

Ministère de la planification
du Développement



Institut Africain d'Informatique,
Représentation du TOGO (IAI-TOGO)
Rue de la Kozah Nyekonakpoè
07 BP : 12456 Lomé 07, Togo
Tél +228 22 20 47 00/+228 22 21 27 06
E-mail : iaitogo@iaitogo.com
Site web: www.iai-togo.com



DORKENOO CONSULTING

Services informatiques et télécoms

22 BP 231 Lomé-Togo
Tél +228 22 46 36 48 / +228 92 52 01 19
E-mail : dg@dorkenoconsulting.com
Site web : www.dorkenoconsulting.com

STAGE PRATIQUE DE PROGRAMMATION

Thème :

*Conception et Réalisation d'une plateforme autonome de transfert
de crédit et d'activation de forfait internet*

Présenté par :

ADUAYOM MESSAN Ekoue Enoc
Etudiant en Deuxième Année Génie Logiciel
Année académique 2015-2016
Période : 20 Juin 2016 au 20 Septembre 2016

Superviseur :

M. KEOULA Edem Lonlon
Informaticien à MEDIASOFT

Maitre de stage :

M. Elvis Edem DORKENOO
Directeur Général de
DORKENOO CONSULTING

DEDICACES

À ma mère, TENGUEY Akossiwa Christine et à mon père ADUAYOM MESSAN Folly Claude, pour avoir toujours cru en moi, pour leur amour, et leur soutien. Je traduis en ce travail toute ma gratitude,

À Mr AKPAKI Siméon pour m'avoir toujours soutenu dans mon travail,

À mes frères, sœurs et à tous mes proches pour leurs encouragements. Vous êtes pour moi la preuve que la persévérance dans le travail porte toujours ses fruits.

REMERCIEMENTS

Mes remerciements vont d'abord à DIEU tout puissant pour la santé et l'intelligence qu'il m'a conféré et pour toute son assistance dans mon travail.

Je remercie tous ceux qui, de près ou de loin, m'ont aidé ou soutenu d'une manière ou d'une autre pour l'élaboration de ce document. Ma gratitude va particulièrement à :

- ❖ M Kodjo AGBETI, Directeur Général de l'IAI-TOGO pour ses conseils;
- ❖ M AMEYIKPO Kossi N. Directeur des études de l'IAI-TOGO pour ses conseils;
- ❖ M Elvis Edem DORKENOO, Directeur Général de DORKENOO CONSULTING qui a bien voulu m'accorder ce stage ;
- ❖ M KEOULA Edem Lonlon, qui a supervisé mon stage et encadré mon travail. J'ai admiré sa disponibilité, ses idées, et ses conseils.

Sommaire

PAGE DE DEDICACE.....	i
PAGE DE REMERCIEMENT	ii
SOMMAIRE	iii
PARTICIPANTS AU PROJET	iv
INTRODUCTION GENERAL	v
PARTIE 1 : RAPPORT D'INSERTION.....	6
INTRODUCTION	7
I. PRESENTATIONS.....	8
II. THEME DU STAGE.....	12
CONCLUSION	15
PARTIE 2 : RAPPORT DE PRE-PROGRAMMATION	16
INTRODUCTION	17
I. ETUDE DE L'EXISTANT.....	18
II. CRITIQUE DE L'EXISTANT.....	18
III. PROPOSITIONS DE SOLUTIONS.....	19
IV. CHOIX DE SOLUTION.....	20
V. EVALUATION FINANCIERE DE LA SOLUTION RETENUE	21
VI. PLANNING PREVISIONNEL DE REALISATION.....	22
VII. PRESENTATION DES METHODES D'ANALYSE.....	23
VIII. PRESENTATION DE L'OUTILS DE MODELISATION	26
IX. ETUDE DETAILLE DE LA SOLUTION	28
CONCLUSION	43
PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE	44
INTRODUCTION	45
I- MISE EN ŒUVRE.....	46
II- PRESENTATION DE L'APPLICATION.....	54
CONCLUSION	86
CONCLUSION GENRERALE	87

PARTICIPANTS AU PROJET

NOM ET PRENOMS	ROLES	FONCTIONS
ADUAYOM MESSAN Ekoue Enoc	Concepteur du Système	Etudiant
M. KEOULA Edem Lonlon	Superviseur	Développeur d'application à Media Soft
M. Elvis Edem DORKENOO	Maitre de Stage	Directeur Général de DORKENOO CONSULTING

INTRODUCTION GENERALE

Dans le but de jauger les compétences acquises au cours de nos deux années de formation, de nous permettre d'accroître nos compétences dans les différents domaines de compétence et d'études, l'Institut Africain d'Informatique, Représentation du Togo (IAI-TOGO) prévoit un stage obligatoire de trois (3) mois pour les étudiants de deuxième année.

De nos jours, l'outil informatique devient plus une nécessité qu'une utilité ; il se fait voir aussi comme un acteur mais aussi un facteur conséquent pour le développement d'une entreprise. L'utilisation des outils informatiques apporte un gain considérable de temps et de moyens financiers. Cet outil garantit de même une sécurité, une fiabilité, la réutilisabilité et aussi la disponibilité des données. Vu les nombreux avantages qu'il nous offre, l'outil informatique se retrouve partout et se fait ressentir dans le domaine de la téléphonie.

La tâche qui nous a été assignée durant notre stage est de développer une application qui sera chargée d'effectuer les transactions telles que le transfert de crédit et l'activation de forfait internet quel que soit le réseau auquel on est affilié. Dans cet ordre d'idées notre thème de stage est intitulé : « **Conception et Réalisation d'une plateforme autonome de transfert de crédit et d'activation de forfait internet** ». Dans le souci d'automatiser certains services en vue de répondre à certaines attentes des clients de ces différents opérateurs Télécoms qui exprime le besoin d'avoir accès à tout instant et en tout lieu aux recharges téléphoniques et/ou à l'activation de forfaits internet, notre centre d'accueil, le cabinet de développement « **DORKENOO Consulting** » à penser au développement d'un système nommé « Robot-Agence » qui fournira automatiquement, sans intervention humaine, des recharges aux clients. Ce projet innovant a été l'objet de notre stage.

Notre document sera divisé en trois grandes parties. Nous présenterons en premier notre rapport d'insertion. Il sera suivi du rapport de pré-programmation. Le rapport de réalisation mettra fin au présent document.

PARTIE 1 : RAPPORT D'INSERTION

Du 20 juin 2016 Au 18 juillet 2016

INTRODUCTION

La phase d'insertion est cette phase où, nous présenterons donc notre institut, l'**Institut Africain d'Informatique, Représentation du TOGO (IAI-TOGO)** ; suivra la présentation du cadre de stage, la société **DORKENOO CONSULTING** et pour finir, le thème de stage.

I. PRESENTATIONS

1. Présentation de IAI-TOGO

L’Institut Africain d'informatique, représentation du TOGO (IAI-TOGO) est ouvert le 22 octobre 2002. IAI-TOGO est une école inter-Etats d'enseignement supérieur en Informatique. Il est membre du réseau IAI créé le 29 janvier 1971 à Fort Lamy (actuel Ndjamenya) en république de TCHAD. L'accord d'établissement entre la République Togolaise et l’Institut Africain d’Informatique a été signé le 12 mai 2006.

IAI-TOGO forme en trois (3) ans des ingénieurs des travaux informatiques dans trois filières : Génie Logiciel (GL), Système & Réseaux (SR) et Multimédia Technologie Web & Infographie (MTWI). IAI-TOGO propose actuellement le cycle de formation des ingénieurs de travaux informatiques (Licence professionnelle en informatique) et de Master. Au terme des trois années de formation, les diplômés peuvent poursuivre leurs études supérieures à l’IAI-TOGO ou dans le pays de leur choix.

Il propose aussi une formation dénommée formation Cisco. Les cours Cisco sont découpés en quatre (4) modules CCNA1, CCNA2, CCNA3 et CCNA4, tous accessibles via Internet.

En ce qui concerne l’organisation structurelle de IAI-TOGO, nous avons:

Mr AGBETI A. B. Kodjo, le Représentant Résident de l’IAI-TOGO. Il est l’intermédiaire entre l’IAI-Gabon et celui du Togo.

Mr AMEYIKPO Kossi N. Directeur des Etudes (DE), qui est chargé des problèmes liés aux études entre autres le respect du règlement intérieur.

Le chargé de la scolarité qui est comme l’indique son titre, chargé du règlement à temps et à terme de la scolarité.

-les secrétaires qui servent d’intermédiaire entre l’administration et les étudiants.

2. Présentation du cadre de stage

❖ Statut

Le centre d'accueil pour notre stage est un cabinet de services informatiques et télécoms au nom de DORKENOO CONSULTING SARLU.

Il s'agit d'une Société à Responsabilité Unipersonnelle dont l'associé unique est Mr DORKENOO Edem Elvis. Le capital social, apporté par l'associé, est fixé à la somme d'un million (1 000 000) de francs CFA.

❖ Mission

Depuis 2013, DORKENOO Consulting offre des conseils, développe et fournit des solutions informatiques logicielles innovantes aux entreprises et institutions togolaises, les aidant à assurer la veille technologique. DORKENOO Consulting dispose d'une équipe professionnelle et dynamique qui œuvre dans diverses branches de l'informatique et des télécommunications.

Nos missions sont les suivantes :

- S'affirmer en tant que partenaire sûr dans le secteur des solutions informatiques appliquées à l'e-gouv.
- Doter les Entreprises et Institutions de solutions avancées et performantes à un coût raisonnable.
- Apporter notre expertise et en faire profiter le Togo et l'Afrique afin de participer à son développement.

❖ Activités

- Conception, développement et déploiement des systèmes informatiques.
- Développement d'applications mobiles, sms, IVR.
- Expertise en conception, administration et maintenance de systèmes informatiques de sondage et de collecte d'informations (web, mobile, sms, IVR, réseaux sociaux).
- Développement web.
- Administration de bases de données.

- Analyse informatique et automatisation des systèmes d'information d'entreprise.
 - Intégration de solutions et ERP.
 - Conception de SIG (Systèmes d'information Géographique).
 - Maintenance informatique.
- ❖ Quelques réalisations
- **MAEH-SIG (Projet PADAT / SAKSS) : Système d'Information Géographique** pour les réalisations du Ministère de l'Agriculture, l'Elevage et de l'Hydraulique.
 - **Projet Togo PAD Infrastructure** en collaboration avec la GIZ, Agetur-Togo, Demos-Europe et de l'UCT (Union des communes du Togo) : **Système de sondage par sms au sein de la population des communes**. Sous-Traitance par Roots-Technologies de la mise en route, maintenance, de l'administration et du support technique en collaboration avec Demos-Europe (Groupe allemand).
 - **HSP Solutions (HSP by Sms et HSP Mobile)** : solution de sondage, de collecte d'information et de communication par sms ou par Smartphone.
 - **ANGE COM** : (avec **NOVATECH**) Système de communication utilisé à l'**ANGE** (Agence Nationale pour la Gestion de l'environnement).
 - **Ange Courrier** : Développement du logiciel de gestion de Courrier utilisé à l'**ANGE** (Agence Nationale pour la Gestion de l'environnement).
 - **GED by Alfresco** (Gestion électronique de documents) et Intégration OODOO.
 - **Tontine Mobile** : Utilitaire de collecte des cotisations pour tontine.
 - **Caisse Mobile** : Utilitaire de gestion de caisse mobile d'une microfinance.
 - **Services SMS Banking** pour microfinance (consultation de solde, demande de relevé, dépôt/ retrait à distance).
 - **Télémétrie (avec Global Services Group)** Développement de la plateforme informatique pour télé jaugeage du fuel des sites BSC pour les différents

opérateurs Télécoms : Moov Togo, Moov Bénin, Moov Niger, Sonatel Guinée, Cellcom Guinée.

- Conception du site web du groupe **WaveSide** (<http://wavesidelab.com>).

❖ Organigramme (Résumé)

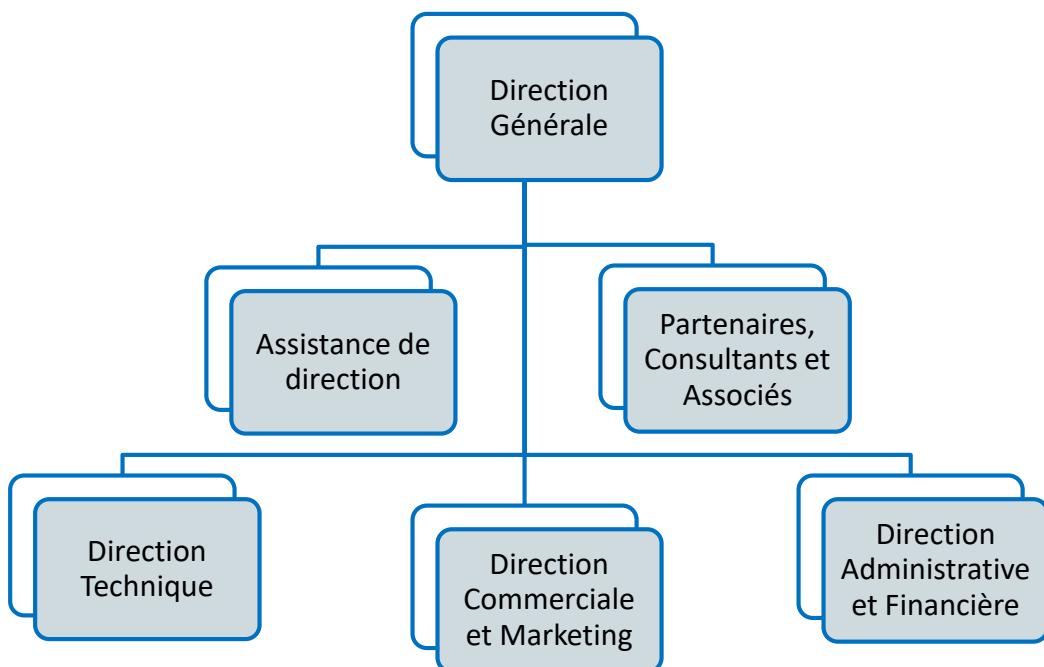


Figure 1: Organigramme de DORKENOO CONSULTING

❖ Plan de localisation

Le siège social se situe à Lomé, au Togo ; en plus d'une filiale ivoirienne situé à Abidjan depuis Mars 2016.

La direction de Lomé se situe à Agoè non loin du Carrefour des Armoiries.

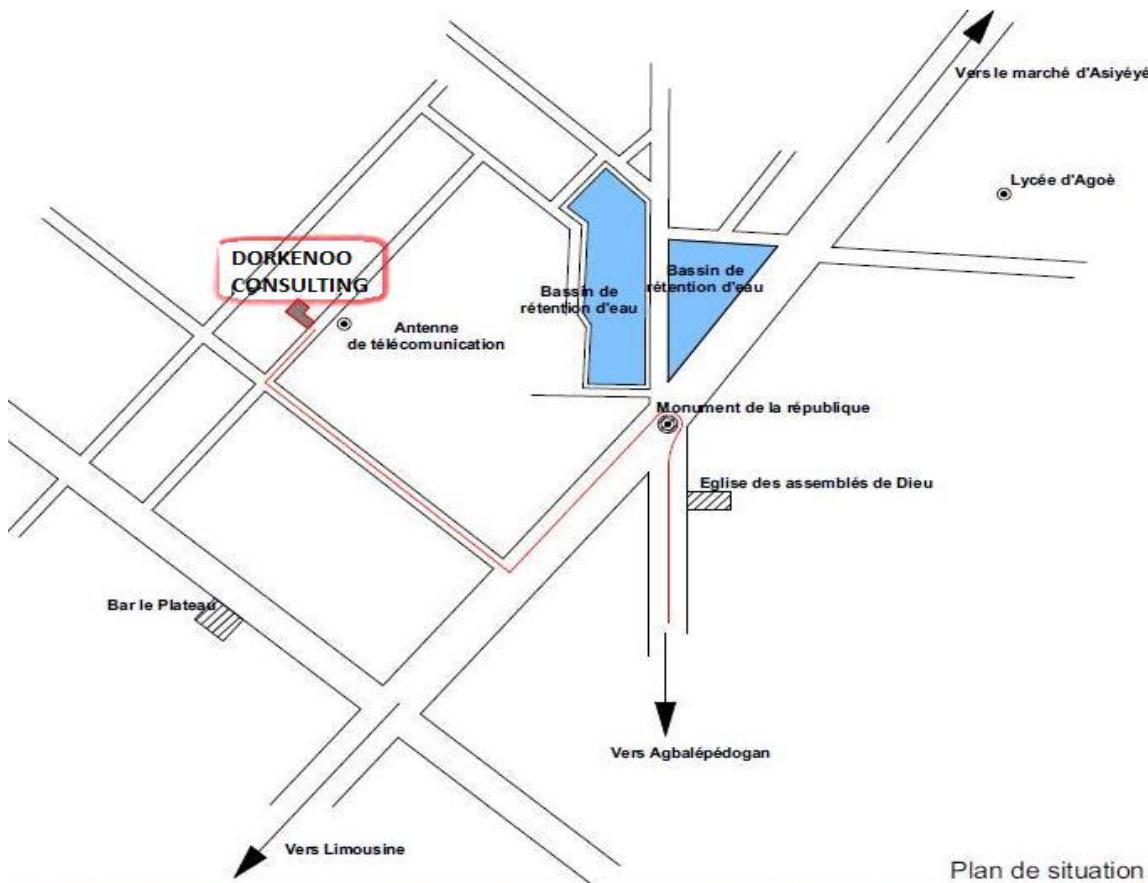


Figure 2 : Plan de situation DORKENOO CONSULTING à Lomé

II. THEME DU STAGE

Les deux premières semaines furent celle d'intégration, de prise de contact et de prise de connaissance du thème de stage. Il s'agit de : « **Conception et Réalisation d'une plateforme autonome de transfert de crédit et d'activation de forfait internet** ».

1. Présentation du sujet

La tâche qui nous est assignée est de développer un système autonome chargé d'effectuer de manière autonome et automatique, le transfert de crédit et l'activation d'un forfait internet. Notre système pourra être utilisé par tous les abonnées de TOGOCEL et MOOV. Nommé « **Robot-Agence** », il fournira, sans l'intervention humaine, les services précédemment identifiés aux clients.

2. Problématique du sujet

Comment, recharger son compte téléphonique, activer ou souscrire à un forfait internet sans se rendre à un point de vente ? Ces points de vente qui ne nous sont pas accessibles, en tout temps et en tout lieu. Se trouver en insuffisance de crédit est souvent une situation aussi indésirable. Ainsi le cabinet à penser mettre en place un système dénommé « **robot-agence** » qui se chargera d'effectuer toutes ces opérations. Cette agence (virtuelle) sera accessible à tout moment : 24 h sur 24 et 7 jours sur 7.

3. Intérêt du sujet

i. Objectifs

Offrir de meilleurs services aux clients permet aux opérateurs d'augmenter leur chiffre d'affaire et surtout la satisfaction de leurs clients. robot-agence sera chargé de :

- Permettre l'inscription des clients disponible en deux (2) modes :
 - **le mode post payé** : le client sera facturé selon sa consommation et suivant une périodicité bien définie. Elle peut être mensuelle, trimestrielle ou personnalisée ;
 - **le mode prépayé** : le client après l'inscription, doit créditer son compte avant usage. Il pourra donc utiliser ce montant à sa guise. Le compte pourra être recharge à volonté. Le solde s'épuise au fur et à mesure de sa consommation.
- Recevoir les demandes des clients par SMS (gratuit éventuellement), de les traiter selon leurs besoins et de renvoyer un accusé suivi de la demande du client. Le système sera aussi capable de distinguer l'opérateur mobile sur lequel la transaction doit être faite ;
- D'autoriser certaines actions à partir d'une interface d'administration et de gestion. Ces actions sont entre autres :

- avoir un historique des opérations ;
- générer les factures ;
- inscrire les clients.

ii. Résultats attendus

Robot-agence doit permettre de :

- gérer l'inscription et le suivi des clients suivants les profils ;
- d'obtenir l'historique de toutes les opérations ;
- garantir la sécurité et la confidentialité des informations ;
- permettre de faire un transfert de crédit aux clients ;
- permettre d'activer des forfaits data pour internet.

CONCLUSION

Le début du stage n'a pas été aisé. Il fallait comprendre et choisir un thème parmi tant d'autres, se familiariser avec ce thème, poser les fondements de notre travail avec nos encadreurs. Après cette étape cruciale, nous avons entamé la conception du « **robot-agence** ». Le rapport de pré-programmation, expose le travail que nous avons effectué durant cette phase.

PARTIE 2 : RAPPORT DE PRE-PROGRAMMATION

Du 28 juin 2016 Au 29 juillet 2016

INTRODUCTION

La phase de pré-programmation est la phase conceptuelle du futur système. Nous allons d'abord présenter le système actuel et son fonctionnement, identifier ses failles puis proposer des approches de solutions. Ensuite nous citerons quelques avantages de la solution retenue, puis la méthode de conception ainsi que l'outil de modélisation choisie.

I. ETUDE DE L'EXISTANT

Togocel, ETISALAT Togo (MOOV), Togo Télécom sont des opérateurs mobiles disposant de deux types de clients. Les clients prépayés et les clients post payés.

- En ce qui concerne les clients en mode prépayé, la consommation n'est possible que lorsque ce compte est crédité d'une valeur minimale qui diffère d'un réseau à l'autre. Lorsque le solde du compte est nul, ces clients sont obligés de recharger leur compte afin de bénéficier des différents services qu'offre leur opérateur. Pour recharger le compte il faudra donc se rendre à un point de vente ou dans une agence.
- Les clients se trouvant en mode post payé disposent aussi d'un compte à la différence qu'il est facturé selon leur consommation. Il reçoit une facture suivant une périodicité qu'il aura au préalable défini. Celle-ci peut être mensuelle, trimestrielle, semestrielle ou personnalisée. Ce dernier profil de client n'a pas accès à certains services comme le transfert de crédit, la consommation en post-paiement de forfait internet. Il doit pour ces différents besoins se rendre dans une agence ou un point de vente.

La gestion actuelle possède certain manquement eu égard aux objectifs de ces compagnies de téléphonie mobile que nous détaillerons dans la partie suivante.

II. CRITIQUE DE L'EXISTANT

Avec ce système déjà établit, on se retrouve face à certaines insuffisances :

- ❖ En ce qui concerne les abonnés prépayés :
 - les agents ou agences accrédités ne sont disponibles, ni en tout temps ni en tout lieu. Il est difficile à tout client ayant urgentement besoin de recharger son compte tard dans la nuit de l'obtenir. Ce qui est pareil pour un autre géographiquement isolé ;

- les agents ainsi que le client sont susceptibles d'entrer des numéros de bénéficiaires incorrects ;
 - les opérateurs sont obligés de payer leurs agents pour des heures supplémentaires ou trouver des artifices pour garantir une continuité de service durant les heures de repos. Ce qui bien évidemment engendre des coûts pour les services dont la qualité est discutable ;
- ❖ En ce qui concerne les abonnés post payés:
- il n'est pas possible de demander un suivi de l'état de sa consommation au cours d'un temps inférieur à un mois ; autrement, elle sera retardée par plusieurs démarches administratives ;
 - le filtre pour le mode post payé est très sélectif au niveau des opérateurs ;
 - Il n'y a pas la possibilité de faire un achat depuis son compte sur un autre numéro.

III. PROPOSITIONS DE SOLUTIONS

En vue de palier à ces insuffisances, nous proposons deux (02) solutions:

• **Solution1** : développer un système similaire au système de Namcredit/Semoa: Il s'agit d'un système qui s'oriente sur l'utilisation d'un service web. Il est basé sur une plateforme Android et web qui permet de faire un transfert de crédit ou d'argent via internet. Pour transférer de l'argent ou du crédit, il suffit donc de renseigner le numéro du bénéficiaire, de choisir le montant et de payer.

L'application présente plusieurs avantages à savoir :

- la possibilité d'envoyer de l'argent partout au Togo à partir de 2 000 f CFA (exemple de Namcredit);
- des frais d'envoi très moindres par rapport aux transferts ordinaires ;
- la possibilité d'envoyer et de retirer de l'argent même les dimanches et les jours fériés ;
- une assistance permanente par téléphone, par mail ou par WhatsApp.

Les inconvénients relevés sur ce type de système sont :

- il est destiné uniquement aux personnes ayant une carte bancaire (Carte Visa, Master Card...), ce qui n'est vraiment pas adapté à notre réalité ;
- tout client n'ayant pas accès à internet ne pourra pas avoir accès à ce service ;
- même si les frais de transaction sont moins coûteux par rapport à ce qui se fait sur le terrain, il demeure encore pour la plupart des clients, un peu cher puisque des frais bancaires seront tout de même prélevés.

- **Solution 2** : développer un système autonome fonctionnant par SMS gratuit.

Cette solution consiste, à développer un système automatique et autonome semblable qui permettra aux clients de faire des recharges d'unités suite à l'envoi d'un sms gratuit sur un numéro court. Pas besoin d'une quelconque carte ou de connexion internet ou d'un téléphone mobile à la pointe de la technologie. Ce système doit pouvoir être déployé sur n'importe quelle technologie mobile.

Un système qui résout toutes les insuffisances de la première solution tout en gardant ses avantages.

IV. CHOIX DE SOLUTION

En faisant le comparatif des deux solutions, notre cabinet a retenu la solution 2. Elle offre l'avantage d'être non seulement ouvert à tous les clients mais aussi, peut s'adapter à tous les milieux que ce soit sociale ou économique, alors que la première solution n'est strictement réservée qu'à une pincée de personnes.

V. EVALUATION FINANCIERE DE LA SOLUTION RETENUE

Cette solution consiste à développer un système autonome fonctionnant sur sms gratuit. Les coûts qu'il engendre sont :

- Approche avec modem sans API :

Matériels	Prix unitaires	Quantité	Prix total
Carte SIM (pour test)	1 000	3	3 000
Ordinateur Ultrabook	786 492,44 ¹	1	786 492,44
Modem professionnel	60 000 ²	3	180 000
Onduleur	72 089,67 ³	1	72 089,67
Groupe électrogène (5.5 KVA)	655 301,04 ⁴	1	655 301,04
Serveur dédié Processeur Intel Xeon Core i7 3.5GHz, RAM 32Go, Disque 512Go SSD + 4*500Go HDD	52 000 FCFA/mois	-	624 000/an
TOTAL			2 320 883,15

- Approche sans modem avec API :

Matériels	Prix Mensuel	Prix annuel
Numéro court	50 000	600 000
Serveur dédié Processeur Intel Xeon Core i7 3.5GHz, RAM 32Go, Disque 512Go SSD + 4*500Go HDD	52 000	624 000
TOTAL		1 224 000

-Sites de collecte des Montants :

1 :<http://www.dell.com/fr/p/xps-13-9350-laptop-ubuntu/pd?oc=cnx9333>

2 :<https://www.siemens.com/global/en/home.html>

3 :<http://www.materiel.net/onduleur-et-prise-parafoudre/apc-onduleur-back-ups-bx-950va-prises-fr-115813.html>

4 :<http://www.kipor-power.fr/>

VI. PLANNING PREVISIONNEL DE REALISATION

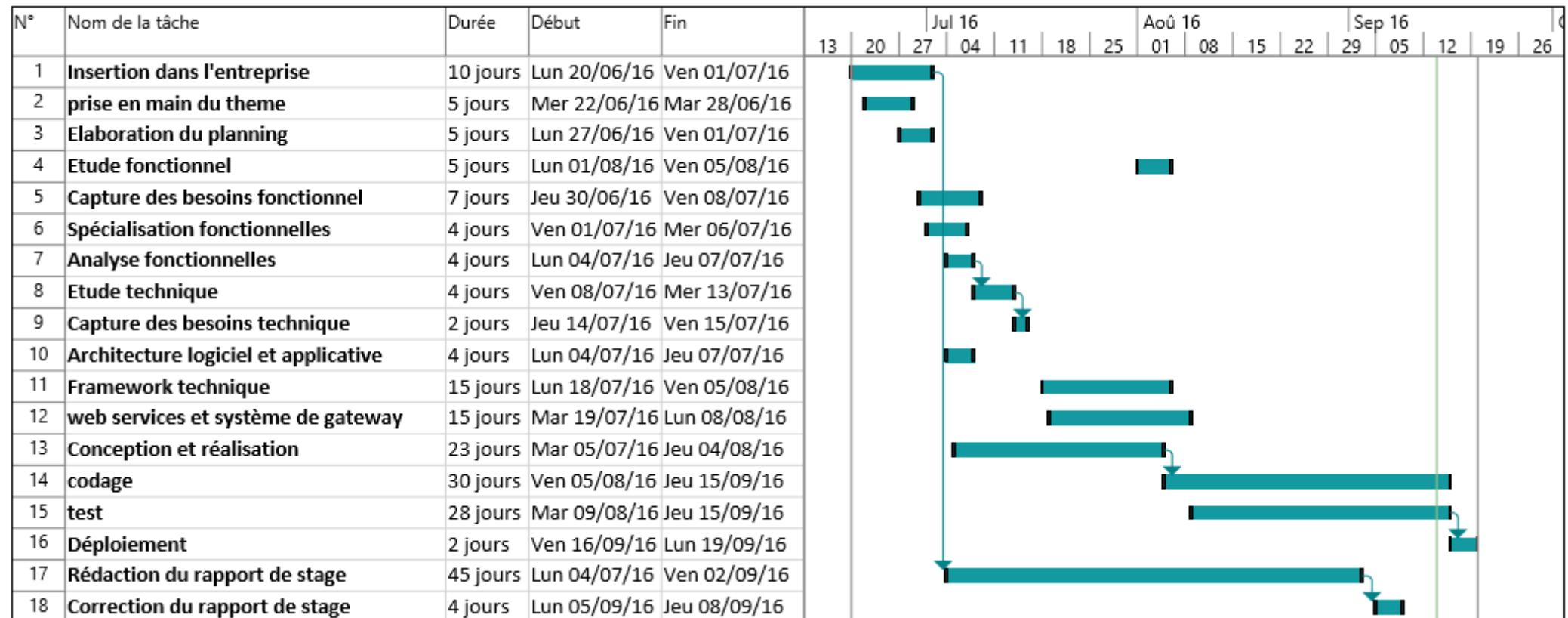


Figure 3: Diagramme de Gantt (planning prévisionnel de réalisation)

VII. PRESENTATION DES METHODES D'ANALYSE

1. Le langage UML



Figure 4: Logo UML

Le **langage de modélisation unifié**, de l'anglais *Unified Modeling Language (UML)*, est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.

L'UML est le résultat de la fusion de précédents langages de modélisation objet : Booch, OMT, OOSE. Principalement issu des travaux de Grady Booch, James Rumbaugh et Ivar Jacobson, UML est à présent un standard adopté par l'Object Management Group (OMG). UML dispose de différents diagrammes pour modéliser le système. Les diagrammes sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie. Il en existe 14 (depuis UML 2.3 contre 9 pour UML 1.3).

❖ Les diagrammes structurels ou statiques (07) :

- diagramme de classes (class diagram) ;
- diagramme d'objets (object diagram) ;
- diagramme de composants (component diagram) ;
- diagramme de déploiement (deployment diagram) ;
- diagramme des paquets (package diagram) ;

- diagramme de structure composite (composite structure diagram) ;
- diagramme de profils (profile diagram).

❖ **Les diagrammes comportementaux (03) :**

- diagramme des cas d'utilisation (use-case diagram) ;
- diagramme états-transitions (state machine diagram) ;
- diagramme d'activité (activity diagram).

❖ **Les diagrammes d'interaction ou dynamiques (04) :**

- diagramme de séquence (sequence diagram) ;
- diagramme de communication (communication diagram) ;
- diagramme global d'interaction (interaction overview diagram) ;
- diagramme de temps (timing diagram).

2. La méthode AGILE

Les méthodes agiles sont des méthodologies essentiellement dédiées à la gestion de projets informatiques. Elles reposent sur des cycles de développement itératifs et adaptatifs en fonction des besoins évolutifs du client. Elles permettent notamment d'impliquer l'ensemble des collaborateurs ainsi que le client dans le développement du projet.

Ces méthodes permettent généralement de mieux répondre aux attentes du client en un temps limité (en partie grâce à l'implication de celui-ci) tout en faisant monter les collaborateurs en compétences. Ces méthodes constituent donc un gain en productivité ainsi qu'un avantage compétitif tant du côté client que du côté du fournisseur.

Il se base sur quatre (4) méthodes et douze (12) principes.

Les méthodes :

- l'équipe et la communication avant les outils et processus : dans la vision agile, l'équipe est bien plus importante que les outils ou les procédures de fonctionnement.
- l'application avant la documentation : il est primordial que le projet fonctionne, c'est la priorité avant toute chose.

- la collaboration avant la négociation : le client doit être impliqué dans le développement.
- l'acceptation du changement et la flexibilité avant la planification : la planification initiale et la structure du projet doivent être flexibles afin de permettre les évolutions attendues par le client.

Les principes :

1. la plus haute priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à grande valeur ajoutée.
2. Il faut accueillir positivement les changements et les nouveaux besoins, même lorsqu'ils arrivent tardivement dans un projet. Les processus agiles exploitent la flexibilité au changement afin de fournir un avantage compétitif pour le client.
3. Il faut livrer régulièrement un logiciel opérationnel (utilisable en production) avec des cycles courts (idéalement entre deux et quatre semaines).
4. les utilisateurs ou leurs représentants et les développeurs doivent travailler ensemble au quotidien et tout au long du projet.
5. Il faut réaliser les projets avec des personnes motivées, leur fournir des environnements adaptés à leur besoin ainsi que le soutien dont ils ont besoin et leur confiance pour atteindre les objectifs fixés.
6. le dialogue en face à face entre les différents acteurs d'un projet agile est la méthode la plus simple et la plus efficace pour transmettre l'information et la connaissance entre ces derniers.
7. l'aspect opérationnel d'un produit est la principale mesure d'avancement de ce dernier.
8. les processus agiles doivent amener à un rythme de développement soutenable pour l'équipe et constant (il ne doit pas y avoir de période de forte montée ou baisse de charge de travail ayant des impacts significatifs sur l'équipe).
9. la recherche de l'excellence et de la performance conceptuelle et technique renforce l'agilité d'un produit.
10. simplifier le travail en minimisant le nombre de tâches inutiles et redondantes est essentiel.
11. les meilleures solutions logicielles émergent d'équipes auto-organisées tant au niveau de la clarté des spécifications, que de la conception et de la mise en place d'architectures performantes et efficaces.
12. l'équipe doit réfléchir à des moyens, à intervalles réguliers, pour devenir davantage efficace et mettre en pratique ces nouvelles méthodes une fois décidées.

VIII. PRESENTATION DE L'OUTILS DE MODELISATION

L'outil retenu pour la modélisation est Sybase PowerAMC Version 15.1.

1. Qu'est-ce que PowerAMC ?



Figure 5: Logo PowerAMC

PowerAMC est un logiciel de modélisation. Il permet de modéliser les traitements informatiques et leurs bases de données associées. Créé par SDP sous le nom AMC Designor, racheté par Powersoft, ce logiciel est produit par Sybase depuis le rachat par cet éditeur en 1995. Hors de France, la version internationale est commercialisée par Sybase sous la marque PowerDesigner.

Il permet d'effectuer les tâches suivantes :

- ❖ modélisation intégrée via l'utilisation de méthodologies et de notation standard:
 - données (E/R, Merise) ;
 - métiers (BPMN, BPEL, ebXML) ;
 - application (UML) ;
- ❖ génération automatique de code via des templates personnalisables :
 - SQL (avec plus de 50 SGBD) ;

- Java ;
- .NET ;
- ❖ fonctionnalités de reverse engineering pour documenter et mettre à jour des systèmes existants ;
- ❖ Une solution de référentiel d'entreprise avec des fonctionnalités de sécurité et de gestion des versions très complètes pour permettre un développement multiutilisateur ;
- ❖ fonctionnalités de génération et de gestion de rapports automatisés et personnalisables ;
- ❖ un environnement extensible, qui vous permet d'ajouter des règles, des commandes, des concepts et des attributs à vos méthodologies de modélisation et de codage.

2. Modélisation avec PowerAMC

PowerAMC fournit un jeu unique d'outils de modélisation professionnels qui associent les techniques et notations standard de la modélisation de processus métiers, de la modélisation des données et de la modélisation d'applications UML et d'autres fonctionnalités complexes afin d'aider à analyser, à concevoir, à construire et à maintenir des applications, en utilisant les techniques les plus élaborées d'ingénierie logicielle. La solution de modélisation PowerAMC permet d'intégrer étroitement la conception et la maintenance des couches de données centrales de l'application et exigences de projet, processus métiers, code orienté objet, vocabulaires XML et informations de réPLICATION de base de données. En mettant à disposition une série complète de modèles à tous les niveaux d'abstraction, PowerAMC permet d'élargir la portée du processus de conception itérative à tous les aspects de l'architecture du système, de la conception au déploiement, et même au-delà.

IX. ETUDE DETAILLE DE LA SOLUTION

1. Le diagramme des cas d'utilisation

Un diagramme de cas d'utilisation définit toutes les fonctionnalités du système à concevoir. Avant de présenter ces fonctionnalités, nous allons d'abord identifier les différents acteurs (entité interagissant avec le système).

i. Identification des acteurs

Nous distinguons deux catégories d'acteurs :

- ❖ les acteurs principaux (qui modifient l'état du système ou qui consultent cet état :
 - l'administrateur ;
 - l'agent commercial
 - le client.
- ❖ les acteurs secondaires (acteurs auxquels le système fait appel pour répondre aux sollicitations d'un acteur principal système de Gateway).

ii. Identification des cas d'utilisation

Les cas d'utilisation identifiés sont les suivants :

Abréviations :

AC : Agent commercial ;

Admin : Administrateur ;

SIWAD : Système d'Information Web d'Administration.

Cas d'utilisation	Acteur(s)
Gérer les inscriptions	Enregistrer un tiers
	Modifier un tiers
	Supprimer un tiers
	Consulter un tiers
Gérer les contrats	Valider un contrat
	Rompre un contrat
Gérer les historiques	Consulter une histoire
	Suivre l'évolution d'un client
Gérer les factures	Editer une facture
	Modifier une facture
	Imprimer une facture
	envoyer une facture
Gérer les paramètres	Consulter un paramètre
	Modifier un paramètre
Gérer son compte	Abonner
	Acheter unité
	Demander transfert
	Désabonner
	Consulter la consommation
Gérer les messages	Vérifier l'existence d'un client
	Vérifier l'état d'un client
	Modifier le solde d'un client
Gérer les messages	Demande emprunt
	Vérifier le message
	Envoyer les requêtes
	Envoyer accusé

Figure 6: Tableau récapitulatif des cas d'utilisation du système

iii. Les diagrammes de cas d'utilisation

-Système d'informations web d'administration(SIWAD)

PARTIE 2 : RAPPORT DE PRE-PROGRAMMATION

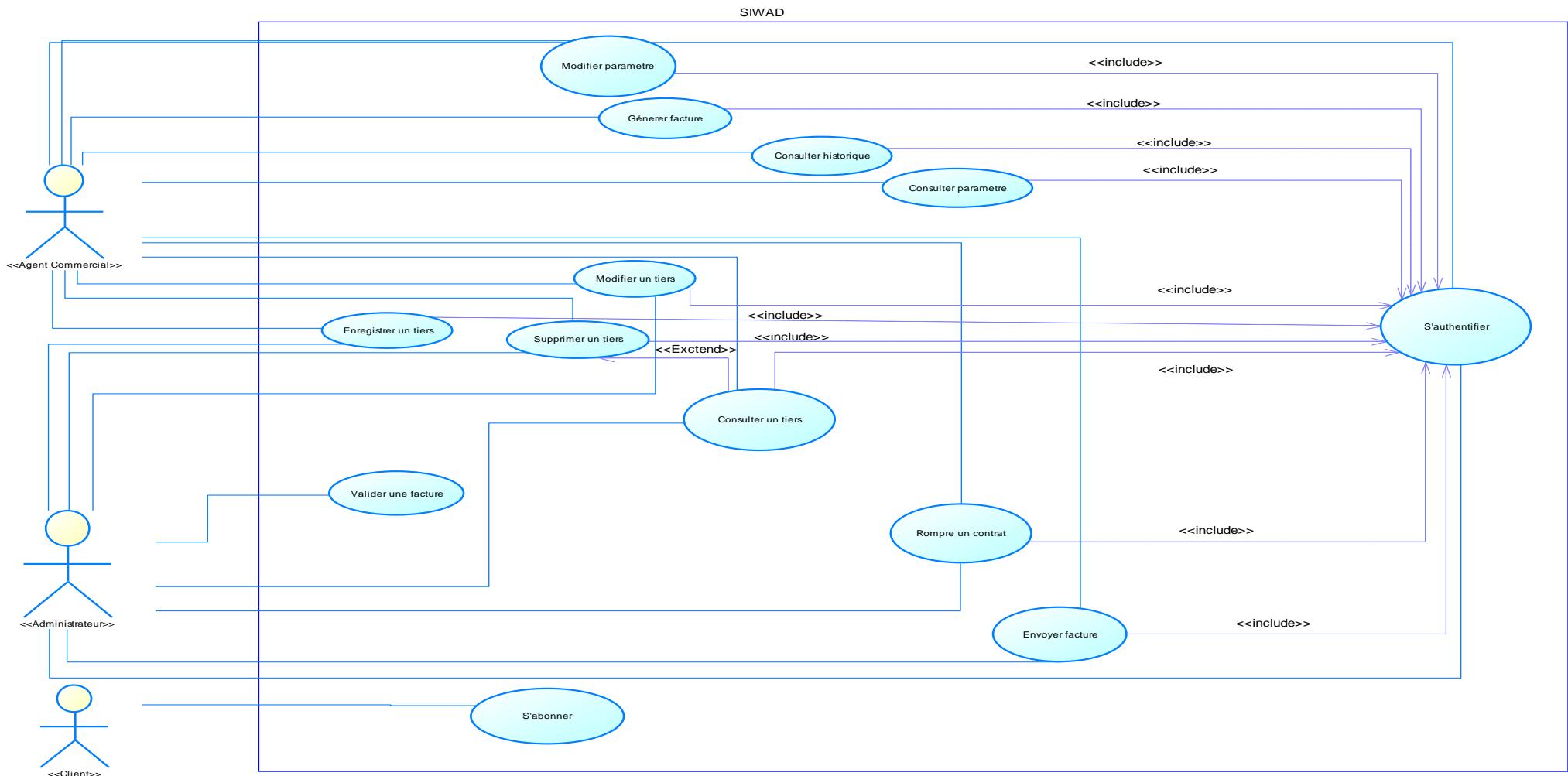


Figure 7: Diagramme de cas d'utilisation : SIWAD

-Système de gestion de messagerie(GATEWAY)

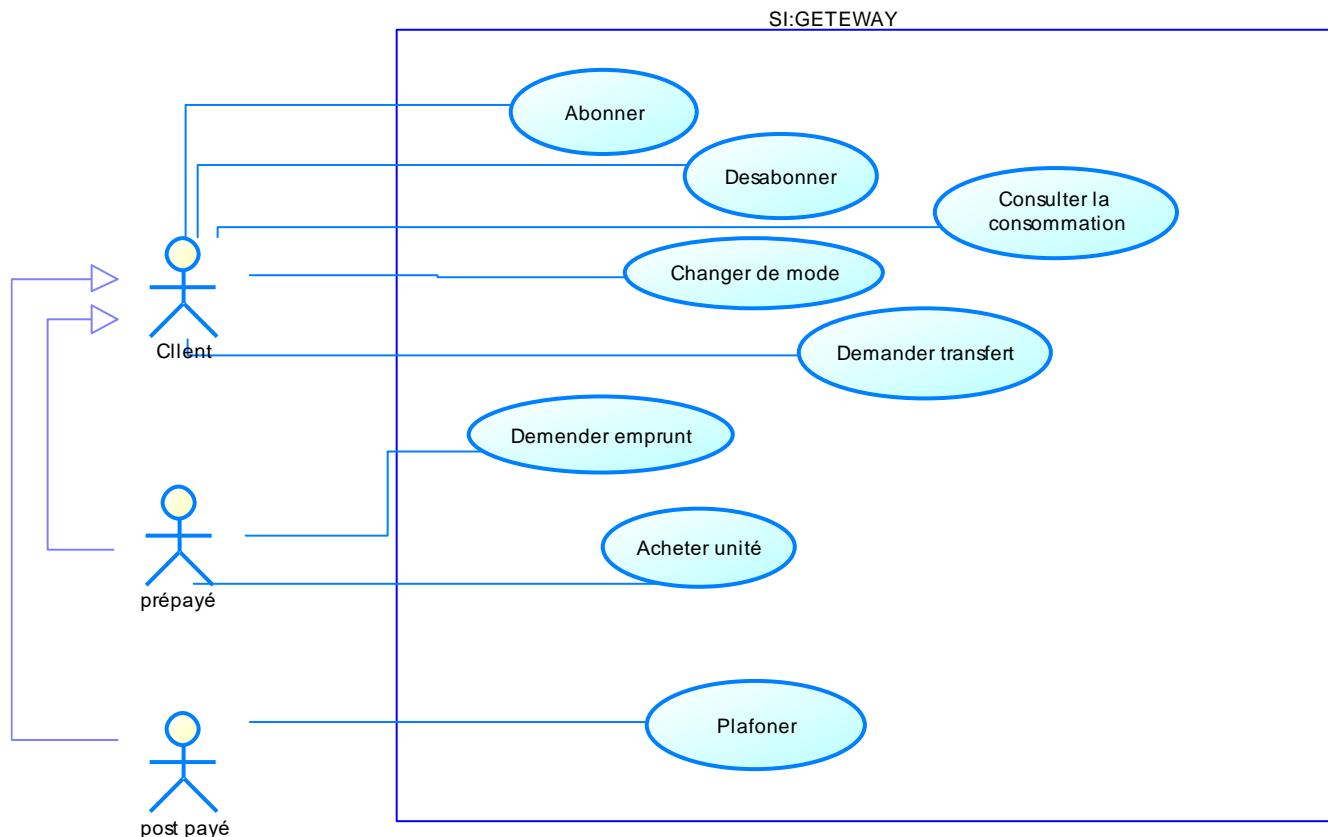


Figure 8: Diagramme de cas d'utilisation : GATEWAY

iv. Description textuelle des cas d'utilisation

a. S'inscrire

Titre : S'abonner

Acteurs : client

Résumé : Se faire reconnaître par le système

Responsable : ADUAYOM MESSAN

Version : 1.0

Date : 30/07/2016

Description des scénarii :

❖ Pré condition :

- ✓ Le système est fonctionnel ;
- ✓ Le client dispose d'un numéro de téléphone.

❖ Scénario nominal :

1. Le client fait une demande d'inscription ;
2. Le système vérifie le format du message ;
3. Le système envoie un message d'attente au client ;
4. L'administrateur valide l'inscription du client ;
5. Le système envoie un message de confirmation.

❖ Scénario d'exceptions :

• **E1 : Le client annule l'inscription**

- ✓ Ce scénario débute au point 1 du scénario nominal lorsque le client annule l'inscription après la demande ;
- ✓ Le scenario nominal est interrompu.

❖ Scénario alternatif :

• **A1 : Le format du message est incorrect**

- ✓ Un message lui est envoyé lui disant qu'il a annulé sa demande ;
- ✓ Le client est invité sur la plateforme web après trois (3) erreurs.

❖ Post conditions :

- ✓ Le client est inscrit ;
- ✓ Le journal est mis à jour ;

- ✓ Le système est toujours fonctionnel.

b. S'authentifier

Titre : S'authentifier

Acteurs : Agents, Administrateur

Résumé : Permet à l'utilisateur de se connecter à la plateforme d'administration du système

Responsable : ADUAYOM MESSAN

Version : 1.0

Date : 30/07/2016

Description des scénarii :

❖ Pré condition :

- ✓ Le système est fonctionnel ;
- ✓ La page de connexion est accessible.

❖ Scénario nominal :

1. L'utilisateur renseigne les informations de connexion ;
2. L'utilisateur valide les informations renseignées ;
3. Le système vérifie la conformité des informations renseignées ;
4. Le système affiche la page d'accueil selon les droits qui sont assignés à l'utilisateur.

❖ Scénario d'exceptions :

• **E1 : L'utilisateur annule la connexion**

- ✓ Ce scénario débute au point 1 du scénario nominal lorsque l'utilisateur annule la connexion ;
- ✓ Le scenario nominal est interrompu.

• **E2 : L'utilisateur saisie les informations de connexion erronées à (3) reprises**

- ✓ Ce scénario commence au point 3 du scénario nominal lorsque le système remarque que les informations saisies sont incorrect ;
- ✓ L'utilisateur est informé de l'épuisement des essais
- ✓ Le système quitte la page de connexion ;
- ✓ Le système verrouille momentanément l'accès à cet utilisateur (compte inactif).

❖ **Scénario alternatif :**

• **A1 : Les informations sont erronées moins de 3 fois**

- ✓ Il débute à l'étape 3 du scenario nominal lorsque l'utilisateur saisi des informations erronés ;
- ✓ L'utilisateur reçoit un message qui lui affiche le nombre d'essais restants.

❖ **Scénario d'exception :**

• **E1 : Les informations sont erronées plus de 3 fois**

- ✓ Ce scenario débute à l'étape 3 lorsque les informations erronées dépassent 3 essais ;
- ✓ Le système informe l'utilisateur que le nombre d'essais maximum est atteint et lui demande de contacter l'administrateur pour plus d'informations ;
- ✓ Le compte est désactivé.

❖ **Post conditions :**

- ✓ L'utilisateur est connecté ;
- ✓ Le journal est mis à jour ;
- ✓ Le système est toujours fonctionnel.

c. **Faire une demande de transfert**

Titre : Faire une demande de transfert

Acteurs : Client

Résumé : Permet aux clients de pouvoir effectuer un transfert vers un numéro donné

Responsable : ADUAYOM MESSAN

Version : 1.0

Date : 30/07/2016

Description des scénarii :

❖ **Pré condition :**

- ✓ Le système est fonctionnel ;
- ✓ Le client doit disposer d'un numéro de téléphone.

❖ **Scénario nominal :**

1. Le client envoie un message définissant ses besoins ;

2. Le système vérifie l'existence du numéro ;
3. Le système vérifie la conformité du message reçu ;
4. Le système vérifie le solde du client ;
5. Le système débite le compte du client.

❖ **Scénario d'exceptions :**

- **E1 : Le client n'existe pas dans la base**
 - ✓ Ce scénario débute au point 1 du scénario nominal lorsque l'utilisateur envoi sa demande ;
 - ✓ Le système lui envoie un message l'invitant à s'inscrire d'abord ;
 - ✓ Le scenario nominal est interrompu.
- **E2 : Le format du message est incorrect**
 - ✓ Ce scénario commence au point 3 du scénario nominal lorsque le système remarque que le format de message n'est pas respecté ;
 - ✓ Le système envoie un message informant le client de l'erreur de formatage et lui propose le bon format.
- **E3 : Le solde du client inférieur à la demande**
 - ✓ Ce scenario débute au point 4 du scenario nominal lorsque le system remarque que la demande du client est supérieure au solde ;
 - ✓ Le système demande au client s'il veut faire un prêt en fonction du type de compte ;
 - ✓ Selon le choix du client le scenario continuera ou pas.

❖ **Scénario alternatif :**

- **A1 : Le client refuse le prêt**
 - ✓ Il débute à l'étape 4 si le solde du client est inférieur à la demande ;
 - ✓ Le scenario nominal est interrompu.
- **A2 : Le client accepte le prêt**
 - ✓ Il débute à l'étape 4 si le solde du client est inférieur à la demande ;
 - ✓ Le scenario nominal continu jusqu'à la fin.

❖ **Post conditions :**

- ✓ La demande du client est satisfaite ;
- ✓ Le système envoi un accusé au client ;

- ✓ Le journal est mis à jour ;
- ✓ Le système est toujours fonctionnel.

2. Les diagrammes de séquences

Ils permettent de décrire COMMENT les éléments du système interagissent entre eux et avec les acteurs.

Indication :

SIWAD : Système d'Information Web d'Administration.

- S'abonner

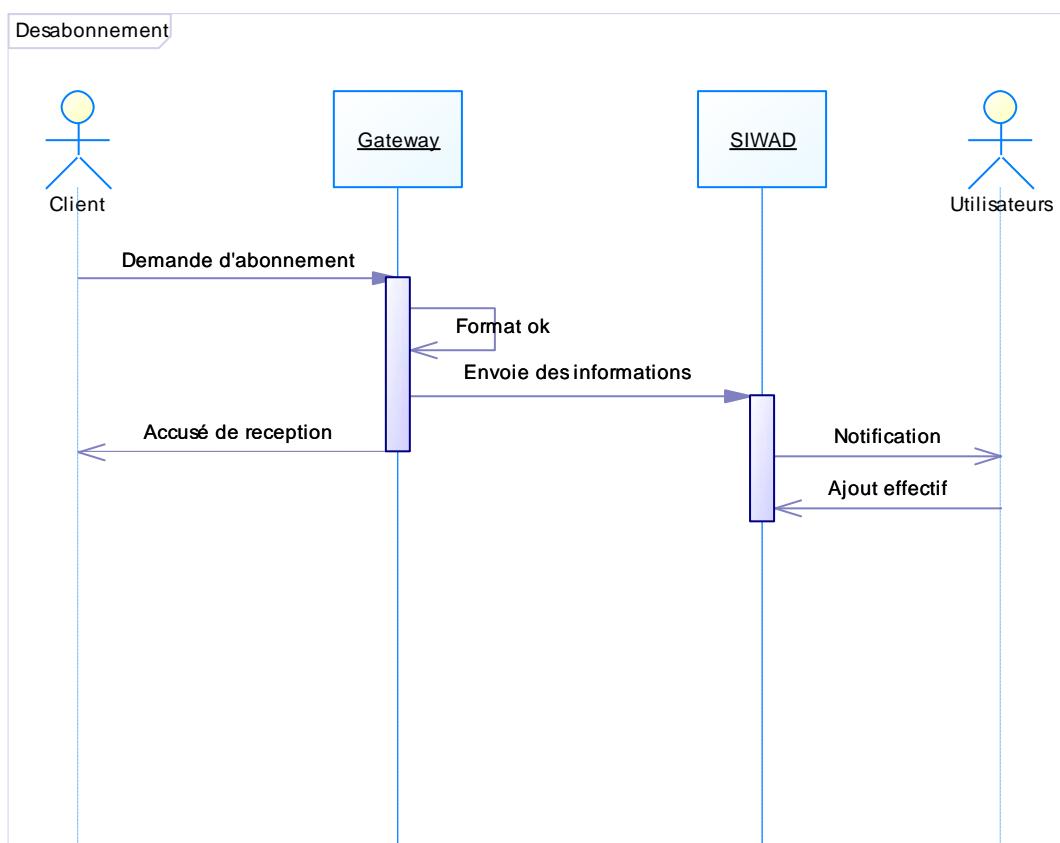


Figure 9: Diagramme de séquence : S'abonner

- S'authentifier

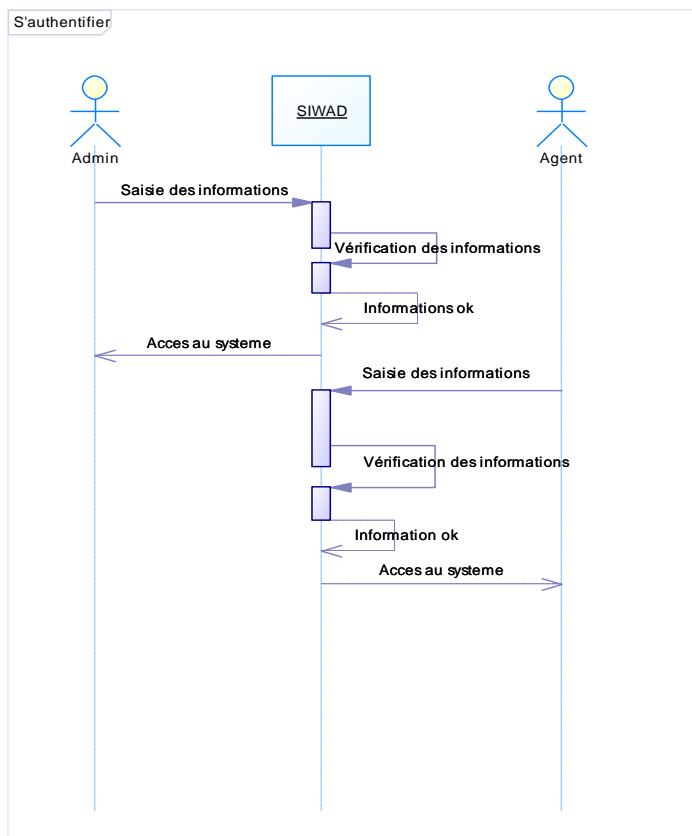


Figure 10: Diagramme de séquence : S'authentifier

- Faire un Transfert

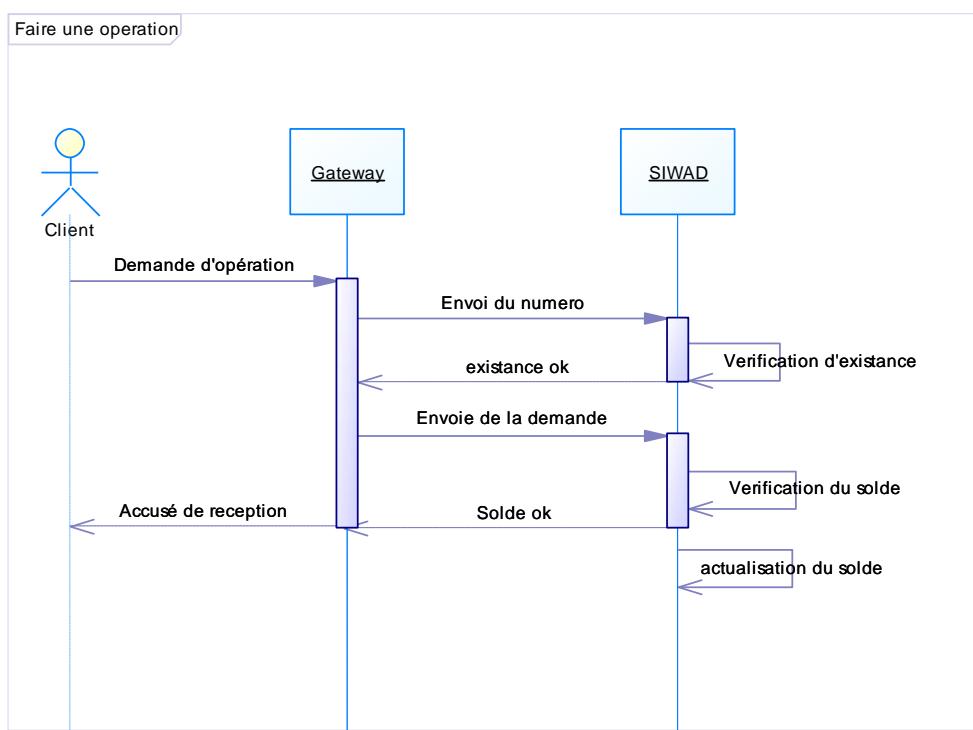


Figure 11: Diagramme de séquence : Faire une Opération

- Désabonnement

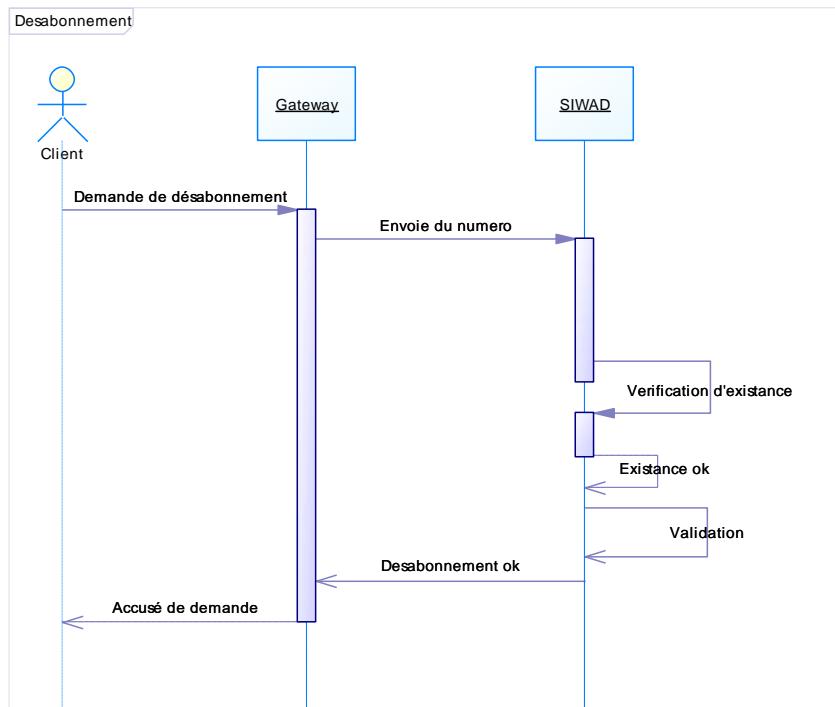


Figure 12: Diagramme de séquence de désabonnement

3. Le diagramme d'activités

Le diagramme d'activités permet de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

- S'abonner

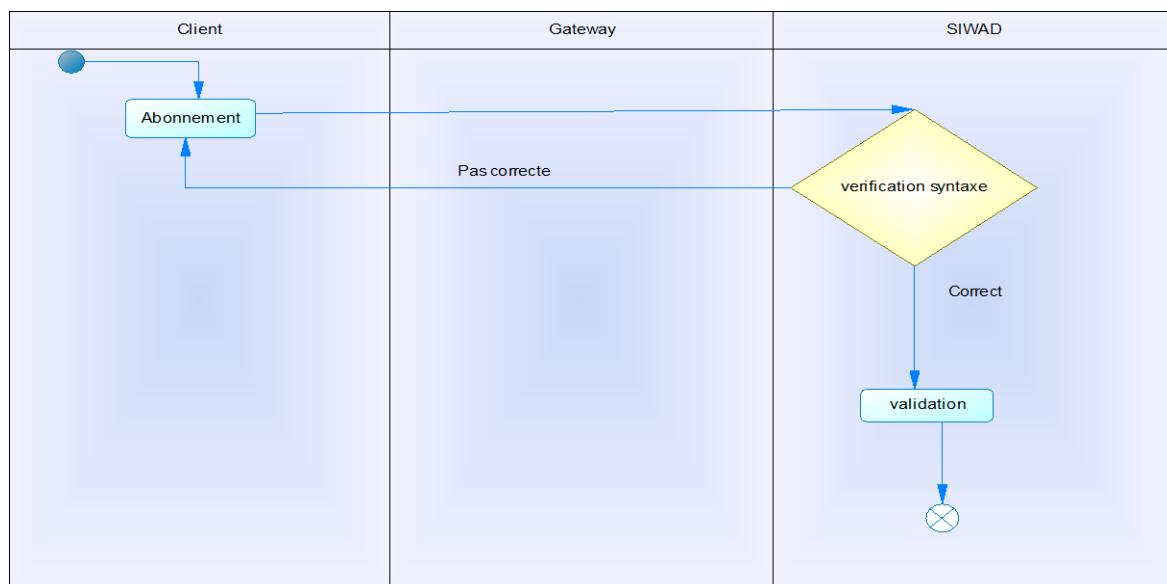


Figure 13: Diagramme d'activité : S'abonner

- Demande

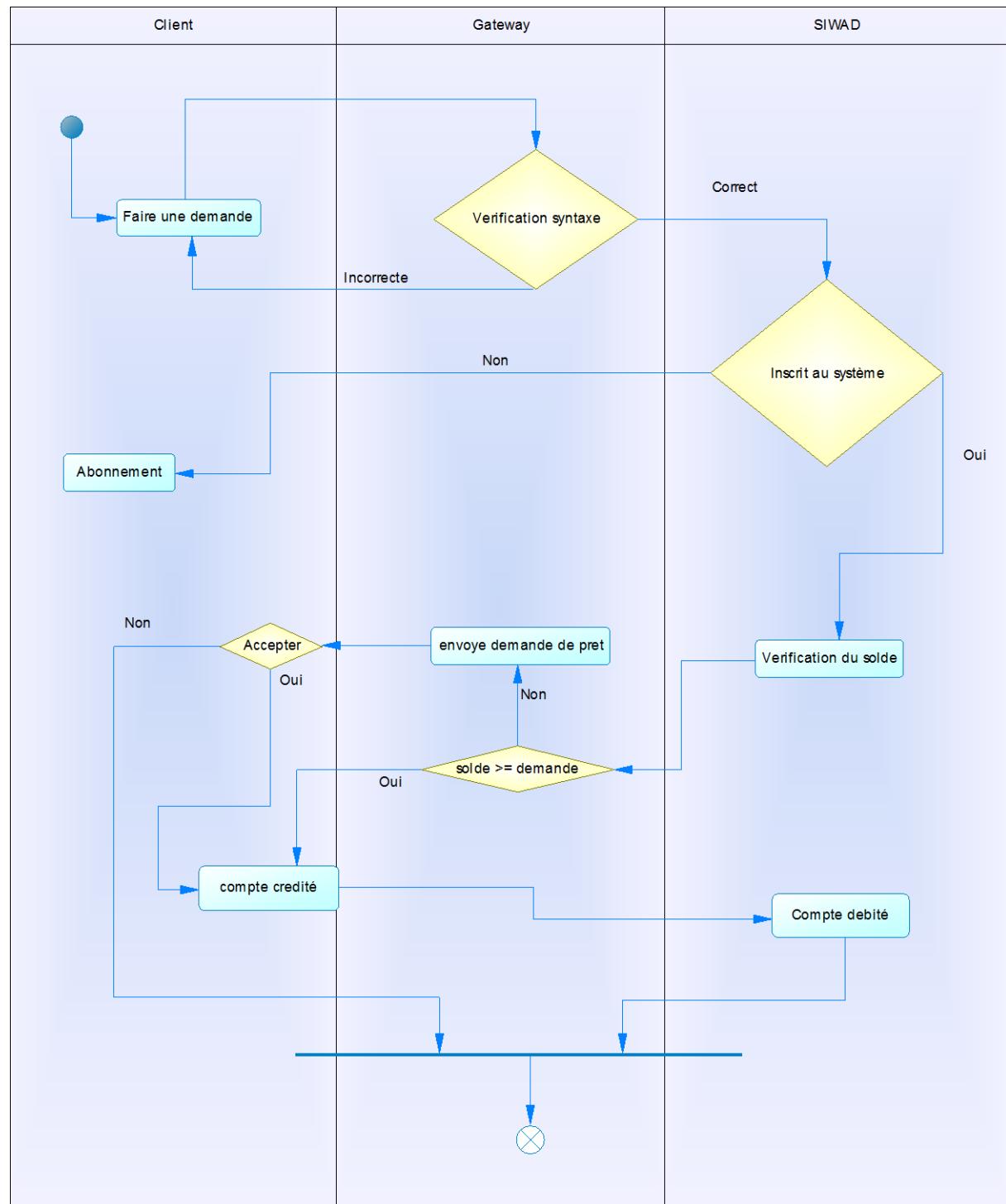


Figure 14: Diagramme d'activité : Demande

4. Le diagramme de classe

- Ils permettent de spécifier la structure et les liens entre les objets dont le système est composé : ils spécifient QUI sera à l'œuvre dans le système pour réaliser les fonctionnalités décrites par les diagrammes de cas d'utilisation.

PARTIE 2 : RAPPORT DE PRE-PROGRAMMATION

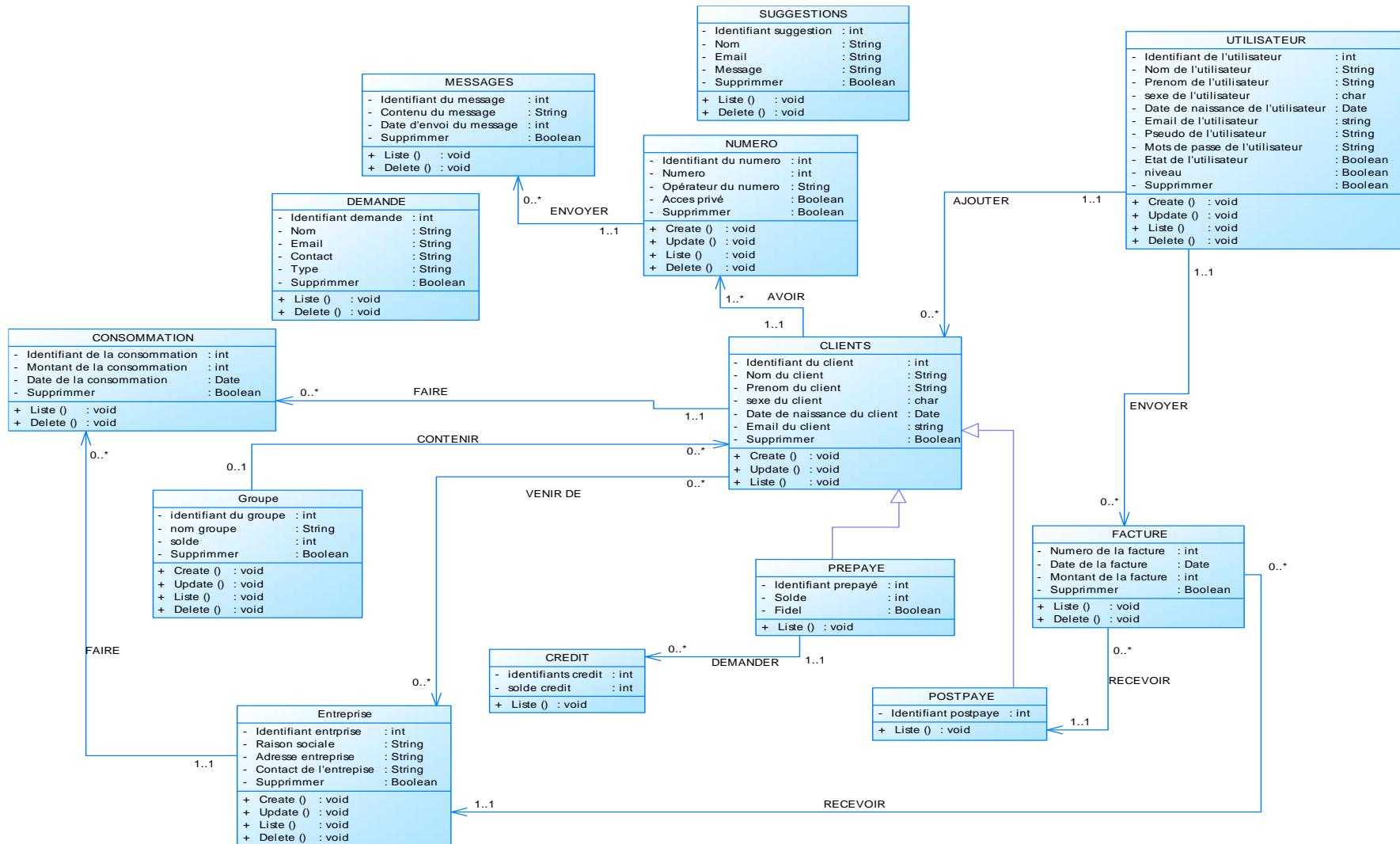


Figure 15: Diagramme de classes

CONCLUSION

Nous venons de terminer la phase de conception ou pré-programmation. Nous allons entamer naturellement la phase de réalisation ou programmation, qui consistera à expliquer les différentes étapes de la programmation.

PARTIE 3 : RAPPORT DE REALISATION ET MISE EN ŒUVRE

Du 01 Aout 2016 Au 13 Septembre 2016

INTRODUCTION

Ce document reflète non seulement la structure interne (les lignes de codes) de notre système mais aussi sa structure externe (l'apparence). On présentera de même l'architecture de du système et le script de création de la base de données.

I. MISE EN ŒUVRE

1. Choix matériels

Pour le développement de notre système, il nous faudra :

- Pour l'approche avec modem sans API
 - d'une carte Sim ;
 - d'un modem ;
 - d'un ordinateur fonctionnel 24h/24h et 7jrs/7 ;
 - d'un hébergeur web ;
 - d'un onduleur ;
 - d'un groupe électrogène.
- Pour l'approche sans modem avec API
 - d'un hébergeur web ;
 - d'un numéro court ;
 - l'accès à l'API de chaque opérateur.

Ce système doit non seulement avoir ces matériels mise à sa disposition mais aussi le développement de ce système doit suivre les normes techniques pour permettre sa maintenance et son évolution.

2. Choix Logiciels

i. Langages

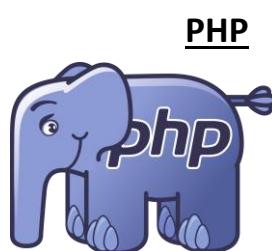


Figure 16: Logo PHP

Pour la partie WEB d'administration et de publicité nous avons jugé bon d'utiliser du PHP parce qu'il est principalement utilisé pour produire des pages WEB dynamiques via un serveur HTTP.

ASP.NET



Figure 17: Logo ASP.NET

ASP.NET est une plateforme Web qui offre tous les services dont vous avez besoin pour générer des applications Web basées sur des serveurs de classe d'entreprise. ASP.NET repose sur le .NET Framework ; toutes les fonctionnalités du .NET Framework sont donc disponibles pour les applications ASP.NET. Vos applications peuvent être écrites dans tout langage compatible avec le Common Language Runtime (CLR), y compris Visual Basic et C#.

Anis nous avons décidé d'utiliser l'ASP.NET pour écrire la partie service de notre système.

ii. Outils et technologies

LARAVEL



Figure 18: Logo LARAVEL

Laravel sera le logiciel que nous allons utiliser pour gérer la partie administration de notre système.

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orienté objet. Laravel est distribué sous licence MIT avec ses sources hébergées sur Git hub.

La cinquième version de Laravel est écrite en PHP 5 et son installation est basée sur le gestionnaire de paquets Composer. Laravel fournit des fonctionnalités en termes de routage de requêtes, de mapping objet-relationnel (un système baptisé Eloquent implémentant Active Record), d'authentification, de vue (avec Blade), de migration de base de données, de gestion des exceptions et de tests unitaires. Pour la réalisation de la plateforme web de notre application, nous avons opté pour l'utilisation du Framework Laravel.

Services XML-RPC

XML-RPC est un protocole RPC (Remote procedure call), une spécification simple et un ensemble de codes qui permettent à des processus s'exécutant dans des environnements différents de faire des appels de méthodes à travers un réseau.

XML-RPC permet d'appeler une fonction sur un serveur distant à partir de n'importe quel système (Windows, Mac OS X, GNU/Linux) et avec n'importe quel langage de programmation. Le serveur est lui-même sur n'importe quel système et est programmé dans n'importe quel langage. Cela permet de fournir un Service web utilisable par tout le monde sans restriction de système ou de langage.

Les processus d'invocation à distance utilisent le protocole HTTP pour le transfert des données et la norme XML pour la structuration des données.

XML-RPC est conçu pour permettre à des structures de données complexes d'être transmises, exécutées et renvoyées très facilement. XML-RPC est une alternative aux Services Web WS-*, dont SOAP.

Services RESTFUL

Nous allons utiliser cette technologie avec du PHP.

RESTFUL est juste un adjectif qui qualifie une architecture REST nous allons donc présenter plutôt le REST. REST (Representational State Transfer) est l'un de ces acronymes qui ne représente pas une technologie comme peuvent l'être Ajax, DHTML, Web 2.0 et autres. REST est un style d'architecture qui repose sur le protocole HTTP : On accède à une ressource (par son URI unique) pour procéder à diverses opérations (GET lecture / POST écriture / PUT modification / DELETE suppression), opérations supportées nativement par HTTP.

NOWSMS

NowSms est l'outil consacré à la gestion de la partie Gateway du système.

NowSms est une solution pour l'envoi et la réception des SMS ou de MMS via un modem GSM connecté à un pc via un port USB.

Les fonctions et avantages de Nowsms sont multiples :

- ❖ Permettre l'envoi des MMS et des SMS ;
- ❖ Utilisable sous Windows et sur des serveurs Windows ;
- ❖ Utilise un ou plusieurs modems GSM ou n'importe quelle connexion TCP/IP utilisant des protocoles comme SMPP, UCP/EMI et ou HTTP du côté du client et des protocoles comme SMTP, SMPP, HTTP du côté serveur ;
- ❖ Notons tout de même que Nowsms ne remplace en aucun cas les services de messagerie ;
- ❖ peut s'utiliser avec internet (d'où les protocoles SMPP, HTTP...) ou dans un réseau local ou privé.

GIT



Figure 19: Logo Git

Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2. En 2016, il s'agit du logiciel de gestion de versions le plus populaire qui est utilisé par plus de douze millions de personnes.

Il nous a permis de mettre en place le Versioning des codes sources et documents de notre application. Pour son implémentation sur le web, nous avons utilisé Bitbucket, plateforme présentée à la suite.

BITBUCKET



Figure 20: Logo BitBucket

Bitbucket est un service web d'hébergement et de gestion de développement logiciel utilisant les logiciels de gestion de versions Git (logiciel de gestion de versions décentralisé) et Mercurial (logiciel de gestion de versions décentralisé disponible) par l'intermédiaire de logiciels comme SourceTree.

Bitbucket est une plateforme web qui permet d'effectuer le clouding. Un petit rappel, au sein de notre entreprise nous avons décidé d'opter pour la méthode AGILE ce qui implique un travail de groupe, Bitbucket nous permet alors de pouvoir travailler quelques soit la distance de suivre l'évolution de nos codes. Il nous permet d'effectuer le Versionning et de faciliter le suivi et le développement du projet en groupe.

A screenshot of the Bitbucket dashboard. At the top, there's a navigation bar with links for Bitbucket, Teams, Projects, Repositories, and Snippets, along with a search bar and user profile icons. Below the navigation is a header titled 'Dashboard'. Underneath is a sub-header with tabs for Overview, **Repositories**, Pull requests, Issues, and Snippets. A 'FILTER BY:' dropdown is set to 'Owner' and 'Project'. To the right of the tabs is a search bar labeled 'Find repositories...'. The main area shows a table of repositories with columns for Repository, Project, Owner, and Last updated. The table lists five repositories: 'ConsoPourTous', 'ROBOT-AGENCE', 'Test', 'Test2', and 'Testdiyama'. On the right side of the dashboard, there are two sections: 'Latest updates' and 'Recent activity'. The 'Latest updates' section highlights 'Improved branch permissions' and 'Code review-focused dashboard'. The 'Recent activity' section shows a single commit from 'ADUAYOM_MESSAN_Eko...' with the message 'Pushed to robot_agence/robot-agence' and the date 'yesterday'.

Figure 21: Interface Bitbucket

iii. Environnements de Développement Intégré (EDI)

PHP Storm



Figure 22: Logo PHP Storm

PHP Storm est un éditeur pour PHP, HTML et JavaScript, édité par JetBrains. Nous l'avons spécialement choisi parce qu'il nous permet d'intégrer le Versionning. Ses avantages sont multiples, il propose :

- Une coloration syntaxique ;
- Affichage des erreurs à la volée ;
- Auto-complétion intelligente du code ;
- Ré usinage du code ;
- L'envoi des fichiers via FTP pour le déploiement;
- Un gestionnaire de version.

Il permet aussi de visualiser l'architecture de bases de données de différentes sources (MySQL, SQLite, ...).

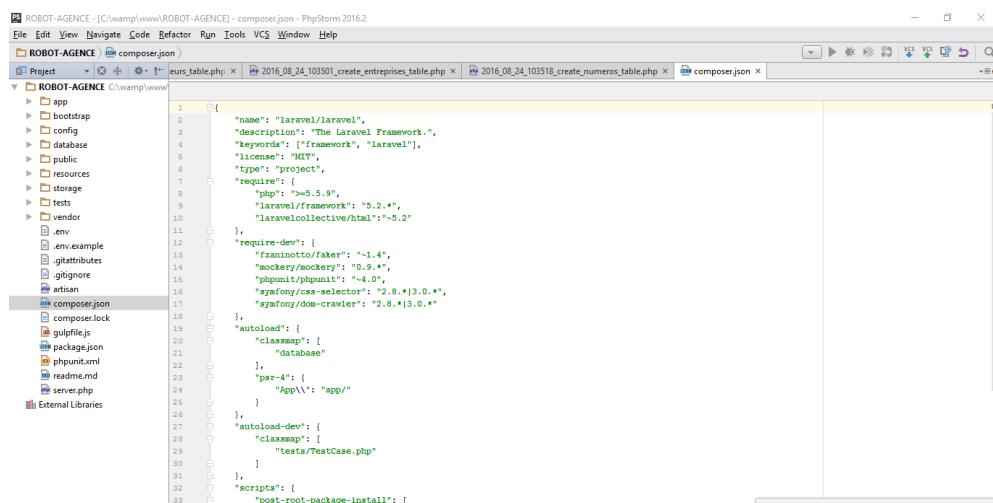


Figure 23: Interface PHP Storm

Visual Studio



Figure 24: Logo Visual Studio

Visual Studio vas nous permettre d'implémenter la partie service de notre système.

Microsoft Visual Studio est une suite de logiciels de développement pour Windows conçue par Microsoft. La dernière version s'appelle Visual Studio 2015.

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# utilisent tous le même environnement de développement intégré (IDE), qui leur permet de partager des outils et facilite la création de solutions faisant appel à plusieurs langages. Par ailleurs, ces langages permettent de mieux tirer parti des fonctionnalités du Framework.NET, qui fournit un accès à des technologies clés simplifiant le développement d'applications web ASP et de services web XML grâce à Visual Web Développer.

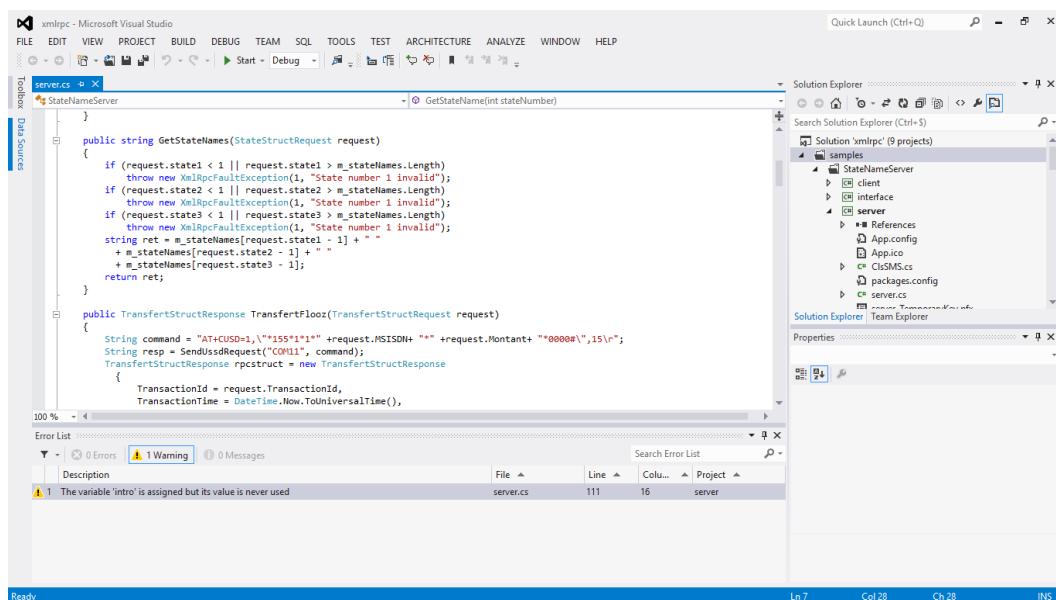


Figure 25: Interface Visual Studio

iv. Base de données

MySQL



Figure 26:Logo MySQL

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server.

Son nom vient du prénom de la fille du Co créateur Michael Widenius, My. SQL fait référence au *Structured Query Language*, le langage de requête utilisé.

MySQL LAB a été acheté le 16 janvier 2008 par Sun Microsystems.

Depuis mai 2009, son créateur Michael Widenius a créé Maria DB pour continuer son développement en tant que projet Open Source.

3. Sécurité de l'application

- Une gestion de l'historique très ferme qui pourra nous permettre de répondre à une quelconque plainte d'un client. N'oublions pas de même la gestion de la sécurité soit de privilèges et de niveaux d'accès pour les différents types d'utilisateurs (Agents commerciaux, administrateurs). Selon leur statut, le contenu des pages varie et l'accès aux informations avec un statut supérieur est interdit. L'identification des différents intervenants devra s'exécuter par un service central d'authentification.
- Chaque opération qui sera effectué par le client sera automatiquement sauvegardée. On pourra suivre à la trace les activités du client pour permettre une gestion plus aiguisee des clients.

- Auront accès au système seul les ayant accès, un service d'authentification est mise en place à cet effets.
- L'application de gestions Web administrative est elle-même hébergé sur un serveur web distant et les données sensibles seront sécurisées par des protocoles de Hachages comme le md5.
- La communication entre les différents serveurs, services et parties du système se fera le protocole HTTPS pour garantir la sécurité des transactions et palier aux risques de piratage.

I- PRESENTATION DE L'APPLICATION

1. Présentation

Robot agence est un système pourra permettre aux clients qui font partie du système de pouvoir effectuer automatiquement sans l'intervention humaine des transferts d'unités et de forfaits internet. Il proposera aussi une gestion efficace des clients en présentant toutes les opérations qu'un client aura à effectuer ainsi que toutes les factures qui lui seront délivrées.

Nous allons vous présenter certaines interfaces de notre application :

- La page d'accueil

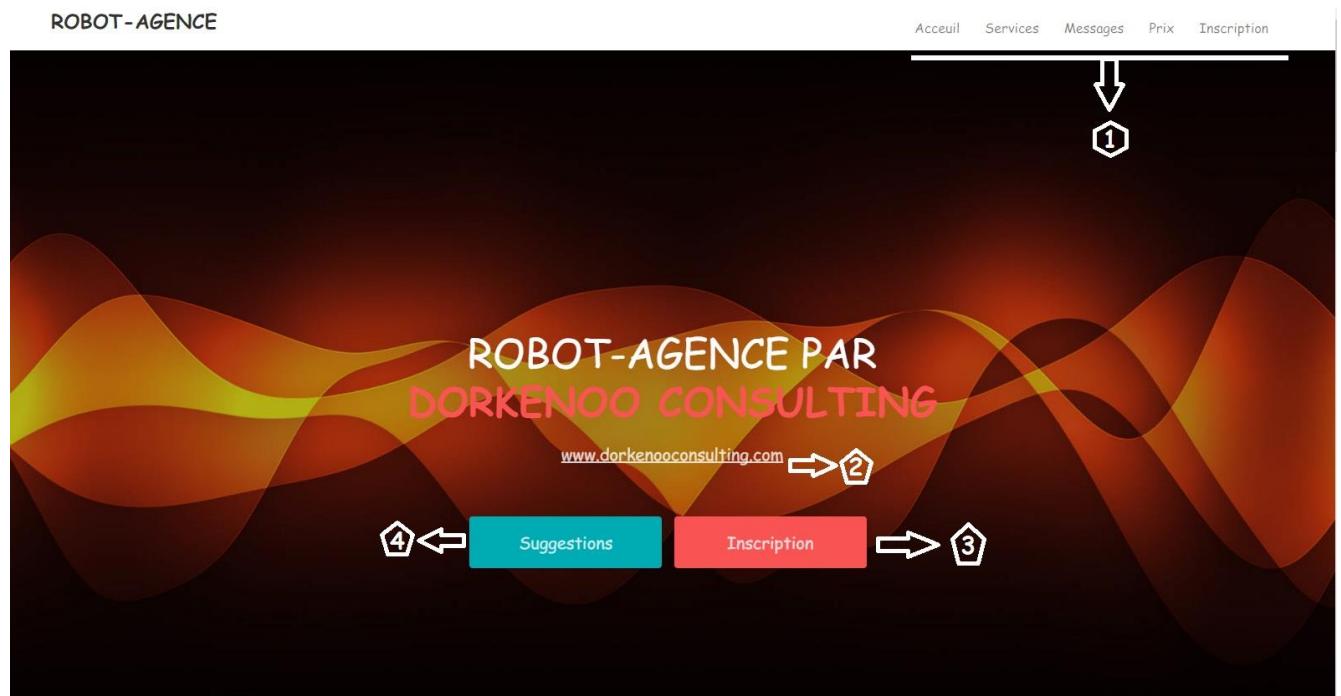


Figure 27: Première section d'accueil

1. Onglet encré renvoyant vers les différentes sections de la page ;
2. Lien renvoyant vers le site de la société ;
3. Bouton renvoyant vers le formulaire de demande d'inscription ;
4. Bouton renvoyant vers le formulaire de suggestions.

- **Page d'accueil publicité**



Figure 28: Deuxième section d'accueil

Présente un peu les avantages du système.

- **Page d'accueil format d'un message**

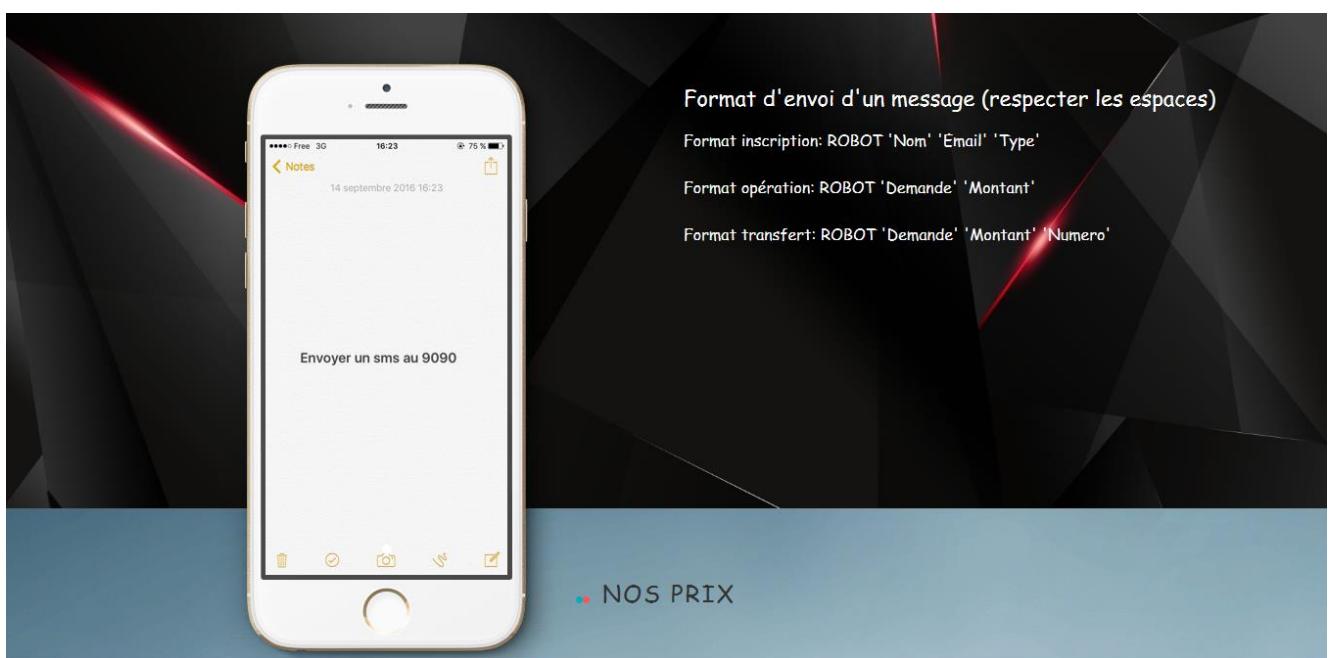


Figure 29: Troisième section d'accueil

- **Page d'accueil contact**



Figure 30: Pied de la page d'accueil

1. Contacts de l'entreprise ;
2. Formulaire de suggestions.

- **La page de connexion**

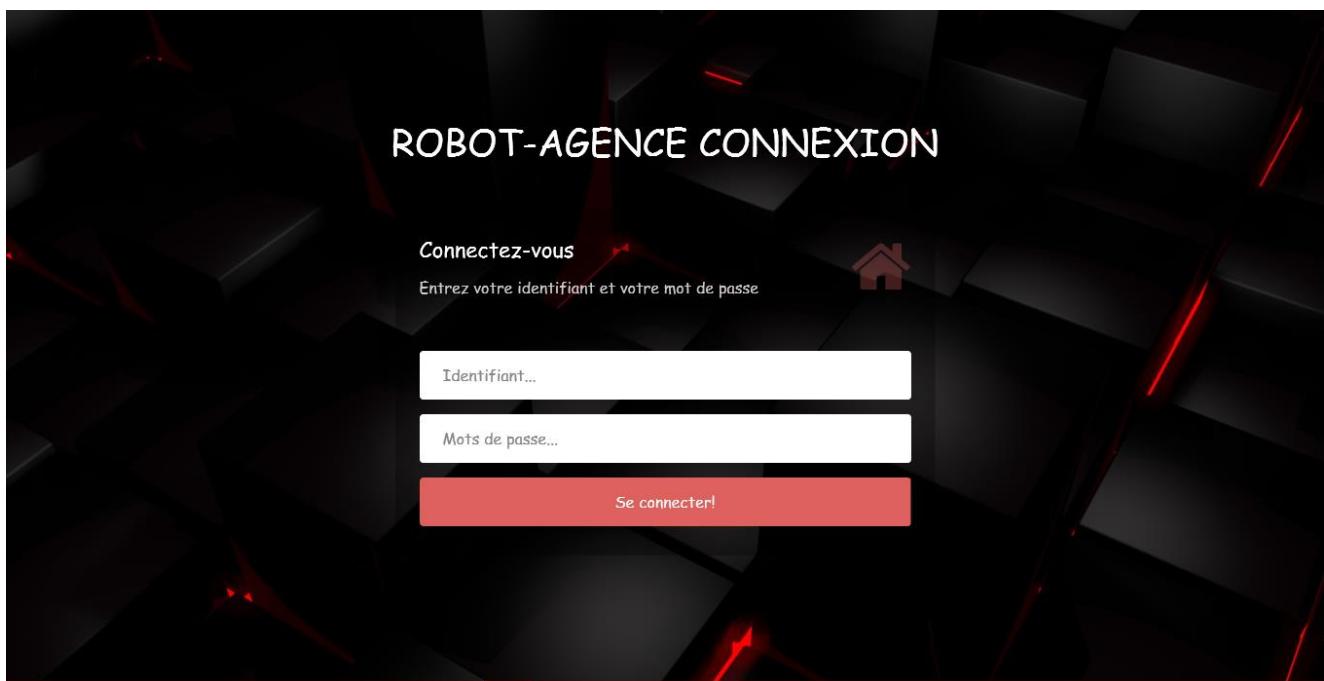


Figure 31: Page de connexion

PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE

- Page d'accueil administrateur

The screenshot shows the 'Espace Administrateur' dashboard. On the left, there's a sidebar with a user profile (ADUAYOM MESSAN, enoc@dorkenoconsulting.com) and a menu with items like Accueil, Liste des Clients, Consommations, Liste des operations, Statistiques, Services, Unites, and Forfait. The main area is titled 'Tableaux de bord' and contains several cards: 'NOUVEAU CLIENT', 'LISTE SUGGESTIONS', 'LISTE OPERATIONS', 'NOUVEAU PERSONNEL', 'NOUVEL ENTREPRISE', 'NOUVEAU GROUPE', 'LISTE DES CONSOMMATIONS', 'STATISTIQUES', 'FIDELISER', 'EDITER FACTURE', 'LISTE DU PERSONNEL', and 'HISTORIQUE DU PERSONNEL'. A copyright notice at the bottom left reads '© 2016 Admin Robot-Agence.'

Figure 32: Page d'accueil Administrateur

- Le formulaire d'ajout d'un client

The screenshot shows the 'Renseigner les informations' form for adding a new client. It includes fields for Nom, Prenom, Email, Date de naissance (with a placeholder 'jj/mm/aaaa'), Mode (with a dropdown arrow), SEXE (radio buttons for Masculin and Feminin), and a 'enregistrer' button. A large blue arrow points downwards towards the 'enregistrer' button, which is circled with a red outline.

Figure 33: Exemple de formulaire d'ajout

1. Renvoi vers la liste de choix de mode ci-dessous.



Figure 34: Liste de choix de mode

- Le formulaire d'ajout d'un nouveau numéro

Figure 35: Formulaire d'ajout de numéro

1. Renvoi la liste des clients vus qu'un client peut avoir plusieurs numéros sur une requête exemple ci-dessous.

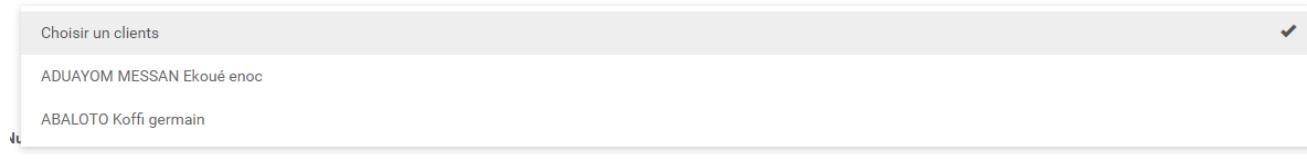


Figure 36: Liste de choix de clients

2. Renvoie la liste des opérateurs mobiles.

- Le formulaire de modification des utilisateurs

The screenshot shows a user modification form titled "Renseigner les informations". It contains the following fields:

- Nom: ADUAYOM MESSAN
- Prénom: Ekoué enoc
- Adresse email: enoc@dorkenoconsulting.com
- Nom de famille: BARON
- Mot de passe: (empty field)
- Date de naissance: 05/04/1995
- Rôle: Administrateur
- Sexe: Masculin féminin

At the bottom left is a blue "Modifier" button.

Figure 37: Formulaire de modification

- La liste des consommations

The screenshot shows a consumption list interface with the following structure:

Espace Administrateur

LISTE DES CONSOMMATIONS

TOUTES LES CONSOMMATIONS

Nom de l'entreprise	Nom du client	Montant	date Heure
DORKENO CONSULTING	ADUAYOM MESSAN	50000	2016-09-01 22:02:18
Acune	ABALOTO	30000	2016-09-01 22:02:43

Menu (Left Sidebar):

- Accueil
- Liste des Clients
- Consommations
- Liste des opérations
- Statistiques
- Services
 - Unités
 - Forfait
 - Gestion

© 2016 Admin Robot-Agence.

Figure 38: Liste des Consommations

- **La liste des clients**

The screenshot shows the 'Espace Administrateur' interface. On the left, there is a sidebar with a user profile (ADUAYOM MESSAN, enoc@dorkenoconsulting.com), a menu with 'Accueil', 'Liste des Clients', 'Consommations', 'Liste des opérations', 'Statistiques', and 'Services' sections, and a footer with '© 2016 Admin Robot-Agence'. The main content area is titled 'LISTE DES CLIENTS' and 'TOUS LES CLIENTS'. It displays a table with two rows of client data:

Nom	Prenom	Email	Mode	Date de naissance	Editor
ADUAYOM MESSAN	Ekoué enoc	enoc@dorkenoconsulting.com	Postpayé	1995-04-05	
ABALOTO	Kaffi germain	germain@yahoo.com	Prepayé	1995-03-16	

Figure 39: Liste des clients

1. Suppression d'un client ;
 2. Modification d'un client ;
- **La liste des messages**

The screenshot shows the 'Espace Administrateur' (Administrator Space) interface. On the left is a sidebar menu with the following items:

- Accueil
- Liste des Clients
- Consommations
- Liste des opérations
- Statistiques
- Services
 - Unités
 - Forfait
 - Gestion

Below the sidebar, the main area is titled 'LISTE DES MESSAGES' and 'TOUS LES MESSAGES'. It displays a table with the following data:

Nom du client	Contenu	Date Heure
ADUAYOM MESSAN	TRANSFERT 10000	2016-09-02 00:48:22
ADUAYOM MESSAN	TRANSFERT 50000	2016-09-02 00:49:08
ABALOTO	TRANSFERT 500	2016-09-02 00:49:37

Figure 40: Liste des messages

2. Architecture de l'application

Notre système se subdivise en trois parties :

- **La partie administration** : cette partie regroupe toutes les fonctionnalités qui ont trait à la gestion, l'édition des factures et le suivi des historiques. Il s'agit là d'une plateforme d'administration que nous avons nommée « **SIWAD** ».
- **La partie GATEWAY** : cette partie se charge de d'effectuer tous les traitements en rapport avec le format des messages, l'envoi mais aussi de la réception des messages et de certains contrôles notamment l'existence d'un numéro dans la base de données.
- **La partie Service** : cette partie est le pivot de tout notre système les opérations effectives de transfert de crédits s'effectueront là.

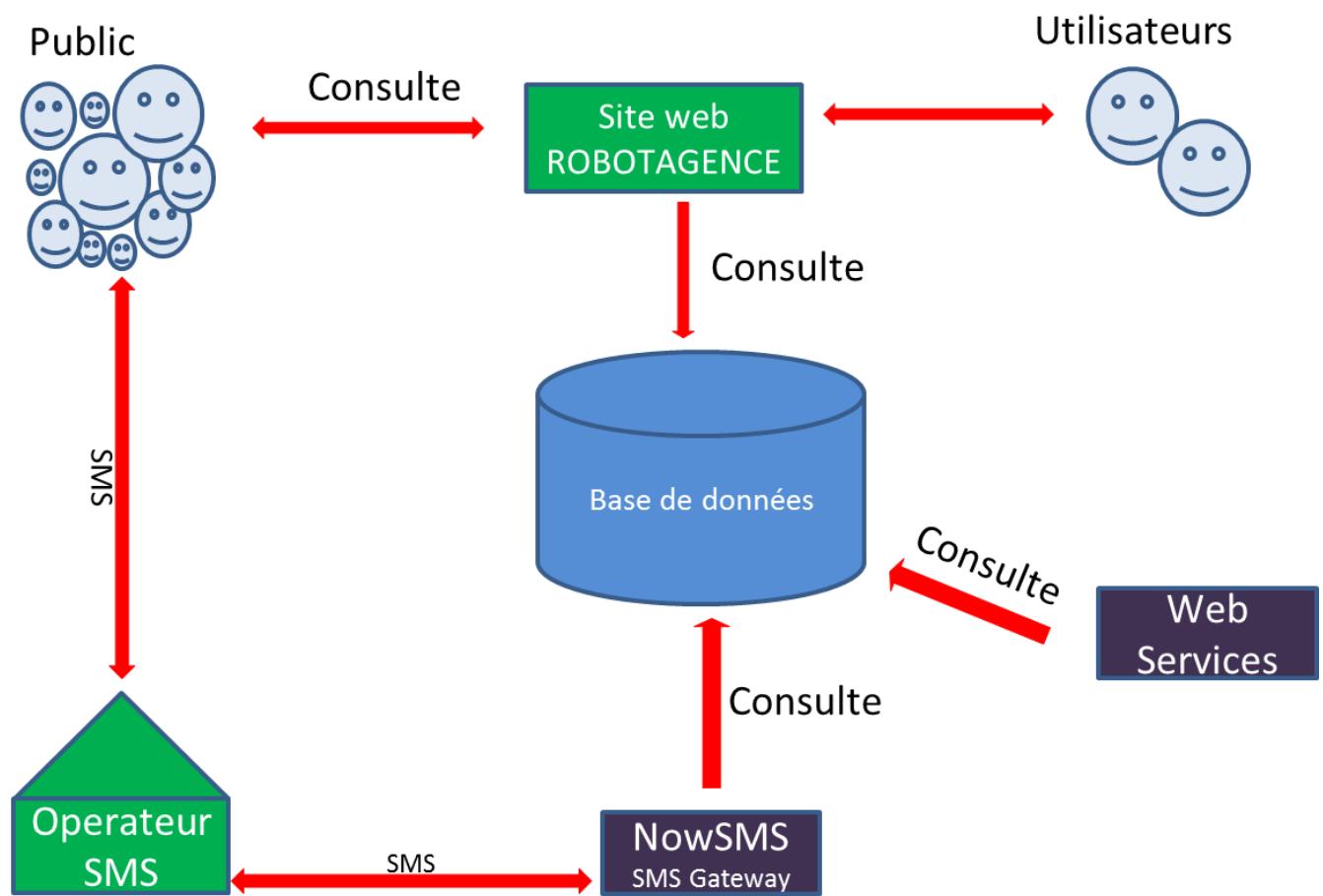


Figure 41: Architecture de l'application

Explication :

- Le site web récupère les données de la base de données, celui-ci est alors consulté par les utilisateurs et les clients.
- Les clients peuvent alors faire une demande par le biais de l'opérateur de messagerie. Le message reçu sera alors traité par notre service de Gateway qui communique avec la base de données. Le service de Gateway se charge de la vérification syntaxique et de l'appartenance du client au système.
- Les web services consultent la base et répondent à la demande effectives du client.

3. Déploiement

Nous avons déployé la partie web de notre système sur un serveur web Apache. La partie suivante décrit les étapes suivies pour le déploiement.

- Nous avons cloné le projet qui était hébergé sur une plateforme git (Bitbucket) ;
- Nous avons importé sur le serveur les tables de notre système dans une base de données créées avec Phpmyadmin ;
- Nous avons adapté le fichier d'extension env du système ;
- Nous avons édité le fichier de configuration du site pour personnaliser le fonctionnement de l'application ;
- Nous avons activé SSL (Secure Socket Layer) avec la commande « a2enmode ssl ». Il permet d'échanger des informations entre 2 ordinateurs de façon sûre ;
- Nous avons activé le site avec La commande « a2ensite » ;
- Nous avons rechargé le serveur apache pour prendre en compte les modifications avec la commande « service apache2 reload ».

La phase de déploiement est terminée. Nous avons maintenant accès au site web.

4. Script de créations de la base de données

```
/*=====
/* Nom de SGBD : Microsoft SQL Server 2008          */
/* Date de création : 02/09/2016 16:59:08           */
/*=====*/
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Clients') and o.name = 'FK_CLIENTS_ASSOCIATI_GROUPE')
alter table Clients
    drop constraint FK_CLIENTS_ASSOCIATI_GROUPE
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Clients') and o.name = 'FK_CLIENTS_INSCRIRE_UTILISAT')
alter table Clients
    drop constraint FK_CLIENTS_INSCRIRE_UTILISAT
go
```

```
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Consommation') and o.name = 'FK_CONSUMMA_CONSOMMER_ENTREPRI')
alter table Consommation
    drop constraint FK_CONSUMMA_CONSOMMER_ENTREPRI
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Consommation') and o.name = 'FK_CONSUMMA_EFFECTUER_CLIENTS')
alter table Consommation
    drop constraint FK_CONSUMMA_EFFECTUER_CLIENTS
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Credit') and o.name = 'FK_CREDIT_ASSOCIATI_PREPAYE')
alter table Credit
    drop constraint FK_CREDIT_ASSOCIATI_PREPAYE
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Facture') and o.name = 'FK_FACTURE_DESTINE_POSTPAYE')
alter table Facture
    drop constraint FK_FACTURE_DESTINE_POSTPAYE
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Facture') and o.name = 'FK_FACTURE_ENVOYER_UTILISAT')
alter table Facture
    drop constraint FK_FACTURE_ENVOYER_UTILISAT
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Facture') and o.name = 'FK_FACTURE_RECEVOIR_ENTREPRI')
alter table Facture
    drop constraint FK_FACTURE_RECEVOIR_ENTREPRI
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Messages') and o.name = 'FK_MESSAGES_ENVOYER_CLIENTS')
alter table Messages
    drop constraint FK_MESSAGES_ENVOYER_CLIENTS
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Numero') and o.name = 'FK_NUMERO_AVOIR_CLIENTS')
alter table Numero
    drop constraint FK_NUMERO_AVOIR_CLIENTS
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Postpaye') and o.name = 'FK_POSTPAYE_GENERALIS_CLIENTS')
alter table Postpaye
    drop constraint FK_POSTPAYE_GENERALIS_CLIENTS
go
```

```
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Prepaye') and o.name = 'FK_PREPAYE_GENERALIS_CLIENTS')
alter table Prepaye
    drop constraint FK_PREPAYE_GENERALIS_CLIENTS
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('Utilisateur') and o.name = 'FK_UTILISAT_ASSOCIATI_UTILISAT')
alter table Utilisateur
    drop constraint FK_UTILISAT_ASSOCIATI_UTILISAT
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('appartenir') and o.name = 'FK_APPARTEN_APPARTENI_CLIENTS')
alter table appartenir
    drop constraint FK_APPARTEN_APPARTENI_CLIENTS
go
if exists (select 1
    from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')
    where r.fkeyid = object_id('appartenir') and o.name = 'FK_APPARTEN_APPARTENI_ENTREPRI')
alter table appartenir
    drop constraint FK_APPARTEN_APPARTENI_ENTREPRI
go
if exists (select 1
    from sysindexes
    where id = object_id('Clients')
    and name = 'INSCRIRE_FK'
    and indid > 0
    and indid < 255)
drop index Clients.INSCRIRE_FK
go
if exists (select 1
    from sysobjects
    where id = object_id('Clients')
    and type = 'U')
drop table Clients
go
if exists (select 1
    from sysindexes
    where id = object_id('Consommation')
    and name = 'EFFECTUER_FK'
    and indid > 0
    and indid < 255)
drop index Consommation.EFFECTUER_FK
go
if exists (select 1
    from sysobjects
    where id = object_id('Consommation')
    and type = 'U')
drop table Consommation
go
if exists (select 1
    from sysindexes
```

```
where id = object_id('Credit')
and name = 'ASSOCIATION15_FK'
and indid > 0
and indid < 255)
drop index Credit.ASSOCIATION15_FK
go
if exists (select 1
            from sysobjects
            where id = object_id('Credit')
            and type = 'U')
    drop table Credit
go
if exists (select 1
            from sysobjects
            where id = object_id('Demande')
            and type = 'U')
    drop table Demande
go
if exists (select 1
            from sysobjects
            where id = object_id('Entreprise')
            and type = 'U')
    drop table Entreprise
go
if exists (select 1
            from sysindexes
            where id = object_id('Facture')
            and name = 'ENVOYER_FK'
            and indid > 0
            and indid < 255)
    drop index Facture.ENVOYER_FK
go
if exists (select 1
            from sysobjects
            where id = object_id('Facture')
            and type = 'U')
    drop table Facture
go
if exists (select 1
            from sysobjects
            where id = object_id('Groupe')
            and type = 'U')
    drop table Groupe
go
if exists (select 1
            from sysindexes
            where id = object_id('Messages')
            and name = 'ENVOYER_FK'
            and indid > 0
            and indid < 255)
    drop index Messages.ENVOYER_FK
go
if exists (select 1
            from sysobjects
```

```
where id = object_id('Messages')
      and type = 'U')
drop table Messages
go
if exists (select I
            from sysindexes
            where id   = object_id('Numero')
              and name = 'AVOIR_FK'
              and indid > 0
              and indid < 255)
    drop index Numero.AVOIR_FK
go
if exists (select I
            from sysobjects
            where id = object_id('Numero')
              and type = 'U')
    drop table Numero
go
if exists (select I
            from sysindexes
            where id   = object_id('Postpaye')
              and name = 'GENERALISATION_2_FK'
              and indid > 0
              and indid < 255)
    drop index Postpaye.GENERALISATION_2_FK
go
if exists (select I
            from sysobjects
            where id = object_id('Postpaye')
              and type = 'U')
    drop table Postpaye
go
if exists (select I
            from sysindexes
            where id   = object_id('Prepaye')
              and name = 'GENERALISATION_I_FK'
              and indid > 0
              and indid < 255)
    drop index Prepaye.GENERALISATION_I_FK
go
if exists (select I
            from sysobjects
            where id = object_id('Prepaye')
              and type = 'U')
    drop table Prepaye
go
if exists (select I
            from sysobjects
            where id = object_id('Suggestions')
              and type = 'U')
    drop table Suggestions
go
```

```
if exists (select 1
            from sysindexes
            where id = object_id('Utilisateur')
            and name = 'ASSOCIATION12_FK'
            and indid > 0
            and indid < 255)
    drop index Utilisateur.ASSOCIATION12_FK
go
if exists (select 1
            from sysobjects
            where id = object_id('Utilisateur')
            and type = 'U')
    drop table Utilisateur
go
if exists (select 1
            from sysindexes
            where id = object_id('appartenir')
            and name = 'APPARTENIR_FK'
            and indid > 0
            and indid < 255)
    drop index appartenir.APPARTENIR_FK
go
if exists (select 1
            from sysobjects
            where id = object_id('appartenir')
            and type = 'U')
    drop table appartenir
go
/*=====
/* Table : Clients
*/
=====*/
create table Clients (
    Id_client      int          null,
    Nom_client     varchar(254)   null,
    Prenom_client   varchar(254)   null,
    Sexe_client    char(1)       null,
    Date_naiss_client datetime    null,
    Email_client   varchar(254)   null
)
go
/*=====
/* Index : INSCRIRE_FK
*/
=====*/
create index INSCRIRE_FK on Clients (
)
go
/*=====
/* Table : Consommation
*/
=====*/
create table Consommation (
    Id_cons        int          null,
    Mtnt_cons      int          null,
    Date_cons      datetime    null
)
```

PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE

```
go
/*=====
/* Index : EFFECTUER_FK */ */
/*=====*/
create index EFFECTUER_FK on Consommation (
)
go
/*=====
/* Table : Credit */ */
/*=====*/
create table Credit (
    id_credit      int          null,
    soldecredit    int          null
)
go
/*=====
/* Index : ASSOCIATION15_FK */ */
/*=====*/
create index ASSOCIATION15_FK on Credit (
)
go
/*=====
/* Table : Demande */ */
/*=====*/
create table Demande (
    identifiantDemande int          null,
    nom              varchar(254)  null,
    email            varchar(254)  null,
    contact          varchar(254)  null,
    type             varchar(254)  null
)
go
/*=====
/* Table : Entreprise */ */
/*=====*/
create table Entreprise (
    Id_entr        int          null,
    Nom_entre      varchar(254)  null,
    adresseEntreprise varchar(254)  null,
    contact         varchar(254)  null
)
go
/*=====
/* Table : Facture */ */
/*=====*/
create table Facture (
    Num_fact       int          null,
    date_Fact      datetime     null,
    Mtnt_fact      int          null
)
go
/*=====
/* Index : ENVOYER_FK */ */
/*=====*/
```

PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE

```
create index ENVOYER_FK on Facture (
)
go
/*=====
/* Table : Groupe
/*=====*/
create table Groupe (
    id_grp      int      null,
    nomGrp      varchar(254)  null,
    solde       int      null
)
go
/*=====
/* Table : Messages
/*=====*/
create table Messages (
    Id_msg      int      null,
    Ctn_msg      varchar(254)  null,
    Date_envoi   int      null
)
go
/*=====
/* Index : ENVOYER_FK
/*=====*/
create index ENVOYER_FK on Messages (
)
go
/*=====
/* Table : Numero
/*=====*/
create table Numero (
    Id_num      int      null,
    numero      int      null,
    Operateur    varchar(254)  null,
    accesPrive   bit      null
)
go
/*=====
/* Index : AVOIR_FK
/*=====*/
create index AVOIR_FK on Numero (
)
go
/*=====
/* Table : Postpaye
/*=====*/
create table Postpaye (
    id_post      int      null
)
go
/*=====
/* Index : GENERALISATION_2_FK
/*=====*/
create index GENERALISATION_2_FK on Postpaye (
```

PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE

```
)  
go  
/*=====*/  
/* Table : Prepaye */  
/*=====*/  
create table Prepaye (  
    id_pre      int      null,  
    solde       int      null,  
    fidel       bit      null  
)  
go  
/*=====*/  
/* Index : GENERALISATION_I_FK */  
/*=====*/  
create index GENERALISATION_I_FK on Prepaye (  
)  
go  
/*=====*/  
/* Table : Suggestions */  
/*=====*/  
create table Suggestions (  
    id_sugg     int      null,  
    nom         varchar(254)  null,  
    email       varchar(254)  null,  
    message     varchar(254)  null  
)  
go  
/*=====*/  
/* Table : Utilisateur */  
/*=====*/  
create table Utilisateur (  
    Id_utilisateur   int      null,  
    Nom_utilisateur  varchar(254)  null,  
    Prenom_utilisateur  varchar(254)  null,  
    Sexe_utilisateur char(1)      null,  
    Date_naiss_utilisateur  datetime  null,  
    Email_utilisateur  varchar(254)  null,  
    pseudo_utilisateur  varchar(254)  null,  
    Mp_utilisateur    varchar(254)  null,  
    Etat_utilisateur  bit      null,  
    niveau          bit      null  
)  
go  
/*=====*/  
/* Index : ASSOCIATION12_FK */  
/*=====*/  
create index ASSOCIATION12_FK on Utilisateur (  
)  
go  
/*=====*/  
/* Table : appartenir */  
/*=====*/  
create table appartenir (
```

```
)  
go  
  
/*=====*/  
/* Index : APPARTENIR_FK */  
/*=====*/  
create index APPARTENIR_FK on appartenir (  
)  
go  
  
alter table Clients  
add constraint FK_CLIENTS_ASSOCIATI_GROUPE foreign key ()  
    references Groupe  
go  
  
alter table Clients  
add constraint FK_CLIENTS_INSCRIRE_UTILISAT foreign key ()  
    references Utilisateur  
go  
  
alter table Consommation  
add constraint FK_CONSUMMA_CONSOMMER_ENTREPRI foreign key ()  
    references Entreprise  
go  
  
alter table Consommation  
add constraint FK_CONSUMMA_EFFECTUER_CLIENTS foreign key ()  
    references Clients  
go  
  
alter table Credit  
add constraint FK_CREDIT_ASSOCIATI_PREPAYE foreign key ()  
    references Prepaye  
go  
alter table Facture  
add constraint FK_FACTURE_DESTINE_POSTPAYE foreign key ()  
    references Postpaye  
go  
alter table Facture  
add constraint FK_FACTURE_ENVOYER_UTILISAT foreign key ()  
    references Utilisateur  
go  
alter table Facture  
add constraint FK_FACTURE_RECEVOIR_ENTREPRI foreign key ()  
    references Entreprise  
go  
alter table Messages  
add constraint FK_MESSAGES_ENVOYER_CLIENTS foreign key ()  
    references Clients  
go  
alter table Numero  
add constraint FK_NUMERO_AVOIR_CLIENTS foreign key ()
```

```
references Clients
go
alter table Postpaye
    add constraint FK_POSTPAYE_GENERALIS_CLIENTS foreign key ()
        references Clients
go
alter table Prepaye
    add constraint FK_PREPAYE_GENERALIS_CLIENTS foreign key ()
        references Clients
go
alter table Utilisateur
    add constraint FK_UTILISAT_ASSOCIATI_UTILISAT foreign key ()
        references Utilisateur
go
alter table appartenir
    add constraint FK_APPARTEN_APPARTENI_CLIENTS foreign key ()
        references Clients
go
alter table appartenir
    add constraint FK_APPARTEN_APPARTENI_ENTREPRI foreign key ()
        references Entreprise
go
```

5. Quelques masques de saisie et code source de l'application

Nous allons présenter à partir de ces quelques codes la notion MVC (Model Vue Controller) de LARAVEL.

Le code suivant concerne les clients :

- Le Model

Les Models en Laravel contiennent certaines fonctions et définissent, la Structure de la base mais aussi contiennent quelques fonctions liées au Controller.

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Clients extends Model
{
    protected $table = 'clients';
    public $timestamps = true;

    /**
     * The attributes that are mass assignable.
     *
```

```

        * @var array
    */
/*Definitions de attributs*/
protected $fillable = [
    'id', 'nom', 'prenom', 'sexe', 'datenaiss', 'mode', 'email',
];

/**
 * The attributes that should be hidden for arrays.
 *
 * @var array
 */
/*permet de garder le Mots de passe*/
protected $hidden = [
    'password', 'remember_token',
];
/*definition de cles etrangères*/
protected function numeros(){
    return $this->hasMany('App\Numeros');
}

protected function consommations(){
    return $this->hasMany('App\Consommations');
}

protected function messages(){
    return $this->hasMany('App\Messages');
}
/*definitions de vers ou la cle de client migre vers*/
protected function groupes(){
    return $this->belongsToMany('App\Groupes');
}

protected function utilisateurs(){
    return $this->belongsToMany('App\Utilisateurs');
}
protected function clients_entreprises(){
    return $this->belongsToMany('App\Clients_Entreprises');
}
}

```

- La Vue

- Formulaire

Le code suivant est le code de la figure 28.

```

@extends('fond')
@section('title', ' LISTE CLIENTS')
@section('bar')
    @parent
@stop
@section('content')
    <section class="content">
        <div class="container-fluid">
            <div class="block-header">
                <h2>
                    LISTE DES CLIENTS
                </h2>
            </div>
            <div class="row clearfix">

```

```

<div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
    <div class="card">
        <div class="header">
            <h2>TOUS LES CLIENTS
            </h2>
        </div>
        <div class="body table-responsive">
            <table class="table table-condensed">
                <thead>
                    <tr class="bg-cyan">
                        <th>Nom</th>
                        <th>Prenom</th>
                        <th>Email</th>
                        <th>Mode</th>
                        <th>Date de naissance</th>
                        <th>Editer</th>
                    </tr>
                </thead>
                @foreach ($clients as $clients)
                <tbody>
                    <tr class="bg-teal">
                        <td>{{$clients->nom}} </td>
                        <td>{{$clients->prenom}}</td>
                        <td>{{$clients->email}}</td>
                        <td>@if($clients->mode == 0) {{ 'Prepayé' }} @endif @if($clients->mode == 1) {{ 'Postpayé' }} @endif</td>
                        <td>{{$clients->datenaiss}}</td>
                        <td> <div class="icon-button-demo">
                            <a href="{{url($clients->id.'/delete')}}"> <button type="button" onclick="return confirm('Voulez vous vraiment supprimer ce client?') " class="btn bg-red btn-circle waves-effect waves-circle waves-float" ><i class="material-icons">delete</i></button></a>
                            <a href="{{url($clients->id.'/modif')}}"> <button type="button" class="btn bg-green btn-circle waves-effect waves-circle waves-float" ><i class="material-icons">update</i></button></a>
                        </div></td>
                    </tr>
                @endforeach
                </tbody>
            </table>
        </div>
    </div>
</div>
</div>
</section>
@stop

```

- La Modification

La figure 32 traduite ces lignes de code même si sur la figure la modification concerne les utilisateurs.

```

@extends('fond')
@section('title', 'CLIENTS')
@section('bar')
@parent
@stop
@section('content')
    <section class="content">

```

```

<div class="container-fluid">
    <div class="row clearfix">
        <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
            <div class="card">
                <div class="body">
                    <form role="form" method="POST"
action="{!!url($Clients->id. '/update')}" accept-charset="UTF-8">
                        {{ csrf_field() }}
                        <h2 class="card-inside-title">Renseigner
les informations</h2>
                        <div class="form-group form-float">
                            <div class="col-sm-12">
                                <div class="form-line">
                                    <input type="text"
class="form-control" name ="nom" value="{!!old('nom',$Clients->nom) }" required>
                                    </div>
                                </div>
                            <div class="col-sm-12">
                                <div class="form-line">
                                    <input type="text"
class="form-control" name ="prenom" value="{!!old('prenom',$Clients->prenom) }" required>
                                    </div>
                                </div>
                            <div class="col-sm-12">
                                <div class="form-line">
                                    <input type="text"
class="form-control" name ="email" value="{!!old('email',$Clients->email) }" required>
                                    </div>
                                </div>
                            <div class="col-sm-12">
                                <div class="form-line">
                                    <input type="date"
class="form-control" name="date" value="{!!old('datenaiss',$Clients->datenaiss) }" required>
                                    <label class="form-label">
                                        </div>
                                    </div>
                                </div>
                            <div class="col-sm-12">
                                <select class="form-control show-
tick" name ="mode" required>
                                    un Mode --<option>
                                    >mode == 0) {{ 'selected' }} @endif>Prepayé</option>
                                    >mode == 1) {{ 'selected' }} @endif>Postpayé</option>
                                </select>
                            </div>
                            <div class="form-group">
                                SEXE :
                                <input type="radio" name="sexe"
id="male" class="with-gap radio-col-blue" value="0" @if($Clients->sexe ==
0) {{ 'checked' }} @endif>
                                    <label for="male">Masculin</label>
                                <input type="radio" name="sexe" value="1" @if($Clients->sexe ==
1) {{ 'checked' }} @endif>
                                    <label for="female">Féminin</label>
                            </div>
                        </div>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

        id="female" class="with-gap radio-col-blue" value="1" @if($Clients->sexe
== 1) {{ 'checked' }} @endif>
        <label for="female" class="m-1-
20">feminin</label>
        </div>
        <button class="btn btn-primary waves-
effect" type="submit">Modifier</button>
        </form >
        </div>
        </div>
    </div>
</section>
@stop

```

- Le Controller

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Foundation\Auth\ThrottlesLogins;
use Illuminate\Foundation\Auth\AuthenticatesAndRegistersUsers;
use App\clients;
use App\Repositories\UserRepository;
use App\Numeros;
use App\Http\Requests;

use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
use Illuminate\Foundation\Auth\Access\AuthorizesResources;
use Illuminate\Support\Facades\Auth;

class ClientsController extends Controller
{
    public function create()
    {
        return view('form_client');

    }

    public function store(Request $request) {

        $clients = new Clients;

        $clients->nom = $request->input('nom');
        $clients->prenom = $request->input('prenom');
        $clients->email = $request->input('email');
        $clients->datenaiss = $request->input('date');
        $clients->sexe = $request->input('sexe');
        $clients->mode = $request->input('mode');

        $clients->save();
        return redirect('/form_num');
    }
}

```

```

    public function liste(Request $request) {

        $clients=Clients::all();
        return view('liste_clients')->with(['clients'=>$clients]);
    }

    public function delete($id){

        $clients=Clients::find($id);
        foreach ($clients->numeros as $numero) {
            $temp = Numeros::find($numero->id);
            $temp->delete();

        }
        foreach ($clients->messages as $message) {
            $temp1 = Messages::find($message->id);
            $temp1->delete();

        }
        foreach ($clients->consommations as $consommation) {
            $temp1 = Consommations::find($consommation->id);
            $temp1->delete();

        }
        $clients->delete();

        return redirect('/liste_clients');
    }

    public function edit($id){

        $Clients=Clients::find($id);
        return view('clients_maj')->with(['Clients'=>$Clients]);
    }

    public function update($id,Request $request){

        $Clients=Clients::find($id);

        $Clients->nom = $request->input('nom');
        $Clients->prenom = $request->input('prenom');
        $Clients->email = $request->input('email');
        $Clients->datenaiss = $request->input('date');
        $Clients->sexe = $request->input('sexe');
        $Clients->mode = $request->input('mode');
        $Clients->save();

        return redirect('/liste_clients');
    }
}

```

- Enregistrement des messages reçu depuis le Gateway

```

<?php

include('connexion.php');

```

```
$message = "";
$sender="";
$date="";

if ($_REQUEST['message']) {
    $message = $_REQUEST['message'];
}

if ($_REQUEST['sender']){
    $sender = $_REQUEST['sender'];
}

if ($_REQUEST['date']){
    $date = $_REQUEST['date'];
}

function SendSMS ($host, $port, $username, $password, $phoneNoRecip, $msgText) {

    $fp = fsockopen($host, $port, $errno, $errstr);

    if (!$fp) {
        echo "errno: $errno \n";
        echo "errstr: $errstr\n";
        return $result;
    }

    fwrite($fp, "GET /?Phone=" . rawurlencode($phoneNoRecip) . "&Text=" .
           rawurlencode($msgText) . " HTTP/1.0\n");

    if ($username != "") {
        $auth = $username . ":" . $password;
        $auth = base64_encode($auth);
        fwrite($fp, "Authorization: Basic " . $auth . "\n");
    }

    fwrite($fp, "\n");

    $res = "";

    while (!feof($fp)) {
        $res .= fread($fp, 1);
    }

    fclose($fp);

    return $res;
}
```

```
}

$req2 = $bdd->execute('select numeros from numeros where numeros ='.$sender.);

if ($sender != "")
{

    $clients_id=2;
    $req = $bdd->prepare('INSERT INTO
messages(contenu,numero,created_at,clients_id)
VALUES(:message,:sender,now(),:clients_id)');
    $req->execute(array(
        'message'=>$message,
        'sender'=>$sender,
        'clients_id'=>$clients_id
    ));
    if ($sender ==$req2)
    {

        SendSMS("127.0.0.1", 8800, "user", "", $sender, "Votre demande est en
cours de traitement");

    }
    else
    {

        SendSMS("127.0.0.1", 8800, "user", "", $sender, "Veuillez d'abord vous
inscrire ROBOT 'Nom' 'Email' 'Type' ");
    }
}
?>
```

- Capture de quelques opérations réussies

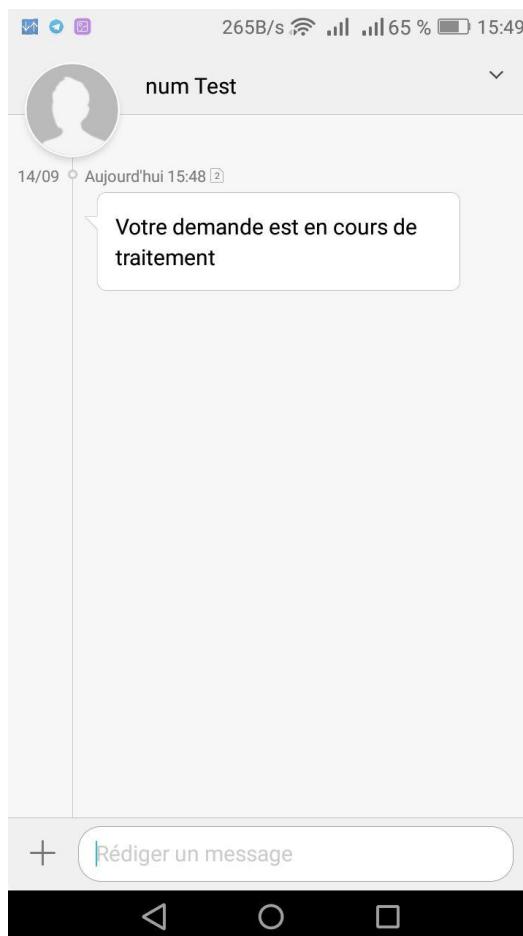


Figure 43: Réponse 1

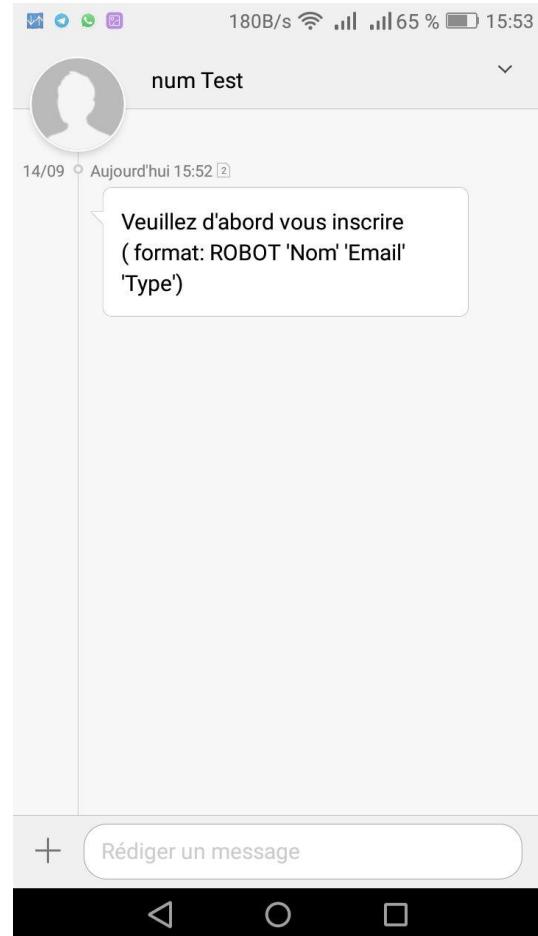


Figure 42: Réponse 2

6. Quelques Etats et Statistiques

- Facture pour un client en Mode Post payé

09/09/2016

N° de facture 5

Facturer à :
TENGUEY
Christine
Lomé,TOGO
+228 90 90 36 20

Pour :
La consommation

Les chèques sont à mettre à l'ordre de :
DORKENOO CONSULTING

Pour toute question
concernant cette facture,
veuillez vous adresser à :
EMS DORKENDO
+228 22 46 36 48
dax@dorken.noconsulting.com

Sous-total	Nombre total d'éléments : 3	Nombre total d'éléments : 3	
Taux des taxes : 5,0%	<input checked="" type="checkbox"/> Exonération fiscale	Taxes versées	8 500
		Montant total	17 000

Indication

TMN = Transfert vers Mon Numéro

Total de la facture

Nous vo

Montant à payer dès réception

Nous vous remercions de votre confiance.

DORKENOO CONSULTING

Tel +228 33 66 36 48

www.dierkenconsulting.com

Page 22 Page 224

dc@darkanconsulting.com

P
C

Figure 44: Facture Client

PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE

- #### - Facture d'une entreprise

09/09/2016

N° de facture 18

Facturer à :
VEGABOND
+228 22 37 45 69
Lomé,TOGO

Pour :
La consommation

Les chèques sont à mettre à l'ordre de :

Pour toute question
concernant cette facture,
veuillez vous adresser à :
EMS DORKENDO
+228 22 46 56 48
dax@dorken.noconsulting.com

DORKENOO CONSULTING

Tel. +33 22 36 48

http://www.dynamilis.com/whitepaper/online-advertising-best-practices

BB 22 BB 224

dgt@darknetconsulting.com

D
C

Figure 45: Facture d'une entreprise

- Graphe de consommation Mensuelle

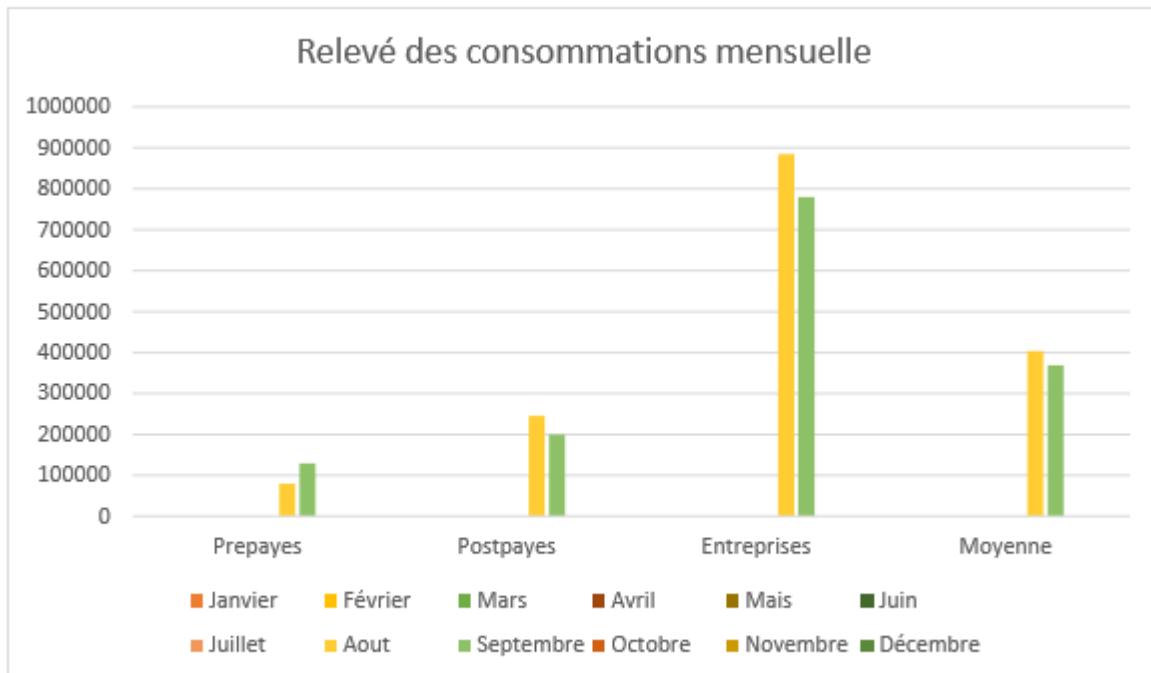


Figure 46: Consommation Mensuelle

- Graphe de consommation Trimestrielle

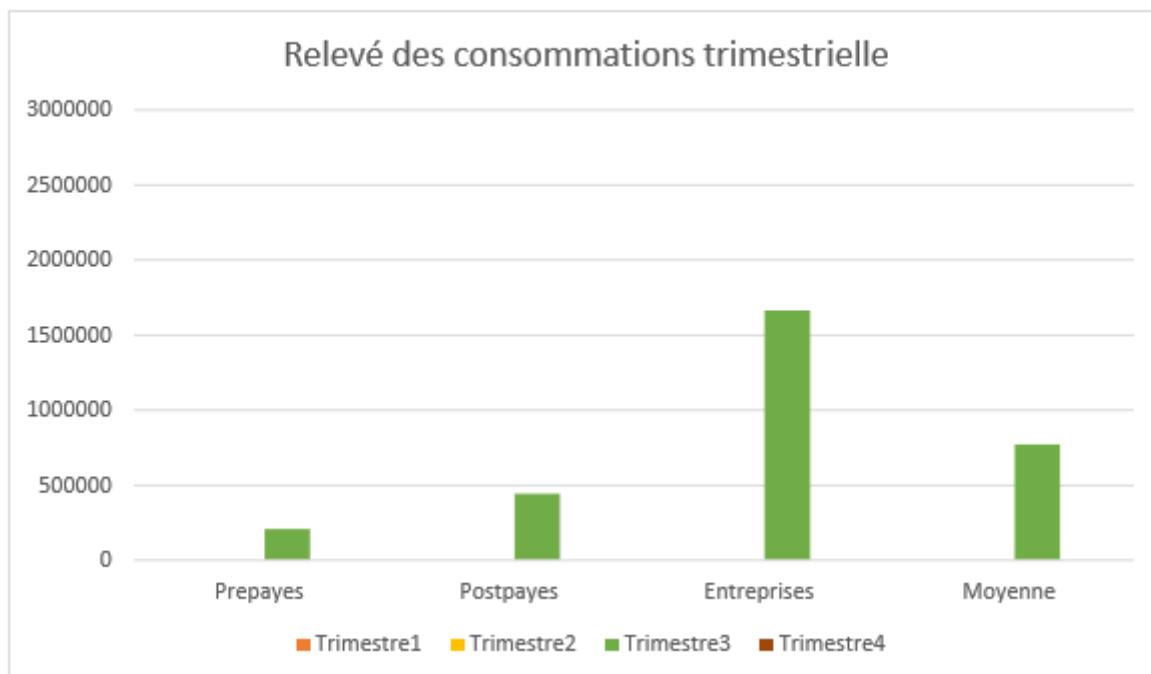


Figure 47: Consommation Trimestrielle

CONCLUSION

La phase de programmation mets fin à cette partie de notre document. Il est vrai que notre système a été développé pour les opérateurs mobiles se situant au Togo mais qu'il peut être adapté aux réalités de d'autres pays aussi.

CONCLUSION GENERALE

Ce stage de programmation a été une formidable occasion qui nous a permis de mettre en pratique les compétences acquises au cours de ces deux dernières années. Il nous a aussi permis de nous imprégner du travail au sein d'un cabinet de programmation.

Le thème de notre stage est en effet la Conception et Réalisation d'une plateforme autonome de transfert de crédit et d'activation de forfait internet. Ce stage s'est déroulé sous l'encadrement et le contrôle de M.KEOULA Edem Lonlon et de M DORKENOO Edem Elvis, nous a permis de mettre en place un système de transfert automatique de crédit et d'activation de forfait internet.

De part, nous avons reçu de nouvelles connaissances grâce à l'intervention de tous ceux qui nous ont assistés tout au long de notre ce stage.

Ouvrages

- UML 2 la pratique

Sites web

- Documentation générale, <http://fr.wikipedia.org/wiki/>
Du 22/06/2016 AU 15/09/2016
- Documentation sur LARAVEL, <https://laravel.com/docs/5.3> et
<https://openclassrooms.com/courses/dcouvrez-le-framework-php-laravel>
DU 25/07/2016 AU 15/09/2016
- Documentation ASP.NET <https://openclassrooms.com/courses/apprendre-asp-net-mvc>
DU 27/08/2016 AU 15/09/2016

Notes de cours

- Analyse (UML), M. GBINU (Année académique 2015-2016) ;
- Programmation Web, M. DIZEWE (Année académique 2015-2016).

Table Des Illustrations

Figure 1: Organigramme de DORKENOO CONSULTING.....	11
Figure 2 : Plan de situation DORKENOO CONSULTING à Lomé.....	12
Figure 3: Diagramme de Gantt (planning prévisionnel de réalisation)	22
Figure 4: Logo UML.....	23
Figure 5: Logo PowerAMC.....	26
Figure 6: Tableau récapitulatif des cas d'utilisation du système	29
Figure 7: Diagramme de cas d'utilisation : SIWAD.....	30
Figure 8: Diagramme de cas d'utilisation : GATEWAY.....	31
Figure 9: Diagramme de séquence : S'abonner	36
Figure 10: Diagramme de séquence : S'authentifier	37
Figure 11: Diagramme de séquence : Faire une Opération	38
Figure 12: Diagramme de séquence de désabonnement	38
Figure 13: Diagramme d'activité : S'abonner	39
Figure 14: Diagramme d'activité : Demande	40
Figure 15: Diagramme de classes	42
Figure 16: Logo PHP	46
Figure 17: Logo ASP.NET	47
Figure 18: Logo LARAVEL	47
Figure 19: Logo Git.....	49
Figure 20: Logo BitBucket	50
Figure 21: Interface Bitbucket.....	50
Figure 22: Logo PHP Storm.....	51
Figure 23: Interface PHP Storm	51
Figure 24: Logo Visual Studio	52
Figure 25: Interface Visual Studio	52
Figure 26:Logo MySQL	53
Figure 27: Première section d'accueil.....	55
Figure 28: Deuxième section d'accueil	56
Figure 29: Troisième section d'accueil.....	56
Figure 30: Pied de la page d'accueil	57
Figure 31: Page de connexion	57
Figure 32: Page d'accueil Administrateur	58
Figure 33: Exemple de formulaire d'ajout	58
Figure 34: Liste de choix de mode	59
Figure 35: Formulaire d'ajout de numéro.....	59
Figure 36: Liste de choix de clients.....	59
Figure 37: Formulaire de modification	60
Figure 38: Liste des Consommations	60
Figure 39: Liste des clients	61
Figure 40: Liste des messages	62
Figure 41: Architecture de l'application	63
Figure 42: Réponse 2	82
Figure 43: Réponse 1	82
Figure 44: Facture Client	83
Figure 45: Facture d'une entreprise	84
Figure 46: Consommation Mensuelle.....	85
Figure 47: Consommation Trimestrielle	85

Table de Matières

PAGE DE DEDICACE	i
PAGE DE REMERCIEMENT	ii
SOMMAIRE.....	iii
LISTE DES PARTICIPANTS.....	iv
INTRODUCTION GENERAL	v
PARTIE 1 : RAPPORT D'INSERTION.....	6
INTRODUCTION	7
I. PRESENTATIONS	8
1. Présentation de IAI-TOGO.....	8
2. Présentation du cadre de stage.....	7
II. THEME DU STAGE	12
1. Présentation du sujet	12
2. Problématique du sujet	13
3. Intérêt du sujet	13
CONCLUSION.....	15
PARTIE 2 : RAPPORT DE PRE-PROGRAMMATION	16
INTRODUCTION	17
I. ETUDE DE L'EXISTANT	18
II. CRITIQUE DE L'EXISTANT.....	18
III. PROPOSITIONS DE SOLUTIONS.....	19
IV. CHOIX DE SOLUTION.....	20
V. EVALUATION FINANCIERE DE LA SOLUTION RETENUE	21
VI. PLANNING PREVISIONNEL DE REALISATION	22
VII. PRESENTATION DES METHODES D'ANALYSE.....	23
1. Le langage UML.....	23
2. La méthode AGILE	24
VIII. PRESENTATION DE L'OUTILS DE MODELISATION.....	26
1. Qu'est-ce que PowerAMC ?	26

2. Modélisation avec PowerAMC.....	27
IX. ETUDE DETAILLE DE LA SOLUTION.....	28
1. Le diagramme des cas d'utilisation	28
2. Les diagrammes de séquences.....	36
3. Le diagramme d'activités.....	39
4. Le diagramme de classe.....	41
CONCLUSION.....	43
PARTIE 3 : RAPPORT DE REALISATION ET DE MISE EN ŒUVRE.....	44
INTRODUCTION	45
I. MISE EN ŒUVRE	46
1. Choix matériels	46
2. Choix Logiciels.....	46
3. Sécurité de l'application	53
I- PRESENTATION DE L'APPLICATION.....	54
1. Présentation	54
2. Architecture de l'application	62
3. Déploiement	64
4. Script de créations de la base de données.....	64
5. Quelques masques de saisie et code source de l'application	74
6. Quelques Etats et Statistiques	83
CONCLUSION.....	86
CONCLSION GENERALE.....	87