HAL
open science

Solution d'interface ergonomique pour la relance client

Ghislain Orekan

To cite this version:

Ghislain Orekan. Solution d'interface ergonomique pour la relance client. Informatique [cs]. 2012. dumas-01160198

HAL Id: dumas-01160198

https://dumas.ccsd.cnrs.fr/dumas-01160198v1

Submitted on 4 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CONSERVATOIRE NATIONAL DES ARTS ET MÉTIERS

CENTRE RÉGIONAL D'AIX EN PROVENCE

MÉMOIRE

présenté en vue d'obtenir le

DIPLÔME D'INGÉNIEUR CNAM

Spécialité : Informatique

Option : Systèmes d'information

par

Ghislain OREKAN

Solution d'interface ergonomique pour la relance client

Soutenu le 11 Juillet 2012

JURY

PRÉSIDENT : Yves LALOUM (Professeur CNAM)

MEMBRES : Noel QUESSADA (Professeur CNAM)

Bastien PESCE (Professeur CNAM)

Alain MAERTEN (Responsable études, Digital Virgo Entertainment)

Sylvain CASAREGOLA (Développeur NTIC, Digital Virgo Entertainment)

le cnam Provence-Alpes-Côte d'Azur

Région PACA

Remerciements

Tout d'abord, je remercie Digital Virgo Entertainment, entreprise pour laquelle je travaille depuis plusieurs années, de m'avoir donné la possibilité de réaliser mon mémoire. Cela a été possible grâce à Ludovic DEMARE, responsable de la direction informatique, ce dernier m'a nommé chef de projet pour la mission support à mon mémoire.

Je suis reconnaissant du travail effectué par les développeurs qui ont participé à la réalisation du projet. Je pense notamment à Sylvain CASAREGOLA et Renaud BEAUFILS.

Je tiens à remercier mon tuteur de mémoire, Bastien PESCE, pour s'être porté garant de mon projet jusqu'à la soutenance.

Enfin, je remercie le CNAM et l'ensemble des professeurs. Je suis persuadé que l'obtention du diplôme d'ingénieur CNAM sera un élément déterminant pour la suite de ma carrière professionnelle.

Sommaire

Remerciements 2
1) Introduction 6
2) L'entreprise 7
2.1) Le groupe Digital Virgo 7
2.2) Les métiers de Digital Virgo Entertainment 10
2.3) Les concurrents de Digital Virgo Entertainment 11
2.4) Mes fonctions 12
2.5) Le développement d'applications chez Digital Virgo Entertainment 14
2.5.1) Les différents acteurs 14
2.5.2) L'infrastructure technique 17
2.5.3) L'environnement technologique 18
2.6) Nos différents moyens de paiement 19
2.6.1) Présentation 19
2.6.2) Profils clients 21
2.6.3) Evolutions 21
2.7) Vers une internalisation complète de notre système d'information 22
3) Présentation de l'application IRC 23
3.1) Le besoin 23
3.2) Architecture technique 24
3.3) Périmètre fonctionnel 25
3.4) Analyse des faiblesses du système existant 26
3.5) Analyse des besoins 26
3.6) Objectif et fonctionnalités 27
3.7) Méthode de conception 31
3.7.1) Définition de SCRUM 31
3.7.2) SCRUM en pratique 33
3.8) Backlogs de produits et sprints 34
3.9) Traitement des livrables 37
3.10) Contraintes liées à l'envoi de relances 38
3.11) Relance via alias 39
3.12) Relance via MSISDN 42

4) Développement de l'application 43
4.1) Mes missions 43
4.1.1) Développement 44
4.1.2) Conception de la base de données 45
4.2) Méthodes de travail 48
4.2.1) Développement web 48
4.2.2) Présentation du Framework 48
4.3) jQuery 50
4.3.1) jQuery le futur du JavaScript? 50
4.3.2) Réalisation d'un plugin jQuery 50
4.4) Une interface ergonomique 56
4.4.1) Les widgets 56
4.4.2) Sécurisation de l'interface 59
4.4.3) Optimisation de l'interface 60
4.5) Talend Open Studio 61
4.5.1) Présentation de Talend Open Studio 61
4.5.2) Utilisation de Talend Open Studio dans IRC 62
4.6) Le cloud computing 64
4.6.1) Présentation 64
4.6.2) IRC, une application de type SaaS? 65
4.6.3) Evolution du SaaS dans le monde de l'entreprise 65
4.7) Utilité d'un développement HTML5 66
4.7.1) Présentation 66
4.7.2) Apport du HTML5 dans IRC 66
4.8) Réalisation de statistiques sous QlickView 68
4.9) Intégration à API_MOBILE 69
4.9.1) Présentation de l'application API_MOBILE 69
4.9.2) Problèmes posés lors de l'intégration 71
4.9.3) Phase de test 71
5) Résultats et évolutions 72
5.1) Résultats 72
5.2) Futur de l'application IRC 75
5.2.1) Evolution du produit 75

5.2.2) Déploiement à d'autres clients 75
6) Conclusion 76
7) Bibliographie 78
7.1) Sites internet 78
7.2) Livres et documents 78
8) Glossaire 79
9) Table des figures 82
10) Liste des tableaux 84

1) Introduction

Ce mémoire est l'élément final de ma formation au CNAM d'Aix en Provence. Mon objectif est d'obtenir un diplôme d'ingénieur CNAM. Afin de compléter mon diplôme de chef de projet en système d'information, de niveau II, obtenu à l'ECML (Ecole de Conception et de Maintenance de Logiciels) de Marseille.

Aussi, je vais m'appuyer sur la réalisation d'un projet réalisé au sein de mon entreprise, Digital Virgo Entertainment. L'objectif de cette mission consiste à réaliser une application disposant d'une interface ergonomique pour des équipes commerciales en charge d'envoyer des messages publicitaires ciblés à nos clients. Cette dernière a débuté en septembre 2011 et s'est achevée en avril 2012. Le nom choisi pour ce projet est IRC (Interface de Relances Clients).

Digital Virgo Entertainment est une entreprise spécialisée dans le secteur du divertissement numérique. Un élément crucial concernant le développement de nos services, est l'envoi de messages publicitaires à nos clients. L'entreprise utilisait jusqu'à présent une application développée par un de nos prestataires, la société Motricity, pour la gestion de ces relances. Suite aux difficultés rencontrées par cette dernière et dans un souci d'internalisation des applicatifs majeurs, il est nécessaire de réaliser une nouvelle application prenant le relais de celle-ci en y ajoutant des fonctionnalités.

Le mémoire ayant pour but de démontrer mes capacités à être ingénieur, le choix de ce projet comporte plusieurs avantages. Désigné chef de projet sur cette mission, j'ai réalisé plusieurs tâches dont :

- analyser les besoins fonctionnels,
- faire des choix de conception et d'optimisation,
- assurer la gestion et la planification d'une équipe.
- développer certains modules de l'application.

Ce document présente les différentes étapes de la création, à la mise en place de cette application dans le système d'information de l'entreprise. Nous verrons donc, la méthode de gestion de projet utilisé, les différents choix technologiques et conceptuels pris lors de la réalisation ainsi que leur répercussion sur le déroulement du projet. Puis, nous analyserons les résultats de l'application après sa mise en place.

2) L'entreprise

2.1) Le groupe Digital Virgo

Créée en décembre 2008, Digital Virgo est présent sur trois continents et dans neuf pays avec des entreprises leaders offrant des services grand public et professionnels et des catalogues complets de contenus.

Historique:

2012 : Création de Digital Virgo Entertainment par la fusion de CellCast Media et de MediaPlazza. Acquisition de ParuVendu.fr et de CarriereOnline.com, sites internet spécialisés dans les petites annonces et les offres d'emplois.

2011 : Acquisition de CellCast Media, n°1 du divertissement numérique (en juillet). Lancement de Virgopass, solution de paiement de Digital Virgo.

2010 : Développement de Digital Virgo sous la marque Jet Multimédia en Afrique de l'Ouest : Côte d'Ivoire, Togo, Bénin et Sénégal.

2009 : Renforcement des activités en Europe Centrale avec l'acquisition de Blue Orange en Slovaquie

2008 : Création de Digital Virgo

Les principales filiales :

- Digital Virgo Entertainment:
  C'est la structure assurant le développement et l'exploitation de différents services de divertissement numérique. Cette entreprise est située en trois secteurs géographiques : Marseille, Paris et Aix en Provence.

- Digital Virgo Facilities:
  Cette filiale gère les besoins logistiques de toutes les filiales du groupe. Elle est située à Lyon.

- Sedeco:
  Le principal rôle de cette filiale est d'assurer le service après vente et les réclamations de nos clients. Cette filiale est située à l'Ile Maurice.

Chiffre d'affaires :

Le chiffre d'affaires du groupe Digital Virgo est en constante progression. Ci-dessous un schéma retraçant son évolution sur les trois dernières années :

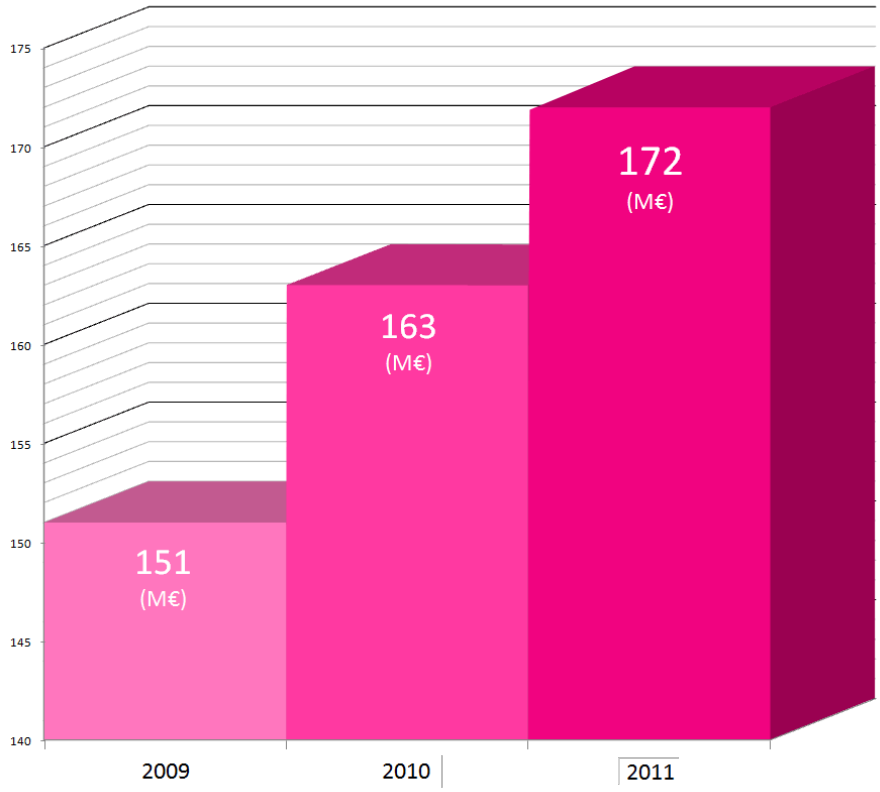

Figure 1 - Chiffre d'affaires du groupe Digital Virgo

En trois ans l'effectif du groupe à presque doublé, pour évoluer de 390 à 750 employés.

Présence géographique :

Par l'intermédiaire de ses nombreuses filiales, Digital Virgo est implanté dans plusieurs pays. Voici un schéma représentant notre présence géographique à l'échelle mondiale :

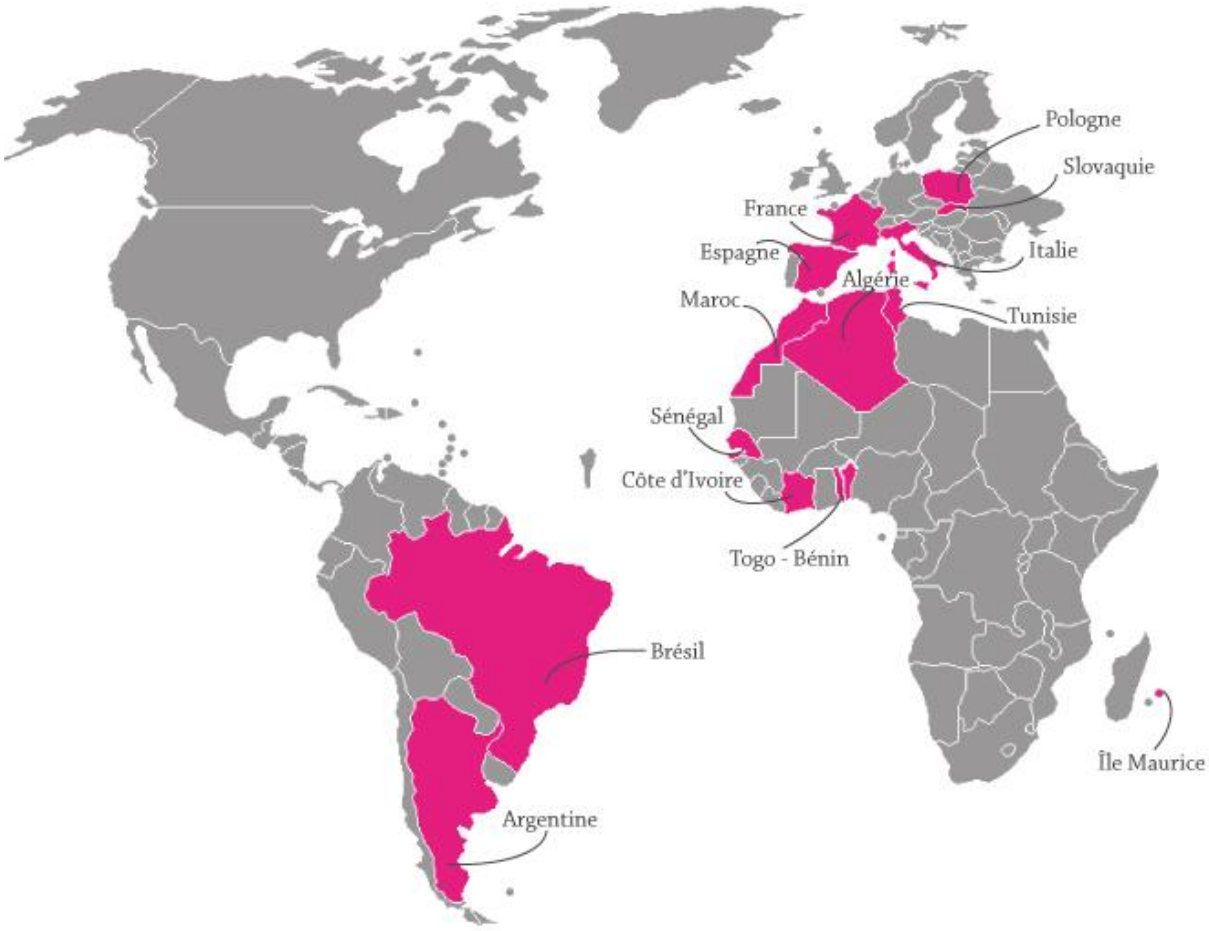

Figure 2 - Présence géographique du groupe Digital Virgo

D'un point de vue personnel, le fait d'avoir intégré le groupe Digital Virgo est positif. En effet, une meilleure progression de carrière est envisageable. Grâce à la multitude de filiale composant le groupe, nous sommes moins vulnérables aux aléas inhérents à notre secteur d'activité. Par exemple, si un opérateur modifie ses règles déontologiques ou de gestion d'abonnements, nos filiales hors France ne seront pas impactées. Nous pouvons d'ailleurs constater que le chiffre d'affaire est assez bien réparti entre la France, l'Espagne et l'Europe Centrale.

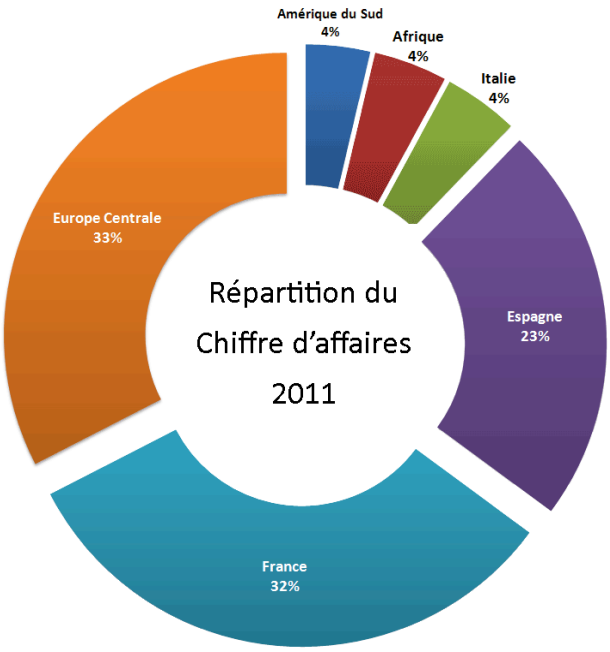

Figure 3 - Répartition du chiffre d'affaires de Digital Virgo

2.2) Les métiers de Digital Virgo Entertainment

Digital Virgo Entertainment est une société spécialisée dans la livraison de divertissements numériques (logos, sonneries, jeux ou films à la demande) pour téléphones mobiles ou ordinateurs. Les contenus sont fournis par des entreprises partenaires. Nous devons ensuite intégrer ces contenus dans notre système d'information afin de les rendre disponibles sur nos différents services (sites internet). Pour les contenus musicaux nous travaillons avec « Sony » et « Universal ». « Boonty » est notre partenaire pour les jeux vidéo alors que « Virgin » assure la livraison de films à la demande.

Nous réalisons de nombreuses applications audiotel pour des clients tels que « Le Parisien », « l'Équipe » ou « TV mag ». Aussi, nous réalisons des sites internet mobiles afin de proposer des contenus et monétiser nos mobinautes (contraction de « mobile » et d' « internaute ». Ce terme désigne tout internaute utilisant un téléphone mobile pour naviguer sur internet). Cependant l'entreprise élargit ses secteurs d'activités. Nous avons récemment développé un réseau social nommé « Woozgo ».

Pour gérer ces services l'entreprise est segmentée en plusieurs sections. Je travail au sein de la division technique. Notre premier rôle est de réaliser les sites internet proposant des services à nos clients. Nous devons aussi mettre en place différentes solutions de micropaiement. C'est notre première source de revenus. Les différentes solutions que nous proposons sont paypal, l'internet+ box, l'internet+ mobile et le SMS+. Ces deux derniers moyens de paiement sont fortement liés à l'application que nous devons réaliser dans le cadre de ce mémoire. Avec l'évolution technologique des terminaux numériques, l'émergence des smartphones, l'internet+ box ou mobile représente le moyen de paiement le plus intéressant. Aujourd'hui le trafic de

l'internet mobile est 10 fois supérieur à celui de l'internet fixe en l'an 2000. Cela présage une forte évolution de ce dernier.

La promotion du service est un élément déterminant quand à la réussite de ce dernier. Ainsi, la division publicité de la société investit chaque mois des centaines de milliers d'euros, afin de nous apporter une audience suffisante et par conséquent un nombre de transformation important. Nous sommes actuellement le premier client Google France en termes de publicité.

Nous utilisons aussi de l'affiliation avec des partenaires tels qu'Advertstream ou Erorezo. Ces derniers nous apportent une source importante de nouveaux clients. De plus, un gros travail d'optimisation du référencement naturel (SEO) est effectué afin que nos sites soient bien classés par les différents moteurs de recherche. Pour cela, plusieurs journalistes travaillant à l'île Maurice, sont chargés de rédiger des articles pour nos différents sites. Avec l'accumulation de tous ces efforts, plusieurs de nos sites internet comptent plus d'un millions de visiteurs par jour. Nos sites sont de plus classés aux alentours de la centième place au classement réalisé par Alexa Rank (Alexa Internet est une entreprise appartenant au groupe Amazon, principalement connu pour fournir des statistiques sur le trafic web). Cela nous situe par exemple au même niveau que des sites comme « France2.fr » ou « Sports.fr ». « Megavod.fr », « Jeu-a-telecharger.com » et « ParuVendu.fr » font partis de nos sites les plus visités.

Pour mesurer l'efficacité de nos campagnes publicitaires menées sur nos différents services, il est nécessaire d'avoir des indicateurs précis. Nous disposons donc d'une application récapitulant le nombre de lead (un lead est un prospect en commerce électronique. Il est au minimum composé d'un email ou d'un numéro de téléphone mobile), de ventes et donc le retour sur investissement de nos différentes campagnes.

Une fois que le client est recruté, il est important de le conserver et de lui proposer d'autres services susceptibles de l'intéresser. C'est ce qu'on appelle la relance client ou le « push »

2.3) Les concurrents de Digital Virgo Entertainment

Le secteur d'activité des divertissements numériques est assez récent. Pourtant, il existe plusieurs entreprises l'exploitant. Ainsi Cellfish est un de nos concurrents principaux. Cependant cette entreprise est plus axée sur l'audience mobile. Un autre grand acteur du marché est Vivendi Mobile Entertainment avec son service "zaoza". Ce dernier est un service pour ordinateur et téléphone mobile donnant accès à différents contenus multimédia.

2.4) Mes fonctions

J'interviens lors du développement d'application web ou dans la mise en place de nouvelles solutions de micropaiement.

Les principales missions sur lesquelles je travaille actuellement sont :

- Réalisation d'application audiotel :

  Une de mes mission au sein de cette entreprise consistent à réaliser des applications audiotel pour des clients tels que « AEF-Le parisien », « l'Équipe » ou « TV Mag ».

- Mise en place de l'internet+ mobile :

  L'internet+ mobile est une nouvelle solution de micropaiement. J'ai accordé le système d'information de l'entreprise avec les trois opérateurs mobiles principaux (Orange, SFR et Bouygues).

  - Gestion des abonnements clients.
  - Mise en place de cette solution de micropaiement sur nos sites internet.

- Réalisation d'un réseau social :

  Digital Virgo développe depuis plusieurs mois un réseau social baptisé « Woozgo », basé sur les sorties (réseau social de type « onvasortir.com »). J'ai participé à la réalisation de plusieurs fonctionnalités de ce site internet. La création d'activité et le listing des fans font partis des tâches que j'ai effectué.

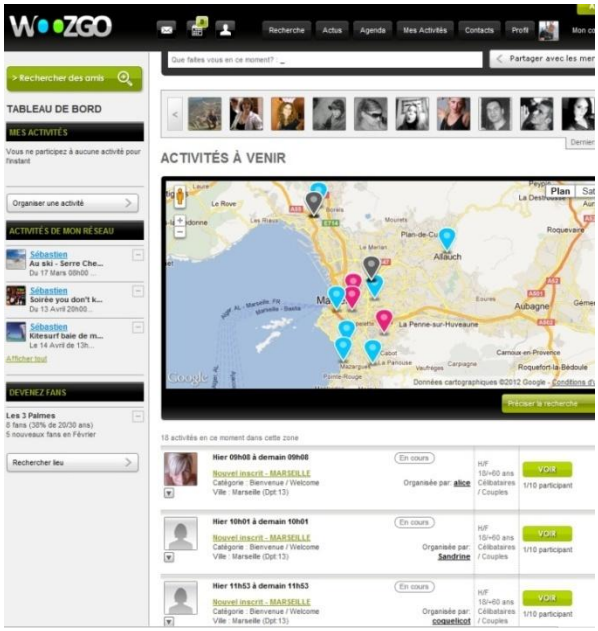

Figure 4 - Page d'accueil de woozgo.fr

- Réalisation d'un générateur de mini-sites :

  Ce projet est venu du constat que la plupart de nos sites possèdent la même structure. Il a été décidé de réaliser un générateur de mini-sites. De ce fait, la maintenance applicative de la plupart de nos sites est simplifiée. Seules les feuilles de style CSS, les scripts JavaScript et les images fournies par les intégrateurs changent. Nous proposons donc des sites ayant le même code source mais des apparences complètement différentes.

  Technologie utilisé: PHP/jQuery/CSS

Figure 5 - Page d'accueil de mobifiesta.com

2.5) Le développement d'applications chez Digital Virgo Entertainment

Lors de la planification du projet IRC, il n'y avait pas de logiciel de gestion de projet défini au sein de l'entreprise. Nous utilisions des fichiers Excel avec les colonnes nécessaires afin de suivre le projet dans le temps. Désormais nous utilisons le logiciel « Redmine » pour la gestion de tous nos projets.

2.5.1) Les différents acteurs

Plusieurs acteurs interviennent tout au long de la vie d'un projet.

En premier lieu, il y'a le chef de produit. Ce dernier recueil les différents besoins et rédige un cahier des charges fonctionnelles. Ces derniers sont réalisés de différentes manières. Cela peut être un document de type tableur (Excel), une présentation PowerPoint ou encore un document texte classique. Ce document récapitule toutes les fonctionnalités auxquels le projet devra répondre. Aucune notion technique n'est abordée.

La rédaction du cahier des charges terminée, la direction détermine les différentes priorités à accorder aux différents projets. Le chef de projet est désigné lors de ces réunions. La direction est composée du directeur informatique, du responsable commerciale ainsi que du responsable de la division projet.

Le chef de projet et le chef de produit adaptent le cahier des charges afin de ne retenir que les besoins techniquement réalisables au sein de l'entreprise. De plus, ces réunions précisent les fonctionnalités afin de bien délimiter le périmètre du projet. Une fois le projet analysé, le chef de projet le planifie et affecte les ressources aux différentes tâches de ce dernier.

Dans le même temps, l'équipe graphique réalise les différentes créations nécessaires à l'application. Une fois ces dernières réalisées, elles sont fournies aux développeurs.

L'équipe de développement s'occupe de la suite du projet. Ces principales missions sont les suivantes:

- Rendre le code dynamique en le liant à une base de données.
- Réaliser les différents développements JavaScript afin d'animer le site.
- Mettre en place les différents journaux d'événements (ou logs) afin de tracer le parcours des clients et mettre en évidence d'éventuels problèmes.
- Configurer et mettre en place les différents moyens de paiement.
- S'assurer que la nouvelle application sera bien surveillée ou monitorée (envoi de mail, SMS ou appel téléphonique en cas de coupure de service).
- Gérer le tracking (provenance de nos clients) afin d'avoir un suivi des différentes campagnes publicitaires.

Une fois ces étapes réalisées, le projet entre dans l'étape de recette fonctionnelle. Le chef de produit ainsi que plusieurs membres de l'équipe commerciale testent le produit final et vérifient que ce dernier répond bien aux besoins initiaux. Les bogues sont remontés à l'équipe de développement qui les corrige. Selon le projet, nous effectuons des tests de non-régression. Si ces derniers sont concluants, nous passons l'application en production.

Après cette étape, le chef de projet s'assure que les différents éléments de l'application fonctionnent comme attendu sous une utilisation normale. En complément, ce dernier reçoit chaque jour un mail avec les diverses erreurs provenant des logs.

En effet, lorsque l'application effectue des opérations imprévues ou anormales une chaine de caractères spéciale est inscrite dans les logs. Cette chaîne de caractères est par la suite envoyée par mail, indiquant à l'équipe projet qu'un problème a été détecté à ce niveau de l'application.

Nous avons également une équipe dédiée à la qualité de service. Celle-ci est chargée de tester quotidiennement nos différents services afin de s'assurer que toutes les fonctionnalités de ces derniers sont en état de marche. Ainsi les clients bénéficient d'un service de qualité.

Si des bogues d'utilisations sont détectés après la mise en production initiale, ces derniers sont consignés dans « Mantis ». Cet outil classe les bogues par projet et par ordre d'importance. Une fois corrigé par l'équipe technique, l'utilisateur qui a ajouté le bogue dans l'outil est informé de sa résolution et le clôture. Nous avons pour objectif d'avoir moins de cinq bogues « Mantis » à traiter par personne. Il faut donc être réactif pour résoudre ces derniers.

Récemment, un architecte de systèmes d'information a été recruté afin de standardiser l'environnement de travail des développeurs. Les développeurs de MediaPlazza utilisent un Framework commun à la réalisation de chaque application tandis que, les applications développées par Cellcast Média fonctionnent avec un minimum de dépendance. Chaque application récupère uniquement les librairies nécessaires à son fonctionnement. L'architecte a été chargé de la mise en place d'un Framework commun afin que tous développeurs puissent travailler sur n'importe quel projet.

Ci-dessous un schéma présentant le cycle de vie d'un projet :

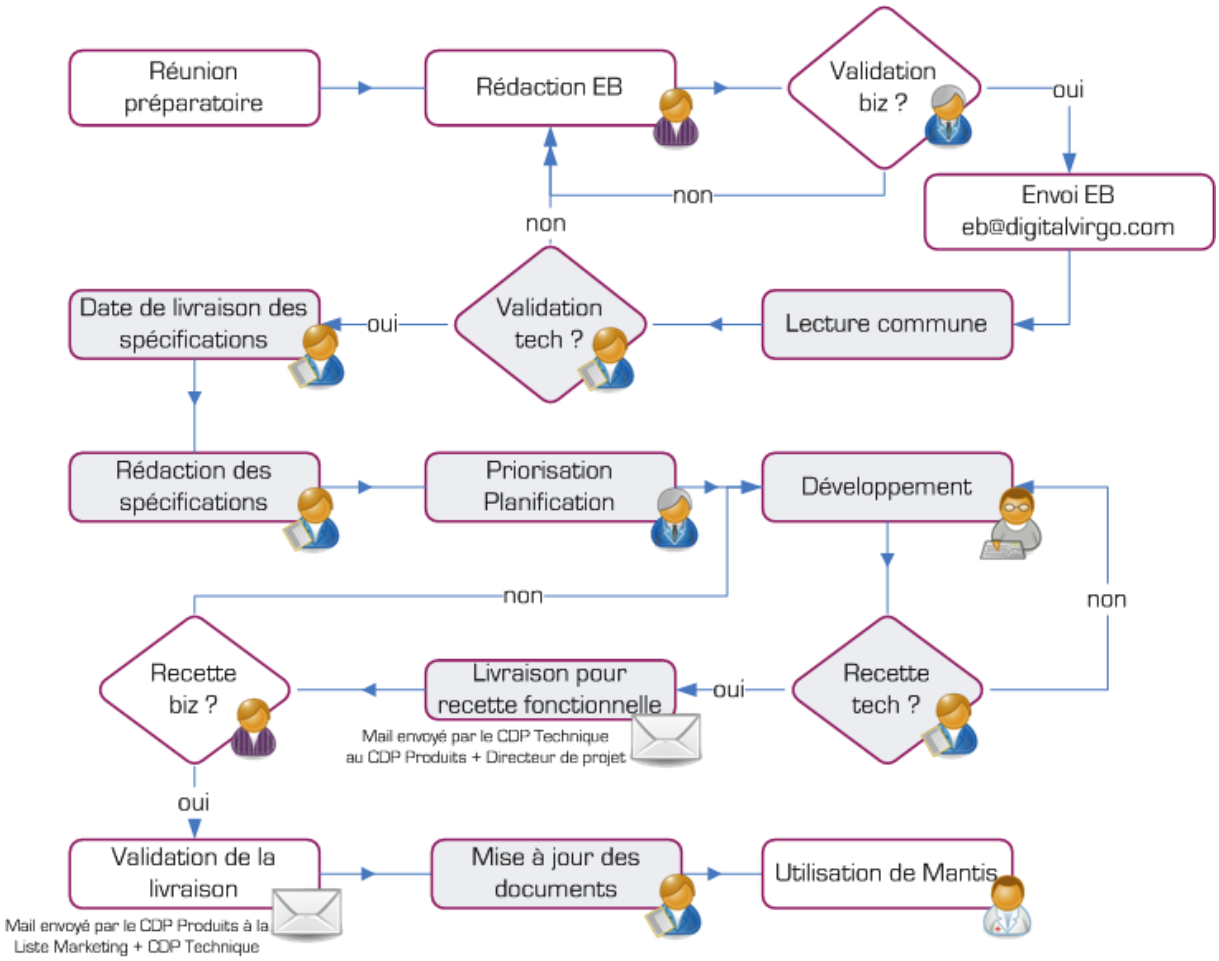

Figure 6 - Cycle de vie d'un projet chez Digital Virgo Entertainment

2.5.2) L'infrastructure technique

Ci-dessous, un schéma simplifié de l'infrastructure technique entre Digital Virgo Entertainment et notre hébergeur LinkByNet. Seules les plateformes de Marseille et d'Aix en Provence sont représentées. Celles de Paris et Lyon ne sont pas impactées par la réalisation du projet.

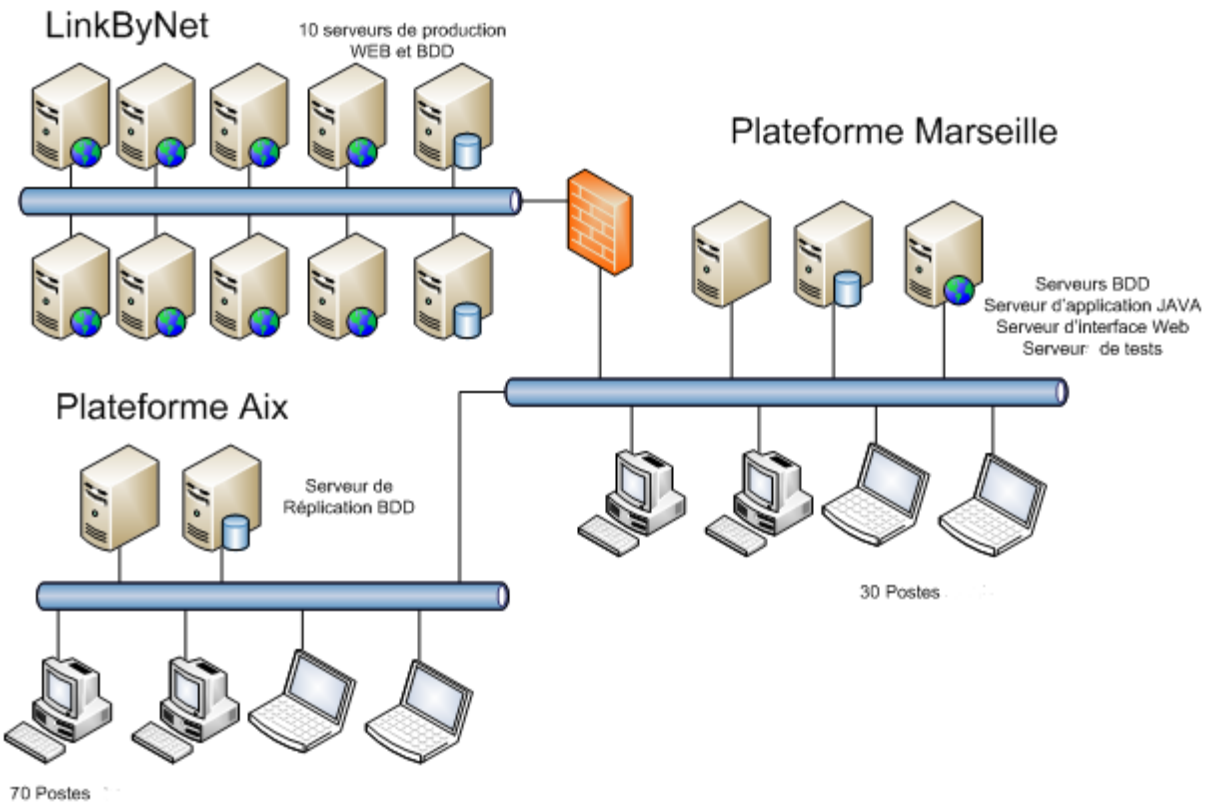

Figure 7 - Infrastructure technique de Digital Virgo Entertainment Marseille - Aix En Provence

Notre hébergeur met à notre disposition une dizaine de serveurs. Certains font parties de leur environnement « Agile » et sont donc virtualisés. La virtualisation de serveur consiste à faire fonctionner un ou plusieurs systèmes d'exploitation sur un ou plusieurs serveurs physiques. Un des avantages principaux de cette fonctionnalité est le fait de pouvoir modifier les capacités (de mémoire ou de stockage) du serveur sans intervention physique et dans un minimum de temps. Cependant, la plupart de nos serveurs à disposition sont des serveurs classiques, non virtualisés. Nous disposons aussi de deux serveurs de base de données. Ces derniers utilisent une configuration de type maître-esclave. La base de données maître est utilisée par tous nos services de production pour les accès de type lecture-écriture. Elle se met à jour en fonction des inscriptions et des commandes de clients. La base de données esclave copie le maître avec cinq minutes d'écart. Cette base de données est souvent utilisée dans le cadre de calculs de statistiques pour l'équipe commerciale.

La plateforme de Marseille regroupe une partie des équipes techniques. Nous pouvons développer sur nos serveurs de tests avant de passer nos services chez notre hébergeur. Cette

plateforme dispose aussi de serveurs de base de données de type maître-esclave. Une partie des informations enregistrées par nos services y figure.

La plateforme d'Aix en Provence regroupe une partie des équipes commerciales. Cette dernière dispose de quelques serveurs de réplications de nos bases de données de production.

2.5.3) L'environnement technologique

Nous utilisons plusieurs logiciels dont voici les principaux :

- Développement d'application
  Les développeurs ont le choix entre « Netbeans » et « Zend » pour le développement d'application.

- Gestion de projet
  La gestion de projet est assurée par « Redmine ». Nous y retrouvons les plannings, la rédaction des spécifications techniques et les différents acteurs du projet.

- Gestion de bogue
  La gestion de bogue est assurée par « Mantis ».

- Navigateur web
  Nos applications sont fonctionnelles avec les trois navigateurs principaux, Mozilla Firefox, Internet Explorer 9 et Google Chrome. Cependant avec l'émergence des tablettes tactiles et des smartphones nous développons des sites adaptés à ces nouveaux appareils technologiques.

- Gestion de BDD
  MySQL est le SGBD utilisé pour toutes nos applications.

- Dépôt SVN
  TortoiseSVN assure le suivi de versions de nos applications.

- Connexion FTP
  Filezilla est utilisé pour l'échange de fichiers avec les serveurs de productions.

- Connexion SSH
  Nous utilisons secureCRT pour se connecter à nos serveurs de production et lancer des scripts.

2.6) Nos différents moyens de paiement

2.6.1) Présentation

Au sein de notre entreprise nous disposons de plusieurs moyens de paiement. C'est à l'aide de ces derniers que nous monétisons l'audience réalisée sur nos différents services. Voici les principaux:

- Internet+ box
  Avec ce moyen de paiement, les clients peuvent s'abonner afin de bénéficier des différents services et régler le paiement sur la facture de leur FAI (fournisseur d'accès à internet). Après s'être identifié le client n'a plus qu'à cliquer sur un bouton pour confirmer son achat et profiter pleinement du service.

- Paypal
  Nous utilisons un mode de transaction à l'abonnement chez paypal. Une fois que le client a donné son accord pour accéder au service, nous pourrons par le biais de tâches planifiées, débiter ce dernier du montant de son abonnement hebdomadaire ou mensuel. En effet, une fois abonné, c'est à l'entreprise de prendre en charge l'abonnement et les transactions associées et non plus paypal.

- Audiotel
  Certains de nos services deviennent pleinement accessibles, après que le client appelle un numéro surtaxé débutant par « 08 ». Par exemple, un service met en relation des clients avec des entreprises après appel sur un numéro surtaxé.

- Internet+ mobile
  L'utilisateur est en mesure de régler l'achat de contenus numériques directement sur sa facture mobile.

- SMS+

  Le client est invité, à agir ou réagir en envoyant un mot clef par SMS à un numéro court (à 5 chiffres). Il reçoit en retour un autre message lui délivrant le contenu mobile désiré ou le connectant au service.

Part du recrutement sur nos différents moyens de paiement:

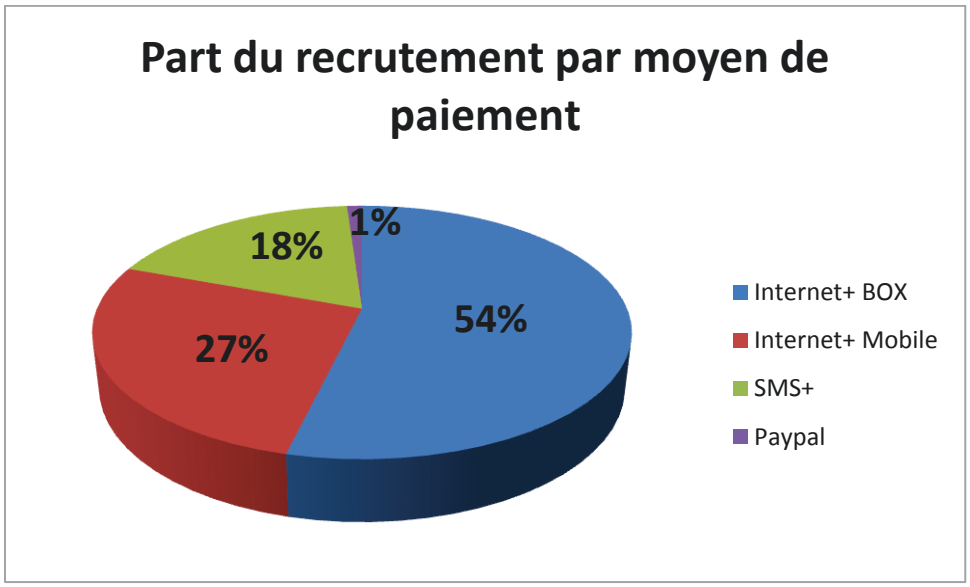

Figure 8 - Part du recrutement par moyen de paiement

L'internet+ box représente la majorité de notre recrutement de clients. Cependant l'internet+ mobile est le deuxième moyen de paiement qui recrute le plus de nouveaux clients. Il occupe déjà 27% de nos recrutements et il n'a pas été mis en place sur l'ensemble de nos sites. Le SMS+ est en phase de déclin. Paypal a été remis en place suite à l'arrivée de Free mobile sur le marché des opérateurs mobiles. Cet opérateur ne propose pas encore de solutions de micropaiement. Cependant, lorsque ce sera le cas, ce moyen de paiement sera de nouveau supprimé. Cela est expliqué par son faible taux de transformation.

2.6.2) Profils clients

Selon une étude menée par la société Médiamétrie 94% des 15-60 ans sont équipés d'un téléphone mobile. La moitié d'entre eux possèdent un smartphone. Ce sont en majorité de jeunes hommes. Plus du quart des mobinautes ont déjà effectué un achat depuis leur mobile. Ces derniers récupèrent majoritairement des contenus de types, jeux, musiques ou vidéos. Certains s'abonnent à des services payants tels que des sites d'information ou encore des sites de rencontre.

2.6.3) Evolutions

Différentes études indiquent que le téléphone mobile sera le premier moyen pour naviguer sur internet en 2020. Au niveau du paiement sur facture, une large majorité de mobinautes estiment que c'est un paiement simple rapide et pratique. Les micropaiements mobiles sont en phase d'expansion et nous allons en retrouver dans différents domaines. Ainsi les clients montrent un grand intérêt quand à la possibilité de régler leurs tickets de transports en commun, les places de stationnement ou encore le lavage de leur voiture avec le micropaiement mobile.

Ce secteur est en pleine évolution, et cible de plus en plus de secteurs différents. Il est important de développer une application qui pourra gérer ce nombre grandissant de clients potentiels.

2.7) Vers une internalisation complète de notre système d'information

Notre système d'information est très hétéroclite et nécessite de travailler en coordination avec plusieurs prestataires. Nous ne pouvons pas être spécialisés dans une multitude de domaines différents. Ainsi la réalisation de site mobile est confiée à Motricity. Une société spécialisée dans ce domaine. En outre cette société doit fournir les bons fichiers à télécharger en fonction du téléphone mobile du client. Par exemple, un téléphone mobile utilisant Android comme système d'exploitation ne sera pas en mesure de lire les mêmes fichiers qu'un téléphone utilisant iOS (iPhone).

Actuellement, le chantier le plus important de l'entreprise concerne l'internalisation des différents outils utilisés. Ainsi nous développons une application web générant des sites mobiles. Une fois cette application en place, nous n'aurons plus besoins des services de notre prestataire en charge de la création de site mobile. De plus, nous pourrons faire évoluer cet outil en interne. Outre l'aspect financier cela apporte un contrôle complet sur la direction à donner à l'application.

Un autre élément qui était externalisé concerne la gestion des performances de nos campagnes publicitaires. La société click2sales fournissait des rapports complets sur les performances de nos campagnes. Ces dernières sont affichées sous forme de rapport récapitulant le nombre de lead, de vente et le retour sur investissement de chaque campagne. Plusieurs statistiques sont disponibles via différents rapports.

Nous avons développé un outil gérant les différentes campagnes publicitaires. Ce dernier donne accès à des rapports affichant les statistiques de nos différentes campagnes. Exactement à l'image de l'outil qui était fourni par click2sales. Une fois l'application développée et mise en place, peu de problèmes ont été rencontré. Le seul souci majeur a concerné une montée en charge inattendue rendant l'application indisponible quelques heures. Cela après avoir mis à disposition l'application à d'autres filiales afin que ces derniers suivent les résultats de leurs propres campagnes publicitaires.

Tous les domaines ne sont pas gérables en interne comme par exemple la gestion de nos serveurs de production. Nous n'avons pas l'infrastructure nécessaire afin de répondre à un tel besoin. Pour de tels services, nous travaillons toujours avec des prestataires.

3) Présentation de l'application IRC

3.1) Le besoin

Suite aux difficultés rencontrées par notre prestataire, Motricity, la gestion du micropaiement SMS+ a récemment été internalisée. Cette internalisation a donné vie à l'application API_MOBILE. Après cette opération, nous n'avons plus de moyen simple d'envoyer des messages publicitaires à nos clients (ou les relancer). En effet, seul le propriétaire du micropaiement est autorisé à réaliser de la publicité sur ses clients. De plus, nous ne disposons d'aucune application destinée à l'envoi de messages publicitaires au sein de l'entreprise. Avec l'application développée par notre prestataire il n'est pas possible de cibler des clients abonnés en internet+ mobile. Nous devons être en mesure de relancer ces derniers car ce moyen de paiement est mis en place sur la majorité de nos sites web ou mobiles.

Actuellement nous ciblons et relançons manuellement les clients qui sont abonnés par le biais de l'application API_MOBILE. L'équipe commerciale donne les critères de sélection des clients ainsi que le message à diffuser. Le développeur en charge de l'application fait les requêtes manuellement dans notre système d'information, afin d'être en mesure d'effectuer les relances et de cibler les bons clients. Ces opérations prennent de plus en plus de temps. Il est donc primordial de fournir aux équipes commerciales une application les rendant autonomes.

3.2) Architecture technique

Pour la mise en place de l'interface de l'application, j'ai choisi d'utiliser un serveur gérant déjà d'autres applications disposant d'interfaces. Il s'agissait généralement d'interface de calculs de statistiques. Afin de faciliter les échanges avec les autres applications de notre système d'information, nous l'avons hébergé dans le même réseau que celui de l'application API_MOBILE. En effet, API_MOBILE sera utilisé pour l'envoi de SMS ou de Wap Push à nos clients.

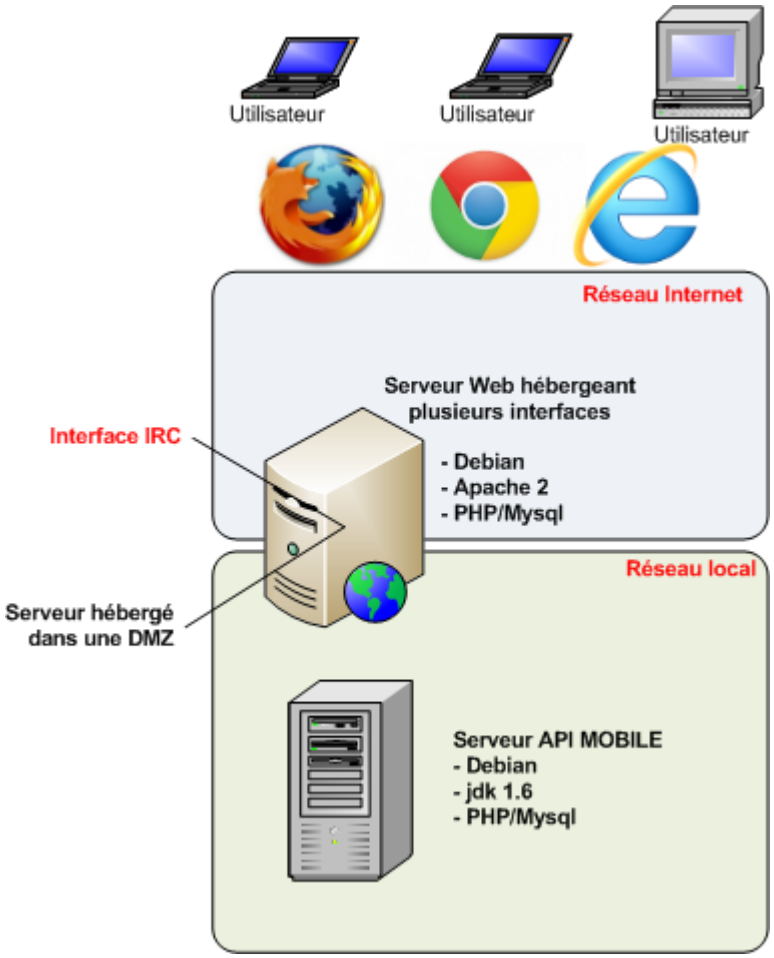

Figure 9 - Architecture technique de l'application IRC

Le serveur hébergeant l'application est dans une zone démilitarisée ou DMZ. De ce fait, ce dernier est accessible via le réseau local et via internet. Les utilisateurs auront le choix entre les trois navigateurs principaux pour consulter l'application, Google Chrome, Mozilla Firefox et Internet Explorer (Supérieur à la version 7).

3.3) Périmètre fonctionnel

Le schéma suivant présente le périmètre fonctionnel ainsi que les différentes interactions entre les acteurs de l'application IRC:

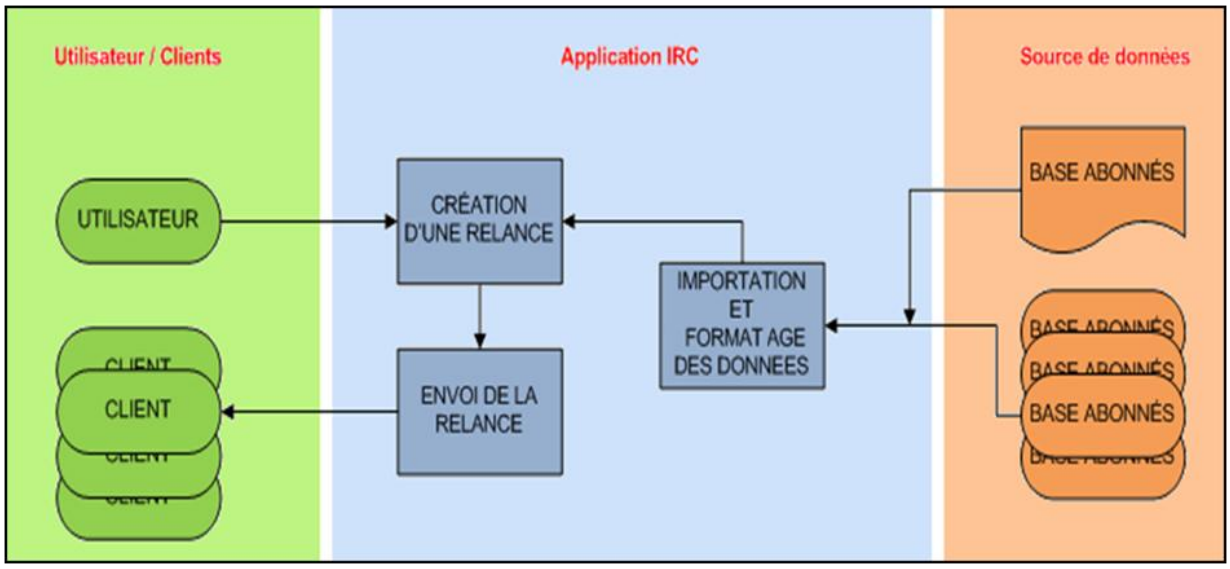

Figure 10 - Périmètre fonctionnel de l'application IRC

Sur ce schéma nous pouvons distinguer les trois missions principales auxquelles l'application devra répondre.

Tout d'abord, l'application devra importer et formater les données provenant des bases de données des services. Elle récupère les données et les rend utilisable au sein de l'application IRC. Ce traitement devra être régulièrement répété pour prendre en compte les modifications de nos différentes bases de données de services.

Ensuite l'application IRC devra proposer à ses utilisateurs la possibilité de créer une relance via l'interface. Les clients ciblés par la relance proviendront de l'import des données de l'étape précédente.

Enfin IRC sera en mesure d'envoyer la relance créée sur les téléphones mobiles de nos clients.

3.4) Analyse des faiblesses du système existant

L’équipe commerciale a mené une analyse de l’application existante. Pour nous informer des points qui doivent évoluer dans la nouvelle application. Un des soucis de l'ancienne application concerne sa mauvaise ergonomie. En effet, trop de clics sont nécessaires afin de réaliser des actions simples. Le positionnement des différents menus n'est pas intuitif. Le manque de sécurité au niveau des différents formulaires constitue également un problème majeur. L'interface ne possède pas de confirmation d'action. Lorsque l'on valide un formulaire, la base de données est directement mise à jour sans contrôle sur les données entrées.

3.5) Analyse des besoins

L'analyse des besoins a été réalisée par le chef de produit. Ce dernier a répertorié les différents besoins des utilisateurs finaux au sein d'un document Excel. Aussi, il s'est appuyé sur l'application existante afin de rédiger une liste de fonctionnalités.

J'ai ensuite parcouru l'expression des besoins avec le responsable de l'infrastructure technique lors de différentes réunions. Cela afin de ne garder uniquement les fonctionnalités réalisables sur notre plateforme technique. De plus, nous avons pu clarifier certains points au cours de ces réunions.

3.6) Objectif et fonctionnalités

L'objectif principal de ce projet consiste à réaliser une application disposant d'une interface interactive et ergonomique pour des équipes commerciales en charge d'envoyer des messages publicitaires ciblés à nos clients. Pour atteindre les meilleurs taux d'efficacité, l'application doit répondre à plusieurs critères.

Gestion de droits utilisateurs

L'interface devra être disponible pour plusieurs types d'utilisateurs différents. Ces utilisateurs représenteront des comptes clients différents. Nous réalisons certains de nos services dans le cadre de partenariat avec d'autres entreprises. Nos partenaires devront avoir accès à l'application. Ils seront eux aussi à même de relancer les clients qui se sont abonnés à leurs services.

Gestion de segments

La gestion de segments est un élément crucial de la réalisation de ce projet. Grâce à cette dernière nous serons en mesure de sélectionner un groupe de client par rapport à plusieurs critères plus ou moins fin. A noter que ces critères pourront être inclusifs ou exclusifs.

Par exemple, nous pourrons créer des segments de clients qui ont pour opérateur mobile « Orange » et qui sont abonnés à un service de téléchargement de jeux.

Gestion de campagnes

Ce module de l'application doit être en mesure de proposer la personnalisation du message de la campagne à envoyer aux clients. Cela peut être un message de type SMS ou un Wap Push avec les spécificités de chaque vecteur.

Un SMS sera limité à 160 caractères alors qu'un Wap Push doit contenir une URL de redirection. Nous pouvons de plus choisir la cible des types de profil de la campagne. Sur notre site internet, il y'a parfois des récupérations des numéros de téléphones mobiles. Lorsque nous demandons un email et qu'un client s'abonne, son numéro de téléphone n'est pas connu par notre système d'information. L'opérateur nous communique alors un alias qui est composé généralement d'une suite de digit. Nous devons donc être en mesure de relancer les clients par l'intermédiaire de leurs alias et leurs numéros de téléphones mobiles ou MSISDN (Mobile Station Integrated Services Digital Network. Désigne le numéro mobile de l'utilisateur).

<u>Rappel sur les messages SMS et leurs spécificités</u> :

Les messages de type SMS (Short Message Service) sont des messages constitués de plusieurs milliers de caractères (en utilisant la concaténation de 7 à 8 messages au maximum). A leur lancement, seul des messages de 160 caractères étaient acceptés. Dans le cadre de ce projet nous allons aussi limiter les messages de nos campagnes publicitaires de type SMS à 160 caractères.

A l'origine, les messages SMS étaient destinés à transmettre des messages de service envoyés par l'opérateur téléphonique. Ces derniers ont évolué et ont donné la possibilité aux utilisateurs de téléphone mobile de s'envoyer des messages entre eux.

Il existe une version améliorée de ce dernier, le MMS (Multimedia Messaging Service). Avec ce service, nous pouvons envoyer des messages plus longs aux contenus plus riches. Ainsi, nous pouvons transmettre des photos, des messages vocaux ou vidéo. Contrairement aux SMS les MMS utilisent des canaux prévus par l'opérateur, alors que les SMS utilisent les canaux de signalisation définis par la norme GSM. Les MMS n'occupent pas la bande passante réservée au transport de la voix.

Rappel sur les WAP Push et leurs spécificités :

Le WAP Push a été intégré dans les spécifications WAP, afin de donner accès à du contenu WAP avec une intervention minimal de la part des utilisateurs. Un WAP Push est composé d'un message encodé incluant un lien vers une adresse WAP et une accroche. Lorsqu'un client reçoit un WAP Push, une option lui propose d'accéder directement au lien. Seule l'accroche du message est visible par le client.

Création de requêtes / gestion de planning

Une requête met en relation un segment avec une campagne, afin de réaliser concrètement la relance. Cette requête sera configurée avec plusieurs informations :

- Les jours de la semaine durant lesquels la relance s'exécutera.
- Le volume de clients à cibler par jour.
- L'envoi de bon à tirer.

C'est par l'intermédiaire de ces derniers (les bons à tirer) que l'équipe commercial s'assure de la validité de la campagne sur des téléphones témoins.
Une fois la requête réalisée, cette dernière apparaît dans la gestion de planning. La gestion de planning est présentée sous la forme d'un calendrier prenant en compte les différentes relances qui vont s'effectuer dans la journée. Elle doit en outre confirmer ou annuler une relance en fonction de la conformité du bon à tirer.
Nous devons proposer l'avancé de l'envoi d'une relance, avec la possibilité d'arrêter cette dernière et de la reprendre plus tard en cas de difficultés.

Importation de données

La réalisation de ce projet nécessite l'importation de données au sein de l'application. La plupart de nos sites récoltent les numéros de téléphone de nos clients. Ils sont enregistrés en base de données. Nous disposons aussi de plusieurs applications audiotel récoltant des numéros de téléphones.

Nous récupérons les numéros provenant de ces applications par le biais d'export de fichiers réalisé une fois par mois. Pour être en mesure de relancer nos clients de manière efficace, il faut réunir tous ces numéros de téléphone au sein de la base de données de l'application IRC.

Réalisation de bon a tirer

S'assurer de la conformité de la cible des clients lors d'une relance est un point crucial. Par exemple, une relance peut être destinée uniquement aux clients d'un opérateur mobile particulier. Aussi, nous pouvons cibler des clients possédant un téléphone mobile d'une marque spécifique. Pour cela nous avons mis en place un système de bons à tirer. Au sein de l'entreprise, nous disposons d'une dizaine de téléphones mobiles possédant des configurations radicalement différentes. Ainsi, nous disposons de téléphones mobiles de plusieurs marques associés aux différents opérateurs mobiles principaux. Avant que la relance soit envoyée aux clients, ces téléphones reçoivent le message de cette dernière. Si ce dernier est conforme, l'équipe commerciale est alors en mesure de valider la programmation de la relance.

Récapitulatif

La section "récapitulatif" de l'interface affiche l'état des différents segments. Les utilisateurs de l'application peuvent surveiller l'évolution du nombre de client disponible par segment avec les mises à jour dues aux nouvelles transactions réalisées sur nos services.

Gestions des URL courtes

Afin de réaliser des campagnes publicitaires de type Wap Push performantes, nous sommes contraints de générer des URL de redirection relativement courtes. Ainsi nous pouvons utiliser un message d'accroche plus long. Un système simple a été mis en place afin de gérer les URL courtes. Un nom de domaine au nom relativement court a été acheté, « lefun.com ». Une table dans la base de données de l'application a pour rôle de faire la correspondance entre les URL courtes générées et les originales. Ci-dessous une explication du système mis en place.

Tableau 1 - Gestion des URL courtes

Id	URL courte	URL redirection
1	http://lefun.com/1	http://www.mobifiesta.com/?mobile=1
2	http://lefun.com/2	http://www.mobijeux.com/?v=mob

Une règle de redirection a été mise en place via la configuration d'un fichier .htaccess afin de passer en paramètre à un script PHP l'identifiant (Id) de l'URL. Les fichiers *.htaccess* sont des fichiers de configuration du serveur Apache (serveur hébergeant l'application), définissant les règles dans un répertoire et dans tous ses sous-répertoires (qui n'ont pas de tel fichier à l'intérieur). On peut les utiliser pour protéger un répertoire par mot de passe, pour changer le nom ou l'extension de la page index, ou encore pour interdire l'accès au répertoire. De plus, nous pouvons y définir une norme de réécriture d'URL. C'est ce point qui est intéressant dans notre gestion d'URL courtes. L'identifiant est passé après le nom de domaine. Voici un exemple « **http://lefun.com/1** ». Via la réécriture d'URL le script récupère donc l'identifiant de l'adresse web courte. En effet, pour le script PHP, l'URL sera de la forme suivante: http://lefun.com/?id=1. Ce dernier récupère alors l'adresse de redirection à l'aide d'une requête en base de données et redirige le client.

Statistiques

Comme pour toutes les applications réalisées au sein de notre entreprise, la réalisation de statistiques est nécessaire. Ces dernières sont assez simples et concerne les résultats des relances. Ainsi, il faudra être en mesure d'afficher le volume de messages envoyés et le segment associé à l'envoi. Nous devons aussi indiquer l'heure d'envoi du premier et du dernier message de la relance. Une autre information demandée concerne les volumes de clients relancés par opérateur. Enfin, il faudra indiquer les retours des envois indiqués par notre facilitateur. En effet, nous sommes informés en temps réel de l'état de l'envoi d'un message. Cette information peut indiquer que le message a bien été reçu par le client ou au contraire que le client n'a pas reçu le message.

Afin de bien cerner la cinématique à réaliser afin de créer une relance, voici un schéma récapitulatif du processus.

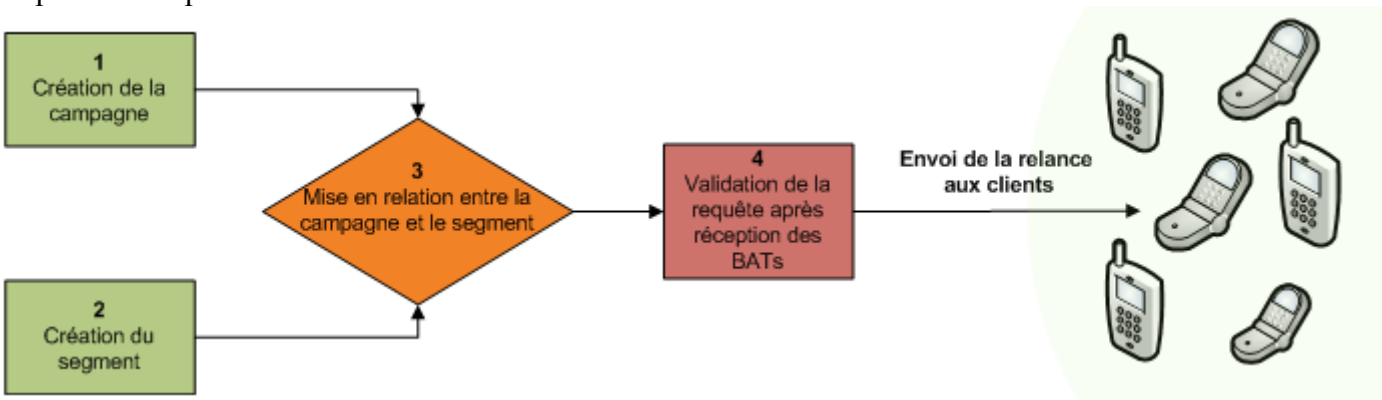

Figure 12 - Processus de création d'une relance

1. Création de la campagne.
2. Création du segment.
3. Mise en relation entre la campagne et le segment.
4. Validation de la requête après réception des BATS.

Une fois ces étapes réalisées, les clients recevront le message publicitaire sur leur téléphone mobile.

3.7) Méthode de conception

J'ai décidé de choisir SCRUM comme méthode de gestion de projet. L'ayant étudié aux cours de ma formation au CNAM, j'ai pensé qu'elle serait adaptée à la réalisation de ce projet.

3.7.1) Définition de SCRUM

Définition

SCRUM est une méthode de gestion de projet très connue. Elle fait partie des méthodes agiles au même titre que l'extrem programming et crystal clear. Le nom de cette méthode provient en fait du rugby. SCRUM a pour traduction française mêlée et parallèlement au rugby une équipe travaillant sous la méthode SCRUM, doit être soudée pour atteindre un but précis (comme pour la mêlée et le gain du ballon de rugby).

Les principaux acteurs de la méthode SCRUM sont le directeur de produit, l'équipe de développement et le ScrumMaster ou facilitateur.

Le ScrumMaster est chargé de résoudre les problèmes n'ayant pas de relation avec la technique. Tous les soucis administratifs lui sont destinés. Il a aussi pour but d'éviter toutes perturbations à l'équipe de développement.

SCRUM ne définit aucune méthode de développement logiciel. Souvent on utilise la méthode agile extrem programming en complément de la méthode SCRUM.

Fonctionnement

Le principe de fonctionnement essentiel de la méthode SCRUM est la découpe du projet en sprint. Un sprint a une durée fixe de deux à quatre semaines et contient des objectifs à atteindre. Lors de ses sprints, l'équipe est focalisée sur les fonctionnalités à réaliser. Les objectifs des sprints sont définis par le directeur de produit dans un backlog de sprint eux même découlant du backlog de produit (réalisé par le client et le directeur de produit). A la fin d'un sprint, l'équipe projet se réunit afin de valider le logiciel réalisé pendant ce dernier (on obtient une version partielle du produit final, appelé « release », qui est entièrement testable et utilisable). Si aucun problème n'est détecté lors de la démonstration du logiciel, le directeur de produit modifie le backlog de produit en fonction des avancées.

Chaque jour l'équipe projet doit réaliser un dailyscrum. Le dailyscrum est une réunion où chaque membre de l'équipe, le directeur de produit et le scrumMaster font le point. Durant ces réunions, chaque interlocuteur indique le travail réalisé la veille, celui à réaliser pour la journée en cours et les difficultés rencontrées. Cette réunion doit rester très courte et ne doit pas excéder cinq minutes.

Schéma de fonctionnement

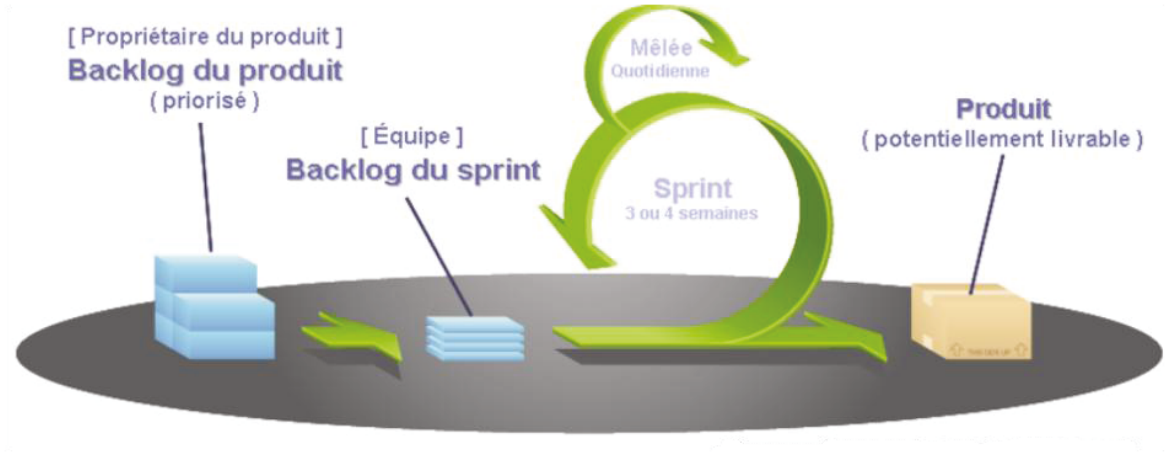

Figure 11 - Schéma de fonctionnement de la méthode agile SCRUM

Principes et valeurs

Les méthodes agiles reposent sur des valeurs. Chacune de ses valeurs sont déclinées en principes et constituent le socle des différentes méthodes agiles. Voici les quatre valeurs des méthodes agiles.

- Équipe soudée et communicante.
- Fournir impérativement une application qui fonctionne.
- Le client doit être souvent consulté et donner des retours (éviter l'effet tunnel).
- Accepter le changement, être réactif.

Les principes issus de ses valeurs pour la méthode SCRUM sont les suivants :

- Esprit d'équipe

La méthode SCRUM se concentre sur l'équipe et ses membres plutôt que sur les outils à utiliser. En effet, même avec les meilleurs outils du monde, sans la qualité dans l'équipe on n'atteindra pas les objectifs fixés.

- Le client au cœur du projet

Dans le but de fournir un projet correspondant aux attentes du client, on évite à tout prix la confrontation client/fournisseur avec négociation de contrat. On cherche à atteindre le même but, la réalisation du projet et on met tous les moyens en œuvre pour y parvenir.

- La communication est la clé

La communication est très importante. Le fait de confronter plusieurs visions pour un problème donné apporte de nombreuses solutions.

- Simplicité, efficacité et qualité

Lorsqu'on applique la méthode SCRUM, on se concentre sur l'essentiel. En effet, on évite de fournir trop de documentations, parfois inutile, et on se focalise sur les fonctionnalités du projet ainsi que leur sûreté.

- Flexibilité aux changements.

Suivre un plan prédéfini à l'avance à tendance à nous rendre insensible aux nouvelles demandes du client (souvent à des fins d'adaptation). De ce fait, nous pouvons être amenés à livrer un produit qui ne correspond pas aux attentes du client. SCRUM, par l'intermédiaire de ses sprints courts s'adapte parfaitement aux changements d'un projet.

- Avancement basé sur le concret.

Via la réalisation de sprint, on obtient relativement tôt dans le projet des livrables fonctionnels.

N'ayant pas de méthode de gestion de projet clairement définie au sein de l'entreprise, j'ai décidé d'utilisé SCRUM pour sa flexibilité et son adaptabilité. Pour cela j'ai donc décidé d'organiser des réunions quotidiennes pour savoir sur quels points mes ressources travaillent. Ainsi, je suis informé des avancés et des difficultés rencontrées. Au niveau des sprints la découpe est assez simple. L'application est découpée en six segments, j'ai décidé de réaliser un segment par sprint.

3.7.2) SCRUM en pratique

Lors de la réalisation de la mission, je n'ai pas mis en place tous les aspects de la gestion de projets avec SCRUM. Ainsi, nous n'avons pas réalisé de burndown charts. De plus, le dailyscrum n'était pas réalisé tous les matins mais à n'importe quel moment de la journée. Cependant la communication ne fut pas un problème dans l'équipe projet. Au moment où un problème était rencontré toute l'équipe en était informée.

Les autres points principaux de la méthode ont étaient respectés. Ainsi, une fois qu'un sprint est réalisé et qu'une release est disponible nous la livrons aux utilisateurs finaux. Cela afin de s'assurer que le projet suit la bonne direction. Parfois, les utilisateurs nous remontent des nouveaux points que nous traitons en fonction de leurs importances.

L'utilisation de SCRUM a été bénéfique lors de la réalisation de ce projet. Cette méthode tend à se généraliser à l'ensemble des développements réalisé au sein de notre entreprise. Cependant nous voulons d'abord l'adapter afin que cette dernière convienne parfaitement aux besoins de l'entreprise.

3.8) Backlogs de produits et sprints

Le backlog de produit se présente sous la forme d'une liste priorisée de tâches et de caractéristiques décrites par le client. En l'occurrence ici le chef de produit. Ce document a été rédigé avec ce dernier, puis présenté et complété avec toute l'équipe de développement, notamment pour le chiffrage du projet.
Ci-dessous le backlog de produit réalisé :

Tableau 2 - Backlog de produit

Importance	Description	Commentaire	Durée (Jour)
95	Connexion avec API_MOBILE	Intégration de l'application avec API_MOBILE afin de pouvoir envoyer les relances de type SMS et Wap Push.	10
90	Conception de la base de données	Conception et mise en place de la base de données.	5
87	Conception de l'interface de base	Réalisation de l'interface qui servira de squelette à tous les modules de l'application.	7
Sprint 1			
85	Synchronisation de la base de données de l'application avec le système d'information.	Importation des données à partir des différents services.	10
75	Conception des fonctions gérant la base de données	Réalisation des requêtes SQL modifiant les tables liées au fonctionnement de l'application.	6
60	Développement du plugin jQuery	Développement du plugin jQuery de recherche qui sera utilisé par tous les modules de l'application.	3
Sprint 2			
67	Réalisation du module « segment »	Réalisation du module de gestion de segments de l'application avec utilisation des filtres.	8
66	Réalisation du module « campagne »	Gestion des campagnes de type SMS et Wap Push avec possibilité de modification de ces dernières. Mise en place des repoussoirs.	5
65	Réalisation du module « requêteur »	Donner la possibilité de mettre en relation une campagne et un segment.	5

Sprint 3			
64	Réalisation du module « planning »	Afficher un calendrier avec les différentes relances planifiées par jour. Donner la possibilité de changer l'état de la relance afin de valider la relance ou l'annuler.	5
62	Développement du module « récapitulatif »	Afficher les différentes informations concernant les segments ainsi que leur disponibilité.	2
61	Affichage de l'avancement des relances	Proposer l'état d'avancement de chaque relance programmée en temps réel.	2
58	Envoi des bons à tirer	Configuration des bons à tirer et envoi de ces derniers sur les téléphones témoins.	3
57	Respect des contraintes	Prise en compte des différentes contraintes régissant l'envoi de relances.	4
Sprint 4			
55	Finition interface et sécurité JavaScript	Sécuriser les soumissions de formulaires avec des pop-up de confirmation JavaScript. Faire un contrôle des champs de formulaire obligatoire non remplis.	5
40	Configuration url courtes	Réalisation d'une interface annexe afin de générer des url courtes.	4
35	Gestion des droits utilisateurs	Donner la possibilité de se connecter à l'interface avec plusieurs comptes différents.	3
Sprint 5			
20	Réalisation des interfaces de statistiques	Donner l'accès aux différentes statistiques de l'application en termes de volume.	3
15	Optimisation de l'interface	Rendre l'interface plus rapide et fluide pour améliorer le confort d'utilisation de cette dernière.	5
Sprint 6			

Il est important que chaque tâche du backlog de produit ait une importance unique afin de créer une hiérarchie claire entre celles-ci. Il faut également laisser un écart suffisant entre chaque tâche à la création du backlog. En effet, il faudra être en mesure de pouvoir insérer de nouvelles tâches que l'on n'aurait pas forcément prévues.

Pour déterminer l'échelle que l'on va utiliser, on peut fixer trois seuils avec un code couleur :

- **Importance** >= 50 : (Rouge) Tâche indispensable à la livraison.
- 20 < **Importance** < 50 : (Jaune) Tâche indispensable mais qui autorise un petit délai de livraison.
- **Importance** <= 20: (Vert) Tâche secondaire qui pourra être livrée dans une version mineure suivante.

Le backlog de produit a été ensuite découpé en plusieurs backlogs de sprint. Le backlog de sprint présente toutes les tâches à mener durant un sprint. Voici une présentation des différents sprints que nous avons réalisés afin de développer l'application.

1er sprint :

Ce sprint a servi de mise en place de l'application. A l'issue de ce dernier, la structure de base de l'interface est réalisée. L'interface est prête à recevoir les différents modules. La base de données est en place. La connexion avec API_MOBILE est aussi en place. Le livrable disponible pour ce sprint n'est pas communiqué aux équipes commerciales car il n'y a pas d'action à tester.

2ème sprint :

A l'occasion de ce sprint, nous avons réalisé la majorité des fonctions gérant le modèle de données de l'application. Toutes les fonctions d'ajout, modification et récupération de campagnes ou segments ont été conçues. Lors de ce sprint, nous avons réalisé l'importation des données de nos différents services au sein de l'application. Un plugin jQuery, très souvent utilisé lors de la réalisation des modules, a aussi été développé durant ce sprint. Il n'y a toujours pas de livraison de release aux utilisateurs finaux. Les travaux réalisés jusqu'a présent ne les concernent pas.

3ème sprint :

Une première partie des modules a été réalisé durant le 3ème sprint. Ainsi les modules concernant la gestion de segments, de campagnes et leur mise en relation ont été développés. Une fois le sprint réalisé, nous avons pu livrer une version de l'application aux utilisateurs finaux. Ils ont pu tester l'application et nous donner les premiers retours concernant cette dernière. Les premiers retours ont concerné l'ergonomie de l'application.

4ème sprint :

La suite des modules a été réalisée durant ce sprint. En particulier, le module de gestion de planning. Evidemment le module envoyant les bons à tirer à nos téléphones témoins fut réalisé durant ce sprint. De plus, un module affichant l'état d'avancement en temps réel des relances a été effectué. C'est le sprint qui a demandé le plus de tests car ce dernier commande l'envoi des relances. Il était important de déceler les bogues amenés par ce sprint avant le développement de la suite de l'application et un éventuel passage en production.

5ème sprint :

Lors de la réalisation du 5ème sprint, l'interface a été consolidée et des fonctions de sécurité utilisant JavaScript ont été ajoutées. Ainsi tous les formulaires d'ajout de modification ou de suppression d'éléments disposent d'un pop-up de confirmation. Les champs incorrects ou vides sont aussi indiqués. Ainsi l'utilisateur connait les éléments bloquant la validation de son formulaire.

6ème sprint :

Lors du 6eme sprint, différents éléments ont été fournis aux administrateurs de QlickView afin qu'ils puissent réaliser les différentes interfaces de statistiques nécessaires. L'optimisation de l'interface afin d'améliorer le confort de son utilisation est le point suivant de ce sprint.

Le développement en plusieurs sprints de courtes durées apportent de nombreux avantages. Tout d'abord, les utilisateurs finaux sont impliqués dès le début de la réalisation du projet. Cela évite l'effet tunnel où l'avis des utilisateurs n'est pas demandé. Ainsi l'application répondra bien aux besoins des utilisateurs. De plus, les demandes d'améliorations et la détection de bogues remontent plus vite au cours du développement du projet.

3.9) Traitement des livrables

Les utilisateurs finaux et le chef de produit sont informés des modifications réalisées lors de la mise à disposition d'une nouvelle livraison de l'application. Ils sont alors en mesure de tester les nouvelles fonctionnalités. Ensuite, ils consignent les bogues décelés lors des tests dans un document. Une fois le document renseigné par tous les utilisateurs, l'équipe projet catégorise les bogues selon leur importance. Cependant, nous retrouvons souvent des demandes d'évolution au sein de ce document. Ce qui n'est évidemment pas la fonction de ce dernier. Dans ces demandes d'évolutions, nous pouvons citer la création d'export CSV.

L'objectif des dernières livraisons est d'obtenir une version stable de l'application disponible en production. Pour cela nous avons limité le traitement des bogues. Nous corrigeons uniquement ceux empêchant l'utilisation de l'application. Les bogues mineurs seront par la suite traités par lot et feront l'objet de mises en production ultérieures. Ceci afin d'éviter de modifier constamment l'application en production.

Voici la présentation du document récapitulant les bogues :

Tableau 3 - Fichier de suivi de bugs

Page : Segments	**Résolution**	**Validation équipe CRM**	**Commentaire**
Ajouter la possibilité de supprimer un segment (en pratique : le cacher. Modification à faire en base de données, à voir)			
Lorsqu'on met comme seul critère "compatibilité Wap push", le calcul de disponibilité ne s'exécute pas			
filtres optin/optout: ne fonctionne pas (prend 0/1/2 au lieu de OUI/NON/NSP)			

Ce tableau indique donc la page ou le bogue se produit, l'état de ce dernier, la validation de sa résolution par le chef de produit ainsi qu'un champ de commentaire.

3.10) Contraintes liées à l'envoi de relances

L'envoi de relances est soumis à plusieurs règles et contraintes. Nous ne pouvons pas envoyer n'importe quels types de messages à nos différents clients. Voici les principales règles régissant l'envoi de relances :

- Un client est éligible à l'envoi de relances à partir du moment où il a envoyé un mot clef à un de nos services. Cependant, les mots clefs demandant le fonctionnement administratif du service ne sont pas pris en compte. Un client envoyant le mot clef AIDE ou CONTACT à un de nos services ne sera donc pas éligible à la réception de relances. Au contraire un client envoyant le mot clef "KADO" au service associé sera susceptible d'en recevoir.

- Un client abonné à un de nos services ayant renseigné son numéro de téléphone mobile est aussi disponible pour l'envoi de relances.

- Un client ne peut pas recevoir plus d'une relance par jour. Si un client est éligible à l'envoi de relances sur plusieurs campagnes différentes, ce dernier recevra un seul message publicitaire.

- Un client n'est ciblé qu'une seule fois par campagne de relances. Nous devons nous assurer qu'un client ne recevra pas plusieurs fois le message d'une de nos campagnes.

- Lors de l'envoi d'un message sur un téléphone, notre facilitateur envoie une notification indiquant l'état du message envoyé. Ainsi l'envoi de message peut réussir, échouer ou encore être en attente. Dans le cas où l'on reçoit trois fois une notification d'échec pour un client, ce dernier est éliminé des prochaines relances.

- Lors de l'envoi d'une relance, nous devons cibler en priorité les clients qui n'en n'ont pas reçu récemment. Il suffit de trier les clients d'une relance avec leur dernière date de réception de message.

Dans certaines situations nous devons inscrire des clients sur liste noire. Dans ce cas précis, le client ne pourra plus être ciblé par notre application de relances. Voici les différentes situations inscrivant un client sur liste noire :

- Lorsqu'un client envoie le mot clef STOP à un de nos services, nous le désabonnons (s'il était préalablement abonné). Dans le cas où ce dernier envoi une deuxième fois le mot clef STOP il entre sur la liste noire.

- Les clients envoyant le mot clef CONTACT ou AIDE à un de nos services sont inscris dans la liste noire.

- Parfois, l'équipe commerciale fourni les informations d'un client (alias ou msisdn) afin que nous l'inscrivions dans cette liste.

- Nos équipes à l'ile Maurice consultent un fichier à plat contenant tous les messages envoyés par nos clients à nos différents services. Ils détectent les clients mécontents, les communiquent et nous les excluons du système de relances.

Parfois les clients ne sont pas éligibles à l'envoi de relances. Voici les différents cas conduisant à cette situation :

- Les clients ayant renseigné leur numéro mobile sur un de nos sites web mais n'ayant pas consommé ne sont pas éligible à l'envoi de relances.

- Les clients qui ont envoyé un mot clef non reconnu sur un de nos services ne sont pas ciblés lors de l'envoi de relances.

L'envoi de relances est soumis à la catégorisation des clients. En effet, nous ne pouvons relancer les clients provenant de certains partenariats selon la contractualisation. De plus, on ne peut pas relancer avec les mêmes critères un client qui s'est abonné à un service adulte d'un autre abonné à un service de jeux.

3.11) Relance via alias

Lorsque les clients s'abonnent en SMS+ ou en internet+ mobile, l'opérateur transmet l'alias du client comme unique information (au retour du paiement). Lorsqu'un client consulte un site mobile en 3G, l'alias est également transmis (le site mobile doit être au préalable inscrit dans un cône d'URL prévu par l'opérateur). L'alias désigne l'identifiant fourni par l'opérateur à la place du MSISDN afin de préserver l'anonymat du client. Ce dernier se présente sous différentes formes selon l'opérateur :

Tableau 4 - Présentation des alias

Opérateur	Type Alias
SFR	270977405656956
ORANGE	379134117060
BOUYGUES	1000000073012360

<u>Cinématique d'abonnement SMS+ :</u>

Au préalable, l'éditeur du service doit réserver un numéro court SMS+. C'est un code à 5 chiffres (88555, 88321) que les clients devront contacter afin de s'abonner.

Un abonnement SMS+ se déroule en 3 étapes :

1. Le client envoi un mot clef ou un ticket à un numéro court. C'est le MO (Mobile Originated).

2. Le client reçoit ensuite un message lui récapitulant les informations du service sur lequel il souhaite s'abonner. De plus, ce message l'invite à répondre « OK » afin de valider l'abonnement. Ce message est envoyé par l'opérateur mobile du client.

3. Une fois que le client a envoyé le mot clef « OK », l'abonnement est effectif. Le client reçoit alors le contenu souhaité sur son téléphone mobile ou sur le site internet. L'opérateur par l'intermédiaire de notre facilitateur nous indique que le client est bien abonné.

Nous utilisons un facilitateur, Netsize, qui assure la gestion des échanges avec les opérateurs. Chaque opérateur dispose de leurs spécificités. Netsize s'occupe de les gérer et nous retourne des informations uniformes. L'alias du client est fourni lors du dernier échange.

Pour arrêter son abonnement et ne plus être facturé par le service, le client doit envoyer le mot clé « STOP » au numéro court auquel il s'est abonné. Une fois cette opération effectuée, Netsize envoi une information indiquant que le client a résilié sa souscription. Il ne reste plus qu'a résilier l'abonnement au sein du service concerné.

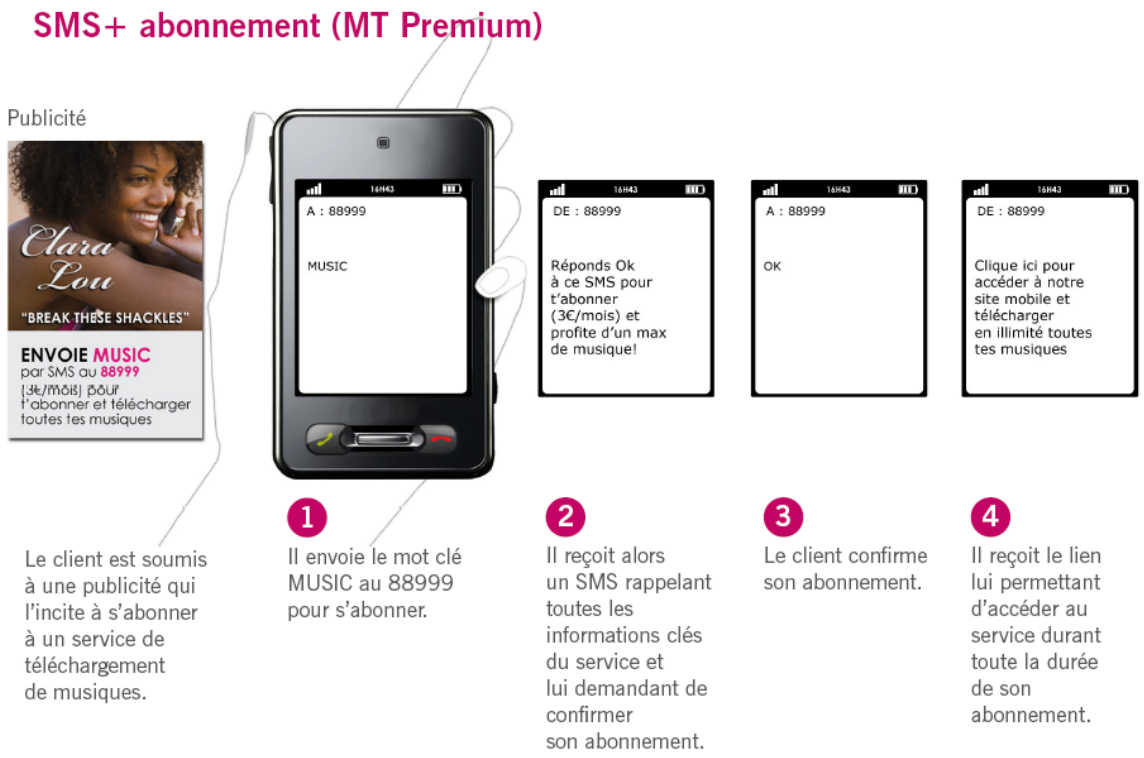

Figure 13 - Cinématique d'abonnement SMS+

Cinématique d'abonnement Internet+ Mobile :

L'internet+ Mobile propose deux solutions de paiement bien distinctes. Une cinématique mobile et une autre adaptée aux sites web. Notre entreprise est connectée avec les trois opérateurs mobiles principaux Bouygues, SFR, Orange. Ce sont les seuls opérateurs proposant cette solution de micropaiement pour le moment. Il suffit alors de créer les offres chez chaque opérateur afin de les proposer sur nos services.

Cinématique mobile :

1. Le client navigue sur un de nos services mobiles et choisit le contenu qu'il souhaite télécharger (sonnerie, logo etc...).

2. Nous redirigeons le client vers la page de paiement de son opérateur. Cette redirection est réalisée grâce à l'adresse IP du client. En effet, chaque opérateur dispose d'une plage d'adresse IP spécifique. Si le client utilise une connexion Wi-Fi, nous lui proposons le choix de son opérateur via un formulaire.

3. Le client valide le paiement puis accède au contenu payant sur le site mobile.

La cinématique Full Web est similaire. Seul l'affichage des panneaux de paiements est adapté aux ordinateurs.

L'alias est communiqué via enrichissement des en-têtes HTTP par les opérateurs. Dans le cadre d'une navigation en Wi-Fi cette information est transmise via l'URL de retour de paiement.

Pour arrêter son abonnement, le client doit se connecter à l'espace client de son opérateur mobile. Il peut ensuite accéder à la liste complète de ses souscriptions et résilier celles de son choix. Lors de la synchronisation entre notre base de données et celle de l'opérateur, l'état de chaque souscription est indiqué.

Figure 14 - Cinématique d'abonnement internet+ mobile

3.12) Relance via MSISDN

Le MSISDN désigne le numéro de téléphone mobile de l'utilisateur. Sur certains de nos services web nous récupérons les numéros de téléphone de nos clients afin de pouvoir leurs envoyer le contenu souhaité.

Une fois le numéro récupéré et le client abonné, nous pouvons le relancer. Les MSISDN récupérés via nos applications audiotel sont aussi sauvegardés. En effet, lors de certaines opérations nous collectons ce type d'information afin de créer le compte client des participants.

4) Développement de l'application

4.1) Mes missions

Les principaux objectifs de ma lettre de mission en tant que chef de projet sont les suivants :

- Valider l'expression des besoins avec le chef de produit.
  Pour cela nous avons organisé plusieurs réunions avec les utilisateurs finaux afin de formaliser au mieux leurs besoins. Une fois ces réunions terminées, le chef de produit rédige un cahier des charges fonctionnel récapitulant tous les besoins auxquels l'application devra répondre.

- Proposer et choisir les différentes technologies à utiliser avec mon directeur technique.
  Le choix des technologies est assez limité. Cela est principalement dû aux différentes interactions que l'application doit avoir avec notre système d'information. De plus, j'ai adapté mes choix en fonction des différentes ressources affectées au projet.

- Organiser le planning et les ressources affectées à chaque tâche du projet.
  Il faut organiser le planning des ressources en estimant la charge de ces dernières et fournir des chiffrages précis. Ce n'était pas évident car les ressources sont parfois mobilisées sur d'autres tâches. Les modifications ou corrections de bogues sur des services de productions sont prioritaires.

- Développer certains modules de l'application.
  J'ai développé certains modules de l'application. Ainsi mes compétences en JavaScript et plus particulièrement sur la librairie jQuery m'ont permises de réaliser un plugin réutilisable (cf « Réalisation d'un plugin jQuery »).

- S'assurer de la conformité des livraisons avec l'équipe commerciale.
  Afin de s'assurer que l'application conviendrait aux utilisateurs finaux, nous la faisions tester à ces derniers dès que nous disposons d'une version stable.

- Faire la recette de l'application.
  Une fois l'application en production, il faut s'assurer que cette dernière n'entraîne pas de bogues collatéraux. Par la suite, il faut capitaliser les informations recueillies lors de ce projet afin de repérer les bonnes méthodes et les erreurs à ne plus reproduire.

4.1.1) Développement

L'équipe projet est composée de trois personnes :

- Ghislain OREKAN : Chef de projet et développeur.
- Renaud BEAUFILS : Développeur et gestion de base de données.
- Sylvain CASAREGOLA : Développeur NTIC (Nouvelles Technologies de l'Information et de la Communication).

Chaque membre de l'équipe projet est responsable de plusieurs applications appartenant à notre système d'information. Nous ne pouvons pas consacrer tout notre temps de travail au développement de l'application IRC. Il faut parfois faire des correctifs ou des évolutions sur les applications qui fonctionnent en production. C'est une notion cruciale à l'heure du chiffrage du projet.

Renaud Beaufils a été chargé de la mise en place de la solution SMS+ au sein de l'entreprise (via l'application API_MOBILE). Au vu de l'interconnexion forte qui lie ces deux projets, il est important que ce dernier soit membre de l'équipe projet.

Pour la conception de la base de données nous avons décidé d'utiliser le logiciel « Power AMC Designer ». A l'aide de son interface graphique, nous pouvons réaliser le schéma de la base de données. Le schéma intègre toutes les relations liant les différentes tables de l'application. Ainsi toutes les clefs primaires et étrangères y sont représentées. Une fois le schéma réalisé, le logiciel proposera de générer le code SQL associé. Plusieurs formats de génération de code sont disponibles. Le format MySQL sera notre choix car c'est le SGBD utilisé par l'ensemble des services de notre entreprise.

4.1.2) Conception de la base de données

La base de données de l'application est relativement conséquente. Cette dernière est composée de 29 tables. Ces tables peuvent être séparées en deux sections bien distinctes. En effet, une partie des tables de la base de données sera destinée à l'importation de données provenant des services. Une autre partie des tables de la base de données sera consacrée au fonctionnement de l'application. Ces tables assurent la construction de segments, de campagnes et d'autres éléments nécessaires au fonctionnement de l'application.

Ci-dessous une présentation des principales tables de l'application :

Tables liées à l'importation de données

CLIENT :

La table client enregistre tous les clients disponibles pour la réception de relances.

MOBILE :

Cette table enregistre toutes les informations concernant les téléphones mobiles de nos clients. Grâce à cette dernière nous pourrons connaître le téléphone mobile utilisé par nos clients ainsi que leurs différentes caractéristiques (réception des Wap Push etc…). Nous collectons ces informations grâce à l'en-tête HTTP « User Agent ». Cette dernière renvoie une chaîne de caractères pour chaque téléphone mobile. Ainsi lorsqu'un mobinaute se connecte à un de nos sites, nous reconnaissons immédiatement son téléphone mobile. Nous comparons la chaine retournée dans les en-têtes avec celles que nous possédons au sein de notre base de données « info_mobile ». Cette dernière enregistre les différentes caractéristiques des téléphones mobiles avec pour clef d'identification l' « User Agent » du téléphone.

SERVICE :

C'est dans cette table que les informations de nos différentes applications sont enregistrées. Par son intermédiaire nous pouvons relier un client et le service auquel il s'est abonné.

PARENTAL_CONTROL_LEVEL :

Cette dernière catégorise le niveau de l'application. Ainsi nous pouvons différencier les applications « tous publics » et les applications de type « adultes ».

OPERATOR :

Dans cette table nous enregistrons l'opérateur téléphonique de nos clients.

REF :

Cette table relie la référence d'un contenu aux clients qui l'ont téléchargé. En effet, tous nos contenus peuvent être identifiés par une référence unique. Il est alors intéressant dans le cadre d'envoi de relances de connaître les contenus abonnant le plus nos clients.

Tables dédiées au fonctionnement de l'application

CAMPAIGN :

Dans cette table, nous enregistrons toutes les informations caractérisant une campagne.

SEGMENT :

La table segment comprend les groupes de clients, filtrés par les critères de recherches définis dans la table « filter ».

FILTER :

Ce sont les filtres liés à la création d'un segment. Ces derniers indiquent si un client doit être pris en compte au sein d'un segment.

PUSH :

Cette table fait la relation entre un jour, un segment, un volume de client et une campagne. Une fois renseignée nous pouvons effectuer la relance en fonction de ces critères.

CLIENT_PUSH :

Cette table lie un client à une relance.

PUSH_STAT :

Lors de l'envoi d'un message de type SMS ou Wap Push sur un téléphone client. Notre facilitateur nous envoie en retour les informations concernant la réception du message par le client. Nous les conservons dans cette table.

Le schéma de la base de données a été réalisé avec Power AMC designer. Voici ce dernier :

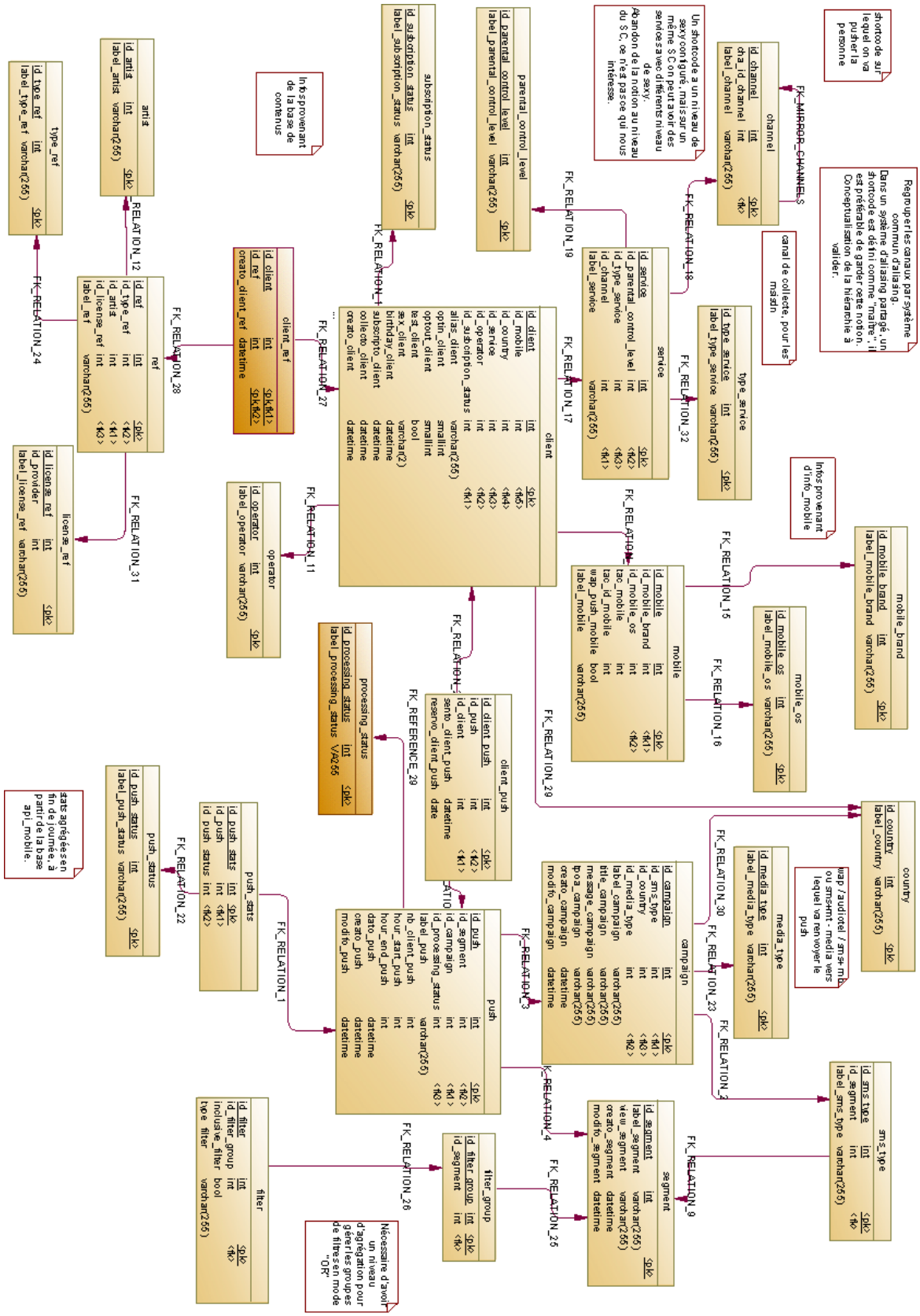

Figure 15 - Schéma de la base de données de l'application IRC

4.2) Méthodes de travail

4.2.1) Développement web

L'application doit être accessible à n'importe quels postes connectés à internet. J'ai donc choisi de la développer sous la forme d'une application web. Ce choix entraine de nombreux avantages. Ainsi les utilisateurs pourront y accéder lors de déplacements. De plus, ils pourront l'utiliser à domicile en cas d'urgence.

Au sein de l'entreprise, la majorité de nos applications sont réalisées avec le langage de programmation PHP. J'ai donc décidé d'utiliser cette technologie pour réaliser l'interface. C'est le langage informatique qui est le mieux maîtrisé par les développeurs de l'entreprise. Aussi, j'ai décidé d'utilisé un Framework développé par l'entreprise donnant entière satisfaction sur quelques une de nos applications.

Pour le travail en équipe, nous utilisions un logiciel de gestion de versions. Ainsi nous évitons le risque d'écraser le travail des autres personnes de l'équipe. En complément, chaque membre de l'équipe projet est informé des développements de chacun. Cela afin d'éviter ce type de problème.

4.2.2) Présentation du Framework

Le Framework que nous utilisons est basé sur un modèle MVC (Modèle Vue Contrôleur). Un des avantages de ce dernier est de séparer la vue du code.

Rappel sur le modèle MVC :

Le principal objectif du modèle MVC est de séparer l'application en trois couches distinctes.

- **Le modèle** encapsule la logique métier et la manipulation des différentes sources de données.
- **La vue** est chargée de présenter les données aux utilisateurs de l'application.
- **Le contrôleur** analyse les requêtes des clients, accède aux données et les préparent afin de les transmettre à la bonne vue.

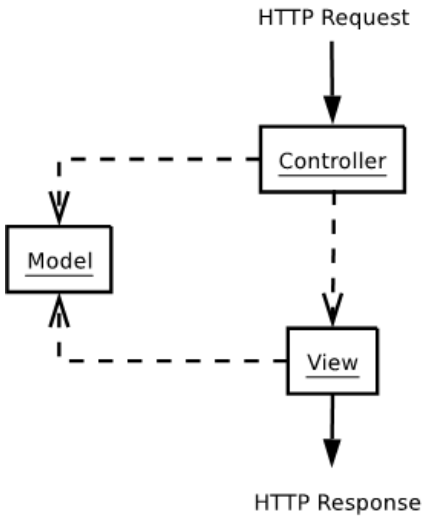

Figure 16 - Présentation du modèle MVC

En utilisant ce Framework, les graphistes peuvent faire des modifications sans impacter les travaux des développeurs. J'ai récupéré du Framework les éléments nécessaires à mon application. Par exemple, j'ai réutilisé principalement les fonctions de communication avec la base de données et le système de gestion de logs. Le Framework est basé sur un routeur qui en fonction des paramètres reçus appelle le model et la vue associée.

Le Framework aura aussi à charge via son routeur de charger les pages JavaScript et feuilles de style CSS en fonction des demandes des utilisateurs.

En termes de design, j'ai souhaité réaliser une interface graphique simple. Il me semble inutile de la surcharger et ainsi perdre les utilisateurs avec des informations inutiles. Il est important que l'aspect graphique de cette dernière se rapproche de celle précédemment utilisé.

L'interface étant assez simple, j'ai pris la décision de ne pas faire réaliser cette dernière par l'équipe graphique. J'ai choisi de la concevoir avec l'équipe projet afin de gagner du temps. En effet, lorsque les graphistes réalisent une maquette, il faut un certains temps afin que les développeurs intègrent et dynamisent cette dernière.

4.3) jQuery

4.3.1) jQuery le futur du JavaScript?

Après une recherche des différentes librairies JavaScript récentes, je me suis rapidement aperçu que jQuery est la librairie JavaScript la plus utilisée et la plus complète du marché. Cette librairie possède de nombreux avantages. Voici les principaux :

- Le code généré est compatible avec les navigateurs principaux. Un point très important à l'heure où tous les navigateurs évoluent afin de s'adapter à HTML5.
- La version compressée de la librairie est très légère.
- Compatible avec CSS3.
- Possibilité de créer ses propres plugins.

Un point fort de la librairie jQuery est sa grande communauté. Outre, les plugins officiels complétant la librairie de base, de nombreux autres sont réalisés par les membres de la communauté couvrant un large spectre des besoins JavaScript d'une application.

De grandes entreprises comme Google, Dell ou NETFLIX utilisent jQuery sur leur site internet. Microsoft a intégré jQuery directement dans Visual Studio pour le développement d'application en ASP.NET.

Comme l'indique sa devise, avec jQuery nous pouvons réaliser plus de choses en écrivant moins de lignes codes. La première version stable de jQuery est sortie le 26 Aout 2006.

4.3.2) Réalisation d'un plugin jQuery

Lors du recueil des besoins, le chef de produit nous a expliqué que les recherches menées au sein de l'interface doivent être simple et efficace. Le nombre de campagne, de segment ou encore de marque de mobile étant très élevés, il est nécessaire de proposer des filtres sur les résultats de recherche. Tout d'abord, j'ai pensé à classer les résultats à l'aide d'une pagination réalisée en Ajax. Elle aurait pour avantage de ne pas recharger la page et d'améliorer le temps des réponses des requêtes. J'ai ensuite fais le choix de lancer la recherche en fonction des premières lettres renseignées par l'utilisateur.

Après une recherche sur les différents plugins de recherches de jQuery, il s'est avéré qu'aucun d'eux ne convenait complètement aux besoins. J'ai donc décidé de réaliser un plugin répondant à notre problématique. De plus, il sera réutilisable pour les prochains projets de l'entreprise.

<u>Objectifs du plugin:</u>

- Afficher les résultats d'une recherche en fonction d'une chaîne de caractères.

- Présenter les résultats sous plusieurs formats différents à l'aide de template fourni à l'appel du plugin. Un template ou gabarit désigne un patron de mise en page où l'on place des images et du texte. Il est souvent utilisé de manière répétitive.

- Configuration du temps entre l'exécution de la requête et la fin de saisie de la recherche par l'utilisateur.

- Possibilité de filtrer les résultats à partir du résultat d'une requête Ajax où d'une source de données fixe.

- Donner la possibilité d'exécuter du code JavaScript, une fois les opérations de filtrage et de mise en place menées par le plugin terminé.

- Lier une action au choix d'un élément de la liste de résultats.

D'un point de vue utilisateur le plugin, effectue une recherche en fonction des données saisies par l'utilisateur. Il classe ensuite les résultats sous une forme ordonnée d'éléments. L'utilisateur n'a pas de besoin de cliquer sur un bouton de recherche avant que cette dernière s'effectue.

Détail du code source du plugin :

Après avoir consulté les principes de bases de la création de plugin jQuery, j'ai pu rapidement mettre en place une première version du plugin. Voici les différentes étapes menant à la création de ce dernier.

Il faut d'abord déclarer le plugin via la fonction jQuery « fn » :

jQuery.fn.tnkfilter = function(params) {...}

Le nom du plugin est donc « tnkfilter ». Le code source du plugin se situera entre les accolades.

Nous allons par la suite, déclarer les variables par défaut qui pourront être modifiées au moment de l'appel du plugin. Ces variables ne sont pas nécessaires lors de l'appel du plugin.

params = $.extend({delay: 300,complete:"",data:"",filter2:"", limit:false,noresult:"Aucun résultat"}, params);

- La variable « ***delay »*** désigne le temps s'écoulant entre la fin de la saisie de l'utilisateur et le début de l'exécution de la recherche.

- Avec la variable « ***complete »*** nous pouvons exécuter du code JavaScript une fois que le plugin a terminé ses opérations.

Il faut ensuite réaliser la logique fonctionnelle du plugin :

```
return this.each(function(){
    // Construction du HTML construisant la liste
    var keyuptimeout ='';
    input_search = $(this).attr("id");
    $(this).attr("autocomplete","off");
    $(this).css("background-color","#C0C0C0");
```

Gestion du filtrage sur une page Ajax ou sur des données JavaScript

```
    if(params.source)
    {
      array_source = params.source;
      generate_display();

    }else if(params.url){
      $.ajax({
          url: params.url,
          data: params.data,
          async: true,
          type: 'POST',
          dataType:'script',
          success: function(){
              generate_display();
          }
        });
    }
```

L'appel Ajax est réalisé en mode asynchrone afin de ne pas bloquer le navigateur de l'utilisateur.

Rafraichir la liste lorsque le client modifie sa recherche

```
    $(this).bind('keyup', function(){
      if(params.source){
          clearTimeout(keyuptimeout);
          keyuptimeout = setTimeout(load_data,params.delay);
      }
      else if(params.url){
          clearTimeout(keyuptimeout);
          keyuptimeout = setTimeout(load_data_ajax,params.delay);
      }
    });
  });
```

Lorsque le client modifie sa recherche dans le champ associé, le délai avant l'exécution de la requête est réinitialisé.

Voici un exemple d'appel au plugin :

```
$( "#search_camp_inclu_pushee" ).tnkfilter({
```

Paramètres par défaut

```
            dest_div:'div_filtered_inclu_pushee',
            url: "../ajax/get_generic_fetch_nfo.php",
            data: "table=campaign&no_del=1",
            delay: 1000,
```

Le paramètre « ***url*** » désigne le chemin du script qui effectue la requête en base de données. Le champ « ***dest_div*** » désigne l'élément de la page HTML qui recevra la liste de résultats générée par le plugin.

Gestion de la sélection d'un résultat

```
            select: function(e){
              var id_campaign=$(this).attr('valeur');
              if(campagnes_incluses[id_campaign]==null &&
campagnes_excluses[id_campaign]==null){
                $( "<li id='inclu_pushee-"+id_campaign+"'></li>" ).html( "<img
src='../img/delete.png' width='15'
onclick='delete_campaign_from_repoussoir("+id_campaign+",\"inclu_pushee\")' /> "+
$(this).text() ).appendTo( '#ul_inclu_pushee' );
                $( '#inclu_pushee' ).find( ".placeholder" ).remove();
                campagnes_incluses[id_campaign]=$(this).text();
              }
            },
```

À la fin du traitement, l'identifiant de l'élément sélectionné est ajouté dans un tableau. Par la suite, nous pourrons transmettre cette information en base de données.

Traitement à la fin du chargement de la liste de résultats

```
            complete: function(){
                        $("#div_filtered_inclu_pushee .tnkfilter li").hover(function(){
                $(this).addClass('selected');
              },function(){
                $(this).removeClass('selected');
              });
            }
          });
```

Une fois la liste de résultats chargée par le plugin, nous ajoutons une surbrillance au survol d'un élément de cette dernière.

Utilisation du paramètre template

template: "<input type='checkbox' value='[valeur]'
***id='id_parental_control_level_[increment]' display='[label]' /> [label]
",***

Ce template inclura un élément HTML de type « checkbox » à côté de chaque élément remonté par la requête.

Exemple du format de données retourné par les scripts Ajax

generic = new Object();
generic.label = "Orange";
generic.valeur = '13;
array_source[0] = generic;
generic = new Object();
generic.label = "SFR";
generic.valeur = '12;
array_source[1] = generic;

Le type de donnée retourné par le script Ajax doit être de type JavaScript. Avec la librairie jQuery il suffit d'indiquer la directive « ***dataType:'script'*** » au moment de l'appel du script.

Voici quelques exemples d'utilisation du plugin au sein de l'application.

Figure 17 - Utilisation basique du plugin jQuery

Cet exemple utilise le plugin avec un template relativement simple.

Figure 18 - Utilisation avancée du plugin jQuery

Dans cet exemple, nous pouvons voir l'utilisation d'un template spécifique prenant en compte la gestion d'un champ de type HTML checkbox.

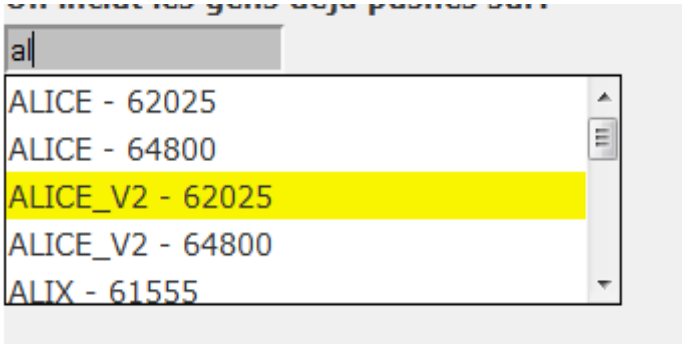

Figure 19 - Utilisation avancée avec habillage du plugin jQuery

Sur cet exemple, l'affichage du plugin est similaire à un sélecteur HTML.

Le plugin est très utilisé au sein du projet (plus de quarante fois). La création du plugin a finalement fait gagner du temps lors de la réalisation du projet.

<u>Utilisation dans d'autres projets:</u>

Satisfait du fonctionnement du plugin, nous l'avons mis en place dans d'autres projets. Ci-dessous, un exemple de l'utilisation de ce dernier dans une application de statistique.

Figure 20 - Utilisation du plugin jQuery dans d'autres projets

La mise en place du plugin dans ce projet s'est réalisée très rapidement. Nous envisageons maintenant de créer une libraire de plugins ou de fonctions JavaScript/jQuery qui sera utilisée par l'ensemble de nos applications.

4.4) Une interface ergonomique

4.4.1) Les widgets

Les utilisateurs finaux ont souhaité que l'interface de l'application soit ergonomique et intuitive. Ces derniers utiliseront très souvent l'application, il est donc important de ne pas les rebuter. Pour cela, j'ai utilisé plusieurs widgets JavaScripts afin d'atteindre cet objectif. En informatique, un widget désigne un composant graphique d'une interface.

Le treeview

Afin de pouvoir repérer aisément les différents filtres segmentant un groupe de clients, il était important de les regrouper par type. Nous pouvons répartir les filtres en deux grandes catégories. Tout d'abord, nous regroupons ceux concernant le parcours de consommation du client. Ensuite nous agrégeons les filtres détaillant le téléphone mobile de nos clients. Nous avons choisi d'afficher les filtres sous la forme d'un « treeview ». Cette arborescence est souvent utilisée pour l'affichage de dossiers et sous-dossiers. Elle est cependant assez rarement utilisée dans les applications web car ce n'est pas un élément natif défini par HTML. Pour sa mise en place nous devons utiliser un plugin jQuery et une feuille de style CSS afin d'afficher les éléments.

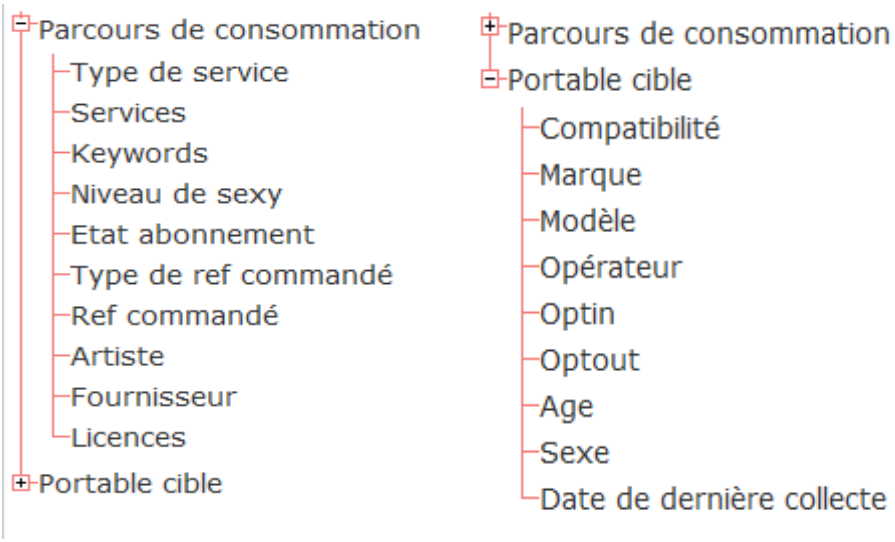

Figure 21 - Mise en place d'un treeview

Nous pouvons donc déplier ou replier la liste de filtres selon leurs types.

Le drag-and-drop

Un autre élément mis en place pour rendre l'application plus intuitive et la notion de « drag-and-drop » (ou de glisser-déposer). Grâce à ce widget nous pouvons sélectionner un filtre, le déplacer et le déposer dans un espace réservé de l'interface. Une fois cette opération réalisée, cet espace se transforme et propose la configuration du filtre choisi. Ci-dessous une présentation de l'opération pour le filtre « Opérateur ».

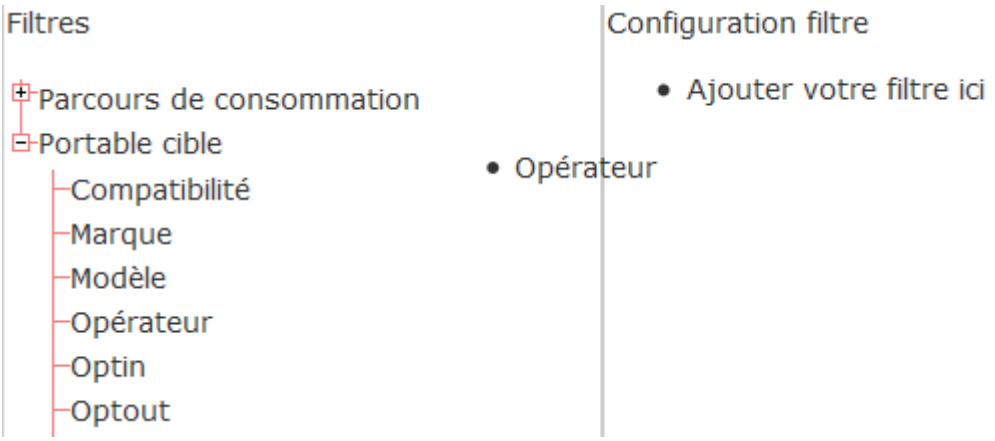

Figure 22 - Gestion du glisser-déposer

Une fois déposé dans l'espace réservé, ce dernier change et propose les options disponibles pour le filtre en cours :

Figure 23 - Présentation d'un filtre

L'utilisateur peut choisir les options souhaitées et valider ses modifications. Il voit ainsi le nombre de clients disponibles sur son segment en cours de création.

La gestion de groupe

Une notion importante lors de la création d'un segment concerne la gestion de groupe de filtre. Grâce à cette dernière nous pouvons segmenter une population ayant des critères différents. Par exemple, nous pourrons créer un segment ciblant note parc de client ayant pour opérateur mobile SFR et ceux ayant Bouygues Telecom et Apple pour marque de téléphone mobile. Les opérations entre les filtres d'un même groupe s'effectuent avec un « ET » logique. Les opérations entre les groupes de filtres sont exécutées avec un « OU » logique. Pour pouvoir répondre aux problèmes posés, nous devons utiliser des groupes.

Ci-dessous la représentation de cette segmentation au sein de l'interface avec une erreur. Tous les filtres sont intégrés dans le même groupe.

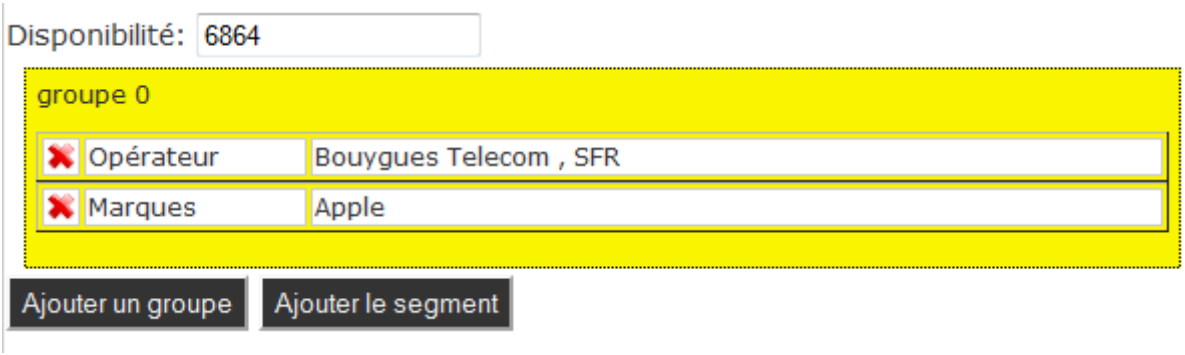

Figure 24 - Gestion de groupe unique

Ce segment ne répond pas au problème car il affiche tous nos clients ayant SFR ou Bouygues pour opérateur <u>et</u> ayant Apple pour marque mobile.

Figure 25 - Gestion de plusieurs groupes

La différence entre les deux segments, en termes de disponibilité de clients est flagrante. Le premier segment cible 6 864 clients alors que le deuxième vise 124 026 clients.

L'utilisation des groupes est très simple. Il suffit de cliquer sur le groupe sur lequel nous souhaitons travailler pour pouvoir le modifier. Une fois sélectionné ce dernier est en surbrillance. Pour ajouter un groupe il suffit de cliquer sur un bouton de l'interface.

Les repoussoirs

Lors de la création d'une campagne les utilisateurs peuvent sélectionner des anciennes campagnes. Ceci afin d'ajouter ou d'exclure de la campagne en cours de création, les clients qui ont déjà été relancées sur les campagnes choisies.

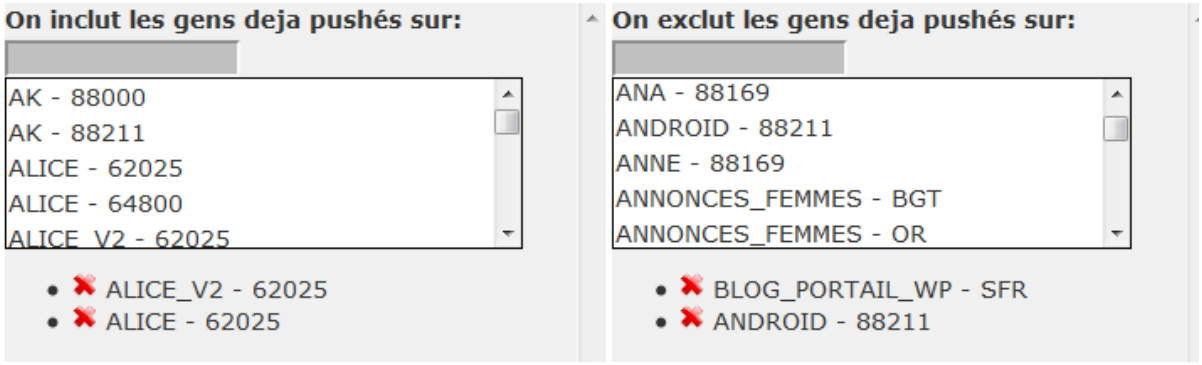

Figure 26 - Présentation des repoussoirs dans la création d'une campagne

Ainsi, dans cette situation, la relance créée avec cette campagne n'intègre pas les clients qui ont déjà été relancés sur les campagnes « BLOG_PORTAIL_WP – SFR » et « ANDROID – 88211 ». En revanche les clients relancés sur les campagnes « ALICE_V2 -62025 » et « ALICE – 62025 » sont inclus.

4.4.2) Sécurisation de l'interface

Pour la sécurisation de l'interface, un contrôle JavaScript sur tous les champs obligatoires a été mis en place. Dès lors qu'un champ obligatoire n'est pas renseigné l'utilisateur en est informé par le biais d'un message (par exemple lors de la validation d'un formulaire). Il peut alors compléter le formulaire afin de remplir les informations manquantes.

Des blocages sont aussi présents sur les champs ayant des contraintes particulières. Lors de la création d'une campagne relançant les clients par l'envoi d'un SMS, il est impossible d'envoyer un message de plus de 160 caractères. Un blocage JavaScript interdit l'écriture de messages avec plus de caractères. De plus, nous avons aussi mis en place une gestion des caractères non autorisés par la norme GSM. Ceci par l'intermédiaire d'un script appelé en Ajax vérifiant les caractères de la campagne.

Une fois que tous les champs du formulaire sont conformes, un pop-up apparaît à l'écran récapitulant les informations saisies par l'utilisateur. Il est alors en mesure de valider ses modifications ou de revenir au formulaire afin de le modifier.

Pour sécuriser la logique fonctionnelle de l'application, notamment au niveau des requêtes en base de données, un test de chaque variable importante est réalisé en amont. Ainsi, si l'on souhaite recevoir des variables de type numérique, on fait appel à la fonction PHP

« ***is_numeric*** ». Si la variable attendue ne peut pas être vide on appel la fonction « ***is_empty*** ». Dans le cas où la variable reçue ne satisfait pas la contrainte imposée, un log est écrit affichant le contenu des variables reçues ainsi que la fonction appelée. Nous pouvons alors rapidement détecter la provenance des erreurs et apporter les correctifs aux différents appels de fonctions.

Lorsque plusieurs tables se mettent à jour lors d'une même opération, des transactions SQL sont utilisées. Si une requête échoue, la base de données reste inchangée.

4.4.3) Optimisation de l'interface

Les premiers tests ont mis en évidence des lenteurs au niveau de l'affichage de certaines pages. Cela était lié aux requêtes sélectionnant les clients en fonction des différents filtres. Nous avons donc décidé de réaliser la plupart de ses requêtes à l'aide d'appel Ajax. De ce fait, le navigateur n'est pas bloqué et les utilisateurs peuvent toujours effectuer leurs modifications.

Les utilisateurs de l'application ont désormais l'habitude d'utiliser des applications web. Nous ne pouvons donc pas nous permettre de réaliser une application statique avec un rafraichissement de page à chaque clic. C'est pour cela que j'ai décidé de rendre l'application la plus dynamique possible. Ce fut d'ailleurs l'objet d'une demande explicite de l'expression des besoins. Le JavaScript combiné avec les technologies Ajax sont au cœur de l'application.

<u>Rappel sur l'Ajax :</u>

En informatique, et plus particulièrement en architecture informatique, Ajax (acronyme d'Asynchronous JavaScript and XML) est une méthode de construction des applications web et des sites web dynamiques basés sur diverses technologies web ajoutées aux navigateurs dès 1995.

Ajax est la combinaison de technologies telles que JavaScript, CSS, XML, le DOM et le XMLHttpRequest dans le but de réaliser des applications web qui offrent un meilleur confort d'utilisation.

L'apport principal de l'Ajax est de pouvoir modifier le contenu d'une page après l'affichage initiale de celle-ci. Sans cette technologie la navigation est moins fluide et on passe obligatoirement par un rafraîchissement complet de la page même quand celui-ci n'est pas nécessaire. Cela est visible lors de la lecture d'une vidéo possédant des commentaires affichés à l'aide d'une pagination. Si lors du clic sur un élément la pagination, la page se rafraîchit en relançant la vidéo du début, les utilisateurs seraient gênés par cette mauvaise ergonomie.

4.5) Talend Open Studio

4.5.1) Présentation de Talend Open Studio

Talend Open Studio est un logiciel proposant plusieurs solutions pour l'intégration de données au sein des entreprises. Voici des exemples de cas d'utilisation :

- **Intégration de données opérationnelle :** dans la plupart des entreprises, l'intégration de données opérationnelle est gérée par la mise en œuvre de programmes ou routines personnalisés, réalisés à la demande pour répondre à un besoin spécifique. La migration et le chargement des données ainsi que la synchronisation et la réplication des données sont les tâches d'intégration opérationnelle les plus fréquentes.

- **Migration de données :** lors de la mise à niveau d'une nouvelle version d'une base de données ou d'une application, ou lors du passage à un nouveau système, les données doivent être préservées dans le nouveau système. L'objectif de la migration de données est de transférer les données existantes dans le nouvel environnement. Elles doivent être converties dans un format adapté au nouveau système, tout en préservant les informations présentes dans l'ancien système.

- **Synchronisation de données :** dans le système d'information, il est très fréquent que les données soient gérées séparément par plusieurs applications ou bases de données. Pourtant, la cohérence des données doit être préservée entre ces systèmes. Le besoin de synchronisation des données peut être permanent (synchronisation entre des systèmes opérationnels) ou temporaire, par exemple lors d'une migration. La synchronisation de données inclut tous les processus qui assurent la synchronisation des données entre les applications et les bases de données.

- **ETL pour le décisionnel et stockage des données :** Les processus ETL (extraction, transformation et chargement) sont les composants les plus critiques - et à valeur ajoutée - d'une infrastructure décisionnelle. Bien qu'ils soient quasiment invisibles pour l'utilisateur de la plateforme d'informatique décisionnelle, les processus ETL récupèrent les données des systèmes opérationnels et les prétraitent pour les outils d'analyse et de reporting.

Plusieurs grandes entreprises utilisent les solutions apportées par Talend Open Studio afin de répondre à leurs problématiques. Ainsi « Alianz Global Investors Distributors », « Sony Online Entertainment » et « Monolith » utilisent Talend Open Studio.

4.5.2) Utilisation de Talend Open Studio dans IRC

Pour l'importation de données, nous avons décidé de réaliser un job avec Talend Open Studio. Ce choix a été fait pour une raison précise. Les sources de données étant hétérogènes, nous aurions dû utiliser plusieurs scripts d'importation de données. La maintenance applicative et les modifications associées seraient peu évidentes.

La solution d'intégration de données opérationnelles proposées par Talend Open Studio semble parfaitement convenir aux besoins de notre application. Ne connaissant que très peu cet outil, il est intéressant de l'utiliser afin de voir si son utilisation pourrait se généraliser au sein de l'entreprise.

La collecte des données de nos différents services s'effectue via l'interface graphique de Talend Open Studio. Plus particulièrement les alias et les MSISDN afin de les restituer dans la base de données de l'application IRC.

Une fois nos flux de données schématisées dans l'interface graphique, nous pouvons générer les jobs associés. Talend Open Studio se charge de créer le code correspondant à ces jobs (le langage informatique utilisé pour la génération de ce code est le Java). Ensuite, nous pouvons les exporter et les exécuter sous forme de tâches planifiées. En effet, les jobs exportés sont exécutables sous forme de scripts shell. Ce sont des fichiers contenant plusieurs commandes lancées séquentiellement.

Les jobs Talend Open Studio à réaliser doivent remplir plusieurs objectifs. Voici les principaux :

- Collecter les alias provenant de l'application API_MOBILE et des différentes bases de données de services.

- Récupérer les MSISDN provenant de fichiers générés périodiquement.

- Récupérer les produits téléchargés par nos clients.

- Récupérer les informations des différents téléphones mobiles avec leurs caractéristiques (compatibilité Wap Push, gestion HTML, WML).

Ci-dessous la présentation d'un job sous Talend Open studio.

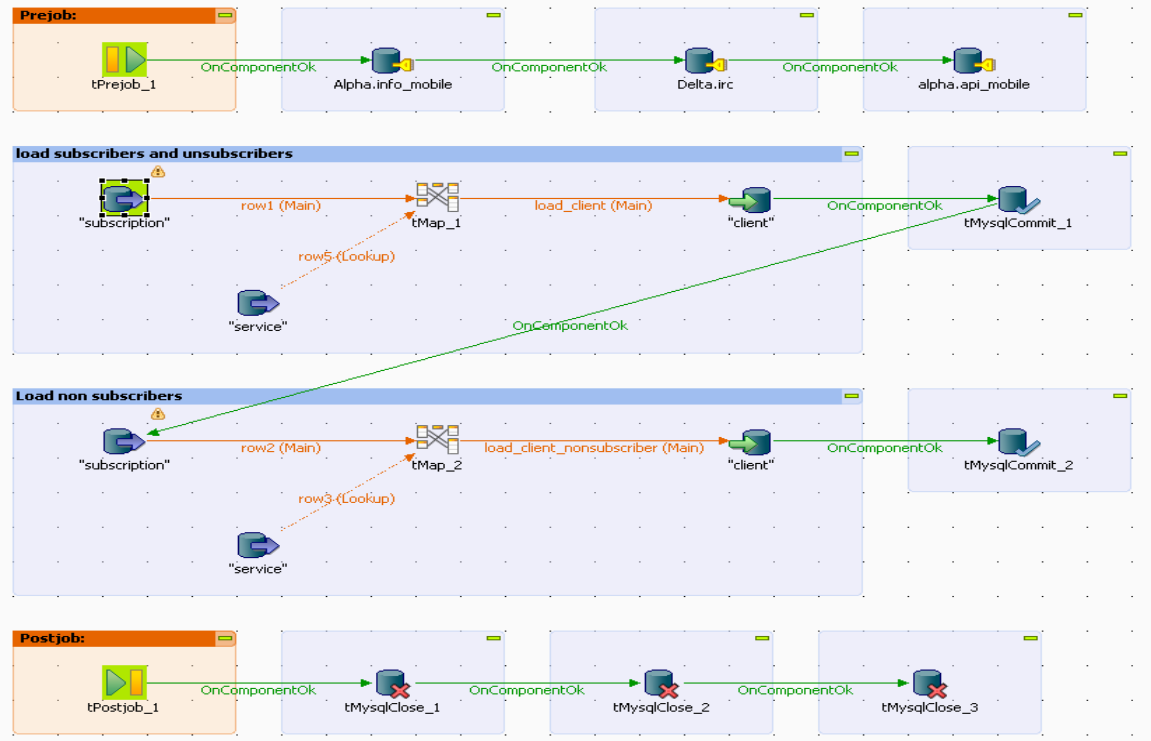

Figure 27 - Réalisation d'un job sous Talend Open Studio

Sur ce schéma, le job commence par l'ouverture des connexions aux différentes bases de données. Les bases de données de services et celle de l'application IRC sont concernées. Nous récupérons au sein de l'application IRC les données des tables des services après les avoir formatées par l'intermédiaire du composant « tMap ». Ce dernier lie les champs des tables sources aux champs des tables de l'application. Il reste ensuite à effectuer un « commit » MySQL afin de valider la transaction. Une fois ces opérations terminées, il suffit de fermer les différentes connexions ouvertes préalablement.

Un job principal sera utilisé afin de coordonner le lancement de tous les jobs d'importation de données.

L'importation réalisée pas les différents jobs a ajouté plus de 500 000 profils clients. 7 000 téléphones mobiles ont été incorporés et répartis en 77 marques différentes. Au niveau des références téléchargées par nos clients nous en comptabilisons plus de 500 000. Les références représentent l'identifiant unique des contenus proposés en téléchargement.

L'utilisation de Talend Open Studio lors de ce projet fut bénéfique. Ceci du point de vue du gain de temps et de la maintenance applicative. L'utilisation de ce logiciel est en train de se généraliser au sein de l'entreprise. En effet, avec la mise en relation des différentes plateformes techniques de Paris (Ex-Mediaplazza) et Marseille, des migrations et des synchronisations de données sont nécessaires. L'outil Talend Open Studio a été retenu pour effectuer ses opérations.

4.6) Le cloud computing

4.6.1) Présentation

Le concept du cloud computing modifie radicalement l'utilisation de l'informatique. En effet, ce dernier met à disposition, par le biais d'entreprises spécialisées, une puissance de calcul et de stockage. Les entreprises n'ont plus à se préoccuper de l'infrastructure nécessaire assurant le fonctionnement de leurs applications. Cette information est à nuancer car dans certains modèles de cloud, l'entreprise garde la charge de son infrastructure. D'un point de vue économique, seul l'utilisation réelle des ressources est facturée.

Le cloud computing peut se mettre en place à l'aide de plusieurs techniques. Nous pouvons en distinguer quatre :

Clouds privés : l'infrastructure est utilisée uniquement pour une entreprise. Elle peut être interne ou externe (gérée par l'entreprise ou un tiers).

Clouds communautaires : l'infrastructure est partagée par plusieurs entreprises et s'appuie sur des intérêts communs (une mission, des exigences en matière de sécurité, une politique de respect des normes). Comme pour les clouds privés elle peut être interne ou externe.

Clouds publics : l'infrastructure est mise à la disposition de plusieurs entreprises différentes.

Clouds hybrides : l'infrastructure est la composition de deux ou plusieurs clouds (privés, communautaires ou publics) liées par une technologie autorisant la portabilité des données et des applications.

Il existe plusieurs modèles technologiques de cloud computing. Ces derniers rendent disponible à la demande les différentes couches pour la réalisation d'un service entre un client et un fournisseur. Cela se décompose de la façon suivante. La couche applicative en contact direct avec le client. La plateforme exécutant les applications et l'infrastructure supportant cette dernière. Ci-dessous leur équivalent en mode cloud :

- Les logiciels fournis sous forme de service ou Saas,
- Les plateformes applicatives ou Paas,
- Les serveurs virtuels ou Iaas.

Rappel sur le SaaS:

Les applications Saas (Software as a service) ou logiciels hébergés sont un nouveau modèle technologique lié au cloud computing. Ce modèle propose à ses clients des applications accessibles à distance via internet. Il n'est donc plus nécessaire d'installer des applications sur les postes des utilisateurs. Le Saas est assez répandu en entreprise car il se rapproche de l'ASP qui s'est développé en entreprise à la fin des années 1990. Une différence entre ces deux modèles repose sur l'aspect économique. En utilisant l'ASP, nous sommes soumis à des versements mensuels alors qu'en Saas, seul l'utilisation réelle du service est facturée.

4.6.2) IRC, une application de type SaaS?

Ci-dessous, un schéma présentant les différents modèles technologiques disponibles dans le cadre du cloud computing. Nous pouvons y voir le rôle de l'entreprise et du fournisseur Cloud en fonction des différents modèles :

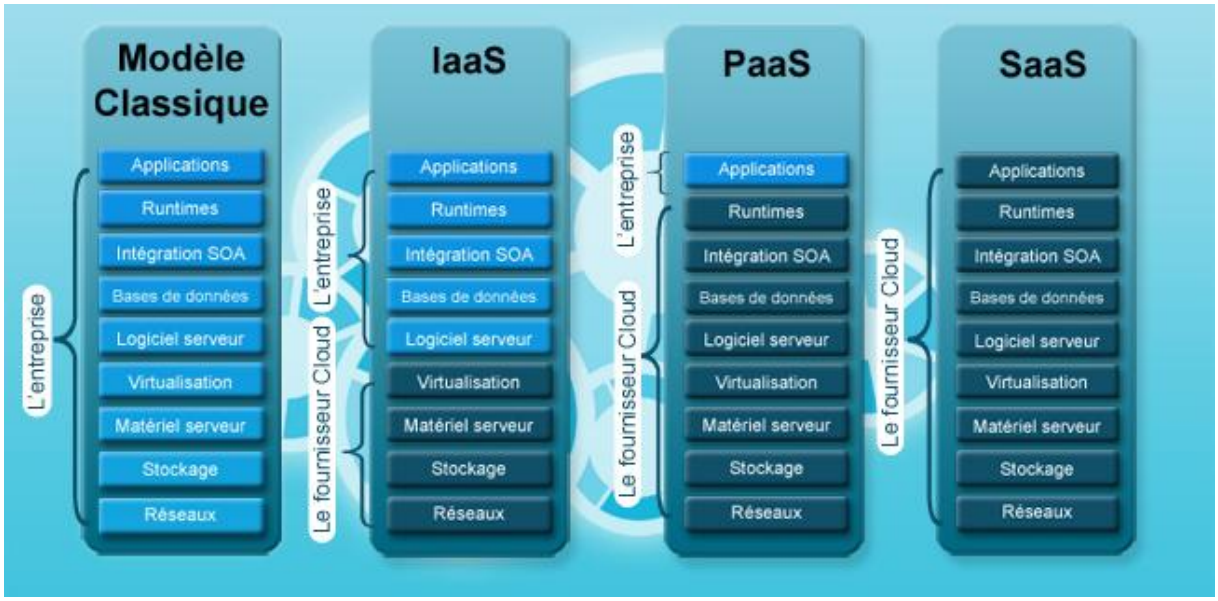

Figure 28 - Les différents modèles de cloud computing

En considérant notre infrastructure technique comme fournisseur cloud on constate que l'application IRC est de type SaaS.

4.6.3) Evolution du SaaS dans le monde de l'entreprise

Les logiciels sous forme de service est le modèle technologique du cloud computing le plus utilisé. Par exemple, les applications telles que Gmail et Google Docs sont connues du grand public. Ce modèle va donc continuer de croître et accroître son influence au sein des entreprises. Il va être développé afin de disposer de services capables de couvrir l'ensemble des métiers de l'entreprise. Ce dernier servira de locomotive aux autres modèles technologiques du cloud computing.

4.7) Utilité d'un développement HTML5

De plus en plus de développeurs utilisent HTML5 comme langage de balisage de leurs différents sites ou applications web. Il serait intéressant de voir quels bénéfices apportent cette version des spécifications HTML au sein de l'interface de l'application IRC.

4.7.1) Présentation

HTML5 est la prochaine révision majeure du langage HTML (le format de données conçu pour représenter les pages web). Les spécifications sont toujours en cours de développement et le W3C (le World Wide Web Consortium est un organisme de normalisation fondé en 1994) envisage la clôture des ajouts de fonctionnalités en 2014. Cependant, il est conseillé aux développeurs web d'utiliser HTML5 dès maintenant. Cette spécification ajoute de nouveaux éléments, de nouveaux attributs et des API de développement. Une API est une interface de programmation facilitant la communication entre deux applications. Ces API peuvent être utilisées dans les situations suivantes :

- Générer des dessins en 2D à l'aide de la balise « canvas ».
- Jouer des vidéos et des sons avec les balises « vidéo » et « audio ».
- Réaliser des applications hors-lignes.
- Edition de contenus avec l'attribut « contenteditable ».
- Gestion du glisser-déposer grâce à l'attribut « draggable ».
- Accéder à l'historique des pages consultées par l'utilisateur.

HTML5 veut rendre prépondérant le rôle des feuilles de style CSS dans le cadre de la représentation des pages web. De ce fait, certaines balises HTML ont été supprimées. C'est le cas de la balise « *center* » assurant l'alignement horizontal d'un texte et de la balise « *u* » soulignant une information. Certains attributs deviennent obsolètes. Par exemple, lors de l'ajout d'un script JavaScript dans un document HTML, il ne sera plus nécessaire de préciser le type de ce dernier. L'attribut « *src* » suffit à l'inclusion.

4.7.2) Apport du HTML5 dans IRC

Avec les différents apports de HTML5, il est facile de voir où son utilisation peut être pertinente au sein de l'application IRC.

Tout d'abord, pour indiquer au navigateur que nous utilisons un document de type HTML5, il faut spécifier la balise « doctype » au début de chaque page de l'application. Pour cela il suffit d'écrire la chaîne de caractères suivante :

<!DOCTYPE html>

Une fois le doctype déclaré nous pouvons utiliser les nouvelles balises et attributs disponibles. La balise « *nav* » peut être utilisée pour placer les liens de navigations vers les différents modules de l'application.

HTML5 apporte un contrôle sur les champs de formulaire de notre choix. Il n'y a plus à gérer la sécurité des formulaires par l'intermédiaire de scripts JavaScript. Cela peut être utile au sein de l'application IRC car nous contrôlons plusieurs champs de formulaire avec des fonctions JavaScript.

Nous pouvons indiquer des patrons concernant le texte admis pour certains champs de formulaire par le biais d'expression régulière. Les noms des campagnes étant formatés, l'utilisation d'un patron semble être adaptée.

Afin d'éviter le problème du format des dates entre la présentation à l'utilisateur et l'insertion en base de données, tous ces champs ne sont pas éditables et proposent un calendrier où il suffit de choisir le jour voulu. Un système de slider à été mis en place afin de choisir les heures. Un slider est un composant graphique choisissant une valeur numérique en déplaçant un curseur sur une échelle graduée. Ce calendrier et ce slider sont disponibles sous la forme de plugins jQuery. HTML5 apporte de nouveaux champs de formulaire de type date et heure réglant ce problème. Dans cette situation, l'utilisation de plugins jQuery est donc inutile.

HTML5 propose la mise en cache de données. Nous pouvons mettre en cache les données nécessaires à la création de campagne et de segments afin de les rendre disponibles en mode non connecté. Une synchronisation peut prendre en compte les modifications réalisées en mode non connecté et les impacter sur la base de données de production. Néanmoins, les navigateurs actuels n'utilisent pas la même méthode pour gérer la mise en cache de données. En effet, certains navigateurs proposent les bases de données de type web SQL alors que d'autres disposent uniquement du local storage. Le web SQL autorise la création de tables et la réalisation de requêtes (de type SQL) afin d'alimenter ou modifier ces dernières. Le local storage propose un simple tableau associatif.

HTML5 peut donc être intégré au sein de l'application IRC. Cependant, il est important de noter que ces attributs ou balise ne sont pas compatibles avec tous les navigateurs. N'ayant pas de politique précise d'utilisation de navigateurs pour les équipes commerciales, nous voulons rendre l'application utilisable par les trois navigateurs principaux. Ainsi IRC est fonctionnel avec Internet Explorer, Mozilla Firefox et Google Chrome. Pour utiliser les nouveaux composants apportés par HTML5, il faut donc vérifier si le navigateur utilisé est compatible avec le composant voulu. Il faudra ensuite réaliser le comportement adéquat en fonction du navigateur. Cela aura pour effet de doubler le travail dans certaines circonstances. L'utilisation de HTML5 au sein de l'application IRC est par conséquent inutile. Une fois que tous les navigateurs principaux prendront en charge la majorité des nouveaux composants apportés, nous pourrons envisager de faire migrer l'application vers les spécifications HTML5.

4.8) Réalisation de statistiques sous QlickView

QlikView
www.qlikview.com

QlickView est l'application générant les rapports de statistiques pour la plupart des applications de notre entreprise. QlickView est un logiciel d'informatique décisionnelle. C'est un outil d'aide à la décision par le fait de donner au responsable concerné une vue d'ensemble de l'activité traitée. Grâce à ce dernier nous obtenons des analyses, reportings et des tableaux de bord intuitifs sur les résultats de nos différents services.

Avec ses bases vectorielles nous pouvons manipuler de grands volumes de données et obtenir les résultats rapidement. Le principe des bases de données vectorielles consiste à extraire les données des tables et les numérotées. Les tables ne contiennent plus les informations mais uniquement leurs numéros. Le principal avantage est d'éviter de stocker plusieurs fois la même valeur.

Ci-dessous un exemple où l'on voit l'avantage des bases vectorielles pour les valeurs « Médecin généraliste » et « Lille ».

Base SQL

Table

Yann	DALAIN	Marseille	Médecin généraliste
Catherine	BAZIOU	La Rochelle	Rhumatologue
Jean-François	ALBERG	Lille	Médecin généraliste
Christian	SOUHAT	Lille	Ophtalmologiste
Nathalie	BAUTRET	Brest	Podologue
Floriane	MARTINO	Paris	Médecin généraliste

Base vectorielle

Table

21	7	12	14
5	2	10	19
9	1	11	14
6	20	11	16
15	3	4	18
8	13	17	14

Données

1	ALBERG
2	BAZIOU
3	BAUTRET
4	Brest
5	Catherine
6	Christian
7	DALAIN
8	Floriane
9	Jean-François
10	La Rochelle
11	Lille
12	Marseille
13	MARTINO
14	Médecin généraliste
15	Nathalie
16	Ophtalmologiste
17	Paris
18	Podologue
19	Rhumatologue
20	SOUHAT
21	Yann

Figure 29 - Comparaison base SQL vs base vectorielle

Depuis sa mise en place, il n'y a plus eu à réaliser d'interfaces de statistiques.

La décision de confier la réalisation des statistiques de l'application IRC à cette application fut rapidement prise. Ainsi les différentes informations concernant l'envoi des relances sont disponibles via des rapports générés par QlickView.

4.9) Intégration à API_MOBILE

4.9.1) Présentation de l'application API_MOBILE

API_MOBILE est l'application qui gère l'ensemble des transactions de type SMS+ de l'entreprise. Elle couvre toutes les opérations menant à l'abonnement d'un client. De plus, avec cette application on peut envoyer des SMS et des Wap Push. C'est pour cette raison que nous devons l'interconnecter avec l'application IRC pour l'envoi des relances.

Pour chaque numéro court (ou short code) que l'on gère, notre facilitateur (Netsize) nous donne accès à des files d'attentes contenant les messages envoyés par les clients. En fonction du mot clef reçu et du numéro court, la réponse est ajoutée dans la file d'envoi à l'attention du client via son alias. Dans le cas d'une relance provenant de l'application IRC, il suffit d'ajouter le message de la campagne dans la file d'envoi et de le destiner aux clients choisis.

Le facilitateur fournit une API en Java afin de communiquer avec leurs différentes applications.

Fonctionnement général

Il existe trois flux de communication entre notre plateforme et celle du facilitateur :

- Flux d'envoi : gère les SMS que nous envoyons vers l'utilisateur final (MT, Mobile Terminated).

- Flux de réception : gère la récupération par notre plateforme des messages envoyés par les utilisateurs vers nos services (MO, Mobile Originated).

- Flux de réception des SR (Status Report).

Chaque flux de communication est géré par un ou plusieurs processus Java, l'API Netsize étant codée avec ce langage de programmation. Pour chaque flux, il y a un ensemble de numéros courts à gérer.

Implémentation technique des flux

A noter que pour avoir accès aux processus actuellement en cours d'exécution sur une machine, il est pratique d'utiliser la commande « jps ».

Par exemple :

```
root@api-mobile:~# jps
21042 PersistentGetSMS88077
27427 PersistentGetSMS88066
21178 PersistentGetSMS88555
6655 Jps
27566 PersistentGetSMS88211
11860 PersistentSendBulkSMS
28199 PersistentGetSR88077
```

Flux d'envoi

```
root@api-mobile:~# jps |grep "PersistentSendBulkSMS"
11860 PersistentSendBulkSMS
```

Le flux boucle sur les numéros courts d'envoi, vérifiant pour chacun la présence dans la file d'envoi de messages arrivés à échéance. En effet, les messages sont insérés dans la file avec une date d'envoi effective. Une fois envoyés, les messages sont supprimés de la file et consignés dans une table de la base de données de l'application.

```
root@api-mobile:~# jps |grep "GetSMS"
21042 PersistentGetSMS88077
3019 PersistentGetSMS
27427 PersistentGetSMS88066
21178 PersistentGetSMS88555
27566 PersistentGetSMS88211
14447 ThreadedGetSMS
```

La récupération des MO (message envoyé par le client) est la tâche qui demande la plus grande réactivité. Chaque fois qu'un nouveau MO est récupéré, il est placé dans une table temporaire puis traité par un script PHP qui détermine la réponse à apporter et place éventuellement le message de réponse dans la file d'envoi.

Flux de réception des SR (Status Report)

```
root@api-mobile:~# jps |grep "PersistentGetSR"
28199 PersistentGetSR
```

Les statuts report contiennent les informations sur la réception, des messages que nous envoyons. Le processus récupère pour chaque short code tous les status report disponibles. Il passe au suivant et repasse au premier lorsque tous les numéros courts ont été traités. Le processus boucle à l'infini sur cette séquence.

4.9.2) Problèmes posés lors de l'intégration

Lors du lancement de l'application IRC, l'envoi en masse de relances à nos clients à poser quelques problèmes. Nous sommes limités par notre facilitateur au niveau du débit d'envoi de messages. Avec la mise en place de notre application, nous arrivons rapidement à un goulet d'étranglement. En effet, l'application IRC pourra générer plus de 10 000 envois de messages par jour.

L'envoi de rebill et retry (représentant généralement plus de 1 000 messages) ajouté aux envois des relances clients ralentissent considérablement la cinématique d'abonnement. Les rebills facturent un client préalablement abonné à un de nos services. Ils sont tentés tous les 8 jours et débitent le client du montant de son abonnement. Cependant, certains rebills échouent (si le mobile du client est éteint ou que ce dernier ne dispose pas d'assez de crédits). Il faut alors faire un retry les jours suivants. Tant que le client n'est pas facturé, nous tentons donc de le refacturer. Ce ralentissement dans la cinématique d'abonnement a entraîné une chute du nombre de nouveaux abonnements. Les clients doivent parfois patienter plus d'une minute avant de recevoir le message leur indiquant les étapes à suivre pour procéder à leur abonnement.

La solution mise en place consiste à espacer l'envoi des rebills et retry. En effet, le délai de livraison de ces derniers est peu important. Il a été décidé de les espacer tout au long de la journée afin de laisser la file d'envoi disponible pour les messages importants. Pour les relances, un système similaire a été mis en place. La seule différence provient du fait que la plage d'envoi est plus restreinte. Les contraintes commerciales nous obligent à envoyer les relances de 9 heures à 17 heures. Il serait contre productif d'importuner nos clients tard le soir avec des messages publicitaires.

Afin d'améliorer la réactivité du système, une logique de priorité de messages a été mise en place. Les envois de messages concernant l'abonnement de clients sont marqués en priorité haute, les envois générés par IRC sont en priorité moyenne alors que les envois des rebills et retry sont en priorité basse.

4.9.3) Phase de test

La phase de test a été réalisée en ciblant les relances sur les téléphones de l'entreprise. Cela enfin de détecter d'éventuels bogues avant de cibler des clients réels. Une fois ces relances de tests passées, nous en avons réalisées sur notre parc de clients. Le volume de clients ciblés par ces relances était faible puis a évolué au cours des différentes relances. Après la validation de ces relances par l'équipe projet, l'équipe commerciale a envoyé des relances ciblant un volume de clients plus conséquent.

5) Résultats et évolutions

5.1) Résultats

Après quelques mois d'utilisation, l'application donne entière satisfaction à ses utilisateurs. Voici par exemple, une des relances qui est envoyée à nos clients.

Configuration de la campagne :

Tableau 5 – Présentation d'une campagne

Pays	France
Type de profil	Alias
Nom de la campagne	APPLIGAG - 88000
Type de message	SMS texte
Message	Mobi >TOP APPLIS scanner deshabillant, detecteur de canons ou de fantomes! Fais halluciner tes potes Envoie APPLIGAG par sms au 88000 Abo 3eur/semaine nopub:stop

C'est donc une relance qui est envoyée via des messages SMS sur des profils clients de type alias.

Le segment choisit s'appelle « 88000_SOFT_NONABO_TOUS ». Voici sa définition :

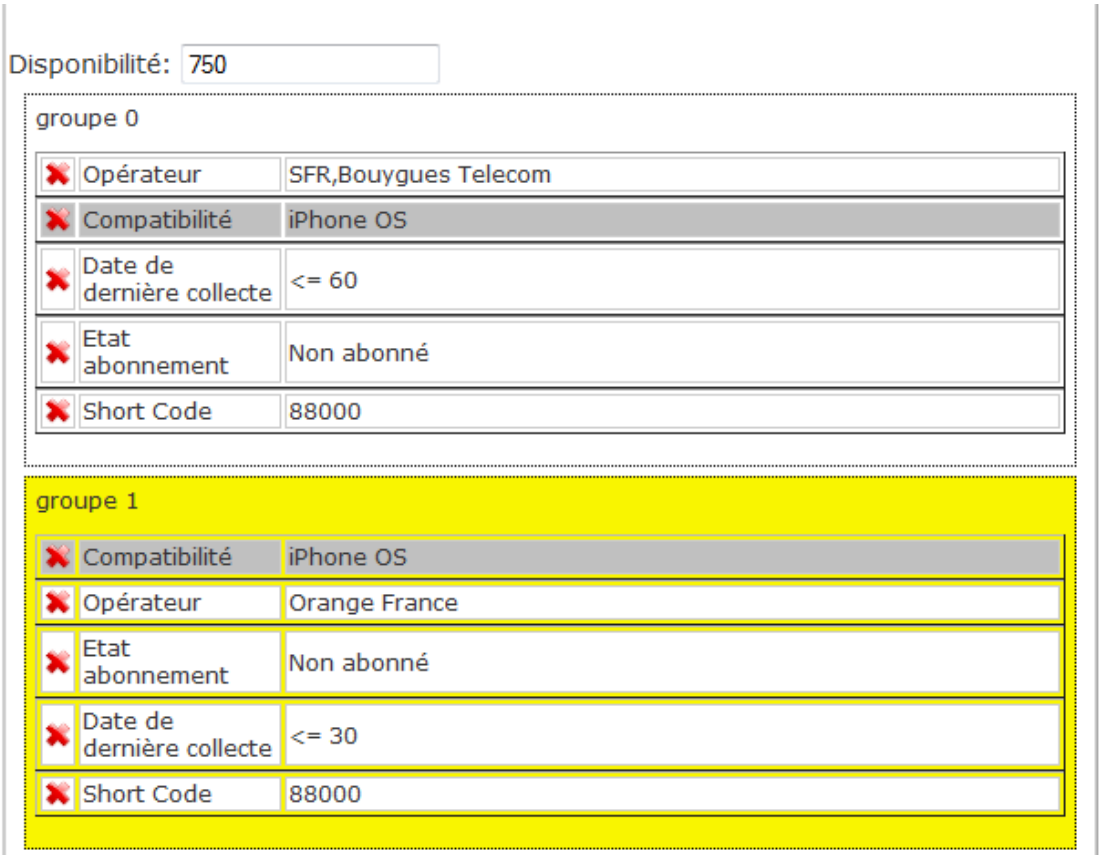

Figure 30 - Présentation d'un segment complet

Ce dernier récupère tous les clients qui ne sont plus abonnés à nos services ayant un téléphone avec un système d'exploitation différent de « iPhone OS ». Pour SFR et Bouygues Télécom,

on récupère les clients qui n'ont plus envoyé ou reçu de messages depuis plus de 60 jours alors que pour Orange 30 jours suffit à intégrer le segment.

La campagne et le segment sont configurés. Il faut maintenant mettre en relation ces derniers afin de planifier les relances par l'intermédiaire du module de requête.

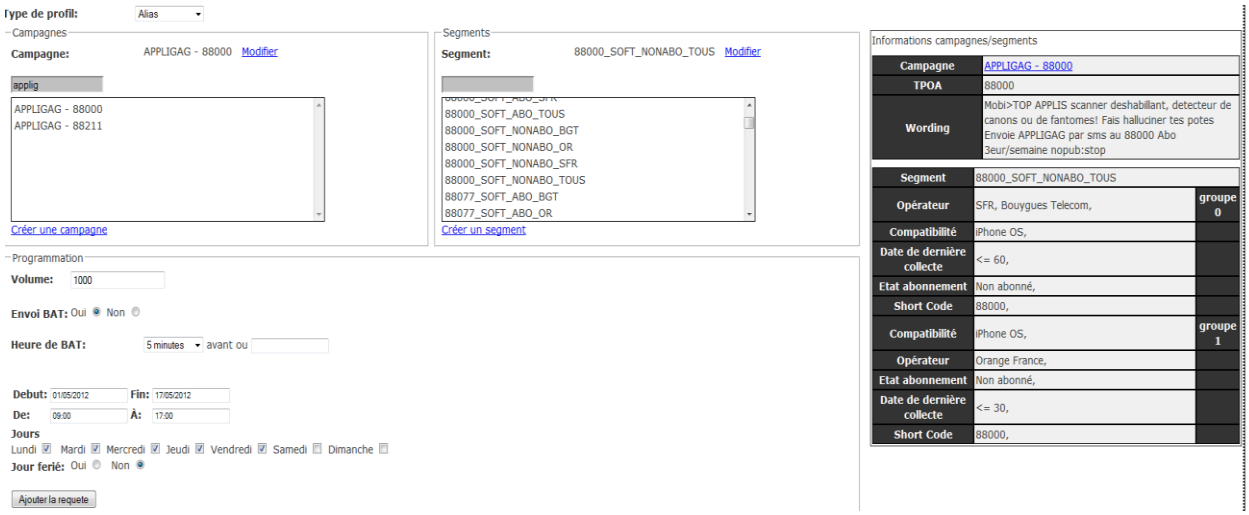

Figure 31 - Création d'une requête

Dans ce module nous choisissons la campagne et le segment à traiter. Il faut ensuite choisir le volume de clients à cibler par jour, 1 000 dans notre exemple. L'envoi d'un bon à tirer est programmé cinq minutes avant le début de la relance. Il reste ensuite à renseigner les dates et heures de début et de fin de relance. Choisir les jours de la semaine où la relance s'exécutera et si cette dernière doit prendre en compte les jours fériés. Dans notre cas la relance s'effectue du 01/05/2012 au 17/05/2012 du lundi au vendredi excepté les jours fériés. La relance est exécutée de 09:00 à 17:00.

Ce module présente succinctement les configurations de la campagne et du segment sélectionnés. Cela afin de détecter un éventuel problème et d'être sur que nous programmons la relance sur les bons éléments.

Une fois la requête ajoutée, il faut la confirmer afin que la relance s'effectue. Il suffit pour cela de modifier l'état de la relance du jour. Les clients recevront alors le message publicitaire sur leur téléphone mobile. Sur l'interface, il suffit de cliquer sur bouton pour confirmer les relances du jour. Les utilisateurs sont informés de l'état de la relance par une coloration spécifique sur le planning.

Semaine **19** - **2012**, du: 7 Mai - 13 Mai 2012 Actualiser les disponnibilités

Campagne	Segment	Type de Profil	Disponibilité	Lun 07 Mai	Mar 08 Mai	Mer 09 Mai	Jeu 10 Mai	Ven 11 Mai	Sam 12 Mai	Dim 13 Mai
APPLIGAG - 88000	88000_SOFT_ABO_TOUS	Alias	Actualiser	09:00-17:00 Volume: 1000		09:00-17:00 Volume: 1000	09:00-17:00 Volume: 1000	09:00-17:00 Volume: 1000		
	88000_SOFT_NONABO_TOUS	Alias	Actualiser	09:00-17:00 Volume: 1000		09:00-17:00 Volume: 1000	09:00-17:00 Volume: 1000	09:00-17:00 Volume: 1000		
	88000_SOFT_ABO_BGT	Alias	Actualiser	09:00-17:00 Volume: 1000		09:00-17:00 Volume: 1000	09:00-17:00 Volume: 1000	09:00-17:00 Volume: 1000		

Figure 32 - Validation des relances

Le fond orange indique que la relance est à confirmer. Le fond vert indique que cette relance est confirmée et le fond gris indique aux utilisateurs que les relances ont déjà été envoyées.

Une fois la relance débutée, les utilisateurs constatent en temps réel l'évolution de cette dernière. Ils ont la possibilité de la mettre en pause en cas de problème et de la réactiver plus tard.

Campagne	Segment	Date	Heure début	Heure fin	Broker	Avancement	Gestion
APPLIGAG - 88000	88000_SOFT_NONABO_TOUS	2012-05-10	09:00	17:00	NETSIZE	332/356 (93.26%)	

Figure 33 - Gestion des requêtes

L'application est très utilisée car elle envoie jusqu'à 15 000 messages publicitaires par jour. Jusqu'à présent, il ya eu 1 000 campagnes créées par l'équipe commerciale. Nous comptabilisons 112 segments. En quelques mois l'application IRC a géré l'envoi de plus de 1 700 relances générant plus d'un million de messages.

Le bilan est donc positif. L'application fait économiser plus de 5 000 euros par mois à l'entreprise. En effet, l'utilisation de l'ancienne application était facturée par notre prestataire.

5.2) Futur de l'application IRC

5.2.1) Evolution du produit

L'une des évolutions majeures de l'application concerne la gestion de plusieurs facilitateurs. L'équipe commerciale voudrait être en mesure de gérer le facilitateur associé à une relance. Ce n'était pas prévu au moment de la conception de la base de données. Cela entraînera certainement l'ajout de nouvelles tables au sein de celles-ci.

D'un point de vu de l'utilisation, l'application va monter en charge et gérer de plus en plus de relances. Le fait de récolter de plus en plus d'alias sur nos différents services (avec à la mise en place de l'internet+ mobile) et la fusion avec Mediaplazza conduisent à ce constat.

5.2.2) Déploiement à d'autres clients

L'application sera bientôt accessible à d'autres clients. Avec les différents partenariats que nous contractons, nous récoltons des abonnés que nous ne pouvons pas ciblés dans nos relances. L'application IRC sera donc accessible à nos clients afin qu'ils puissent relancer leurs clients.

6) Conclusion

D'un point de vu personnel, la réalisation de ce projet fut bénéfique. Tout d'abord, j'ai été nommé chef de projet. J'étais jusqu'a présent développeur NTIC. J'ai pu voir toutes les responsabilités inhérentes à ce rôle dans le cadre de la réalisation d'un projet. C'est un rôle qui est totalement différent de celui de développeur. Ce poste nécessite une connaissance bien plus transversale des différents métiers de l'entreprise. Néanmoins, j'ai développé certains modules de l'application.

En qualité de chef de projet, nous devons organiser des réunions afin de cadrer le projet et de bien cerner les besoins des équipes commerciales. Il faut ensuite réussir à planifier le projet en fonction des ressources qui sont allouées à celui-ci. La planification du projet était peu évidente car il faut bien garder à l'esprit que les ressources n'étaient pas totalement disponibles à sa réalisation. Ces dernières assurent également des travaux récurrents ou de la maintenance applicative. En outre, nous devons présenter le projet de manière claire et concise afin que l'équipe développement en comprenne les principaux objectifs.

Une fois ces phases préparatoires achevées, nous entrons dans les phases de réalisation du projet. Il faut alors suivre son avancé par rapport aux dates prévues lors de la planification. Il est important de faire des réunions avec l'équipe de développement afin d'identifier rapidement les points bloquants. C'est un point à ne pas négliger car en discutant ensemble des problèmes, nous trouvons généralement, rapidement les réponses à apporter.

En utilisant la méthode agile SCRUM, nous avons dû livrer l'application suite aux développements réalisés lors des sprints. Les utilisateurs finaux sont donc impliqués dès le début des développements. Cela évite de commettre des erreurs de compréhension et de réadapter l'application aux besoins de ces derniers. De plus, les utilisateurs nous communiquaient une liste de bogues que nous devons prioriser. Après avoir vu SCRUM en théorie via les cours du CNAM, il était assez intéressant de la mettre en pratique au sein de mon entreprise. Ce fut une expérience positive et la direction informatique envisage de l'utiliser dans le cadre de la réalisation d'autres projets.

J'ai pu développer mes compétences en jQuery grâce à la réalisation d'un plugin. C'est un point important car jQuery tend à s'imposer comme une librairie standard dans le monde du développement web. Par exemple, il est devenu courant que la connaissance de jQuery soit un élément déterminant lors de l'embauche d'un développeur web.

L'utilisation de Talend Open Studio fut aussi très importante au cours de ce projet. Ce dernier nous a évité la réalisation de plusieurs scripts d'import de données et par conséquent des risques d'erreurs potentiels. Désormais, ce logiciel est utilisé dans d'autres projets afin de migrer ou synchroniser des données de différentes plateformes.

Le projet en production, il faut valider son fonctionnement. Pour cela il faut tester les différents éléments de l'application, afin de détecter ceux nécessitant des optimisations. L'application sera alors plus confortable à l'utilisation.

La réalisation de ce projet m'a fait comprendre les différents besoins auxquels sont confrontées les équipes commerciales dans le cadre de la relance client.

Aussi, j'ai acquis une meilleure connaissance des impératifs à assurer lors de la réalisation de projets.

7) Bibliographie

7.1) Sites internet

Tutoriels HTML5
URL: *http://www.w3schools.com/html5/default.asp*

Cloud computing encyclopedia article
URL: *http://en.citizendium.org/wiki/Cloud_computing*

Association Française du Multimédia Mobile
URL: *http://www.afmm.fr/*

Site officiel jQuery
URL: *http://jquery.com/*

Tutoriels Talend Open Studio
URL: *http://www.talendforge.org/tutorials/menu.php*

Protocole WAP
URL: *http://en.wikipedia.org/wiki/Wireless_Application_Protocol*

Web 2.0, allez plus loin avec AJAX et XMLHttpRequest
URL: *http://siddh.developpez.com/articles/ajax/*

7.2) Livres et documents

PETRI Gregor, Éclairage sur le cloud computing, Janvier 2010, 60 pages

DELACROIX Jérôme, Cloud computing : 30 entreprises détaillent leurs retours d'expérience, Mai 2011, 169 pages

AFMM, Etude AFMM, 18 Janvier 2011, 41 pages

AFMM, Guide du Micro-Paiement, Juin 2010, 25 pages

AFMM, Guide du Marketing Mobile (édition 2010), Janvier 20120

8) Glossaire

ALIAS	Identifiant unique pour un utilisateur sur un short code donné. Il est transmis par l'opérateur en lieu et place du numéro de téléphone. Il peut éventuellement être commun sur plusieurs short codes, selon la configuration choisie à l'ouverture du short code. Il commence par +1 pour Bouygues, +2 pour SFR et +3 pour Orange. Un alias a une durée de validité d'un mois (Orange) ou de deux mois (SFR et Bouygues), renouvelé chaque fois qu'un utilisateur envoie un MO, ou qu'il reçoit un MT surtaxé.
CROSS	Relance publicitaire par SMS, incitant l'utilisateur à s'inscrire sur un service différent de celui auquel il s'était intéressé initialement.
CSS	Cascading style sheet : il s'agit des feuilles de style adjointes aux fichiers HTML, séparant les données de leur présentation.
DMZ	Demilitarized zone : c'est un sous-réseau séparé du réseau local et isolé de celui-ci et d'Internet par un pare-feu.
FACILITATEUR	Société intermédiaire mettant à disposition de société comme la notre une API permettant de communiquer avec les différents opérateurs téléphoniques, afin d'échanger des SMS avec les utilisateurs finaux.
FIREWALL	Un pare-feu est un logiciel ou matériel faisant respecter la politique de sécurité du réseau.
FRAMEWORK	Un Framework est un kit de composants logiciels structurels servant à créer les fondations de tout ou d'une partie d'un logiciel.
GSM	Global system for mobile communications (GSM) (historiquement « Groupe spécial mobile ») : c'est une norme numérique de deuxième génération (2G) pour la téléphonie mobile.
HTACCESS	Les fichiers .htaccess sont des fichiers de configuration des serveurs web Apache. Ils servent à modifier les droits d'accès, créer des redirections ou encore écrire des messages d'erreur personnalisés.
HTML	Hypertext markup language : c'est le format de données conçu pour représenter les pages web. C'est un langage de balisage.
HTTP	Hyper text transfer protocol : Il s'agit du protocole de communication utilisé par le web.
IHM	Interface homme-machine : Elle définit les moyens et outils mis en œuvre, afin qu'un humain puisse contrôler et communiquer avec un système.
INTERNET+	Solution de paiement sécurisé sur facture Internet et sur facture mobile.
JAVA	Le Java est un langage de programmation informatique orienté objet présenté officiellement en mai 1995.
jQuery	jQuery est une bibliothèque JavaScript libre qui porte sur l'interaction entre JavaScript et HTML, La première version date de Janvier 2006.
MSISDN	Mobile station integrated services digital network : désigne le numéro mobile de l'utilisateur. Ce MSISDN n'est pas transmis à l'éditeur ; il est remplacé par un alias (identifiant).
Multithreading	Capacité d'un programme à lancer simultanément plusieurs processus de traitement.

MVC	Le modèle-vue-contrôleur est un patron d'architecture et une méthode de conception organisant l'interface homme-machine d'une application logicielle.
PHP	Hypertext Preprocessor : langage de scripts libre principalement utilisé pour produire des pages web dynamiques via un serveur HTTP.
PUSH	(Relance) De manière générale, envoi d'un MT non facturé présentant un message publicitaire vers un utilisateur final (marketing direct).
REBILL RELOAD	Action consistant à facturer l'utilisateur à chaque date anniversaire.
RETRY	Action consistant à retenter de facturer un client lorsque la tentative précédente à échoué.
SAAS	Software As A Service : modèle technologique du cloud computing proposant à ses clients des logiciels accessibles via internet.
SCRUM	Méthode agile de gestion de projet.
SEO	Search engine optimization : ensemble de techniques visant à favoriser la compréhension de la thématique et du contenu d'une ou de l'ensemble des pages d'un site web par les moteurs de recherche.
SGBD	Système de gestion de base de données : il s'agit d'un logiciel système destiné à stocker et à partager des informations dans une base de données, en garantissant la qualité, la pérennité des informations tout en cachant la complexité des opérations.
SHORT CODE	Numéro à 5 chiffres vers lequel l'utilisateur peut envoyer un MO afin d'initier l'achat d'un service. Le premier chiffre indique le palier tarifaire (prix du service), le 2e indique qu'il s'agit. Cf http://www.afmm.fr/Solutions/SMS-et-MMS/Principes/Referencement/Principes-tarifaires
SI	Système d'information : dans le contexte de ce mémoire, il s'agit des systèmes informatiques consacrés au traitement de l'information relative à toute l'activité de Digital Virgo Entertainment
SMS+	Le consommateur est invité, via la communication d'un éditeur, à agir ou réagir en envoyant un mot clé par SMS à un numéro à 5 chiffres, commun à tous les opérateurs mobiles. Il reçoit en retour un autre message lui délivrant le contenu mobile désiré ou le connectant au site mobile de l'éditeur.
SMS-MO	Short Message Service Mobile Originated : désigne un SMS émis par un utilisateur depuis son terminal mobile.
SMS-MT	Short Message Service Mobile Terminated : désigne un SMS reçu par utilisateur sur son terminal mobile.
SQL	Structured query language : Langage commun à toutes les bases de données relationnelles, servant à effectuer des requêtes sur une base de données pour en extraire des informations.
SR	Status report. Information envoyée par l'opérateur et relayée par le broker, indiquant le statut de réception d'un message par son destinataire, et en cas d'échec, la raison de l'échec.
SVN	Subversion est un logiciel de gestion de versions. Il a été conçu pour remplacer CVS.

URL	Uniform resource locator : désigne une chaîne de caractères utilisée pour adresser les ressources du web.
WIDGET	Désigne un composant d'une interface graphique
WML	Wireless markup language : c'est un langage à balises conçu spécifiquement pour le WAP, de manière à pouvoir s'afficher sur un écran de téléphone mobile. Sa syntaxe est proche du HTML.

9) Table des figures

Figure 1 - Chiffre d'affaires du groupe Digital Virgo

Figure 2 - Présence géographique du groupe Digital Virgo

Figure 3 - Répartition du chiffre d'affaires de Digital Virgo

Figure 4 - Page d'accueil de woozgo.fr

Figure 5 - Page d'accueil de mobifiesta.com

Figure 6 - Cycle de vie d'un projet chez Digital Virgo Entertainment

Figure 7 - Infrastructure technique de Digital Virgo Entertainment Marseille - Aix En Provence

Figure 8 - Part du recrutement par moyen de paiement

Figure 9 - Architecture technique de l'application IRC

Figure 10 - Périmètre fonctionnel de l'application IRC

Figure 11 - Schéma de fonctionnement de la méthode agile SCRUM

Figure 12 - Processus de création d'une relance

Figure 13 - Cinématique d'abonnement SMS+

Figure 14 - Cinématique d'abonnement internet+ mobile

Figure 15 - Schéma de la base de données de l'application IRC

Figure 16 - Présentation du modèle MVC

Figure 17 - Utilisation basique du plugin jQuery

Figure 18 - Utilisation avancée du plugin jQuery

Figure 19 - Utilisation avancée avec habillage du plugin jQuery

Figure 20 - Utilisation du plugin jQuery dans d'autres projets

Figure 21 - Mise en place d'un treeview

Figure 22 - Gestion du glisser-déposer

Figure 23 - Gestion de groupe unique

Figure 25 - Gestion de plusieurs groupes

Figure 26 - Présentation des repoussoirs dans la création d'une campagne

Figure 27 - Réalisation d'un job sous Talend Open Studio

Figure 28 - Les différents modèles de cloud computing

Figure 29 - Comparaison base SQL vs base vectorielle

Figure 30 - Présentation d'un segment complet

Figure 31 - Création d'une requête

Figure 32 - Validation des relances

Figure 33 - Gestion des requêtes

10) Liste des tableaux

Tableau 1 - Gestion des URL courtes

Tableau 2 - Backlog de produit

Tableau 3 - Fichier de suivi de bugs

Tableau 4 - Présentation des alias

Tableau 5 – Présentation d'une campagne