être installé. L'ensemble de ses calculs sont effectués sur un serveur qui envoie les pages une fois générées sur la machine de l'utilisateur.

#### 1.1.2.1 Ses forces

- Comme expliqué ci-dessus, le client léger facilite l'installation et le déploiement de mises à jour. En effet, cette tâche devient obsolète côté client, il suffit de mettre à jour le code côté serveur.
- Étant affiché au travers d'un navigateur internet, le client léger dispose d'une meilleure compatibilité que le client lourd puisqu'il se rend disponible sur n'importe quel système disposant d'un navigateur.

#### 1.1.2.2 Ses faiblesses

Le client léger, malgré sa correction de l'une des principales carences du client lourd, dispose d'autres limitations liées à son architecture :

- L'accès aux ressources du système de l'utilisateur est limité, par conséquent il devient bien plus compliqué de dialoguer avec certains services.
- Les performances sont dépendantes de la charge du serveur, mais aussi de la vitesse de connexion.
- L'ergonomie n'est pas optimale car pour chaque demande d'informations, la page doit être recalculée et rechargée côté serveur.
- Bien que proposant en théorie une compatibilité sur l'ensemble des navigateurs, l'affichage de notre client léger peut diverger d'un navigateur à l'autre.

### **1.1.3 Le client riche**

Le client riche arrive donc avec comme objectif d'allier les forces du client lourd avec celles du client léger. Pour ce faire, l'application riche reprend la structure de base du client léger : une architecture client-serveur avec un accès au travers d'un navigateur web. Le client riche comble également les carences du client léger, à savoir une interaction et une ergonomie limitées.

Pour cela, il s'appuie sur de nouvelles technologies permettant d'une part de déléguer une partie du traitement à la machine utilisateur afin de moins charger le serveur, et d'autre part de récupérer des informations sur le serveur sans pour autant avoir à régénérer toute la page.

De ce fait l'application devient plus réactive et fonctionne aux yeux de l'utilisateur comme s'il s'agissait d'une application installée sur sa machine (client lourd).

Cependant l'application riche n'est pas sans défauts. Le premier d'entre eux se traduit par sa complexité de conception. Pour fonctionner de manière interactive tout en étant sécurisée, l'application riche rajoute une couche de code qui augmente la complexité par rapport au client léger. La seconde limitation est celle de la connexion internet, bien que de moins en moins restrictive grâce aux progrès effectués dans le domaine. En effet, l'ensemble des données transitent par le réseau ; par conséquent, une mauvaise connexion pourrait donc directement impacter le temps de récupération des données affichées. Finalement les technologies étant assez nouvelles et pas toujours parfaitement standardisées, certains soucis de compatibilité, principalement avec d'anciens navigateurs, peuvent survenir.

#### 1.1.4 Technologies existante

Jusque récemment, lorsque l'on évoquait les technologies destinées à la création d'un client riche, on pensait tout de suite à Adobe Flex, Microsoft Silverlight ou encore JavaFX.

Ces technologies, pour la plupart propriétaires, reposent toutes sur un plug-in installé soit dans le navigateur, soit directement sur la machine de l'utilisateur. Ce plugin permet ensuite d'exécuter le code sur la machine et de proposer donc des contrôles semblables à ceux d'une application lourde.

Si aujourd'hui les versions récentes des navigateurs sont pour la plupart disposées à proposer ces différents plugins, il subsiste tout de même de réels soucis de compatibilité lorsque l'on tente d'accéder à ce genre d'applications. En effet, les appareils de type smartphones, tablettes, télévisions connectées mais aussi une grande partie des navigateurs installés sur des appareils un peu plus exotiques tels que les ordinateurs de bords de certains véhicules, n'autorisent pas l'installation de tels plugins.

Conscient du réel gain que peut apporter une application riche, les différents acteurs œuvrant pour le HTML et le Javascript ont également commencé à réfléchir sur des solutions pour enrichir les simples clients légers auxquels ils étaient confinés.

De ce travail ont émergés différents Framework Javascript qui permettent alors de proposer des interactions côté utilisateur très proches de ce que les clients riches pouvaient offrir jusque-là. Les récents développements du HTML5 avancent également dans le sens d'une meilleure interaction avec l'utilisateur en proposant nativement certaines actions autrefois réservées au plugin, telles que le « glisser-déposer », la gestion des médias, etc.

Cette solution basée sur des technologies HTML/Javascript (+ AJAX pour les interactions avec la base de données) a comme avantage de ne nécessiter aucune installation côté client.

Aujourd'hui, deux mouvements s'opposent : ceux qui prônent le client riche avec un bout de code installé sur la machine utilisateur et ceux qui défendent le client riche sans aucune installation.

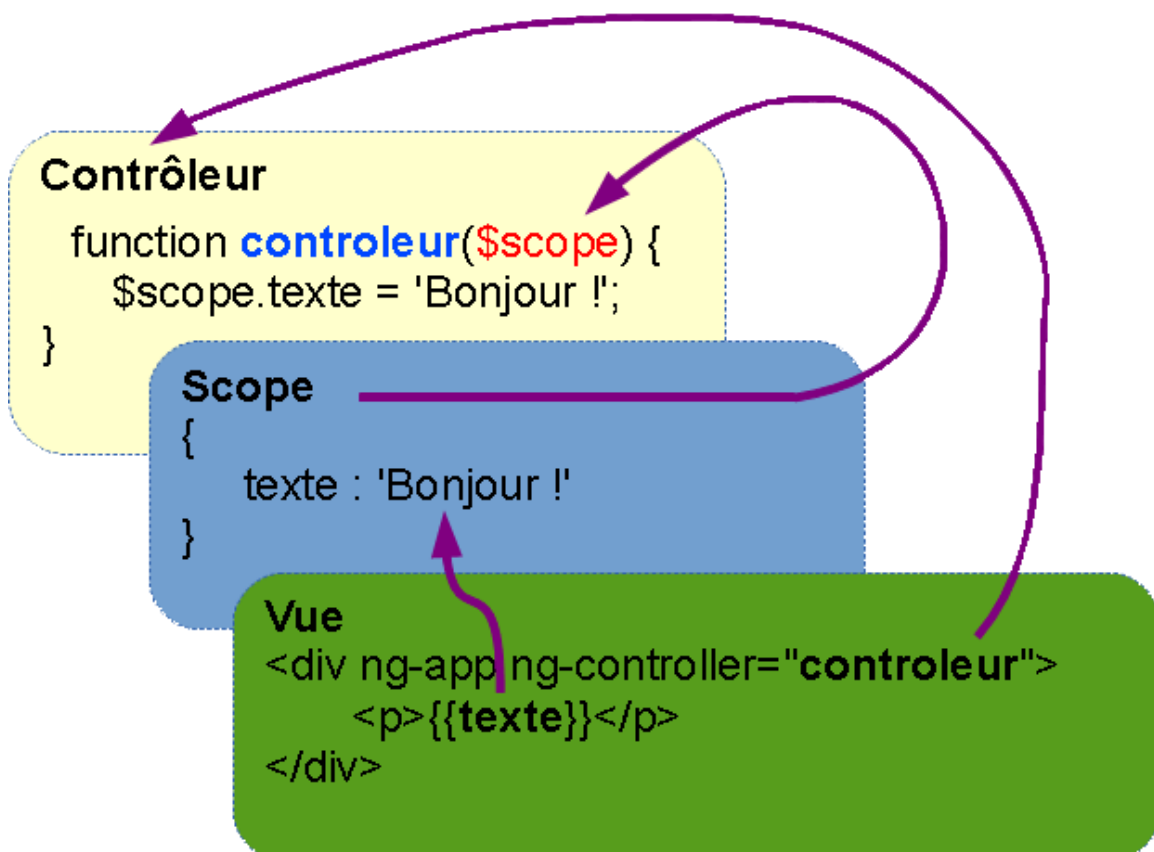
Dans le cadre de ce projet, j'ai fait le choix de partir sur une solution basée sur des technologies HTML/Javascript (+AJAX). Ce choix a notamment été motivé par la plus grande compatibilité de cette solution du fait de son architecture qui ne repose pas sur

un plugin installé. Ma seconde motivation a été la découverte du Framework Javascript : AngularJs.

Ce Framework open-source, soutenu par Google et très en vogue en ce moment, propose de concevoir des applications web en étendant les fonctionnalités du HTML.

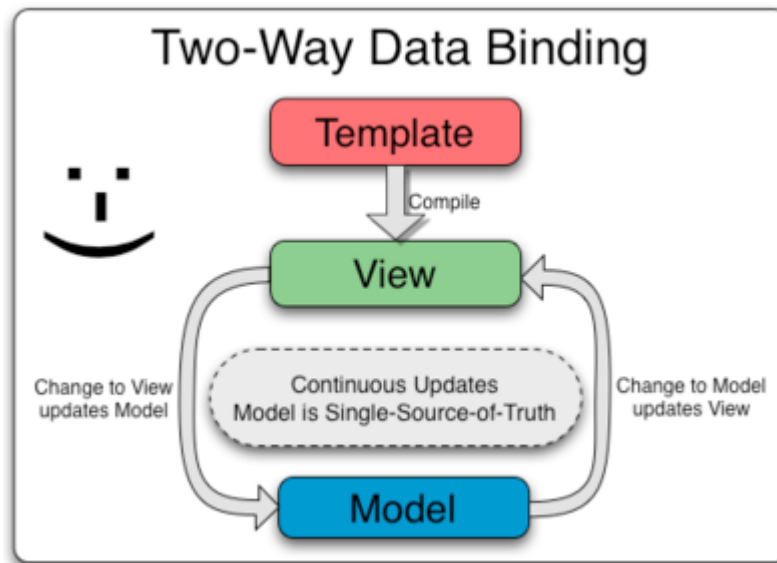
Il repose sur un modèle MVC (modèle-vue-contrôleur) qui a l'avantage de découpler les données de l'application, de l'interface utilisateur ainsi que des traitements.

Figure 1 : Découpage MVC de AngularJS



La principale force d'AngularJS est de proposer une liaison des données fonctionnant dans les deux sens entre la vue et le modèle, également appelée « two-way data-binding ». Cette liaison permet donc de mettre à jour automatiquement notre page affichée sur la machine utilisateur lorsque les données sont modifiées, mais également l'inverse.

Figure 2 : AngularJS, synchronisation modèle-vue



<http://www.inqbaton.com/wp-content/uploads/2013/11/binding.png>

Par exemple, lorsque l'utilisateur saisit un texte dans un formulaire, les données sont mise à jour en direct dans le modèle. Plus besoin de recharger la page contrairement aux applications de type « client léger ».

La synchronisation du modèle avec la base de données peut ensuite s'effectuer à l'aide d'une requête AJAX et fonctionner de manière totalement transparente pour l'utilisateur puisque les données, une fois récupérées, sont mises à jour dans la vue sans aucun rechargement de la page.

## 1.2 Une application multiplateforme ?

Le terme « application multiplateforme » peut être interprété de différentes manières. Dans le cadre de cette recherche, le terme multiplateforme ne signifie pas seulement une disponibilité sur toutes les machines pourvues d'un navigateur internet, mais également une proposition d'ergonomie cohérente et adaptée à chaque machine. Cela implique donc une adaptation en temps réel de notre application à l'environnement de l'utilisateur. Qu'il soit sur iPhone, Android, Windows Phone, ordinateur, Mac ou tablette, l'utilisateur doit pouvoir utiliser l'application comme si elle avait été développée pour son appareil. L'avantage pour le développeur réside dans le développement d'une application unique qui comporte également l'avantage de n'avoir à mettre à jour qu'un seul code en cas de modification. En revanche, le défaut de cette méthode devient la difficulté de développement. Ce dernier devient forcément plus complexe qu'un développement destiné à une plateforme unique.

Le terme en vogue actuellement est le « responsive design ». Ce dernier consiste à développer une seule et unique interface qui s'adaptera en fonction de la taille de l'écran de l'utilisateur.

Pour réaliser cette adaptation, le principe du responsive design est de proposer un design sous forme d'une grille de plusieurs colonnes de large occupant la totalité de l'écran de l'utilisateur.

Le développeur peut ensuite, pour chacun des éléments de son design, définir à l'aide de classes CSS le nombre de colonnes que prend l'élément. Lorsque la taille de l'écran se réduit, la taille des colonnes se réduit également et par conséquent l'élément diminue. Jusque-là rien de nouveau me direz-vous, autant fixer la taille de l'élément comme un pourcentage de la page.

Mais là où le responsive design propose une avancée, c'est lorsqu'il couple ce système de grilles à l'utilisation des media-queries disponibles dans CSS3. Les media-queries nous donnent accès en temps réel à certaines données telles que la taille de la fenêtre du navigateur. Cette donnée va ainsi nous permettre de définir des classes en fonction de la taille de l'écran.

Afin de vous donner un exemple un peu plus parlant, je vais prendre l'exemple de fonctionnement de Bootstrap, le Framework le plus populaire à ce jour proposant des fonctionnalités de responsive design.

Bootstrap base sa grille de design sur un nombre de 12 colonnes puis, à l'aide des media-queries, définit 4 catégories de tailles d'écran : très petit, petit, moyen et grand. Les classes définies par Bootstrap nous permettent donc pour chaque élément de spécifier le nombre de colonnes que l'élément doit prendre en tenant compte de la catégorie de résolution de l'écran actuel.

Par exemple si je décide d'afficher un élément sur la moitié d'un écran de bureau je vais utiliser une classe : grand-6 (taille de l'écran – nombre de colonnes). Si je désire que ce même élément prenne toute la largeur de l'écran de téléphone je vais alors lui ajouter la classe : trèsPetit-12.

Bootstrap propose également des classes permettant de masquer les éléments sur certaines résolutions d'écran. Par exemple, si je souhaite masquer mon élément sur le format mobile, il me suffit d'ajouter la classe : masqué-trèsPetit. Lorsque les media-queries détecteront un écran de très petite taille à l'aide de la CSS, elles masqueront mon élément.

Figure 3 : Exemple de responsive design



[http://.wikimotive.com/wikiblog/wp-content/uploads/sites/2/2013/11/Responsive\\_Design\\_Kinetic\\_Knowledge1.jpg](http://.wikimotive.com/wikiblog/wp-content/uploads/sites/2/2013/11/Responsive_Design_Kinetic_Knowledge1.jpg)

## 2. Contexte de mon projet

Comme expliqué précédemment, j'ai eu la chance de pouvoir effectuer mon travail de Bachelor pour Qim Info, société de services informatiques située à Genève.

Durant 4 mois, à mi-temps, j'ai pris part aux activités du plateau de développement au sein duquel j'ai été chargé de développer de A à Z une nouvelle version d'une application destinée principalement aux responsables des différentes unités opérationnelles de la société.

Cette application nommée SIRH (en version 4.0 à mon arrivée dans la société) est une application de type client lourd, basée sur une technologie Microsoft Access. Mon objectif était donc de prendre connaissance des fonctionnalités de l'application, ainsi que des nouveaux besoins en termes de fonctionnalités pour pouvoir ensuite développer une version 5.0 de cette application sous la forme d'un client riche. Cette version 5.0, reposant sur la structure d'une application riche, est également appelée SIRH Web.

Avant de vous détailler plus amplement l'application existante (SIRH 4.0) à mon arrivée dans la société, je vais vous présenter brièvement la société Qim info, sa structuration mais également les différents termes métiers utilisés par l'entreprise, qui sont par conséquent nécessaires à la compréhension des fonctionnalités de l'application.

### 2.1 La société Qim info

Figure 4 : Logo de la société Qim info



<http://www.qiminfo.ch/templates/templateqiminfo/images/logo.png>

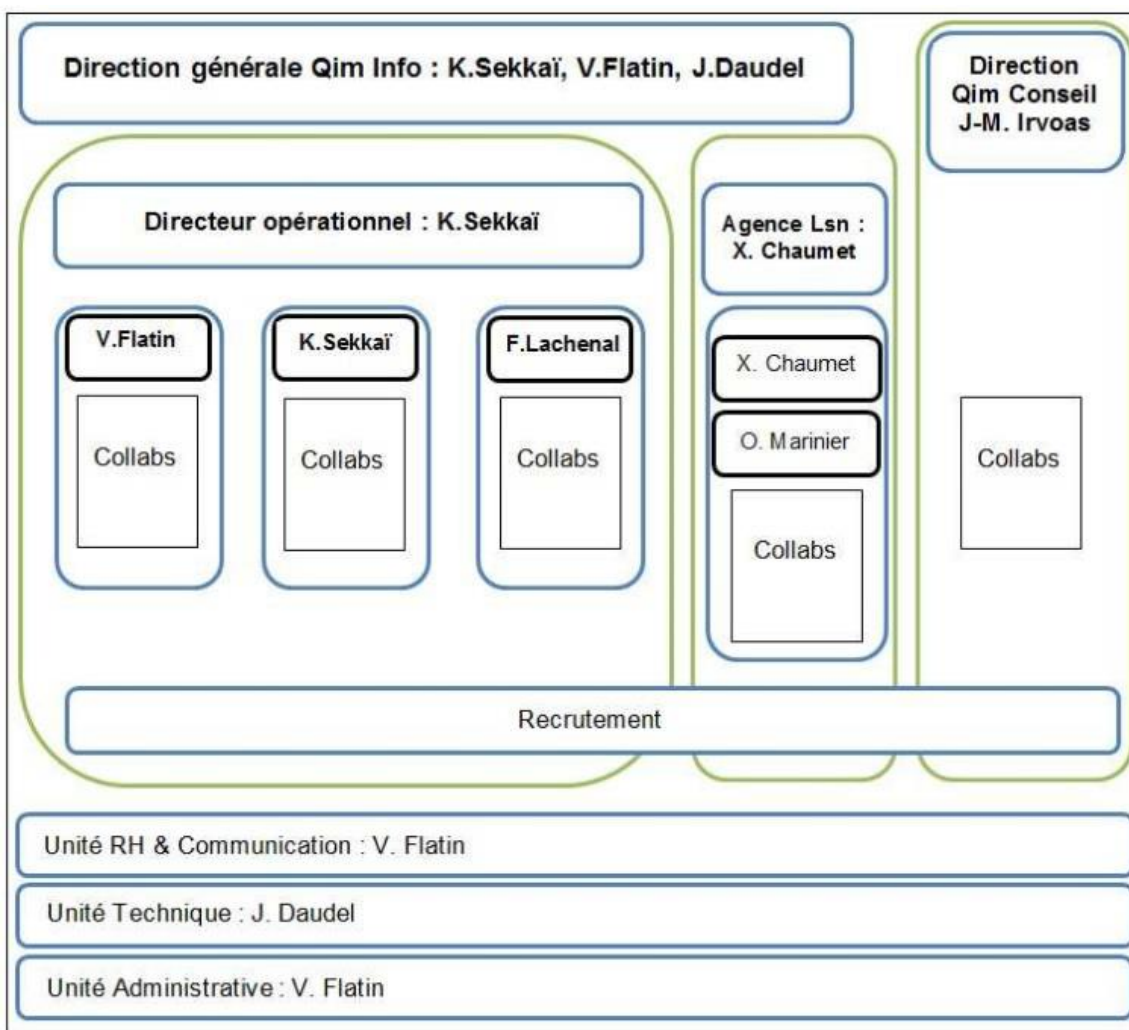
La S.A Qim info a été fondée en 2004 par des personnes issues du domaine du service informatique. Ses locaux sont situés dans la ville de Carouge au sein du canton de Genève. Depuis 2010, la société dispose également de locaux dans la ville de Lausanne.



A ce jour, la société compte plus de 120 collaborateurs mais seule une trentaine d'entre eux sont présents dans les locaux. En effet, la plupart des collaborateurs de Qim info sont envoyés pour des missions de plus ou moins longue durée chez le client. Les prestations proposées par la société sont diverses et variées : développement d'applications mobiles, conception de site internet en passant par l'analyse et la maintenance de solutions existantes, etc.

### 2.1.1 Organigramme

Figure 5 : Organigramme Qim info



Livret d'accueil, page 7, Intranet Qim info

## **2.1.2 Les objets métiers**

### **2.1.2.1 Collaborateur**

Un collaborateur est un employé qui travaille pour la société. Ce dernier a un statut : soit il est actif et travaille donc encore à ce jour pour Qim info, soit son statut est inactif, ce dernier signifiant qu'il ne travaille plus pour la société.

Dans le cadre de notre application, nous allons également nous intéresser à certains collaborateurs qui sont responsables d'unités opérationnelles.

### **2.1.2.2 Unité opérationnelle**

Une unité opérationnelle (également appelée UO) est représentée par un groupe de collaborateurs, un portefeuille de clients ainsi qu'un responsable.

La société Qim info est composée de 6 unités opérationnelles. A noter qu'il est tout à fait possible qu'un collaborateur réalise des missions dans plusieurs unités opérationnelles.

### **2.1.2.3 Trigramme**

Le trigramme est un identifiant composé de 3 lettres découlant directement du nom et du prénom du collaborateur. Cet identifiant est unique et global à l'ensemble des applications Qim info. Il a l'avantage d'être plus compréhensible et plus facilement mémorisable qu'un simple identifiant numérique.

### **2.1.2.4 Tâche**

Une tâche est une composante d'une mission. Elle est réalisée par un collaborateur.

Une tâche est également définie par une date de début ainsi qu'une date de fin, elle comporte de plus un montant.

### **2.1.2.5 Mission**

Une mission est le produit que vend la société Qim info. La mission se déroule sous la direction d'un responsable d'unité opérationnelle et sera réalisée par un ou plusieurs collaborateurs de cette même unité opérationnelle. La mission peut être effectuée pour un client externe mais elle peut aussi être interne. Dans ce deuxième cas le client n'est autre que la société Qim info.

Les missions sont marquées « terminées » une fois la livraison effectuée.

A ce jour nous distinguons deux types de missions :

- La mission régie  
C'est une mission composée d'une seule tâche et donc d'un seul collaborateur.
- La mission forfait  
C'est une mission avec un prix forfait, qui peut être composée d'une ou plusieurs tâches et qui, par conséquent, peut comporter un ou plusieurs collaborateurs.

#### **2.1.2.6 Revue (de mission)**

Une revue de mission est un point de contrôle réalisé par un responsable d'unité opérationnelle. Cette revue concerne le travail d'un collaborateur sur une mission. Le responsable prend alors connaissance de l'état de la mission en compagnie du collaborateur et du client.

La revue peut être de différents statuts :

- A planifier
- Planifiée
- Réalisée
- Terminée

La différence entre « réalisée » et « terminée » est assez subtile. Une fois que la revue s'est déroulée, elle est « réalisée ». Et c'est uniquement lorsque le rapport de revue est envoyé qu'elle peut prendre le statut de « terminée »

#### **2.1.2.7 Formation**

Qim info propose à ses collaborateurs de se former. La formation qui est donc associée à un collaborateur peut avoir différents statuts :

- Évoquée  
La formation évoquée n'a pas encore été planifiée, elle représente une idée de formation à effectuer.
- Planifiée  
La formation est planifiée sur des dates précises, elle est en attente d'être réalisée.
- Réalisée  
La formation a été réalisée.
- Abandonnée  
La formation a été abandonnée avant d'être réalisée.

Les formations sont classifiées en catégories.

### **2.1.2.8 Certification**

Une certification est liée à un collaborateur, elle ne comporte pas de statut particulier mais elle contient une date de début ainsi qu'une date de fin.

Tout comme les formations, les certifications sont classifiées en catégories.

### **2.1.2.9 Contact**

Le contact est un rendez-vous informel entre un responsable d'unité opérationnelle et l'un de ses collaborateurs.

## **2.2 SIRH v4.0 (l'ancienne)**

Maintenant que nous avons pris connaissance de l'ensemble des objets métiers nécessaires à la compréhension de la suite de ce travail, je vais vous présenter l'application SIRH en version 4.0. Cette application était donc celle utilisée à mon arrivée dans la société au premier avril 2014.

Il faut savoir que SIRH comporte principalement deux axes de travail : l'axe « candidats » et l'axe « collaborateurs ».

L'axe « candidats » concerne la gestion des nouveaux candidats, de leur dossier de candidature ainsi que des différentes interactions et rendez-vous liés aux recrutements.

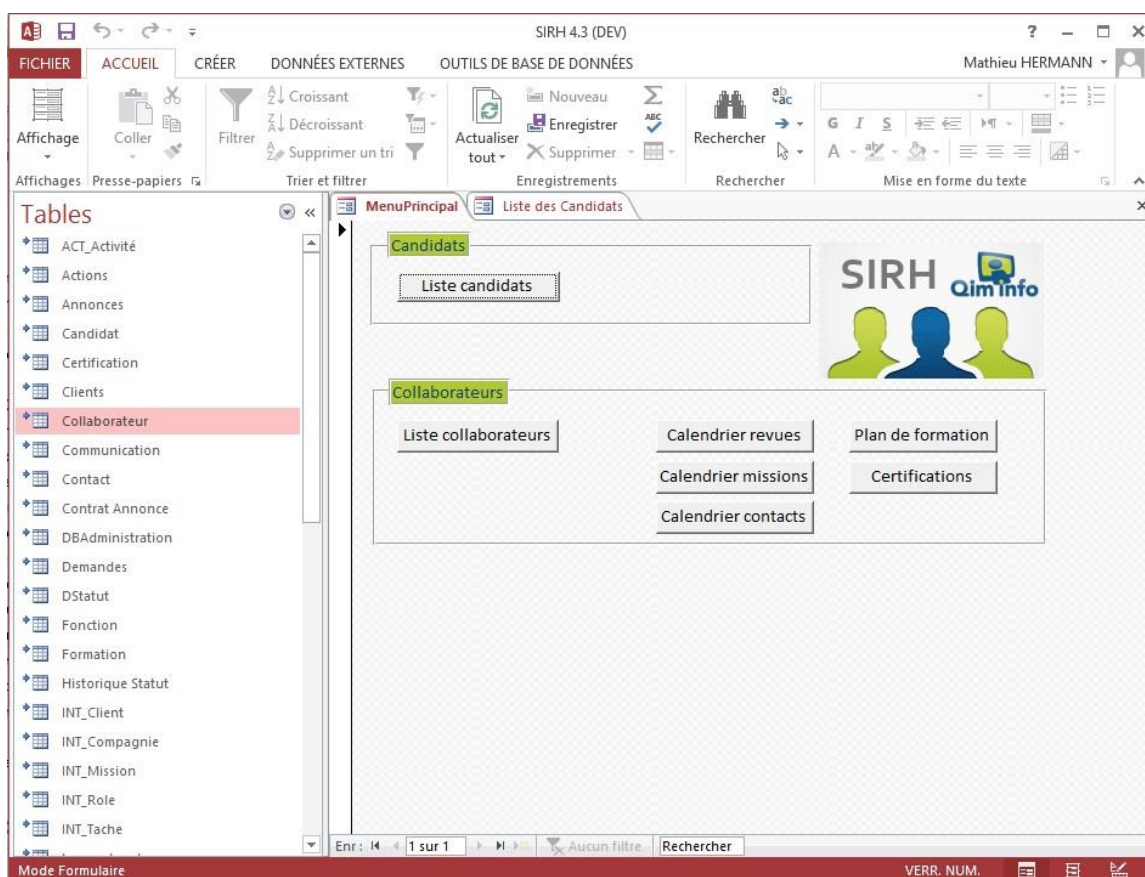
L'axe « collaborateurs », lui, concerne l'ensemble des collaborateurs de la société, les missions, revues, formations, certifications et contacts. C'est sur cet axe que s'est focalisé mon travail durant ces 4 derniers mois.

### **2.2.1 Architecture de l'application**

L'application est un client lourd installé sur un disque réseau de la société Qim info. Elle est disponible au travers du réseau sur l'ensemble des postes de la société.

Ce client lourd est développé au sein de l'application Microsoft Access, un système de gestion de base de données faisant partie de la suite Microsoft Office.

Figure 6 : Menu principal de SIRH v4.0

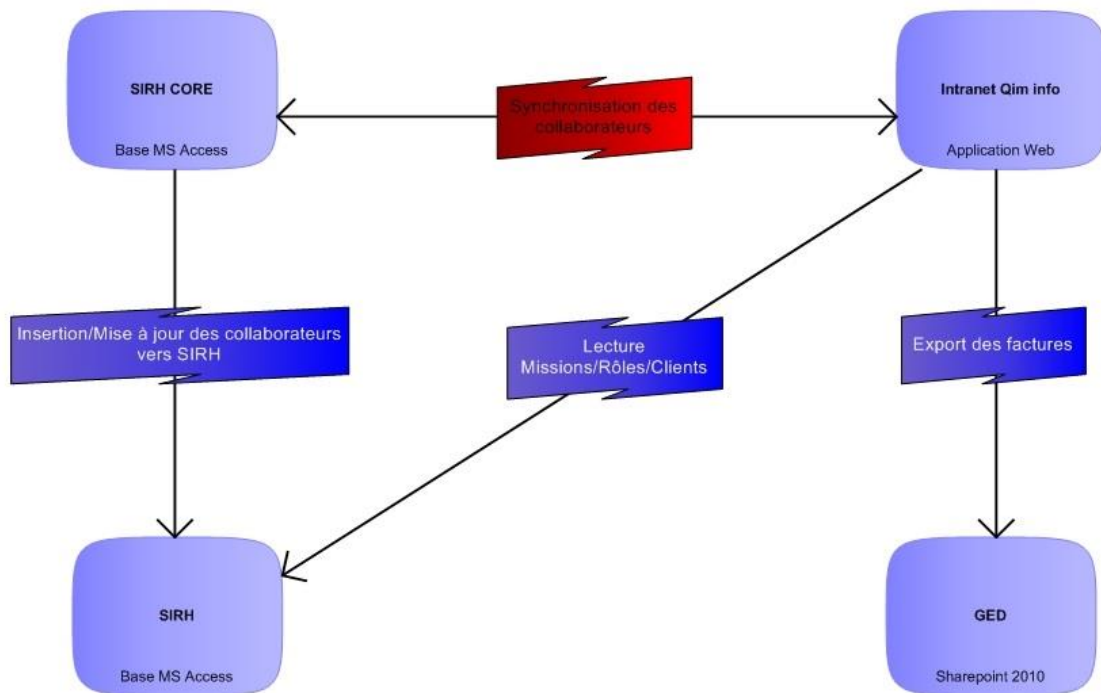


### 2.2.1.1 Synchronisation des données

On distingue deux types de données dans la base de données de SIRH : celles récupérées depuis la base de l'intranet Qim info et celles contenues directement dans la base Access de l'application SIRH.

La gestion des missions ainsi que celle des clients est déléguée à l'intranet. Pour cette raison, un script a été mis en place afin que chaque soir, les données présentes sur l'intranet soient synchronisées dans la base de données SIRH. Pour pouvoir les reconnaître, les tables préfixées : « INT\_ » sont générées par l'intranet. La table « Collaborateur », elle, se synchronise sur SIRH Core, le gestionnaire des collaborateurs de la société.

Figure 7 : Schéma de synchronisation des données



### 2.2.2 Fonctionnalités

Je vais à présent vous exposer les fonctionnalités de l'application SIRH v4.0. Je n'évoquerai pas du tout l'axe candidats cité précédemment, je me baserai ici uniquement sur l'axe collaborateurs.

## 2.2.2.1 Afficher le détail d'un collaborateur

Figure 8 : Détails du collaborateur sur SIRH v4.0

The screenshot displays the SIRH v4.0 web application interface. At the top, there is a navigation bar with tabs: FICHIER, ACCUEIL, CRÉER, DONNÉES EXTERNES, and Outils de base de données. Below this is a ribbon with various icons for actions like 'Couper', 'Copier', 'Coller', 'Filtrer', 'Actualiser tout', 'Nouveau', 'Enregistrer', 'Supprimer', and 'Rechercher'. The left sidebar contains a 'Tables' menu with a list of database tables, including 'Collaborateurs'. The main content area shows the details of a collaborator named 'Ima MALOT'. At the top, there are input fields for 'Trigramme' (Ima), 'Nom' (MALOT), and 'Prénom' (Loïc). Below these are checkboxes for 'Collab. actifs uniquement', 'Collab. sans mission', 'Collab. avec mission mais sans revue prévue', and 'Sans contact envisagé'. The 'Formations' tab is selected, showing a list of training entries. The first entry has N° 225, Trigramme: Ima, Budget, Catégorie: Roles, Sous-catégorie: Chef de projet, Intitulé: Gestion de Projets Informatiques (GPI) : http://www.orsys, Statut: Réalisée, Période: 01.07.2013 - 03.07.2013, Durée: 3, and Organisme: Orsys. The second entry has N° 67, Trigramme: Ima, Budget, Catégorie: Méthodes & Outils, Sous-catégorie: Itil, Intitulé: Certification foundation v2 ou plutôt v3, Statut: Evoquée, and Organisme: [empty]. The bottom of the screen shows a status bar with 'Mode Formulaire' and 'VERR. NUM.'.

A l'aide de la barre supérieure, il est possible de rechercher un collaborateur par son trigramme, nom ou prénom afin d'afficher les informations le concernant.

Une fois le collaborateur sélectionné, le programme affiche un détail du collaborateur sous forme d'onglets pour les missions, revues, formations, certifications et contacts.

## 2.2.2.2 Calendriers de revues, formations, certifications et missions

Figure 9 : Calendrier des certifications sur SIRH v4.0

Date obtention	Initiale	Sujet
01.01.2004	MCP 70-320	Développement de services Web XML et de composants / Langages / CF ASP.net
01.01.2004	Visual C#	Visual C# .Net et de Microsoft .Net Framework / /
01.01.2003	Microsoft Certified Trainer	Microsoft Certified Trainer - Certification CEGOS / Langages / CF ASP.net
01.01.2003	MCP 70-315	Développement et mise en œuvre d'applications Web à / Langages / CF ASP.net
01.01.1999	MCP	Développement d'applications avec Microsoft Visual Basic 5.0 / /
30.06.2013	ITIL Foundation	ITIL Foundation Certificate in IT Service/Service Management / Methodes & Outils / Iti
01.09.2009	Certification Sun SCBCD 5	Sun Certified Business Component Develop / Langages / J2EE
01.06.2008	Certification Sun SOMCD 5	Sun Certified Web Component Develop / Langages / J2EE
01.10.2008	Certification Sun SCP 5	Sun Certified Java Programmer / Langages / Java
04.02.2011	Iti V3 foundation	Methodes & Outils / Iti
01.01.2002	MCP Windows 2000	Systeme / Windows
27.01.2012	ITIL Foundation v3	Methodes & Outils / Iti
01.01.2011	International Software Testing	International Software Testing Qualification Board / Methodes & Outils / STQB
07.06.2013	Technologies XML (Apollo formation)	Langages / XML
13.07.2011	ITIL v3 Foundation	Methodes & Outils / Iti
01.01.2005	MCSAM Exchange 2003	Groupware / Admin Exchange
01.01.2004	MCSÉ Windows 2003	Systeme / Windows
01.01.2002	MCSA Windows 2000	Systeme / Windows

L'application SIRH v4.0 permet également d'afficher un PDF sous forme de calendrier/liste de l'ensemble des revues, formations, certifications ou missions.

La liste est triée par statut, excepté pour les certifications qui sont elles triées par collaborateur.



### 2.2.2.3 Afficher les contacts d'un responsable

Figure 10 : Calendrier des contacts sur SIRH v4.0

Statut	Date Contact	Collaborateur	Manager	Commentaire
[redacted]	15.11.2001	[redacted]	[redacted]	Ropes CTI
[redacted]	15.11.2001	vic-[redacted]	[redacted]	Ropes CTI
[redacted]	15.11.2001	[redacted]	[redacted]	Ropes CTI
[redacted]	15.11.2001	gm-[redacted]	[redacted]	Ropes CTI
[redacted]	15.11.2001	[redacted]	[redacted]	Ropes CTI

SIRH v4.0 propose également la possibilité d'afficher les contacts d'un responsable spécifié. Pour ce faire, l'application invite à saisir le trigramme d'un responsable et affiche sous forme de liste triée par statut l'ensemble des contacts liés au responsable.

## 2.3 Objectifs à atteindre

La tâche qui m'a été confiée par la société Qim info consistait en le développement d'une application SIRH de type client riche multiplateforme, afin de rendre cette application disponible sur ordinateur, Mac, smartphone, tablette. Cette application se devait d'être compatible avec les différents systèmes que sont Windows 8, iOS et Android ainsi que les 4 principaux navigateurs : Internet Explorer, Safari, Mozilla Firefox et Google Chrome.

La nouvelle application devait implémenter les fonctionnalités présentes dans la version 4 de SIRH.

Mon travail avait également comme but de proposer une solution technologique compatible avec l'environnement informatique de Qim info.

## **2.4 Ressources à disposition**

Afin de réaliser ce travail dans les meilleures conditions, la société Qim info a mis à ma disposition un bureau sur leur plateau de développement dans leurs locaux situés à Carouge.

J'ai donc eu à ma disposition un ordinateur équipé de deux écrans sur lequel Microsoft Visual Studio était installé.

J'ai également disposé d'un serveur web de développement.

Ce projet a été encadré par Loïc Malot, chef de projet. Il m'a suivi tout au long de sa réalisation en optant pour une méthode agile. L'application finale a été découpée en différents lots, chaque lot comprenant plusieurs tâches. Cette méthode efficace, combinée à l'outil interne de suivi de projet, a permis à M. Malot de connaître à chaque instant l'avancée de mon travail par rapport au planning prévu.

## 3. Développement de SIRH Web

Je vais maintenant vous présenter le développement de l'application SIRH Web, en abordant les principaux axes de ce projet.

### 3.1 Les besoins métiers

Afin d'être sûrs de cerner les besoins du client, nous avons organisé une première réunion pour définir les différentes fonctionnalités de l'application actuelle. Cette réunion a aussi été l'occasion de prendre connaissances des différents termes métiers et du fonctionnement global de la société Qim info.

En nous basant sur l'application existante, nous avons identifié les fonctionnalités qui devaient être implémentées dans la nouvelle application : SIRH Web.

- Collaborateurs, Missions
  - Afficher tous
    - Mode vignettes / mode liste
    - Filtrer par unité opérationnelle
    - Trier
    - Rechercher
  - Afficher le détail
- Revues, formations, certifications, contact
  - Afficher tous
    - Mode vignettes / mode liste
    - Filtrer par unité opérationnelle
    - Trier
    - Rechercher
  - Modifier
  - Ajouter
  - Supprimer

Après avoir défini les fonctionnalités, j'ai pu passer à la seconde étape : la conception de la base de données.

## 3.2 Base de données

### 3.2.1.1 Système de gestion de base de données

Dans un premier temps, j'ai commencé à chercher quel système de gestion de base de données serait le plus adapté. L'application SIRH Web ne gérant pas une quantité énorme de données, j'ai exclu le choix d'une base non-relationnelle.

Partant de là, la société Qim info étant partenaire Microsoft et effectuant la plupart de ses développements sur l'outil Microsoft SQL Server, je me suis tout naturellement tourné vers cette solution technologique.

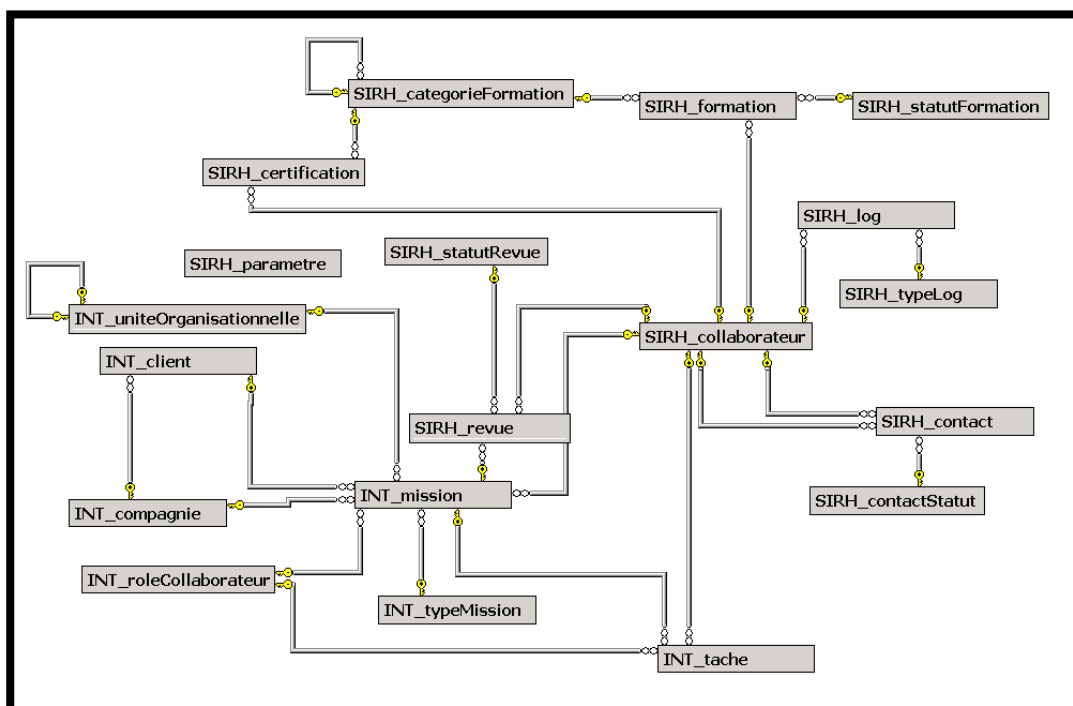
### 3.2.1.2 Analyse et conception

Pour concevoir la nouvelle base de données de SIRH Web, je me suis ensuite intéressé à l'ancienne base. A l'aide de la liste des fonctionnalités que nous avons définie un peu plus haut, j'ai tenté d'identifier les tables dont j'aurais besoin pour le bon fonctionnement de mon application. J'ai également rajouté des tables de statut afin de ne plus avoir à coder en dur le statut dans un attribut.

J'ai, par la suite, modifié l'ensemble des noms afin de respecter les normes de nommage de la société, normes qui n'étaient pas respectées par l'ancienne base.

Pour finir j'ai produit, à l'aide de l'outil intégré à Microsoft SQL Server, un schéma de base de données dans le but d'avoir un aperçu visuel de l'état de la base. Ce dernier aura été d'une aide précieuse pour faire valider mon schéma de base de données.

Figure 11 : Aperçu des tables de la base



### 3.3 Conception des maquettes

La troisième étape de mon projet a été de concevoir les maquettes des différents écrans de l'application SIRH Web. Le but de ces maquettes est de poser les bases du design de l'application.

#### 3.3.1 Technologie

Il existe une multitude d'outils permettant de produire des maquettes d'écrans. Dans mon cas, j'ai fait le choix d'utiliser balsamiq, un outil que j'ai déjà eu l'occasion d'utiliser à plusieurs reprises dans le cadre de projets informatiques.

Cet outil, disponible en ligne gratuitement dans une version d'essai amplement suffisante, permet de faire des maquettes de type « wireframe » (fil de fer) très aisément.

#### 3.3.2 Aperçu des maquettes

Figure 12 : Maquette des collaborateurs sur SIRH Web

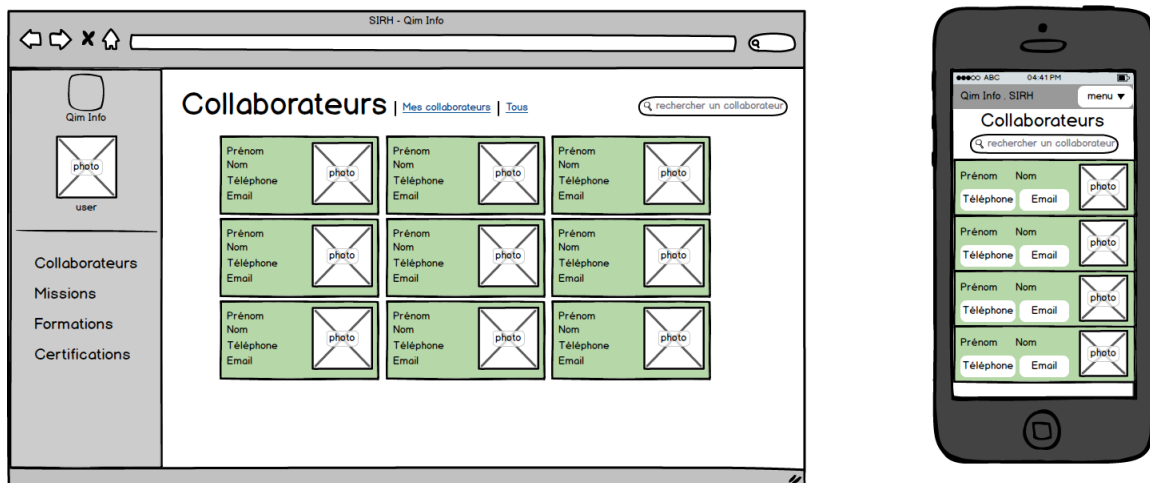


Figure 13 : Maquette de détail d'une mission sur SIRH Web

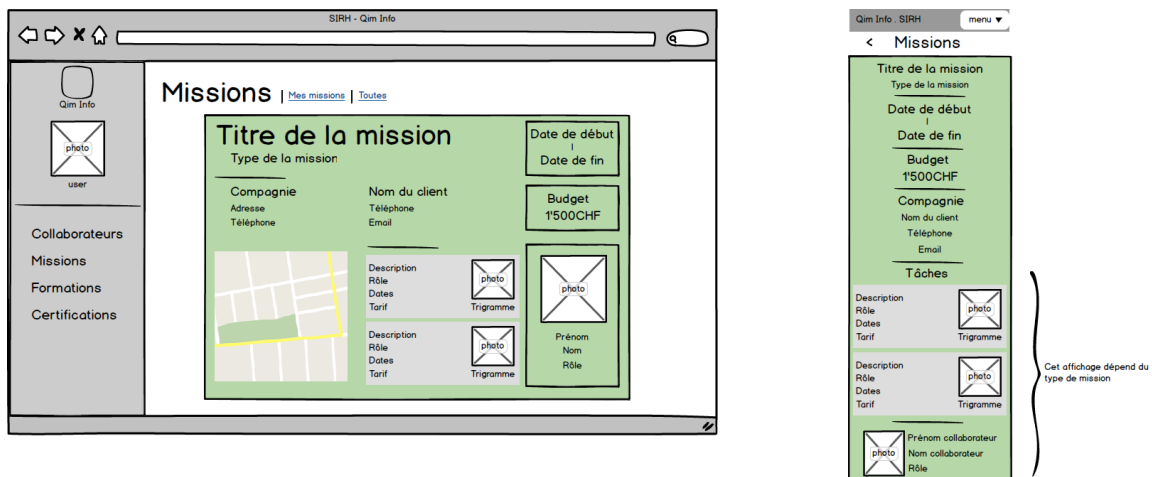
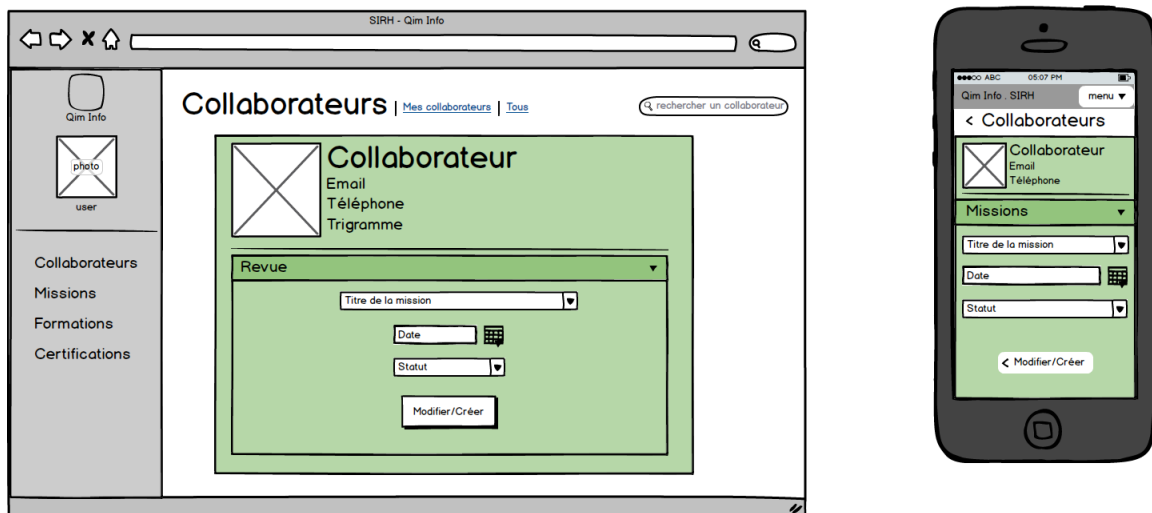


Figure 14 : Maquette de modification d'une revue sur SIRH Web



### 3.3.3 Validation

Comme énoncé précédemment, les maquettes doivent être validées avant de débiter le code à proprement parler. Nous avons donc organisé une courte réunion afin de discuter des différentes maquettes en salle de réunion. Certaines modifications ont été proposées par les futurs utilisateurs de l'application.

Suite aux quelques modifications effectuées j'ai, avec l'accord de mon chef, passé l'étape de conception de réels écrans sur Photoshop. Nous avons considéré que cette étape n'apporterait pas grand-chose et avons choisi de directement continuer sur la création du cœur même de l'application : sa structure.

## 3.4 Structure de l'application

Ne connaissant pas très bien au début de ce projet les technologies ASP.net, nous avons quand même fait le choix de partir sur une application basée sur le Framework

Modèle-Vue-Contrôleur (MVC) ASP.net de Visual Studio afin de limiter le nombre de technologies au sein de l'entreprise.

Cette architecture MVC permet, comme son nom l'indique, de séparer le modèle (données) de la vue (l'interface utilisateur) et du traitement (contrôleur). C'est un grand avantage pour la maintenance de l'application, car on peut aisément intervenir sur une couche sans avoir à modifier le code d'une autre couche.

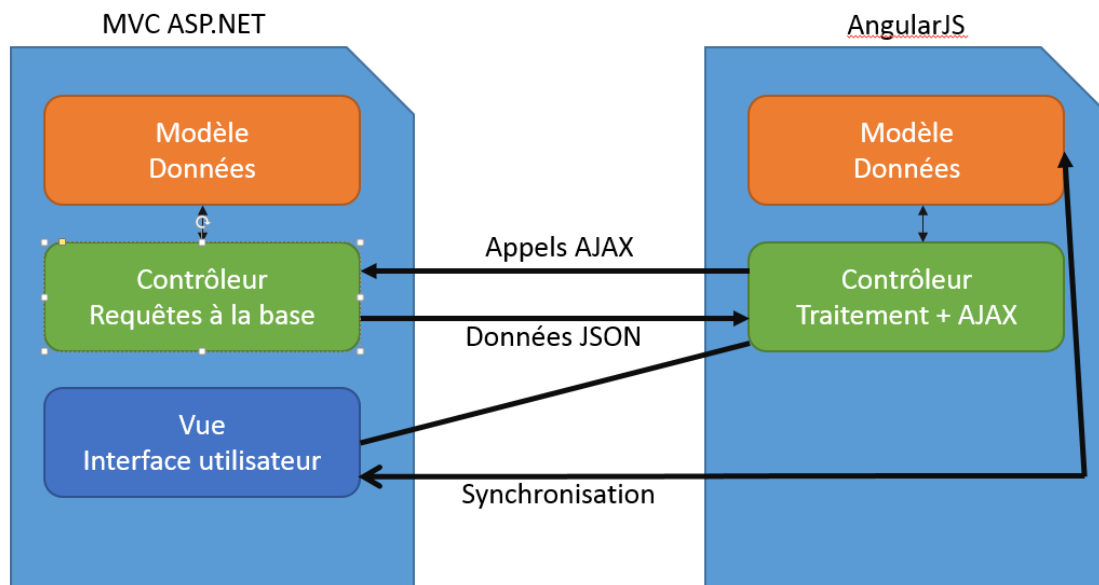
L'environnement Visual Studio est, de plus, plutôt simple à utiliser mais également très intuitif. De plus, le langage C# employé pour le codage des pages est vraiment facile à prendre en main une fois le background en programmation d'un étudiant de la 3<sup>ème</sup> année de la HEG acquis.

Peu après le début du projet, j'ai reçu une demande de la part du plateau de développement me demandant de regarder si il y'avait une possibilité d'ajouter AngularJS, un Framework Javascript très en vogue en ce moment.

Le souci à ce moment-là résidait dans la cohabitation entre AngularJS, et la structure MVC ASP.NET. En effet, les deux Frameworks sont basés sur un modèle MVC. De ce fait, ces derniers sont à la base développés pour fonctionner en tant qu'unique MVC. Par conséquent, j'ai été contraint de remanier grandement le code développé jusqu'ici. Ce remaniement consistait en choix de déléguer certaines tâches à l'un des Framework et certaines tâches à l'autre.

La gestion de changement des pages (routage) et l'interaction (requêtes) avec les données du modèle est laissée aux MVC ASP.NET, alors que la partie de traitements entre l'interface utilisateur et les données est assurée par le contrôleur AngularJS.

Figure 15 : Schéma d'interactions de l'architecture



Avec cette nouvelle structure, on continue de jouir des possibilités de notre MVC ASP.NET, tout en profitant de la synchronisation bidirectionnelle des données entre le modèle et l'interface utilisateur proposée par AngularJS.

Cette structure nous permet donc d'atteindre l'objectif de notre client riche. Les pages seront rechargées lors du changement de page par le MVC ASP.NET, alors que toutes les interactions à l'intérieur des pages seront réalisées par le contrôleur AngularJS qui, à l'aide de requête AJAX, récupèrera les données à la volée dans notre base de données. Ce traitement effectué côté client renforcera la caractéristique riche de notre application et rapprochera son utilisation de celle d'un client lourd.

### 3.5 Design de l'application

Après avoir dessiné les maquettes de l'application et créé la structure de base servant à l'application, j'ai pu m'attaquer au design. Le but était alors de proposer une application capable de s'adapter sur les différentes tailles d'écran comme présenté dans les maquettes.

Afin de simplifier la mise en œuvre et la maintenance de mon application dans le futur, j'ai choisi de partir sur le très populaire Framework Bootstrap. Ce dernier propose des classes de styles CSS par défaut, permettant de réaliser rapidement un design « responsive ». Entendez donc par là un design qui s'adapte à la résolution de l'écran de l'utilisateur.



Pourquoi ce choix ? Il vrai qu'il existe une multitude de Frameworks, et on reproche souvent à Bootstrap le fait que tous les sites se ressemblent. Mais Bootstrap dans ce projet constitue uniquement la base du design, il gère très bien les résolutions avec son système de grilles. Ainsi, il aurait été dommage de perdre du temps pour arriver à un résultat plus ou moins équivalent.

En revanche, j'ai grandement personnalisé Bootstrap à l'aide d'un fichier CSS afin de parfaire le design de l'application, avec comme objectif la reproduction du design proposé dans les maquettes.

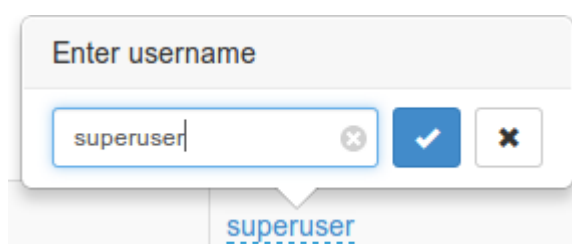
### 3.6 Edition d'objets

L'édition d'objets a été un point très important dans cette application. En effet, la saisie et la modification d'informations au sein de SIRH Web constituera l'un des principaux modes d'utilisation.

Le but a donc été de trouver une façon ergonomique d'ajouter, de modifier mais également de supprimer des objets. Après quelques recherches, j'ai fait la connaissance d'un plugin adapté à AngularJS nommé : Angular-xeditable.

Angular-xeditable est un plugin destiné à proposer une solution d'édition directe également appelée « edit-in-place » ou « click-to-edit ». Cette solution consiste à proposer un mode d'édition d'un objet en cliquant simplement sur ce dernier.

Figure 16 : Exemple d'édition du plugin Angular-xeditable



[http://ryanlanciaux.github.io/files/original\\_xeditable.png](http://ryanlanciaux.github.io/files/original_xeditable.png)

D'un point de vue technique, notre modèle affiche dans notre vue un objet. En cliquant dessus Angular-xeditable affiche un champ de saisie faisant référence à l'objet du modèle. Lorsque l'utilisateur saisit une nouvelle valeur et l'enregistre, du fait du fonctionnement de AngularJS, l'objet contenu dans le modèle est mis à jour en conséquence. Mise à jour qui va ensuite se répercuter sur la vue. Cette dernière va donc afficher la nouvelle valeur de l'objet. Cette technique propose donc une modification rapide ne nécessitant aucun rechargement de page.

Angular-xeditable propose également un ensemble de méthodes destinées à effectuer des contrôles de saisies avant ou après la modification d'un champ.

Dans le cadre de l'application SIRH Web, j'ai, dans un premier temps, utilisé une autre fonctionnalité également proposée par ce plugin : l'édition en ligne. Cette dernière repose sur le même principe de fonctionnement d'un point de vue technique.

Figure 17 : Exemple d'édition d'une ligne avec Angular-xeditable

Name	Status	Group	Edit
awesome user1	status2	admin	<input type="button" value="edit"/> <input type="button" value="del"/>
<input type="text" value="awesome user2"/>	<input type="text" value=""/>	<input type="text" value="vip"/>	<input type="button" value="save"/> <input type="button" value="cancel"/>
awesome user3	status2	<i>Not set</i>	<input type="button" value="edit"/> <input type="button" value="del"/>

Capture d'écran réalisée sur le site officiel : <http://vitalets.github.io/angular-xeditable>

Cette solution intuitive et ergonomique comporte quelques défauts :

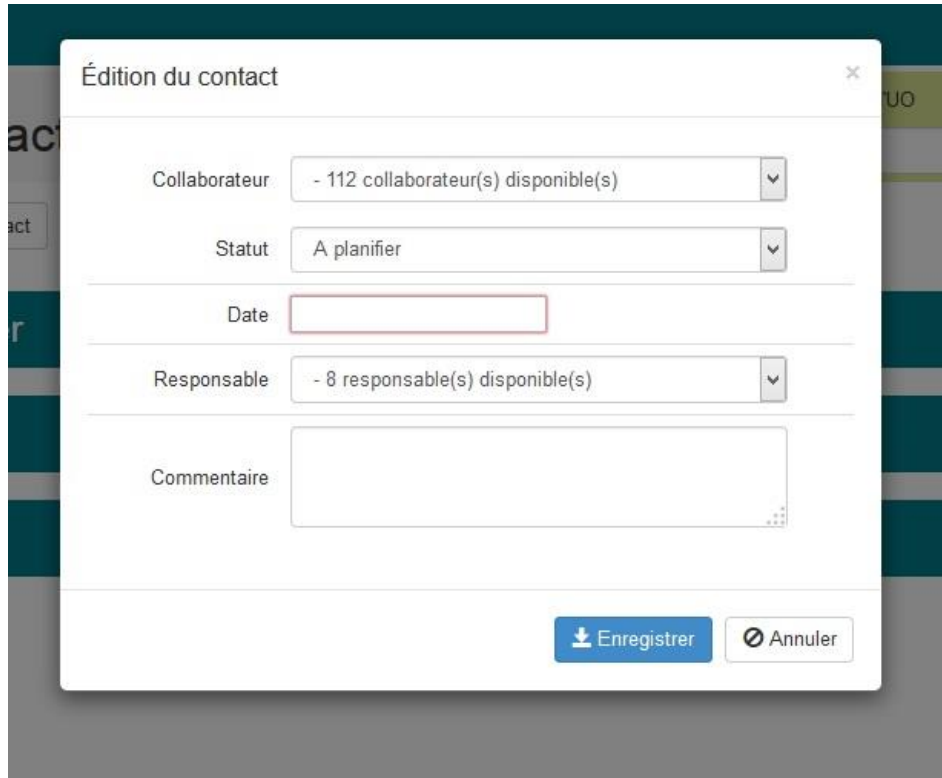
- Si l'objet à modifier comporte beaucoup de champs, l'affichage en ligne n'est pas idéal.
- La solution est assez lourde car intégrée à chaque ligne du tableau. Par conséquent si le tableau contient beaucoup d'objets. Les performances d'affichage sont impactées.
- L'ergonomie de ce genre de tableau sur les petits écrans est très discutable.
- L'obligation d'afficher un tableau pour l'édition d'objets.

Par conséquent, j'ai, suite à l'analyse de ces défauts, pris une autre direction afin d'implémenter l'édition au sein de l'application SIRH Web.

La solution que j'ai choisi d'implémenter est basée sur les fenêtres « modals » proposées par Bootstrap. Ces fenêtres qui s'affichent en surimpression de la page permettent d'afficher toutes sortes d'informations. SIRH Web fait appel à ces fenêtres dans deux cas d'utilisation.

Le premier est lorsque l'utilisateur désire ajouter un objet. La fenêtre modal s'ouvre et affiche un formulaire (proposé par le plugin Angular-xeditable) invitant l'utilisateur à renseigner les champs nécessaires.

Figure 18 : Ajout d'un objet à l'aide de la fenêtre modal



Édition du contact

Collaborateur - 112 collaborateur(s) disponible(s)

Statut A planifier

Date

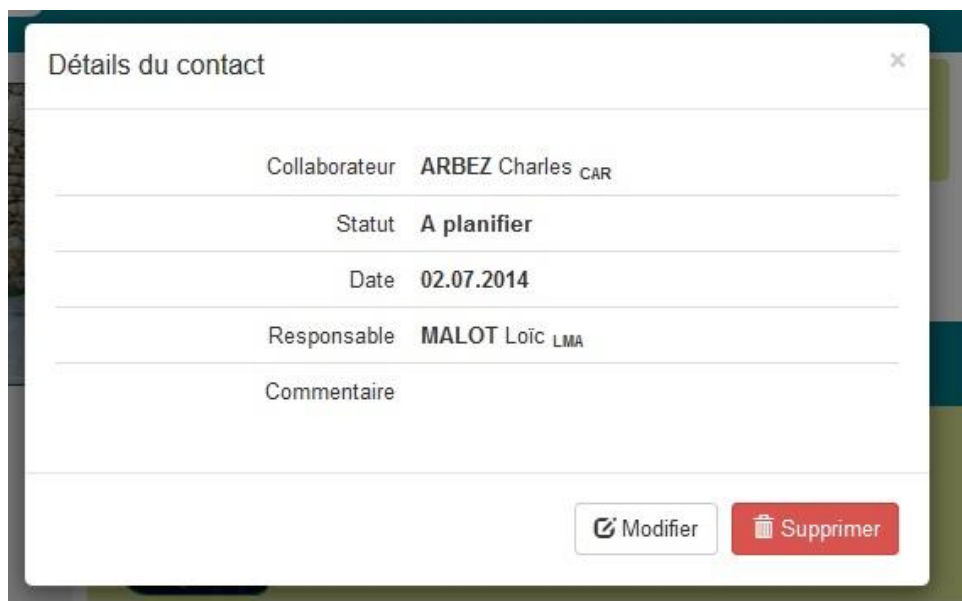
Responsable - 8 responsable(s) disponible(s)

Commentaire

Enregistrer Annuler

Le second cas d'utilisation est lorsque l'utilisateur clique sur un objet. La fenêtre modal s'ouvre et affiche les détails de l'objet.

Figure 19 : Affichage des détails d'un objet



Détails du contact

Collaborateur ARBEZ Charles CAR

Statut A planifier

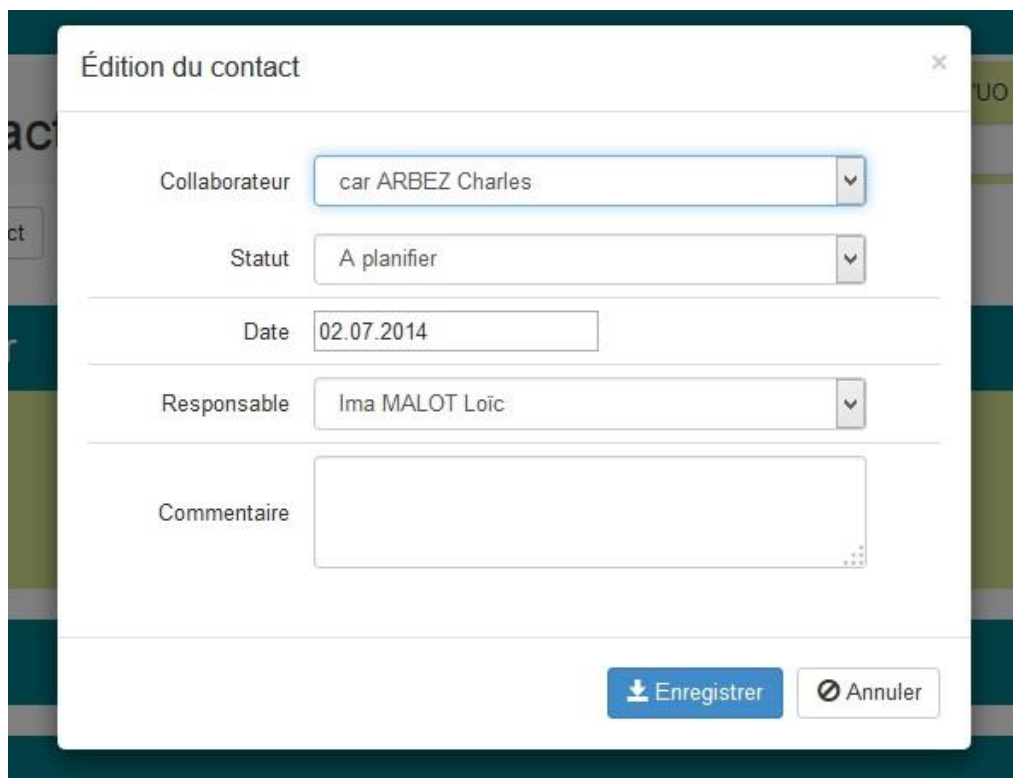
Date 02.07.2014

Responsable MALOT Loïc LMA

Commentaire

Modifier Supprimer

L'utilisateur peut à ce moment-là cliquer sur le bouton d'édition et dès lors, un formulaire contenant les données de l'objet s'affiche et permet la modification de ce dernier.



The image shows a modal window titled "Édition du contact" with a close button (X) in the top right corner. The form contains the following fields:

- Collaborateur:** A dropdown menu with the selected value "car ARBEZ Charles".
- Statut:** A dropdown menu with the selected value "A planifier".
- Date:** A text input field containing "02.07.2014".
- Responsable:** A dropdown menu with the selected value "Ima MALOT Loïc".
- Commentaire:** A large text area for entering notes.

At the bottom of the modal, there are two buttons: a blue "Enregistrer" button with a save icon and a white "Annuler" button with a cancel icon.

La fenêtre de type modal a le grand avantage par rapport à l'édition en ligne d'être parfaitement compatible et ergonomique sur les petits écrans de type smartphone. Cette solution m'a également permis de proposer directement l'édition d'objets depuis le mode vignettes sans nécessiter d'afficher les objets sous forme de liste.

## 4. Présentation de SIRH Web

Je vais maintenant vous présenter l'application SIRH Web dans sa version 1.0. Dans un premier temps, je vous exposerai l'interface utilisateur et les différentes interactions possibles avec cette interface. Dans un second temps, je vous présenterai une vue d'ensemble des fonctionnalités de SIRH Web.

## 4.1 Interface utilisateur

Figure 20 : Aperçu de l'interface de SIRH Web sur un écran d'ordinateur

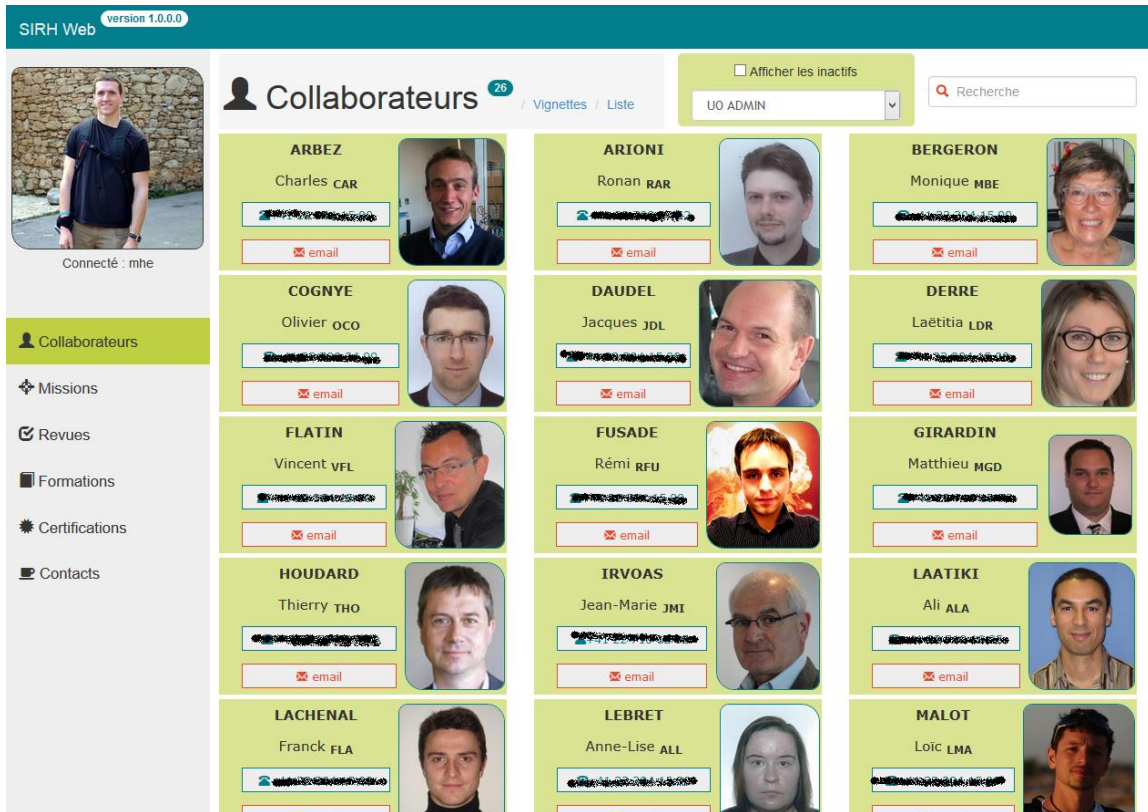
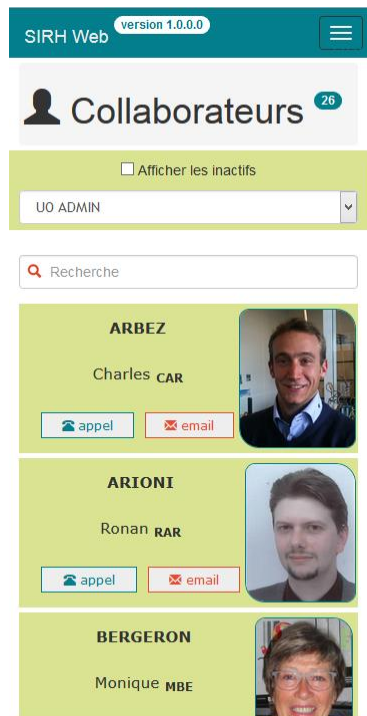


Figure 21 : Aperçu de SIRH Web sur un écran de smartphone



## 4.1.1 Interactions possibles

Figure 22 : Interaction de l'interface SIRH Web

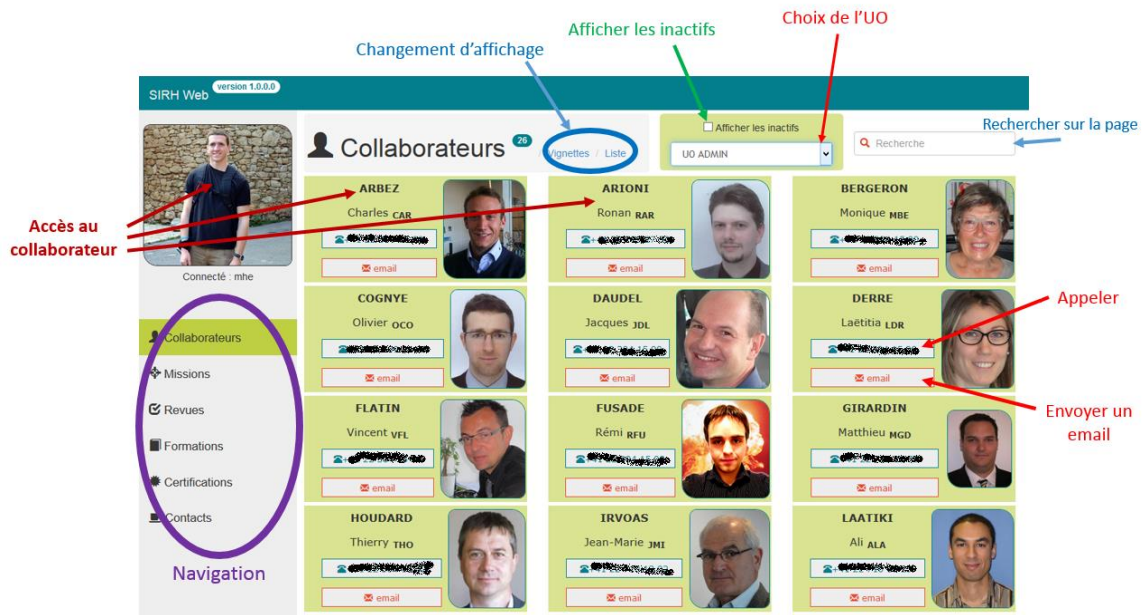
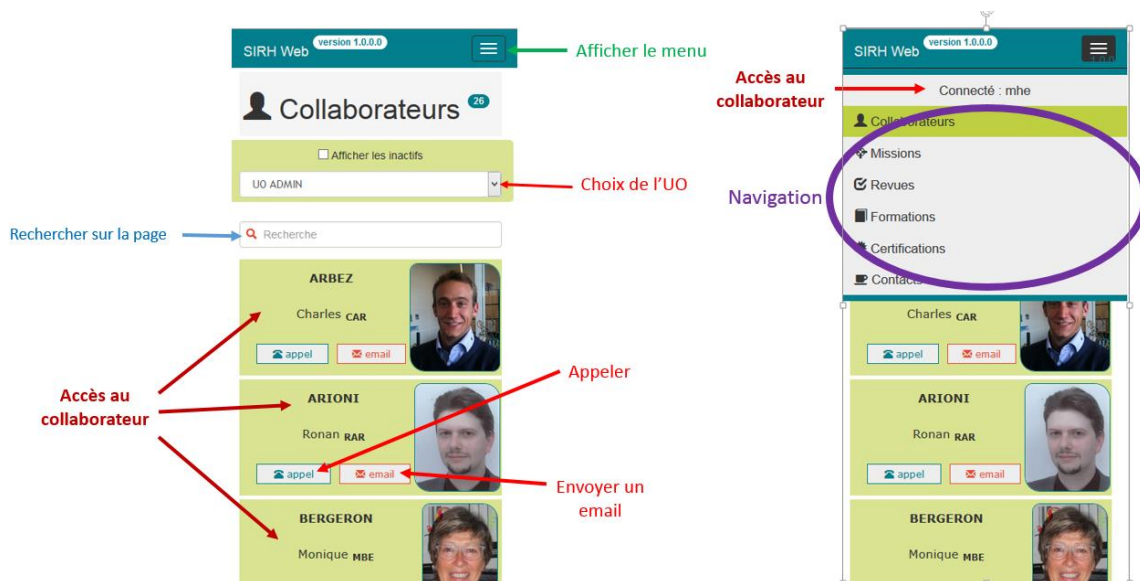


Figure 23 : Interaction de l'interface SIRH Web sur mobile



## 4.2 Fonctionnalités

### 4.2.1 Le filtrage par unité opérationnelle

Le filtrage par unité opérationnelle est un concept que vous allez retrouver dans chacune des pages ayant comme fonction principale le listing d'objets métiers (missions, collaborateurs, etc.). En effet, comme expliqué un peu plus tôt dans ce document, l'application SIRH Web est destinée principalement à des responsables d'unité opérationnelle qui, par conséquent, doivent pouvoir accéder rapidement aux

données de leur unité. La valeur de ce filtre persiste à l'aide d'un cookie enregistré sur le navigateur de l'utilisateur.

D'un point de vue technique, lorsque l'utilisateur change l'unité opérationnelle sélectionnée, le contrôleur AngularJS enregistre la valeur de la nouvelle unité opérationnelle, puis récupère les nouvelles données de l'unité opérationnelle à l'aide d'une requête AJAX.

L'interface graphique affiche un message de chargement si nécessaire, puis affiche la nouvelle liste d'éléments.

#### **4.2.2 La recherche syntaxique**

Tout comme le filtrage par unité opérationnelle, la recherche syntaxique est elle aussi présente sur l'ensemble des pages de type listing.

Cette recherche est disponible sur l'ensemble des champs des objets du listing.

D'un point de vue technique, le modèle AngularJS contient l'ensemble de nos objets. Lorsque l'utilisateur modifie le champ de recherche sur l'interface (grâce à la synchronisation que propose AngularJS entre le modèle et la vue), le modèle de recherche prend la nouvelle valeur introduite par l'utilisateur. A l'aide de cette nouvelle valeur, les objets du modèle sont filtrés. Une fois de plus grâce à la synchronisation modèle-vue, la vue est automatiquement mise à jour avec les objets filtrés du modèle.

Aux yeux de l'utilisateur, lorsqu'il modifie le champ de recherche, la liste des objets à l'écran se filtre afin d'afficher uniquement les objets contenant les caractères recherchés.

### 4.2.3 Les modes d'affichage

L'ensemble des pages du site internet proposant un listing d'objets dispose de deux modes d'affichage : le mode « vignette » et le mode « liste ».

Figure 24 : Aperçu du mode « vignettes »

Figure 24 shows the 'vignettes' (card) view of the 'Missions' page. The interface includes a sidebar with navigation options like 'Collaborateurs', 'Missions', 'Reuves', 'Formations', 'Certifications', and 'Contacts'. The main content area displays two mission cards. The first card, 'Agence - Activités internes', is associated with 'UO ADMIN' and has a value of 0 CHF. The second card, 'Test Mission UO', is associated with 'Lausanne 1' and has a value of 1,000 CHF. Both cards show dates, renewal status, and a project manager profile for 'MALOT Loïc LMA'.

Figure 25 : Aperçu du mode « liste »

Figure 25 shows the 'liste' (table) view of the 'Missions' page. The interface is similar to Figure 24, but the main content area displays a table with the following data:

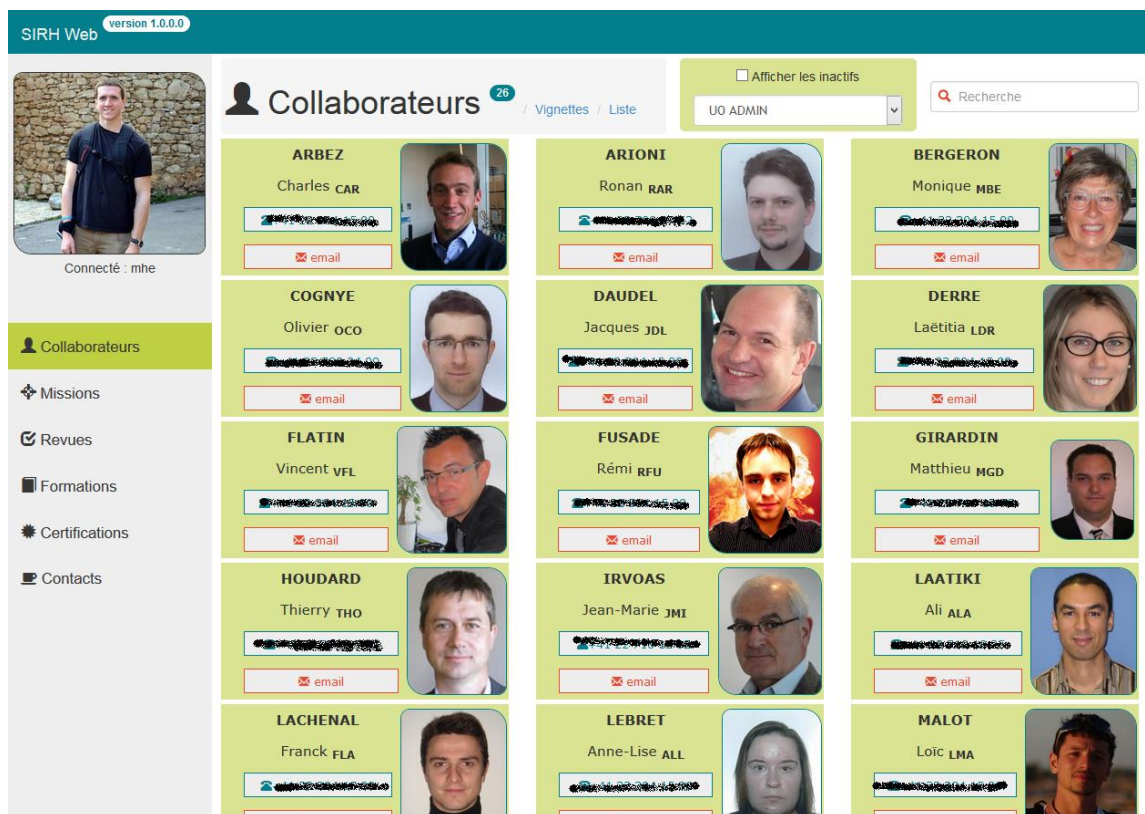
Titre	Fin	Renouv.	Client	Collaborateur	Montant
Agence - Activités internes	non définie	non définie	Qim info SA	MALOT Loïc LMA	0 CHF
Test Mission UO	non définie	non définie	[redacted]	MALOT Loïc LMA	1,000 CHF

L'affichage « liste » propose un plus grand nombre d'objets à l'écran et apporte également la possibilité de trier la liste selon chaque champ. Cependant j'ai fait le choix de masquer le mode liste sur les petits écrans qui, du fait de leur plus petite résolution, ne permettent pas d'afficher un nombre suffisant d'attributs à l'écran tout en proposant une ergonomie convaincante.



## 4.2.4 Page des collaborateurs

Figure 26 : Aperçu de la page des collaborateurs



La page des collaborateurs permet d'afficher un listing de l'ensemble des collaborateurs répondant au filtre. Il est également possible d'afficher les collaborateurs inactifs. Ces derniers sont par défaut masqués.

En cliquant sur la vignette ou la ligne (mode liste) d'un collaborateur, on accède à sa page de détails. Il est également possible d'appeler directement le collaborateur en cliquant sur le bouton contenant son numéro de téléphone ou alors, de lui envoyer un email à l'aide du bouton « email ».

## 4.2.5 Page de détail d'un collaborateur

Figure 27 : Aperçu de la page de détail d'un collaborateur

The screenshot displays the SIRH Web interface (version 1.0.0.0) for a collaborator's details page. The page is titled "Détail du collaborateur" and includes a navigation menu on the left with options: Collaborateurs, Missions, Revues, Formations, Certifications, and Contacts. The main content area shows the collaborator's profile for MALOT Loïc LMA, with contact information (loic.malot@giminfo.ch) and a list of expandable sections: Missions (32), Revues (5), Formations (3), Certifications (1), and Contacts (1). The user is logged in as "mhe".

La page de détails du collaborateur, comme son nom l'indique, affiche l'ensemble des détails du collaborateur sous forme de liste déroulante.

On retrouve donc sur cette page l'ensemble des revues, formations, certifications et contacts du collaborateur. On peut également, directement depuis la page du collaborateur, modifier, ajouter et supprimer la plupart de ces éléments. Seules les missions ne sont pas modifiables car ce n'est pas le rôle de l'application SIRH.

## 4.2.6 Page des missions

Figure 28 : Aperçu de la page des missions

The screenshot displays the SIRH Web interface for mission management. At the top, the version is 1.0.0.0. The main header shows 'Missions' with 493 items, a search bar containing 'malot', and a filter for 'Toutes les UOs'. A sidebar on the left lists navigation options: Collaborateurs, Missions (selected), Revues, Formations, Certifications, and Contacts. The main content area is divided into 'Forfaits' (collapsed) and 'Régies' (expanded). Under 'Régies', two mission cards are visible:

Mission	UO	Montant	Statut	Renouvellement	Info	Projet	
Agence - Activités internes	UO ADMIN	0 CHF	01.10.2012	non définie	Pas de renouvellement	Qim info SA	MALOT (Loïc, LMA, Chef de projet)
Test Mission UO	Lausanne 1	1,000 CHF	13.01.2014	non définie	Pas de renouvellement	CHUV	MALOT (Loïc, LMA, Chef de projet)

La page des missions affiche l'ensemble des missions répondant aux critères de filtrage. Comme vous pouvez le voir ci-dessus, elle distingue les missions forfaits des missions régies. Les missions forfaits étant moins amenées à être consultées que les missions régies, elles sont par défaut repliées.

## 4.2.7 Page de détail d'une mission

Figure 29 : Aperçu de la page du détail d'une mission

SIRH Web version 1.0.0.0

Connecté : mhe

Collaborateurs

Missions

Revues

Formations

Certifications

Contacts

### Détails de la mission / Retour aux missions

## Test Mission UO

Lausanne 1 régie

#### Dates & Budget

Date de début : 13.01.2014

Date de fin : non-définie

Date de renouvellement : non-définie

Budget : 1,000 CHF

#### Collaborateur

**MALOT**  
Loïc LMA  
Chef de projet

#### Compagnie

[Redacted]

#### Client

[Redacted]

#### Tâches liées <sup>1</sup>

#### Revues <sup>1</sup>

+ Ajouter une revue

Description	Collaborateur	Statut	Date	Auditeur
Test Mission UO	MALOT Loïc LMA	A planifier	03.03.2014	LACHENAL Franck FLA

La page de détails d'une mission affiche l'ensemble des informations associées à une mission. On retrouve donc dans un premier temps les différentes dates ainsi que le budget.

Sur la droite se trouve(nt) le ou les collaborateurs participant à cette mission. Les informations concernant la compagnie et le client sont également affichées. Un lien Google Maps pointant sur l'adresse de la compagnie est également présent.

On retrouve ensuite la liste des tâches qui composent la mission. Dans le cadre d'une mission régie, une seule et unique tâche est présente. Mais dans le cadre d'une mission forfaits, nous avons souvent affaire à plusieurs tâches.

Pour finir, cette page affiche les revues qui concernent cette mission. Il est également possible d'ajouter, de modifier ou bien de supprimer une revue pour cette mission directement depuis cette page.

## 4.2.8 Page des revues

Figure 30 : Aperçu de la page des revues

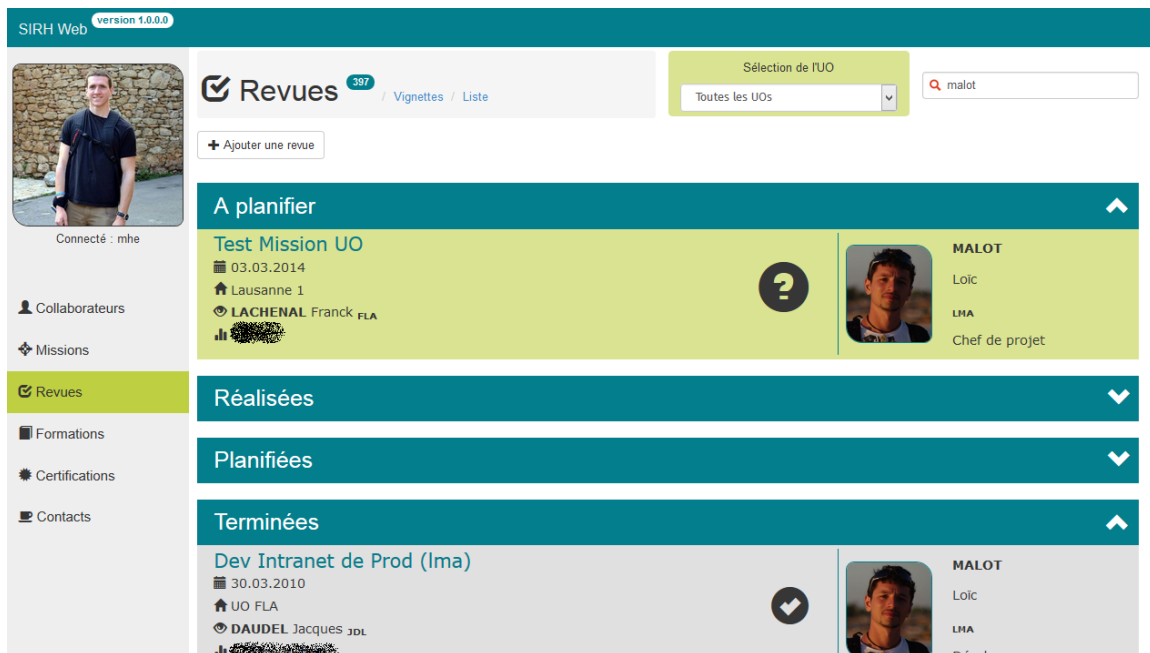
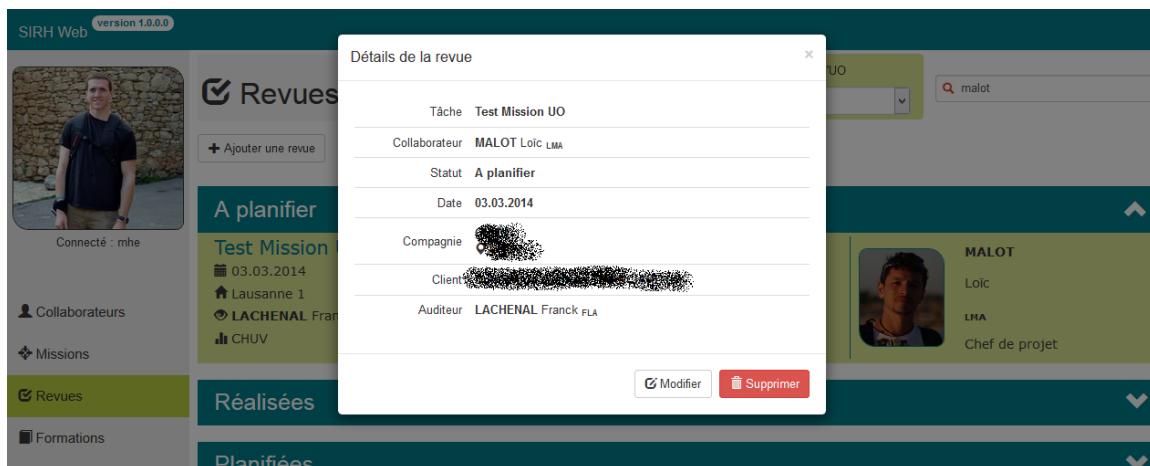


Figure 31 : Aperçu du détail de la revue



La page des revues affiche l'ensemble des revues répondant aux critères de filtre. Ces revues sont triées par statut et disposées sous forme de menu déroulant.

En cliquant sur l'une des revues, une fenêtre modale s'ouvre. Cette fenêtre contient l'ensemble des détails de la revue. Cette fenêtre nous permet également de modifier ou supprimer la revue. En cliquant sur modifier, les champs se transforment en formulaire et nous invitent à modifier les éléments désirés.

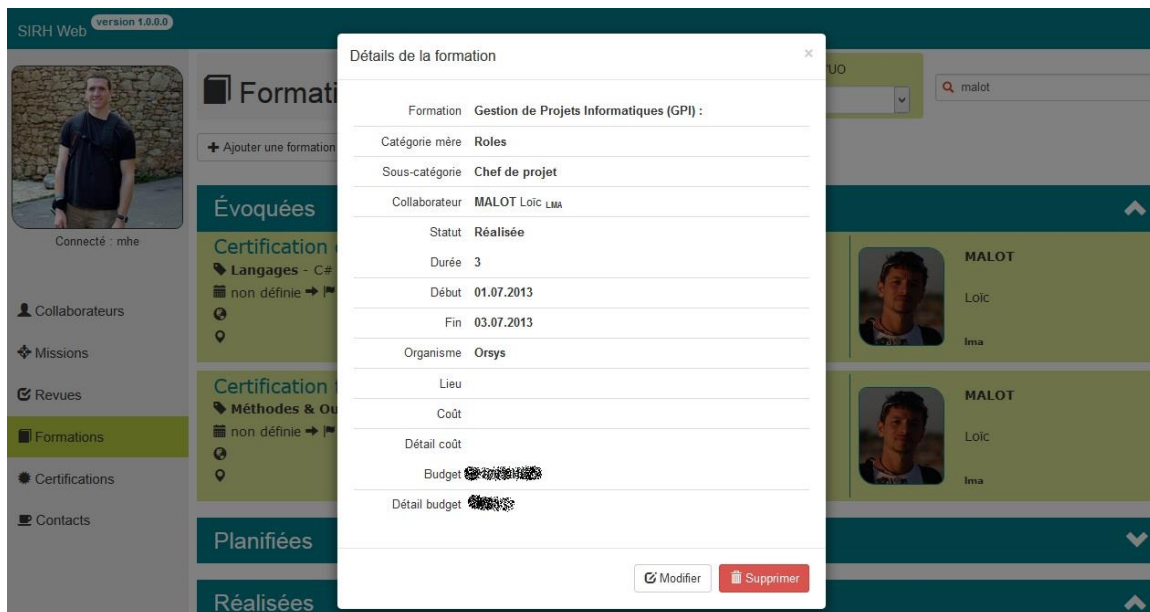
L'ajout de la revue se fait à l'aide du bouton situé en haut de la page : « ajouter une revue ».

## 4.2.9 Page des formations

Figure 32 : Aperçu de la page formations



Figure 33 : Aperçu du détail de la formation



Cette page comporte exactement les mêmes fonctionnalités que la page des revues, excepté que l'ensemble des opérations se passe sur des formations.

## 4.2.10 Page des certifications

Figure 34 : Aperçu de la page des certifications

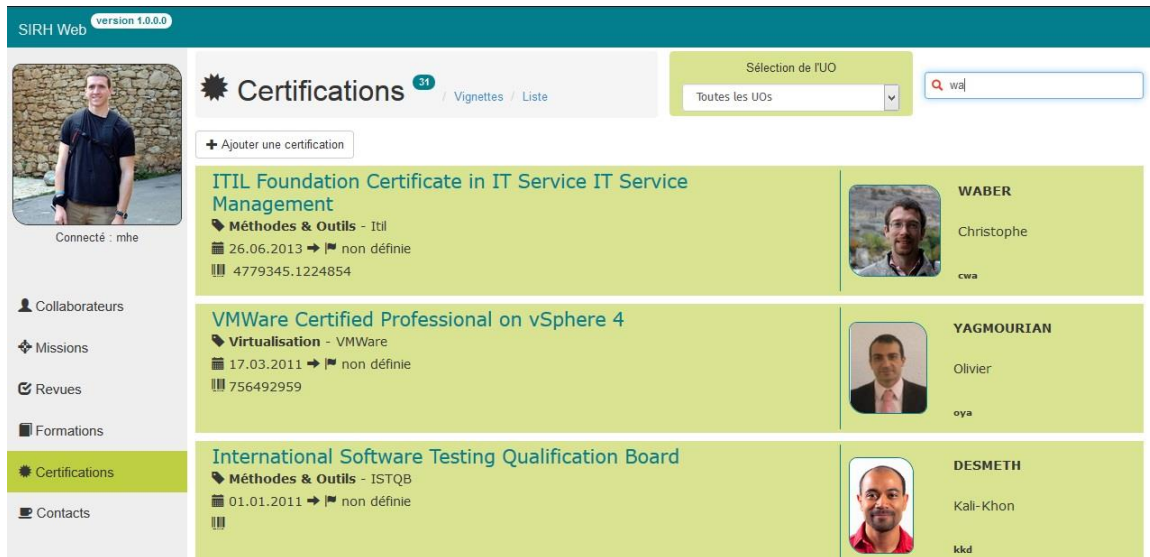
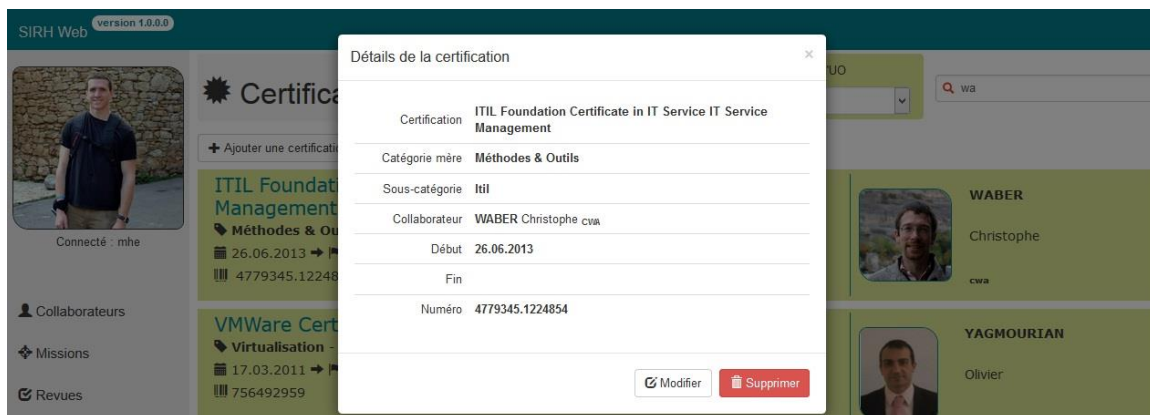


Figure 35 : Aperçu du détail d'une certification



Cette page dispose également des mêmes fonctionnalités que la page des revues. La seule différence ici est que les certifications ne disposent pas de statut. Par conséquent, le tri par statut n'est pas présent sur cette page.

## 4.2.11 Page des contacts

Figure 36 : Aperçu de la page des contacts

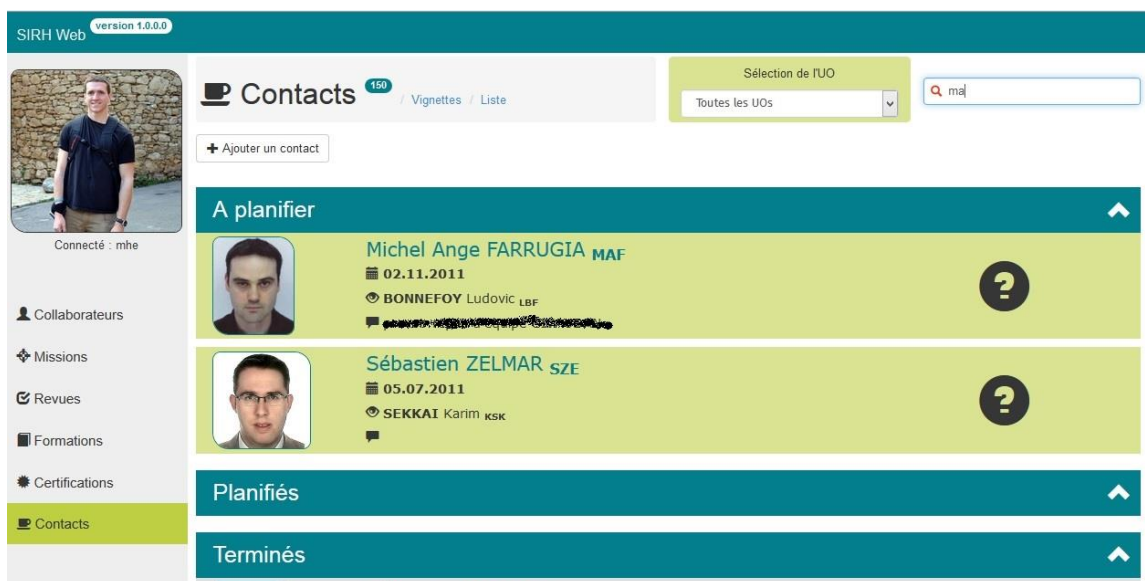
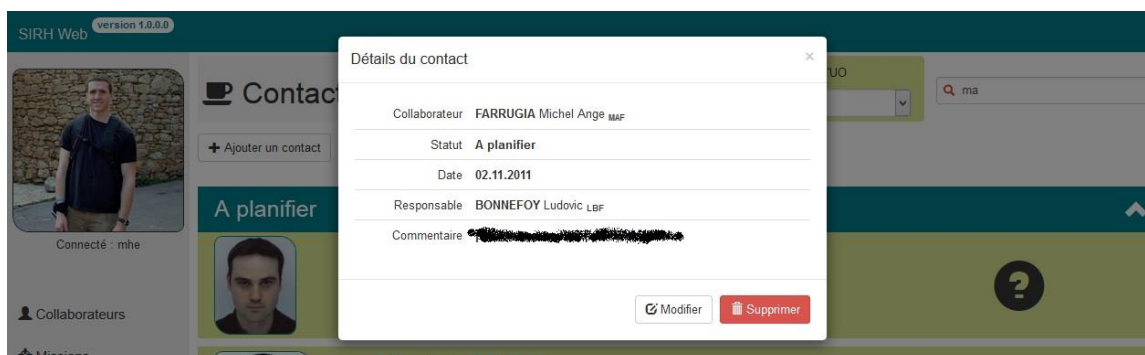


Figure 37 : Aperçu du détail d'un contact



Cette page fonctionne exactement de la même manière que la page des revues, la seule différence ici est que l'objet traité est un contact.

## 4.3 État de développement

L'application est passée en version 1.0 le 14 juillet 2014. Cela signifie que cette dernière est prête à passer en production et prête à être utilisée par les responsables des unités opérationnelles. Une fois en production, elle remplacera l'axe « collaborateurs » de SIRH V4.0 et travaillera par conséquent en parallèle de l'axe « candidats » de cette dernière.

A la suite de ce travail, la société Qim info continuera le développement sur SIRH Web afin d'y inclure l'axe « candidats ». Cela permettra à court terme, de se passer totalement de l'application SIRH 4.0.



L'application, à moyen terme, pourra également profiter du gestionnaire centralisé d'habilitation QimAuthz. Ce dernier, actuellement en cours de développement au sein de la société, utilisera SIRH Web comme première plateforme de test.

A plus long terme, l'application SIRH Web pourrait également, suivant l'évolution de l'infrastructure chez Qim info, se voir doter de certaines tâches qui, aujourd'hui, sont réalisées par l'intranet.

## 5. Conclusion

A l'heure où le nombre d'appareils connectés atteint son point le plus haut jamais enregistré et ne cesse encore de croître. Il devient nécessaire de se préoccuper de la compatibilité des applications développées avec l'ensemble de ces nouveaux systèmes. Le coût de développement de multiples versions du même produits ainsi que la difficulté de maintenance de ces solutions à vite fait de décourager les développeurs, causant une segmentation du marché de l'application. Cette segmentation se fait au détriment de l'utilisateur qui veut pouvoir, en tout temps, et ce depuis n'importe quel appareil, pouvoir avoir accès à son application et l'utiliser de manière optimale.

Aujourd'hui le client riche propose une solution concrète à ces problèmes. En effet, le temps où le client riche proposait une compatibilité limitée sur les systèmes nomades est sur le point d'être révolu. En effet, comme nous avons pu le voir tout au long de ce travail, nous disposons aujourd'hui d'outils capables d'apporter de la richesse à nos applications sans nécessiter une quelconque installation.

L'application que j'ai créée reprend l'ensemble des fonctionnalités de l'application SIRH v4.0. Elle permet d'afficher un annuaire des collaborateurs ainsi que des missions. Elle permet également une gestion complète des revues, formations, certifications et contacts. Les contraintes liées à la caractéristique « multiplateforme » de l'application m'ont fait imaginer des solutions ergonomiques adaptées à chaque situation tout en gardant les fonctionnalités de l'application intactes et ce : peu importe la plateforme. La notion de client riche m'a également permis d'explorer les différentes solutions actuellement disponibles et de choisir la plus adaptée.

Les avantages qu'apporte mon application sont nombreux. Tout d'abord, son architecture de client riche multiplateforme lui permet de proposer une expérience d'utilisation identique sur l'ensemble des plateformes de la société. Cette architecture simplifie également les futures maintenances de l'application. La recherche effectuée sur le design et l'ergonomie aboutissent à une navigation facilitée. Pour finir cette

application constitue une base solide de développement pour les futurs modules de la société Qim info.

Cette expérience m'aura énormément apporté sur le plan personnel. Bien que déjà très intéressé par le sujet, j'ai eu l'occasion de mettre en œuvre des technologies qui jusque-là m'étaient inconnues. En combinant des technologies récentes et en imaginant des solutions peu expérimentées jusqu'ici, j'ai pu non seulement concevoir une application répondant aux objectifs fixés mais également une application innovante de par ses composants.

J'ai également eu l'occasion pour la première fois de travailler en situation réelle, et je retire beaucoup de bénéfice de cette expérience. J'ai côtoyé des professionnels du métiers et pas uniquement dans le domaine du développement. J'ai également pu profiter, observer et prendre part à des processus de gestion de projet grandeur nature.

Je conclurai ce travail en exprimant ma grande satisfaction d'avoir eu l'occasion de développer une application répondant aux attentes de la société et qui d'ici quelques jours sera employée quotidiennement. Ce fût une expérience très enrichissante et j'attends avec impatience les premiers retours de la part des utilisateurs.

# Bibliographie

## Articles en ligne

PUSSINEAU, Jérôme, 2013, Les avantages et inconvénients du client riche. Le blog des experts IT [en ligne]. 1<sup>er</sup> janvier 2013 [Consulté le 1<sup>er</sup> juillet 2014]. Disponible à l'adresse : <https://blog.axopen.com/2013/01/les-avantages-et-inconvenients-du-client-riche/>

BOEAVENTURA, Raoni, 2014, A Step-by-Step Guide to your first AngularJS app [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://www.toptal.com/angular-js/a-step-by-step-guide-to-your-first-angularjs-app>

DBM web design, 2014, Réaliser une interface avec Bootstrap de Twitter [en ligne] [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://dbmwebdesign.fr/Tutos/rwd/tuto-bootstrap.html>

GOETTER, Raphaël, 2013, C'est quoi le responsive design [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://www.alsacreations.com/article/lire/1615-cest-quoi-le-responsive-web-design.html>

## Livres

GREEN, Bras, 2014, AngularJS. O'Reilly, 2014. ISBN 978-1-449-34485-6

## Blogs

AngularJS pour les nuls, 2014, AngularJS pour les nuls [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://angularpourlesnuls.blogspot.ch/>

## Sites web

WIKIPEDIA, 2014, AngularJS [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://en.wikipedia.org/wiki/AngularJS>

WIKIPEDIA, 2014, ASP.NET MVC Framework [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : [http://en.wikipedia.org/wiki/ASP.NET\\_MVC\\_Framework](http://en.wikipedia.org/wiki/ASP.NET_MVC_Framework)

WIKIPEDIA, 2014, Rich Internet Application [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : [http://en.wikipedia.org/wiki/Rich\\_Internet\\_application](http://en.wikipedia.org/wiki/Rich_Internet_application)

BOOTSTRAP, 2014, Bootstrap [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://getbootstrap.com/>

ANGULAR-XEDITABLE, 2014, Angular-xeditable [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://vitalets.github.io/angular-xeditable/>

MICROSOFT, 2014, ASP.NET MVC Overview [en ligne]. [Consulté le 1<sup>er</sup> juillet 2014] Disponible à l'adresse : <http://msdn.microsoft.com/en-us/library/dd381412%28v=vs.108%29.aspx>