

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abderrahmane Mira-Bejaïa
Faculté des Sciences Exactes
Département d'Informatique



Mémoire de fin de cycle
En vue d'obtenir le diplôme de master professionnel en génie logiciel

Thème

Conception et réalisation d'une application web dédié à une entreprise du MLM



www.newlifeopportunity.com

Réalisé par :

BADJA Riad

BENALLAOUA Ryma

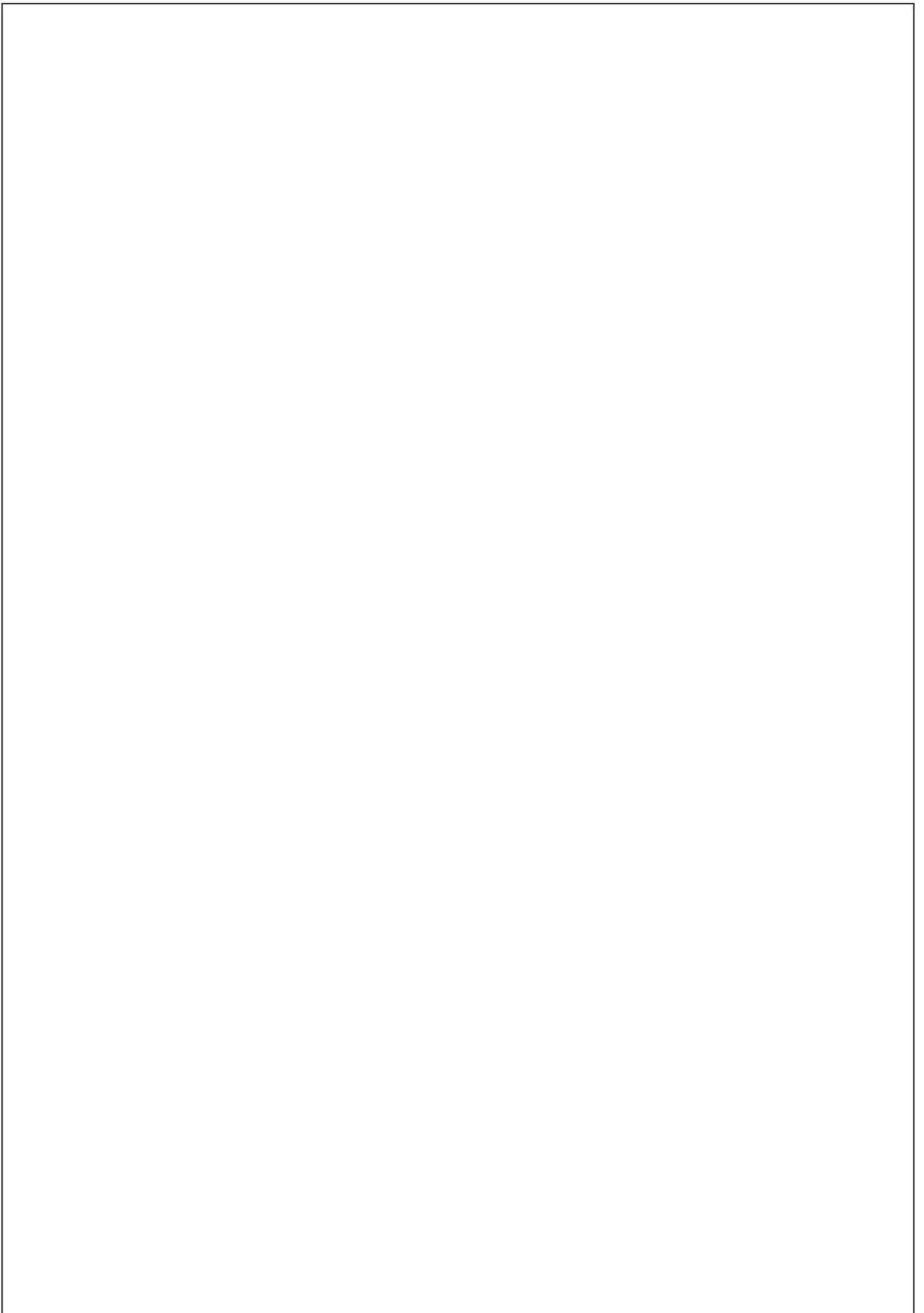
Devant le jury composé de :

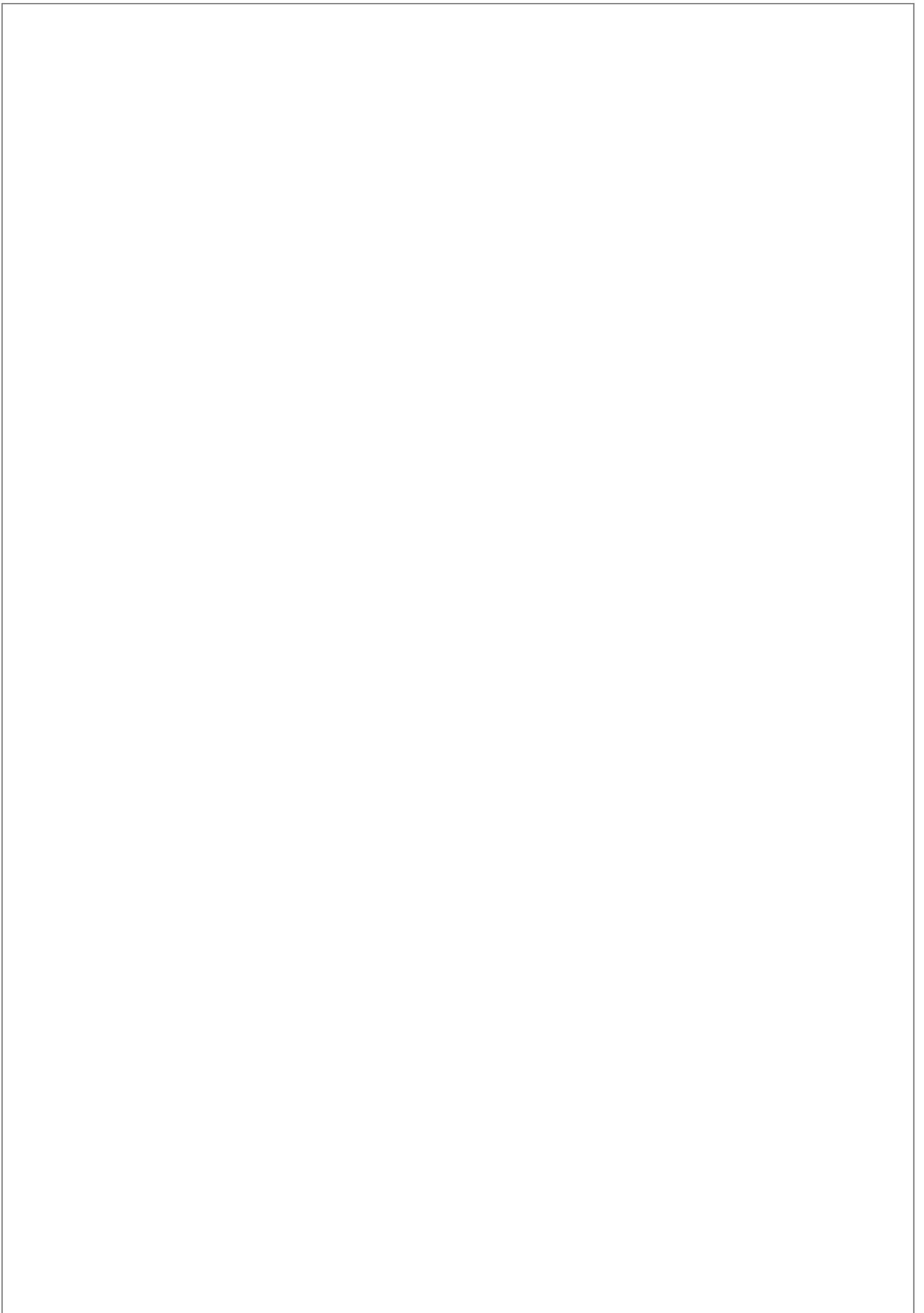
Président : BELKHIRI Louiza

Examineur : KADJOUH Nabil

Encadreur : ACHROUFENE Achour

Promotion : 2015-2016





Remerciements

Nous remercions en premier lieu Dieu tout puissant qui nous a dotés d'une grande volonté et d'un savoir adéquat pour mener à terme notre projet.

La rédaction de ce mémoire fut un exercice quotidien stimulant qui n'aurait pu se concrétiser sans l'apport déterminant et apprécié de plusieurs personnes.

Notre profonde gratitude et sincères remerciements à notre encadreur, en l'occurrence M^r ACHROUFENE Achour qui nous a inculqués une grande confiance et nous a orienté dans le bon sens quant à l'élaboration de ce projet.

Nous tenons fermement à mentionner le plaisir que nous avons eu à faire notre stage pratique au sein de l'entreprise « New-Life » d'Alger. Nous en remercions ici le directeur générale M^r BOUNECER Biaz, M^r HABHOUB Chemseddine et M^r BOUNECER Nadir qui nous ont aidés, soutenus durant la période du stage.

Aux membres de la commission qui jugeront notre travail à savoir :

M^{me} BELKHIRI Louiza pour nous avoir fait l'honneur de présider l'honorable jury.

M^r KADJOUH Nabil pour avoir été parmi les membres du jury et d'avoir examiné notre travail.

Nos remerciements sont adressés également à nos chers parents pour tous les sacrifices consentis à notre égard et leur énorme soutien.

A tous nos proches amis (e)

A tous nos enseignants et membres du département informatique de l'université ABDERRAHMANE MIRA.

Dédicaces

Ce modeste travail est dédié :

*A nos chers parents qui nous ont soutenus et encouragés durant
toute notre scolarité.*

A nos frères et sœurs

A nos familles

A nos enseignants

A nos amis(e)

A toutes les personnes qui nous ont apportés de l'aide.

Table de matières.

Table des matières.....	i
Liste des figures.....	v
Liste des tableaux.....	vii
Liste des abréviations	ix
INTRODUCTION GENERALE	1

Chapitre I : Présentation du cadre du projet.

I.1. Concept de distributions	03
I.2. Définition	04
II.2.1. Distributeur.....	07
II.2.2. Affilié	08
I.3. Principe du MLM.....	09
I.4. Avantage et inconvénients	10
I.5. Présentation de l'entreprise.....	10
I.5.1. Plan en ligne.....	10
I.5.2. Plan binaire	11
I.6. Fonctionnements de l'entreprise	11
I.6.1. Inscriptions.....	11
I.6.2. Système de rémunération	13
I.6.3. Les contraintes de New-Life	17

I.6.4. Système de paiements	18
I.6.5. Facilité de paiements	18
I.7. Problématique	19
I.8. Objectif de travail	19

Chapitre II : Spécification et analyse des besoins.

II.1. Spécification des besoins	21
II.1.1. Les besoins fonctionnels.....	21
II.1.2. Les besoins non fonctionnels.....	24
II.2. Analyse des besoins	26
II.2.1. Identification des acteurs.....	26
II.2.2. Identification des messages échangés.....	27
II.2.3. Diagramme de contexte du système à réaliser	30
II.2.4. Identification des cas d'utilisations	30
II.2.5. Diagramme des cas d'utilisations	32

Chapitre III : conception.

III.1. Modélisation dynamiques.....	35
III.1.1. Les diagrammes d'états transitions.....	35
III.1.2. Les diagrammes de séquence détaillés	36
III.2. Modélisation statiques	52
III.2.1. Le diagramme de classe.....	52

III.2.2. Modèle logique des données	54
III.2.3. Dictionnaires de données	55

Chapitre IV : Réalisation et mise en œuvre.

IV.1. Phases de déploiement.....	62
IV.1.1. Architectures client/serveur	62
IV.1.2. Diagramme de déploiement	63
IV.1.3. Mise en ligne.....	63
IV.1.4. Validation	64
IV.1.5. Guide d'utilisation	64
IV.2. Les langages utilisés	65
IV.2.1. HTML (HyperText Markup Language)	65
IV.2.2. JavaScript.....	65
IV.2.3. AJAX.....	65
IV.2.4. PHP(Hypertext Preprocessor)	66
IV.2.5. MySQL.....	66
IV.2.6. CSS	66
IV.2.7. JSON	66
IV. 3. Outils et logiciels utilisés	67
IV.3.1. Aptana Studio	67
IV.3.2. Environnement Apache/MySQL/PHP(WampServer)	67
IV.3.3. Paint.Net.....	68
IV.3.4. Visio.....	68
IV.3.5. cPanel	68

IV.3.6. VPS (Serveur Privé Virtuel)	68
IV.3.7. Adobe Illustrator	69
IV.4. Interface Homme-Machines	69
IV.4.1. Structure du site web	69
IV.4.2. Présentations des interfaces du site	70
IV.5. Sécurité de l'application	78
IV.6. Phases de maintenance	79
IV.6.1. Modification des packs d'inscription	79
IV.6.2. Méthodes de livraison	80
IV.6.3. Autre	80
IV.6.4. Etude statistique	80
CONCLUSION GENERALE	83

Liste des figures.

Figure 1 : Circuit de distribution traditionnel	03
Figure 2 : Circuit de distribution relationnel.....	04
Figure 3 : Exemple du « MLM »	06
Figure 4 : Principe du « MLM ».....	09
Figure 5 : Logo de « New Life »	10
Figure 6 : Plan binaire	11
Figure 7 : Pack pro	12
Figure 8 : Système binaire	13
Figure 9 : Les commissions d'équilibre.....	15
Figure 10 : Les Gift bonus.....	16
Figure 11 : Les acteurs du site	27
Figure 12 : Diagramme de contexte du système à réaliser	30
Figure 13 : Diagramme des cas d'utilisations	33
Figure 14 : Diagramme d'états/transition des états du compte client.....	35
Figure 15 : Diagramme d'états transitions du parcours des demande de livraisons	36
Figure 16 : Diagramme de séquence détaillé du cas d'utilisation «Se connecter»	38
Figure 17 : Diagramme de séquence détaillé du cas d'utilisation « service de paiements»	40
Figure 18 : Diagramme de séquence détaillé du cas d'utilisation «Consulter les informations»	41

Figure 19 : Diagramme de séquence détaillé du cas d'utilisation « Parrainage»	44
Figure 20 : Diagramme de séquence détaillé du cas d'utilisation «Transfert EM ».....	46
Figure 21 : Diagramme de séquence détaillé du cas d'utilisation «Gestion panier »	48
Figure 22 : Diagramme de séquence détaillé de cas d'utilisation «Validation d'achat»	50
Figure 23 : Diagramme de séquence détaillé des cas d'utilisations «validation produit » ...	51
Figure 24 : Diagramme de classes.....	53
Figure 25 : Diagramme de déploiement du site.....	63
Figure 26 : Structure du site web	70
Figure 27 : IHM de la page d'accueil.....	71
Figure 28 : La page « Formulaire d'inscription 1 ».....	72
Figure 29 : La page « Formulaire d'inscription 2 ».....	73
Figure 30 : IHM du client.....	74
Figure 31 : Le page « Arbre de l'utilisateur ».....	74
Figure 32 : Le page « parrainage de l'utilisateur ».....	75
Figure 33 : IHM de l'administrateur « Gestion compte ».....	76
Figure 34 : IHM de l'administrateur « Gestions des catégories de produits »	76
Figure 35 : IHM de l'espace du service commercial.....	77
Figure 36 : Le page « Liste des nouveaux inscrits».	78
Figure 37 : Diagramme représentant l'évolution du nombre de visites en fonction des heures.....	81
Figure 38 : Diagramme représentant le nombre de visites par rapport aux URL (arbre / connexion).	82

Liste des tableaux

Tableau 1 : les packs d'inscription.....	13
Tableau 2 : Commissions directes	14
Tableau 3 : Commissions d'équilibre.....	14
Tableau 4 : Bonus mensuel.....	16
Tableau 5 : Bonus de démarrage.....	16
Tableau 6 : Premier retrait	17
Tableau 7 : Max out.....	17
Tableau 8 : Facilité de paiements.....	18
Tableau 9 : Les états de transition du «client »	26
Tableau 10 : Les messages entrants échangés avec le système	28
Tableau 11 : Les messages sortants échangés avec le système.....	29
Tableau 12 : Identification des cas d'utilisations.....	31
Tableau 13 : Description du cas d'utilisations « Se connecter »	37
Tableau 14 : Description du cas d'utilisations «service de paiement ».....	38
Tableau 15 : Description du cas d'utilisations «Consulter les informations ».....	41
Tableau 16 : Description du cas d'utilisations «Parrainage »	42
Tableau 17 : Description du cas d'utilisations «Transfert EM ».....	45
Tableau 18 : Description du cas d'utilisations «Gestion panier »	47
Tableau 19 : Description du cas d'utilisations «Validation d'achat »	49
Tableau 20 : Description du cas d'utilisations «Validation produit ».....	51
Tableau 21 : Le glossaire de la table « CompteCommercial».....	55

Tableau 22 : Le glossaire de la table « Utilisateur ».....	55
Tableau 23 : Le glossaire de la table « ClientPackPro ».....	56
Tableau 24 : Le glossaire de la table « Personne »	56
Tableau 25 : Le glossaire de la table « Compte »	56
Tableau 26 : Le glossaire de la table « Panier ».....	57
Tableau 27 : Le glossaire de la table « Produits »	57
Tableau 28 : Le glossaire de la table « CategorieProduits»	58
Tableau 29 : Le glossaire de la table « LigneCommande»	58
Tableau 30 : Le glossaire de la table « Transfert EM»	58
Tableau 31 : Le glossaire de la table « Message».....	59
Tableau 32 : Le glossaire de la table « Retrait»	59
Tableau 33 : Le glossaire de la table « Historique ».....	59
Tableau 34 : Le glossaire de la table « Accueil ».....	60
Tableau 35 : Le glossaire de la table « Bonus »	60
Tableau 36 : Le glossaire de la table « Newsletter »	60
Tableau 37 : Le glossaire de la table « pubPageInscription».....	60
Tableau 38 : Le glossaire de la table « VerificationEmail»	61
Tableau 39 : Le glossaire de la table « Publicité »	61
Tableau 40 : Les nouvelles valeurs du système de « New Life ».....	79
Tableau 41 : Evolution des inscriptions.....	80
Tableau 42 : Evolution du trafic (visite / visiteur)	81

Liste des abréviations

MLM : Multi Level Marketing.

XP : eXtreme Programing.

UML : Unified Modeling Language.

EM : Electronique Monnaie.

CD : Commissions Directes.

CO : Commissions d'équilibre.

FI : Frais d'Inscription.

FL : Frais de Livraison.

EB, E-Bank : Electronique Bank.

GB, G-Bonus : Gift Bonus.

HTML : HyperText Markup Language.

DOM : Document Object Model.

AJAX : Asynchrones JavaScript And XML.

PHP : Hypertext PreProcessor.

SGBD : Système Gestion De Base Donnée.

MySQL : Structured Query Language.

CSS : Cascading Style Sheets.

VPS : Serveur Privé Virtuel.

VDS : virtual Dedicated Server.

IHM : Interface Homme-Machines.

Introduction générale

Depuis des millénaires, la vente et l'achat étaient primordiaux pour assurer la survie et la convivialité entre les êtres humains, ce qui garde jusqu'à aujourd'hui cette importance, tout se vend et tout s'achète. Le marketing a suivi une évolution d'une vitesse surprenante au point où on ne peut s'en priver. Les consommateurs sont donc, confronté à un mode de vie qui les pousse à toujours vouloir acheter plus.

Après l'arrivée d'internet, le marketing a suivi une évolution d'une autre envergure, le commerce est devenu une arène où s'affrontent de nombreuses entreprises avec une toute nouvelle arme, la vente en ligne. A sa découverte, les sites d'achat et de vente se sont multipliés, du fait de leurs facilités d'utilisation et de leur grand choix qu'ils offrent. Mais dans une autre part les entreprises qui n'ont pas les moyens de mettre leurs produits en ligne doivent trouver des solutions nouvelles pour que leurs produits restent dans le marché courant. Pour cela, une autre sorte de marketing a vu le jour.

Les entreprises ont remarqué que les produits qui se vendent le plus, sont ceux qui ont été découvert par la famille et amis des consommateurs. Là, nous parlons du marketing de réseaux qui est une nouvelle manière de vendre, où le produit se transmet du producteur jusqu'au consommateur avec une opportunité d'emploi que nous détaillerons dans la suite de ce présent rapport.

« New Life » est une entreprise qui veut jumeler la vente en ligne et le marketing de réseaux afin de tirer part des deux avantages, mais le seul problème, c'est qu'elle n'a pas l'infrastructure informatique nécessaire pour gérer les deux à la fois, d'où la nécessité d'une solution informatique, où notre objectif est de développer une application web pour remédier à cela.

Ce travail est divisé en quatre chapitres :

Le premier chapitre intitulé « Présentation du cadre du projet » sera dédié en premier lieu à la base théorique, les concepts et le fonctionnement du marketing de réseaux. En second lieu, nous enchaînerons avec la présentation de « New Life », où sera décrit le fonctionnement de l'entreprise, l'admission (inscriptions), la gestion des produits, le système de rémunération et leur contrainte. Nous présenterons aussi les problèmes auxquels « New Life » fait face, et nous proposerons une solution pour y remédier.

Le deuxième chapitre intitulé « Spécification et analyse des besoins » sera consacré à l'identification des besoins fonctionnels et non fonctionnels auxquels doit répondre notre application, ainsi que à l'identification des acteurs et leur cas d'utilisation associés. Dans ce chapitre les messages échangés entre les acteurs et le système seront modélisés par un diagramme de contexte, quant aux cas d'utilisations seront décrit textuellement avant d'être modélisés avec un diagramme de cas d'utilisation.

Le troisième chapitre intitulé « Conception » portera sur le fonctionnement du futur système en répondant aux besoins identifiés dans le chapitre précédent. Il décrira les cas d'utilisations les plus significatifs sous forme de scénario et de diagramme séquence détaillé. Un diagramme de classe sera aussi présenter pour modéliser le côté statique de notre application. Ce diagramme sera transformer par la suite en model logique de données.

Dans le dernier chapitre intitulé « réalisation et mise en œuvre», nous détaillerons les étapes à suivre pour déployer, tester et maintenir l'application web. Où nous allons présenter l'architecture adoptée ainsi l'implémentation de l'application, la manière que nous allons choisir pour héberger notre future système, et les pratiques que nous allons mettre en œuvre pour garantir une sécurité optimale.

Nous terminerons ce mémoire par une conclusion générale, pour résumer l'essentiel du travail accompli, ainsi que quelques perspectives.

Chapitre I

Présentation du cadre du projet

MLM

New Life

Problématique

Objectif

Il existe de nombreuses sortes de marketing sur le web (T-Shirt Marketing, E-commerce, Blogging, etc...) et ça reste juste de la vente en ligne (e-commerce), contrairement au marketing multi niveaux offrant une autre manière d'acheter des produits, et des services offerts.

Ce chapitre portera sur une étude théorique sur le marketing de réseau, nous commençons par introduire le concept de distributions produit/service pour aboutir à la définition du marketing relationnel et ses principes appropriés, nécessaires pour la réalisation de notre travail.

Dans ce chapitre, nous présentons en premier lieu l'organisme d'accueil «New Life Opportunity » au sein duquel nous avons effectué le stage relatif au présent de notre projet. Nous effectuons ensuite une étude de l'existant ou nous décrivons le fonctionnement de cette entreprise. Cette étude permettra de déceler les problèmes rencontrés par cette entreprise de proposer des solutions qui sont présentées à la fin de ce chapitre.

I.1. Concept de distributions

La fonction de distribution peut être définie comme un ensemble d'opérations et d'étapes faisant suite à la fonction de production, mises en œuvre par le producteur lui-même ou par un ou plusieurs intermédiaires spécialisés, appelés distributeurs, pour mettre à la disposition du consommateur final (particulier ou entreprise) les biens produits ou services, et aussi pour faciliter et encourager leurs achats.

Cette définition met en exergue plusieurs points. Dans nos sociétés il existe deux concepts de distribution :

- Le marché traditionnel qui correspond à la vente aux magasins aux grandes surfaces ou sur internet avec les boutiques en lignes, la compagnie investit un coût énorme pour promouvoir leurs produits (Spot publicités à la télé, panneaux) Dans la distribution traditionnelle, 60 à 90 % du prix de vente hors taxe est absorbé par les différents intervenants (Distributeurs, Grossistes, Revendeurs ...), en effet les intermédiaires coûtent cher au final.

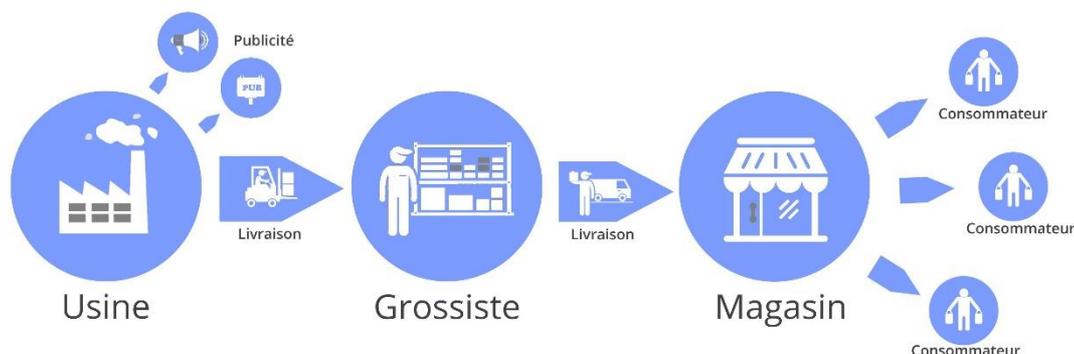


Figure 1 : Circuit de distribution traditionnel.

- Dans Le marché relationnel, les produits sont vendus directement sans publicité par le fabricant au consommateur final ; la publicité est remplacée par un travail d'information de bouche à oreille effectué par une partie des consommateurs qui choisissent d'être aussi distributeurs (aussi appelés réseauteurs). Les économies substantielles réalisées (jusqu'à 300% du prix de fabrication !) par la suppression de tous les intermédiaires et de la publicité permettent au fabricant de :
 - Réduire substantiellement le prix au détail de ses produits (à qualité égale).
 - Rémunérer les distributeurs, c'est-à-dire des clients satisfaits qui choisissent de se créer des revenus en recommandant le produit autour d'eux.

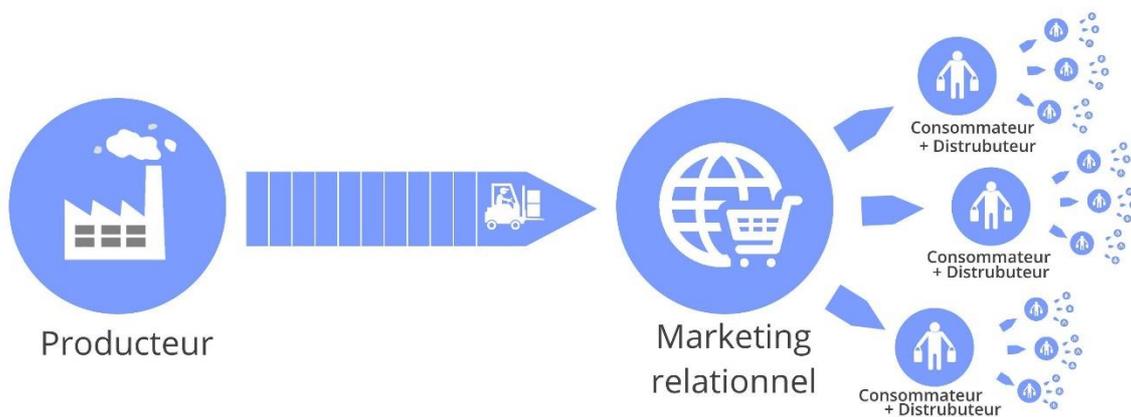


Figure 2 : Circuit de distribution relationnel.

Toute la stratégie de croissance du fabricant consiste donc à encourager le mieux possible le développement de réseaux de clients et de distributeurs/réseauteurs satisfaits de ses produits et de son éthique de marché, d'où l'appellation de « marketing de réseau. »

I.2. Définition

MLM est l'abréviation anglaise de « *Multi Level Marketing* » [1]. Ce terme est traduit en français par « marketing de réseau » ou « marketing multi niveaux ». Cette pratique de vente a vu le jour aux Etats-Unis dans les années 40 et connaît un essor fulgurant.

Le mot « Marketing » désigne l'action de diriger un produit au service d'un fabriquant ou fournisseur vers un consommateur. L'association au mot « Réseau » signifie qu'un système de rémunération est proposé aux personnes qui en assurent la distribution.

- **Marketing = vente**

Le MLM a pour mission d'assurer la distribution d'un ou plusieurs produits / services de bouche à oreille. De plus en plus de sociétés utilisent cette méthode pour vendre leurs produits et ainsi éliminer les intermédiaires entre le producteur et les clients. Le producteur vend au client qui devient vendeur à son tour, en échange d'une commission, et ainsi de suite.

- **Marketing de réseau = vente à notre entourage**

Nous achetons et apprécions un produit et nous en parlons autour de nous, à notre réseau (famille, amis, voisins, collègues ...). Séduites également, certaines personnes de notre entourage intéressées par l'achat du produit deviennent clientes à leur tour.

Inutile d'être un grand vendeur, le MLM n'est pas vraiment une affaire de vente. Il faut juste acheter en petites quantités pour nous, notre famille et amis proches. Pas de stock, pas d'investissement financier important.

- **Marketing multi niveau = parrainage**

Dans un réseau, certaines personnes sont intéressées par les produits et aussi par l'opportunité d'affaire et souhaitent devenir distributeurs. En les affiliant à la société, ces personnes deviennent nos filleuls et nous sommes leur parrain / marraine.

Comme tout le monde a un réseau, chacune des personnes de mon réseau possède également son propre réseau et ainsi de suite Progressivement, nous construisons une équipe de partenaires sur plusieurs niveaux.

Pour une meilleure compréhension, nous avons décidé de vous montrer un exemple concret du fonctionnement du MLM :

Vous arrivez dans un restaurant pour la première fois. Ce restaurant est excellent à votre goût. Le lendemain vous vous rendez à un dîner entouré de vos amis. Une fois à table, le premier sujet qui vous vient à l'esprit est votre sortie au restaurant. Quelques jours plutôt vous avez essayé un restaurant médiocre, l'expérience vous a marqué car elle était vraiment désagréable et votre premier réflexe était de prévenir vos amis pour qu'ils n'y mettent jamais les pieds. Mais cette fois ci c'était différent, vous avez vécu une agréable expérience et vous vous sentiez obligés de partager cette expérience avec vos amis : " Les gars ce resto était vraiment excellent, je vous le recommande beaucoup !"

Là, vous avez mis en pratique l'art du Marketing de réseau !



Figure 3 : Exemple du « MLM ».

Le résultat, Vous avez parlé de ce restaurant à trois personnes, qui, à leur tour en parleront à trois personnes, ainsi s'en suit. Le message s'est bien transmis et ces personnes iront donc vivre l'expérience agréable que propose ce restaurant.

Parties de ce constat, certaines entreprises ont donc décidé d'utiliser ce filon, en sous-traitant leur distribution à des consommateurs, qui deviennent alors des distributeurs indépendants.

Participer au marketing de réseau consiste à devenir un affilié ou un distributeur [2], quelle est la différence entre les deux ? [2]

I.2.1. Distributeur

Un distributeur en Marketing de Réseau est un représentant officiel dûment autorisé par contrat d'une compagnie MLM qui crée ses propres services ou produits (physiques ou virtuels).

Le distributeur va commercialiser ses produits ou services auprès de particuliers ou entreprises.

Il va proposer à d'autres personnes l'opportunité de devenir distributeurs à leur tour !

Il propose à ces nouveaux distributeurs un suivi personnalisé et de les former sous la forme de (Informer – Former – Dupliquer) en partageant sa propre expérience et/ou en utilisant les outils mis à disposition par la compagnie MLM voire par d'autres outils de développement.

Il va aider ses filleuls à apprendre le métier du MLM, inviter ses distributeurs à se former aux produits ou services de la compagnie et les aider à dupliquer ce qu'ils ont appris pour qu'eux-mêmes apprennent à leurs filleuls ces mêmes concepts, il va ainsi dupliquer sa force de vente et son organisation.

Le distributeur en Marketing de Réseau génère des revenus par ses propres ventes et principalement par la production de son groupe de distributeurs qui lui assure de la récurrence sur le long terme et génère un portefeuille clients et distributeurs.

Le nombre de niveaux sur lequel le distributeur peut être rémunéré en Marketing de Réseau est supérieur à 2 et peut aller jusqu'à 10 niveaux et plus (Filleuls qui enrôlent des nouveaux filleuls qui enrôle des filleuls... et ainsi de suite...)

De par son propre groupe, le distributeur met en place ainsi un revenu résiduel pour une longue période.

Un Leader en marketing de réseau fait une seule action pour laquelle il peut être rémunéré à vie.

Un Leader en marketing de réseau, recevra tous les mois les fruits de son travail, pour avoir aidé ses partenaires à réussir.

L'industrie du marketing de réseau fait partie de celles qui créent le plus de millionnaires dans le monde.



I.2.2. Affilié

Un Affilié va mettre en place une stratégie de marketing Internet pour se constituer une liste ciblée de prospects sans pour cela avoir un contact direct avec ceux-ci.

Il va proposer à ses prospects des produits ou services d'entreprises par Internet pour la plupart du temps virtuel du type (E-book, cours audio ou vidéo, formation en ligne, outils marketing etc...)

La vente se réalisera par l'entremise de l'entreprise recommandée (et non par l'Affilié)

Il percevra une commission ou un pourcentage sur la vente réalisée octroyée par la société suivant accords définis dans les conditions générales "Entreprise/Affilié".

Il ne paie pas de redevance ou de commande automatique mensuelle pour proposer les produits ou services de ces entreprises ou comme conditions pour percevoir ses commissions.

Il n'est pas autorisé à représenter l'entreprise ou d'agir en son nom.

Il est libre de toutes obligations quant à un rendement minimum mensuel ou annuel ou de chiffre d'affaires pour percevoir ses commissions.

Il n'a pas de contact privilégié avec ses clients ou filleuls et ne forme pas ceux-ci aux produits ou services proposés.

L'entreprise s'occupe du suivi des clients et filleuls Affiliés.

Il est hors contrainte de proposer à ses mêmes prospects des produits similaires ou directement concurrentiels.

Il ne fait que relayer l'information sans être un représentant officiel de ces entreprises.

Il perçoit des commissions uniquement sur ses propres ventes, voire sur un niveau supplémentaire au plus ! (premier niveau ses propres ventes, deuxième niveau les ventes de ses filleuls directs).

Le Sponsor ou le Parrain n'a aucune importance, puisque l'Affilié ne reçoit rien de lui, c'est l'entreprise qui s'occupe de tout.

Un affilié n'est pas responsable d'un réseau de partenaires qu'il doit aider.



I.3. Principe

La spécificité du MLM est d'inciter le vendeur (l'affilié ou le distributeur) à parrainer d'autres vendeurs sur le principe gagnant-gagnant :

- Plus il y a de vendeurs plus il y a de ventes, **Le producteur du produit est gagnant.**
- Plus un vendeur parraine et plus il vend, plus il gagne des commissions, **Le vendeur est gagnant**

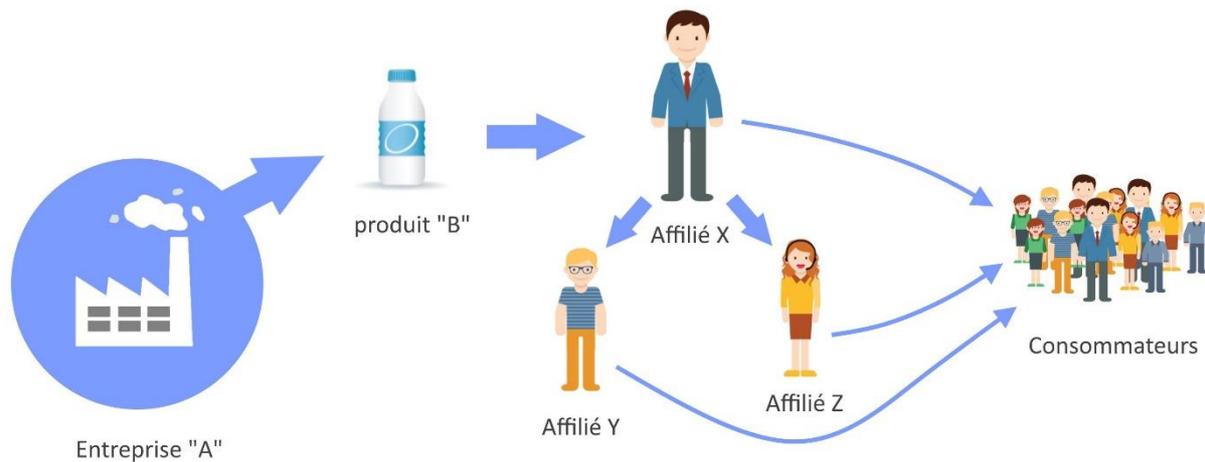


Figure 4 : Principe du « MLM ».

Légende : La société « A » propose un produit « B ».

L'affilié X parraine les affiliés Y et Z, et tous vendent le même produit aux consommateurs. Plus il y a de vendeurs plus le nombre de consommateurs sera élevé.

Pratiquer le Marketing de Réseau permet de générer un revenu complémentaire à domicile, nous distinguons 3 types de revenus :

- Revenu tiré des ventes (pourcentage ou fixe par vente)
- Revenu tiré du parrainage (fixe par affilié)
- Revenu tiré des ventes des filleuls (pourcentage ou fixe par vente)

Les entreprises qui se lancent dans le marketing de réseau, sous-traitent des consommateurs déjà convaincus par le produit. Ces derniers travaillent en indépendants, de chez eux, et sont rémunérés sur un pourcentage de leurs propres ventes de produits/services aux clients de leurs réseaux, et touchent une commission sur tous les produits vendus par les affiliés recrutés par son intermédiaire [3].

I.4. Avantages et inconvénients

La solution du Marketing de Réseau est l'assurance pour une entreprise de toucher le maximum de prospects (personne susceptible d'acheter le produit) en un minimum de temps grâce à l'effet de vague de ce système.

L'inconvénient est qu'il faut rémunérer tous ses affiliés et surtout d'avoir l'infrastructure suffisante pour supporter le poids de milliers d'affiliés et de clients : site Internet, plateforme logistique, etc... [3].

I.5. Présentation de l'entreprise

« New Life » est une entreprise de vente en ligne avec système de marketing de réseau, qui est en phase de création et siégeant au Boulevard Colonel AMIROUCHE, CLAIRVAL, dans la commune d'Alger centre.

L'entreprise est composée d'une équipe de professionnels partout en Algérie, passionnés par le marketing et le travail de groupe, qui travaillent depuis maintenant plusieurs années à développer un nouveau concept de marketing de réseau de telle sorte à le rendre plus simple et plus accessible afin que n'importe quelle personne puisse le comprendre et le pratiquer.



Figure 5 : Logo de « New Life ».

« New Life » présente un nouveau système innovateur qui est une puissante combinaison entre les modes de ventes, le plan en ligne et le plan binaire, qui sont décrit ci-après :

I.5.1. Plan en ligne

Le plan en ligne, vente en ligne ou E-commerce, s'agit de la vente de produit à travers le réseau internet, c'est-à-dire que les clients peuvent passer leurs commandes sans devoir se déplacer, par cela « New Life » désire toucher un maximum de public et augmenter de façon significative leurs bénéfices et ventes respectifs.

I.5.2. Plan binaire

« New Life » propose aussi un plan de rémunération binaire qui limite à n'avoir que deux filleuls directs (deux jambes) sur la première ligne. Dans le cas où d'autres personnes se joignent au réseau, elles seront placées en dessous des deux premiers filleuls.

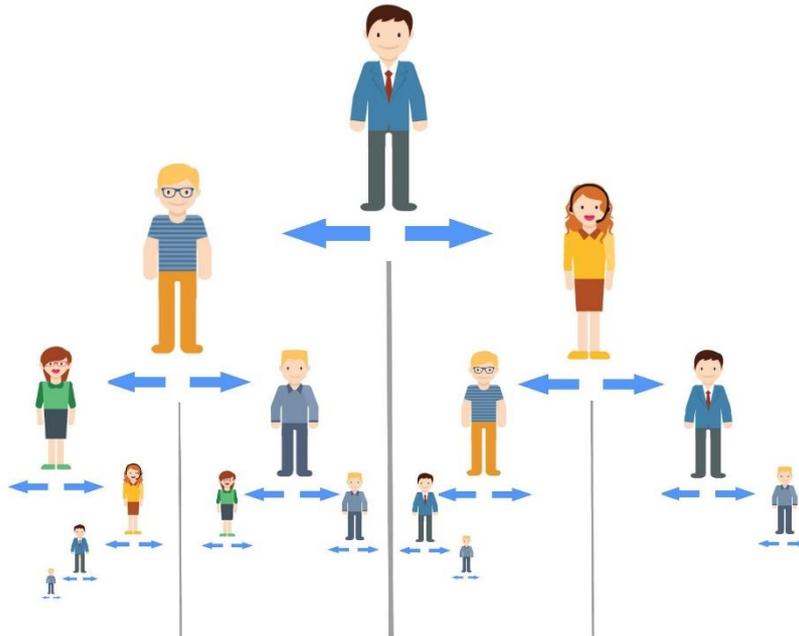


Figure 6 : Plan binaire.

En ce qui concerne la rémunération offerte par « New Life », celle-ci se concentre sur la jambe faible (Le côté où il y a moins de fils), le client doit se consacrer sur cette dernière, afin de maximiser ses gains en équilibrant les deux côtés.

I.6. Fonctionnements de l'entreprise

« New Life » a su fusionner le plan binaire avec le plan en ligne. Par ailleurs, elle élabore la façon dont les clients s'inscrivent à « New Life », l'avantage qu'offre « New Life » pour ses clients, ainsi que la procédure générale du paiement. Nous retrouvons tout de même une certaine incompréhension de la part du client même si le système en lui-même se montre assez simple.

I.6.1. Inscription

« New Life » offre le choix entre une multitude de produits de haute qualité à des prix compétitifs. Pour s'inscrire à « New Life », il faudrait acheter au moins un produit tout en payant des frais d'inscription.

1.6.1.1. Choix du produit

- L'entreprise s'est alliée à plusieurs fournisseurs par le biais de conventions de produits (Iris SAT [4] et JATA électronique [5]),
- Tous les produits sont sous garantie.
- Dès l'achat d'un produit, un service de livraison prend en charge la livraison jusqu'au domicile du client, grâce à une convention avec UPS [6] et EMS [7] des sociétés de livraison.

1.6.1.2. Pack d'inscription

- Quatre packs d'inscription sont offerts au client pour s'inscrire (pack classic, pack évolution, pack premium, pack initiation).
- Il y a plusieurs différences entre les packs d'inscription qui font que ces derniers sont flexibles par rapport au pouvoir d'achat du client.
- Parmi les différences des packs, nous trouvons la quantité du produit acheté ou l'investissement du client, la valeur des frais etc...

1.6.1.3. Frais d'inscription

- Les frais d'inscription varient selon le pack d'inscription du client.
- Les frais d'inscription couvrent la sécurité du compte du client, l'opportunité de pouvoir vendre les produits de « New Life » et bien plus encore.

1.6.1.4. Pack pro

Dès son inscription le client disposera directement d'un contrat de travail plus un pack pro correspondant à son pack d'inscription. Il contient différents outils de travail (carte de visite, CD, block note ...).



Figure 7 : Pack pro.

« New Life » dispose d'une monnaie propre à elle, intitulée « EM » (électronique monnaie) elle diffère selon les pays, 1EM = 100DA pour l'Algérie

PACK D'INSCRIPTION	INVESTISSEMENT	FRAIS D'INSCRIPTION	ACHAT
INITIATION	66 EM	66 EM	0 EM
CLASSIC	286 EM	160 EM	126 EM
EVOLUTION	587 EM	210 EM	377 EM
PREMIUM	986 EM	310 EM	676 EM

Tableau 1 : les packs d'inscription.

Dans ce tableau, l'investissement est égal à la somme des frais d'inscription plus le prix d'achat du produit.

I.6.2. Système de rémunération

« New Life » offre à ses clients deux moyens différents de gain, soit le client désire simplement vendre les produits de « New Life » avec un pourcentage de gain, et là nous parlons de la vente directe, ou il préfère parrainer d'autres personnes pour rentrer dans un autre système complètement différent qui est « le système binaire ».

I.6.2.1. Vente direct

C'est un système simple qui consiste à vendre les produits de « New Life » avec un pourcentage de gain, qui diffère selon le produit acheté. Les gains du client sont limités à ce pourcentage.

I.6.2.2. Système binaire

C'est un système très complexe qui ne pourra pas être mis en œuvre sans une aide informatique. Lors de l'inscription, le client disposera d'un compte avec deux cotés un gauche et un droit. Là, le client pourra parrainer d'autres personnes, et construire son propre réseau, afin d'encaisser ses commissions et d'effectuer ses achats.

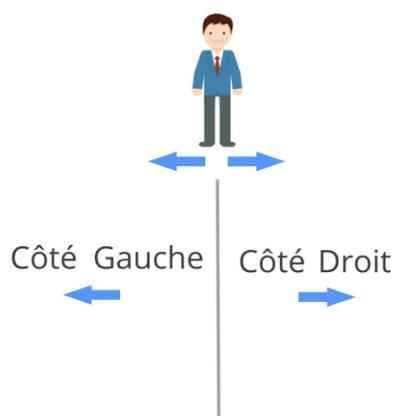


Figure 8 : Système binaire.

I.6.2.2.1. Les commissions

Lors d'un parrainage, « New Life » dispose de trois commissions différentes, les commissions directes, les commissions d'équilibre, et les gift bonus.

- **Les commissions directes :**

Lors d'un parrainage, le client encaisse directement les commissions directes selon le pack du compte parrainé.

PACK D'INSCRIPTION	COMMISSIONS DIRECTES
INITIATION	10 EM
CLASSIC	15 EM
EVOLUTION	30 EM
PREMIUM	35 EM

Tableau 2 : Commissions directes.

- **Les commissions d'équilibres :**

Contrairement aux commissions directes, les commissions d'équilibres ne sont pas encaissées directement par le client, mais juste lors de l'équilibre des deux côtés « gauche et droit », ils utilisent le concept de un point à gauche plus un point à droite égale à 10 EM « 1/1=10EM »

Les points différents selon le pack du compte parrainé

PACK D'INSCRIPTION	POINT D'EQUILIBRE
INITIATION	1 point
CLASSIC	3 points
EVOLUTION	5 points
PREMIUM	7 points

Tableau 3 : Commissions d'équilibres.

« New Life » ne dispose pas de limite verticale, donc si le *n*ème client parraine, alors tous ses ascendants encaissent des points.

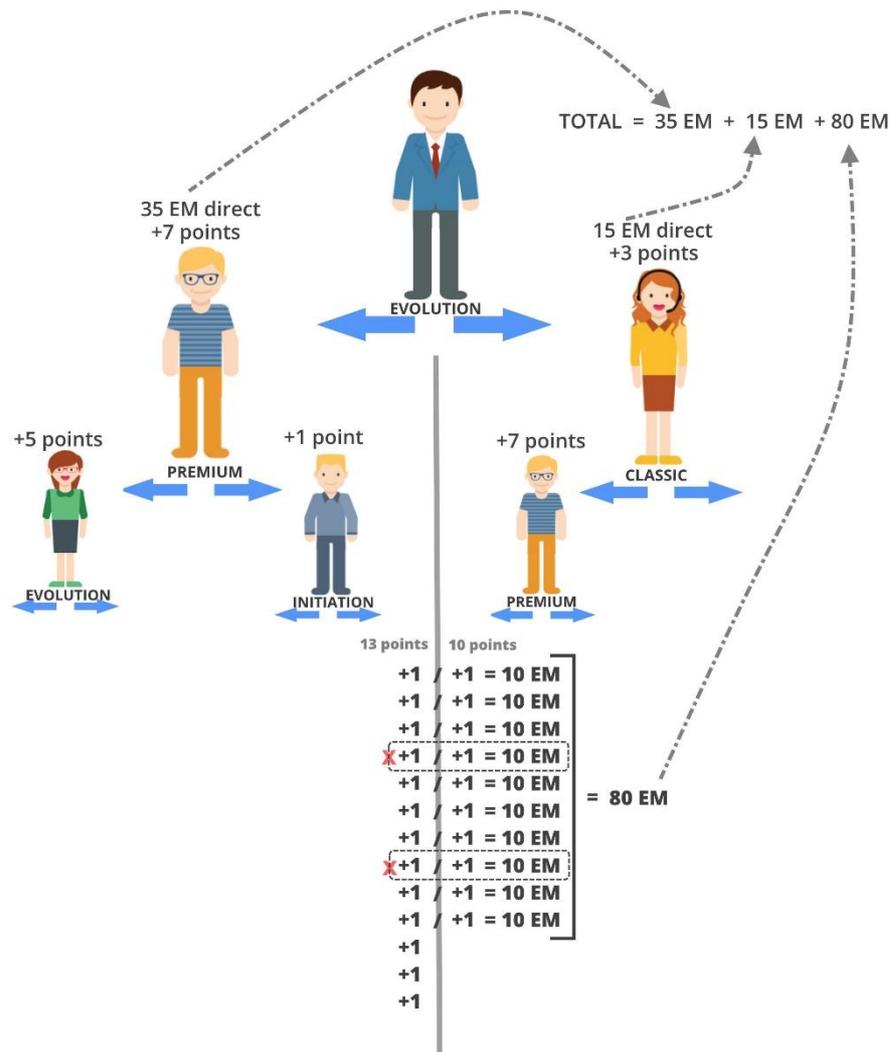


Figure 9 : Les commissions d'équilibre.

- **Les Gift bonus :**

Les commissions gift bonus sont en relation avec les commissions d'équilibre, à chaque trois commissions d'équilibre encaissées, la quatrième se transforme en gift bonus.

Avec les gift bonus les clients peuvent juste consommer des produits gift bonus (voyage formation ...)

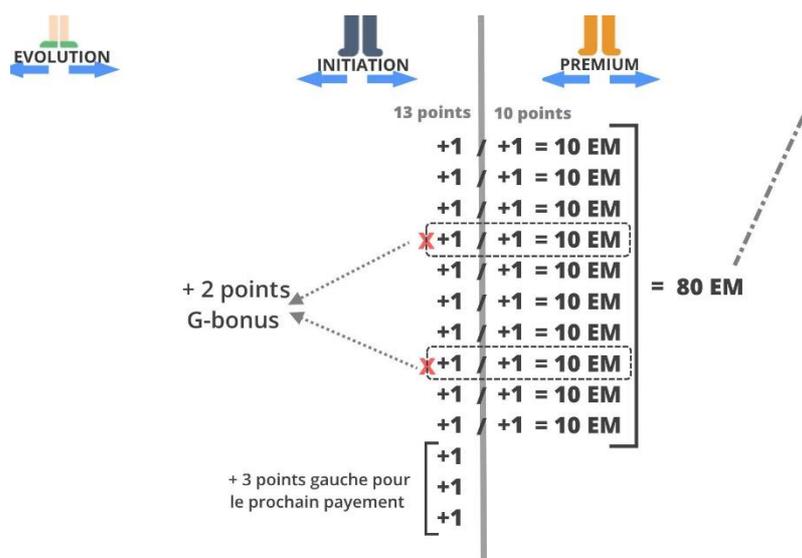


Figure 10 : Les Gift bonus.

I.6.2.2.2. Bonus mensuel

Le but de ce bonus est de motiver les clients de « New Life », si un client arrive à avoir cinq inscriptions directes et quinze inscriptions de son groupe pendant une période d’un mois, il aura un bonus à la fin du mois selon son pack d’inscription. Les bonus sont comme suit :

PACK D’INSCRIPTION	BONUS MENSUEL
INITIATION	0 EM
CLASSIC	46 EM
EVOLUTION	107 EM
PREMIUM	166 EM

Tableau 4 : Bonus mensuel.

I.6.2.2.3. Bonus de démarrage

Pour favoriser les niveaux « évolution » et « Premium », « New Life » a ajouté un bonus de démarrage en gift bonus, qui est additionné automatiquement au compte gift du client.

PACK D’INSCRIPTION	BONUS DE DEMARGE
INITIATION	0 G-Bonus
CLASSIC	0 G-Bonus
EVOLUTION	7 G-Bonus
PREMIUM	17 G-Bonus

Tableau 5 : Bonus de démarrage.

I.6.3. Les contraintes de « New Life »

« New Life » présente plusieurs avantages mais elle a aussi quelques contraintes concernant les gains et la méthode de travail, cela est mis en place afin de permettre la continuité du système.

I.6.3.1. Premier retrait

L'une des contraintes de « New Life » est le premier retrait. Chaque client inscrit à « New Life » est obligé d'atteindre une somme pour pouvoir retirer ses gains, cette somme varie selon le pack du client inscrit.

PACK D'INSCRIPTION	PREMIER RETRAIT
INITIATION	1 EM
CLASSIC	300 EM
EVOLUTION	300 EM
PREMIUM	300 EM

Tableau 6 : Premier retrait.

I.6.3.2. Max Out

Le Max out est une contrainte qui permet de définir le maximum de gain. C'est le maximum de commissions qu'un client peut recevoir par semaine. Ce principe s'adapte pour les trois derniers packs. Par rapport au premier pack, c'est un autre système, le client peut juste recevoir une somme par réactivation jusqu'à ce qu'il paye les frais d'inscription. C'est à dire que le compte du client restera bloqué à une somme jusqu'à ce que le client paye les frais d'inscription, et cela à chaque fois que le compte atteint cette somme.

PACK D'INSCRIPTION	MAX OUT
INITIATION	660 EM / réactivation
CLASSIC	600 EM / semaine
EVOLUTION	1200 EM / semaine
PREMIUM	2000 EM / semaine

Tableau 7 : Max out.

I.6.3.3. Frais d'inscription annuel

Pour garantir une continuité de l'entreprise, les clients de « New Life » sont obligés de payer un pourcentage (20%) de frais d'inscription chaque année. contrairement, la totalité de la somme sera retirée automatiquement de son compte, et cela après un délai fixé par « New Life ».

1.6.4. Système de paiement

Pour satisfaire ses clients, « New Life » dispose de plusieurs moyens de retrait afin de leur offrir un large choix.

- **Retrait cash :**

Un simple retrait cash en échangeant les EM en dinars.

- **Sous forme de produit :**

Si le client désire retirer ses gains en produit, il peut juste faire une commande à son nom et payer avec ses commissions.

- **Virement ccp :**

Les clients peuvent toujours demander un virement CCP en payant une taxe de 10% en contrepartie.

- **Virement bancaire :**

Même chose que le virement CCP avec une taxe de 20%.

- **Virement PayPal, Visa, master carte ou Flexy :**

Les clients peuvent contacter « New Life » pour d'éventuels moyens de retrait.

1.6.5. Facilités de paiements

« New Life » a mis au point un système d'inscription flexible pour donner une chance à toute catégorie de personne de s'inscrire, et de bien profiter des différents avantages de l'entreprise. Dans le but de rendre encore ce système plus flexible, elle a ajouté un moyen de paiement par facilité, qui diffère selon le pack d'inscription choisi :

PACK D'INSCRIPTION	MAX OUT	VALEUR RESTANTE DANS LE COMPTE	RESTE A AJOUTER
INITIATION	Pas existant	/	/
CLASSIC	160 EM	0 EM	126 EM
EVOLUTION	300 EM	90 EM	287 EM
PREMIUM	400 EM	90 EM	586 EM

Tableau 8 : Facilités de paiements.

Le client a deux moyens pour compléter la somme restante : soit il rajouter directement cette somme avec ses propres moyens, ou il la complète avec ses gains de « New Life ».

I.7. Problématique

Comme l'entreprise « New life » trouve des difficultés pour gérer son système de vente et son plan binaire. C'est pour cela que « New Life » affronte les anomalies suivantes :

- Difficultés majeures dans la gestion des comptes clients (parrainages, rémunération ...).
- Difficulté de gérer ses produits.
- Difficultés de communication entre les services (service commercial, service de paiement).
- Difficulté de recherche d'informations due à l'utilisation d'une grande masse de données.

La prise de contact par Internet est souvent très efficace et valorisable pour un client souhaitant obtenir une offre ou un renseignement rapidement. Cela permet à l'entreprise d'améliorer les fonctions et les processus de travail, c'est pour cela que nous allons réaliser un système informatique qui consiste à mettre en place une application web pour l'entreprise qui garantira au mieux la bonne gestion de l'entreprise.

I.8. Objectif du travail

Les responsables de la société ont exprimé leurs besoins et leurs souhaits concernant l'application lors d'une réunion, et afin d'améliorer le système de travail, il nous revient à proposer une solution en mettant en place une application web. En créant cette application, nous réaliserons les objectifs suivants :

- Informatiser le système de fonctionnements de « New Life ».
- Donner à l'entreprise une image professionnelle.
- Gestion à distance (commandes de produit, paiements).
- Gestion des comptes, des paiements, des produits et l'intégralité du service de « New Life ».
- Fournir des informations riches, à jour et facilement accessibles pour les clients.
- Rendre les produits de « New Life » disponibles à tout instant pour ces clients.

Cette application doit être donc aisément administrable, grâce à une interface simple et intuitive. Celui-ci devra bien entendu être accessible de l'extérieur.

Dans ce chapitre nous avons présenté «le marketing de réseau », en particulier, nous avons présenté ses principes, son fonctionnement et les différents services qu'il offre, pour garantir une meilleure compréhension du sujet du stage et des fonctionnalités du système.

Dans un deuxième temps ce chapitre nous a permis de nous imprégner du thème d'étude. Nous avons pu définir les objectifs de l'organisme d'accueil, d'avoir une vue globale du système existant, et de détecter les problèmes majeurs auxquels l'entreprise est confrontée sans un système informatisé.

Dans le prochain chapitre, nous allons aborder l'analyse et la spécification des besoins de notre projet. En particulier, nous allons élaborer un cahier de charge qui va résumer les besoins de l'entreprise et qui va définir les différents éléments comme les cas d'utilisations qui vont servir de base à la conception de notre application web.

Chapitre II

Spécification et analyse des besoins

Identification des acteurs

Identification des messages

Diagramme de contexte

Diagramme des cas d'utilisations

Depuis quelques années, la modélisation objet avec le langage UML est devenue incontournable pour la plupart des projets informatiques, c'est pourquoi nous l'appliquerons à notre projet web. En effet, l'UML nous permet de maîtriser la situation et de la rendre plus confortable avec ses différents diagrammes tout en s'appuyant sur La méthode Agile « eXtreme Programing ».

Ce chapitre est consacré à la description des besoins de notre application en utilisant le langage naturel puis le langage semi formel UML qui permet de pallier toutes les ambiguïtés dans l'expression des besoins.

II.1. Spécification des besoins

C'est la description des besoins et les conditions qui doivent être respectés, nous distinguons alors deux types de besoins :

II.1.1. Les besoins fonctionnels

Les besoins fonctionnels expriment les actions que doit effectuer le système en réponse à la demande des gérants de « New Life » qui ont réussi à exprimer leurs besoins à l'aide des user stories que nous leur avons défini, nous avons pu les subdiviser comme suit :

Le système de l'application doit être conforme aux fonctionnements du système de « New Life ».

Préinscription : un nouveau client peut se préinscrire pour être parrainé par la suite, en remplissant ses informations personnelles telles que son nom, prénom, sexe, email, numéro de téléphone, ainsi que son moyen de paiement (Cash, Facilité).

User stories :

Acteur : client
Action : préinscription
Objectif : pour être parrainée par la suite

Transfert EM : tout client parrainé et le gérant lui-même peuvent envoyer des « EM » afin de payer les frais de parrainage (frais d'inscription avec l'investissement).

User stories :

Acteur : client parrainé, gérant
Action : transfert du « EM »
Objectif : régler les conditions de parrainage.

Parrainage (validation) : chaque client déjà parrainé et même le gérant lui-même peuvent parrainer de nouveaux clients, à condition que ces derniers soient déjà préinscrits, Ainsi le client parrainé dispose des « EM ».

User stories :

Acteur : client parrainé, gérant
Action : Parrainage
Objectif : un nouveau client parrainé.
Condition :

- nouveau client doit être préinscrit
- Client parrainé dispose des « EM » afin de régler les frais de parrainage du nouveau client

Inscription finale : un nouveau client parrainé doit compléter ses informations (son adresse exacte, code postal et numéro de sa carte d'identité) pour profiter des avantages de « New Life » et également d'autres informations facultatives à savoir (compte CCP, compte bancaire, informations de l'héritier).

User stories :

Acteur : Nouveau client
Action : inscription finale
Objectif : le nouveau client sera un client parrainé qui profitera des avantages de « New Life »

Gestion des informations : un client parrainé peut à tout moment modifier ses informations personnelles.

Migration : un client parrainé peut migrer d'un pack au second une seule fois.

Visionnages de l'arbre : un client parrainé peut consulter son arbre qui permet de vérifier ses filleuls.

Paiement (commissions d'équilibre) : chaque dimanche à 5h du matin, le système paye les clients parrainés dont les commissions sont équilibrées.

User stories :

Acteur : système
Action : déclenchement automatique de la vérification des points d'équilibre
Objectif : payer les commissions d'équilibre.
Condition : déclenchement chaque dimanche à 5h du matin

Commissions direct : un client parrainé peut choisir, s'il désire, retirer ses commissions directes ou les ajouter aux commissions du premier retrait.

Conversion des commissions : un client parrainé peut convertir ses commissions afin de parrainer de nouveaux clients, ou les transférer à des clients parrainés.

User stories :

Acteur : client parrainé

Action : conversion et transfert des commissions

Objectif : parrainer de nouveaux clients.

Note : un client parrainé peut convertir ses commissions, à condition d'atteindre la somme du premier retrait et d'avoir l'achat validé.

Demande de retrait : un client parrainé peut demander de retirer ses commissions à travers des virements ccp (plus 1000 EM) ou virement bancaire (plus de 2000 EM) avec moins 10% prélevé comme taxe sur la somme de son retrait.

Consultation : un client parrainé peut consulter ses commissions ainsi que son bonus mensuel.

Cash : un nouveau client inscrit en cash (ayant payé la totalité du produit), peut demander la livraison de son produit à domicile, en payant les frais de livraison afin de valider son achat.

User stories :

Acteur : client parrainé en cash.

Action : pouvoir demander la livraison de son produit à domicile.

Objectif : valider son achat.

Facilité : un nouveau client inscrit en facilité de paiement a le choix de payer le reste de la somme avec ses commissions reçues ou de les rajouter directement en contactant le gérant et cela pour avoir les droits et les actions d'un client parrainé en cash.

User stories :

Acteur : client parrainé avec facilité de paiement

Action : payer le reste de la somme avec ses commissions ou rajouter le reste de la somme directement

Objectif : avoir les droits et les actions du client parrainé en cash.

Panier : un client parrainé peut gérer son panier.

Historique : le système doit enregistrer chaque trace des transactions pour chaque client parrainé afin de consulter son historique.

Administrateur : L'administrateur peut :

- Ajouter des EM au system.
- Envoyer des Messages au client.
- Gérer totalement les comptes des clients.
- Gérer les sommes des bonus mensuels.
- Gérer totalement les produits de « New Life ».

Service commercial : le service commercial peut :

- Afficher les nouveaux clients pour leur envoyer leur pack pro.
- Valider les produits en les envoyant au service de livraison.
- Imprimer des étiquettes à coller sur chaque produit.
- Gérer les produits de « New Life » à part les catégories de produit.

Gestion des stocks : le système doit être capable de gérer les stocks et de prévenir lors d'une rupture de stock.

Sponsor : réserver un espace au bas de page pour les logos des sponsors de « New Life ».

Newsletters : les internautes peuvent s'inscrire sur la newsletter pour recevoir les informations de « New Life ».

Publicité : le gérant de « New Life » peut ajouter des publicités sur les pages les plus visitées.

II.1.2. Les besoins non fonctionnels

Ce sont des fonctionnalités qui participent à l'amélioration du site. Ces besoins regroupent les contraintes techniques, les contraintes de sécurité, l'esthétique et les contraintes ergonomiques.

Technique : la création d'une application web nécessite l'utilisation de différents langages de programmation et techniques d'implémentation qui facilitent le développement de l'application tels que :

- L'utilisation du langage de programmation PHP [8].
- L'utilisation de java script pour manipuler les formulaires.
- L'utilisation de la base de données MySQL [9].
- La méthode agile XP [10] (Voir ANNEXE 1) basée sur le langage de modélisation UML [11] (Voir ANNEXE 2).
- L'utilisation de CCS3 [12] pour bien structurer l'application.

- L'utilisation d'Ajax [13] pour améliorer la rapidité de l'application et la qualité des interfaces.
- Trouver un système simple et efficace pour le paiement en ligne.
- Ajouter des métas data pour que l'application soit facilement identifiable par les moteurs de recherche ainsi que les réseaux sociaux.

Sécurité :

- Les mots de passes des clients seront hachés pour assurer la sécurité des données.
- Toutes les transactions importantes seront plus sécurisées (droit d'accès, code PIN etc...)
- Chaque page va disposer d'une vérification des droits d'accès.
- Une sécurisation des transferts des données est primordiale pour la sécurité des transactions.
- Assurer une grande fiabilité du système.

Esthétique :

- Les couleurs de l'application doivent être pâles pour ne pas agresser l'œil et de manière à améliorer la lisibilité.
- La police utilisée doit avoir un caractère bien intelligible et perceptible puisque notre travail est orienté vers une population.
- La navigation doit être logique, simple et intuitive, l'internaute ne doit pas s'y perdre.
- Garder la même couleur de l'arrière-plan pour toutes les pages.
- Effet de zoom sur les photos des produits.

Ergonomie : Il faut réaliser une interface qui respecte les règles ergonomiques afin de :

- Donner à l'administrateur la possibilité de gérer les informations dans la page d'accueil.
- Communication homme machine à travers des messages d'explication et de prévention avant les transactions importantes, et des messages expliquant l'état du compte du client.
- Familiariser rapidement le nouveau client avec le site.
- Recevoir des notifications lors des transactions.
- Placer le logo dans un lieu bien visible et identifiable.
- Rapidité du service, vu le système de « New Life », les inscriptions vont se multiplier rapidement, il faut que notre application puisse prendre en charge une telle surcharge du système sans perdre en efficacité et en performance, pour cela nous devons chercher un serveur puissant et avec une grande bande passante.
- Avoir une vue globale de l'enchaînement des états de livraison du produit.
- un système de recherche efficace.

II.2. Analyse des besoins

Dans cette partie, nous effectuerons une analyse qui se base sur les résultats de l'étude réalisée au cours de notre stage à « New life ».

II.2.1. Identification des acteurs

Un acteur est un utilisateur (humain, dispositif matériel ou autre système) qui désire effectuer des tâches précises sur l'application.

D'après la spécification des besoins, nous pouvons distinguer les acteurs du système qui sont :

L'internaute : représente le simple visiteur qui navigue sur l'application web sans avoir besoin de s'authentifier, comme il peut se préinscrire.

Client : personne possédant un compte, ayant les mêmes droits d'accès que le client parrainé (parrainage, gestion compte, gestion panier ...). Pour bien distinguer entre les différents états du client, nous les avons subdivisés en transitions comme suit :

<i>Etat</i>	Explication	Evènement déclencheur du passage à l'état suivant
<i>A</i>	pré-inscription	La validation par le parrain (parrainage)
<i>B</i>	compte validé (parrainer)	L'inscription finale (remplir les informations : adresse, numéro de CNI...)
<i>C</i>	compte complété	Les commissions atteignent la somme d'achat (compte inscrit en facilité de paiement)
<i>D</i>	sommes atteinte	Consommation de la somme d'achat correspondant au pack d'inscription (compte inscrit en cash)
<i>E</i>	la valeur du produit est achetée	Les commissions atteignent un minimum pour un premier retrait
<i>F</i>	premier retrait atteint	Migration
<i>G</i>	une migration est effectuée	Fin

Tableau 9 : Les états de transition du «client ».

Administrateur (gérant de « New Life ») : L'administrateur est une personne qui ouvre droit à tout accès, et contrôle de manière intégrale l'application web (la maintenance, le contrôle des conditions d'accès etc...)

Compte commercial : c'est le compte géré par le service commercial, celui-ci permet de gérer de manière générale tout le fonctionnement du service (l'impression des étiquettes, la validation des produits et packs pro avant la livraison, et la gestion des produits).

Service de paiement : c'est l'acteur qui se charge des fonctionnalités automatiques du système. Ce dernier est chargé de veiller au paiement des commissions d'équilibres et à la gestion des notifications et de l'historique.

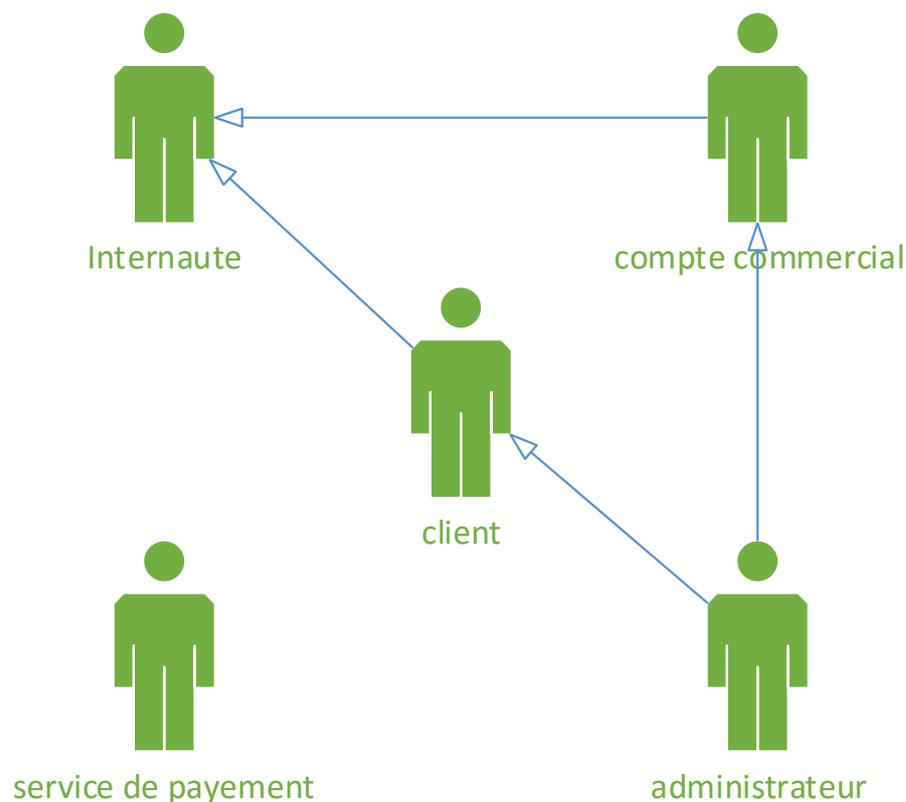


Figure 11 : Les acteurs du site.

II.2.2. Identification des messages échangés

Un message est un moyen de communication entre acteurs. Il définit un événement, c'est-à-dire, une information envoyée à un acteur et provoquant en réponse le commencement d'actions associées à ce dernier

Les acteurs de notre système, peuvent échanger des messages qu'ils soient entrants ou sortants. En effet, les messages entrants représentent les demandes qu'un acteur effectue tandis que les messages sortants représentent la réponse du système à une demande donnée.

II.2.2.1. Messages entrants échangés avec le système

ACTEUR	MESSAGES ENTRANTS AU SYSTEME
<p style="text-align: center;">INTERNaute</p>	<ul style="list-style-type: none"> • Demande de la liste des produits • Inscription aux newsletters • Demande des informations relatives au pack d'inscription • Préinscription
<p style="text-align: center;">COMPTE COMMERCIAL</p>	<ul style="list-style-type: none"> • Demande de la liste des demandes de livraison • Demande de la liste des nouveaux inscrits • Demande d'une version imprimable des étiquettes des produits • Gestion des produits
<p style="text-align: center;">ADMINISTRATEUR</p>	<ul style="list-style-type: none"> • Demande de la gestion de l'application (page accueil, publicité, bonus, etc...) • Demande de modification des informations des clients • Ajoute des EM • Envoi des messages
<p style="text-align: center;">SERVICE DE PAYEMENT</p>	<ul style="list-style-type: none"> • Envoi des commissions d'équilibre, des Gift-Bonus et des bonus mensuel. • Retrait des frais de renouvellement annuel • Envoi des notifications et gestion de l'historique
<p style="text-align: center;">CLIENT</p>	<ul style="list-style-type: none"> • Complément des informations • Parrainage • Transfert EM • Consultation des commissions • Consultation de l'historique

	<ul style="list-style-type: none"> • Demande de la liste des feuilles sous forme d'un arbre • Gestion du compte • Validation de l'achat et les achats de la vente directe ainsi que la livraison de ses derniers • Gestion du panier • Demande de la liste des messages reçus par l'administrateur
--	---

Tableau 10 : Les messages entrants échangés avec le système.

II.2.2.2. Messages sortants échangés avec le système

ACTEUR	MESSAGES SORTANTS DU SYSTEME
ITERNAUTE	<ul style="list-style-type: none"> • Liste des produits • Liste des packs et leurs informations
COMPTE COMMERCIAL	<ul style="list-style-type: none"> • Liste des demandes de livraisons des produits • Liste des demandes des livraisons de pack pro (les nouveaux inscrits) • Version imprimable des étiquettes
ADMINISTRATEUR	<ul style="list-style-type: none"> • Toutes les informations relatives aux clients • Liste des publicités déjà ajoutées • Liste des demandes de retrait
SERVICE DE PAYEMENT	/
CLIENT	<ul style="list-style-type: none"> • Liste des filleuls • Liste des transactions (Historique) • Liste des produits dans le panier • Informations des produits

Tableau 11 : Les messages sortants échangés avec le système.

II.2.3. Diagramme de contexte du système à réaliser

C'est un diagramme représentant les acteurs externes qui interagissent avec le système. Il est considéré comme la plus haute vue de ce système.

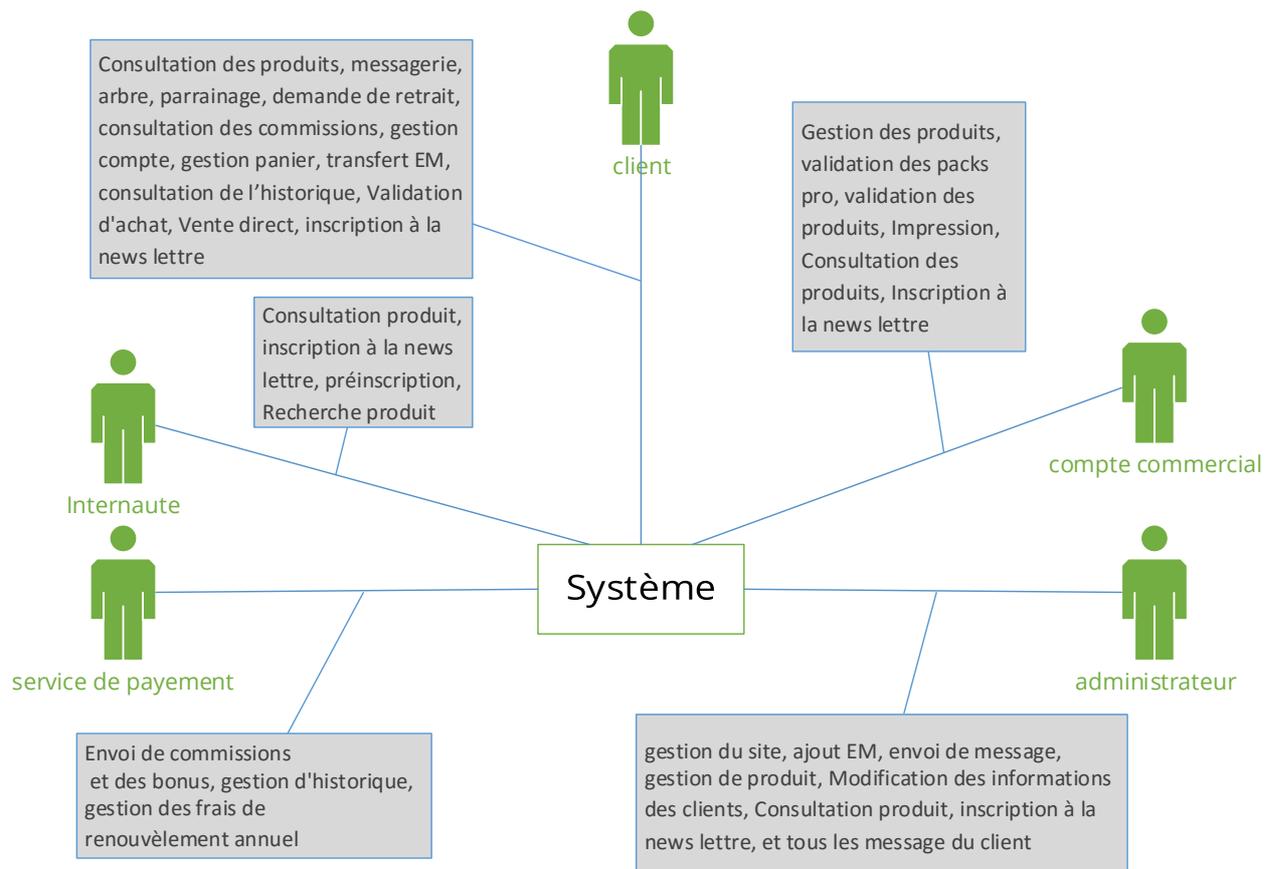


Figure 12 : Diagramme de contexte du système à réaliser.

II.2.4. Identification des cas d'utilisations

Un cas d'utilisation est un ensemble d'actions et de séquences qui définit la relation entre un système et ses acteurs. Pour notre application, nous avons identifié les cas d'utilisation suivants :

N°	Cas d'utilisations	Acteur	
01	Se connecter	Administrateur/ client/ compte commercial	
02	parrainage	Client/Administrateur	
03	Gestion de compte		Modifier les informations
			Modifier le mot de passe
			Modifier le code PIN
04	Consultation et demande de retrait		
05	Transfert EM		
06	Historique		
07	Gestion panier		Modifier le panier
			Valider l'achat
			Vente direct
08	Arbre		
09	Messagerie		
10	Préinscription	l'internaute	
11	S'inscrire aux newsletters	Administrateur/ client/ compte commercial/l'internaute	
12	Consulter recherche produits	Administrateur	
13	Gestion du site	Modifier les publicités	
		Modifier la page d'accueil	
		Modifier les valeurs des bonus mensuels	
		Gestion des comptes des utilisateurs (clients, compte commercial)	
		Rechercher des clients	
14	Ajout EM	Administrateur/ compte commercial	
15	Envoi des messages		
16	Gestion des produits		
17	Valider pack pro	Administrateur/ compte commercial	
18	Valider produits		
19	Impressions		
20	Gestion de l'historique	Envoi des notifications	Service de paiement
		Enregistrer les transactions	
21	Envoi des commissions et des bonus		
22	Gestion des frais de renouvellement annuels		

Tableau 12 : Identification des cas d'utilisations.

II.2.5. Diagramme des cas d'utilisation

C'est un formalisme permettant de modéliser le fonctionnement d'un système par un découpage en fonctionnalités. Il illustre de plus, la nature des interactions avec ces fonctionnalités offertes, à titre de services, à des acteurs externes au système. Chaque fonctionnalité est appelée un cas d'utilisation.

Dans ce qui suit, nous décrivons le diagramme détaillé des cas d'utilisation associés à chacun des acteurs du système à réaliser.

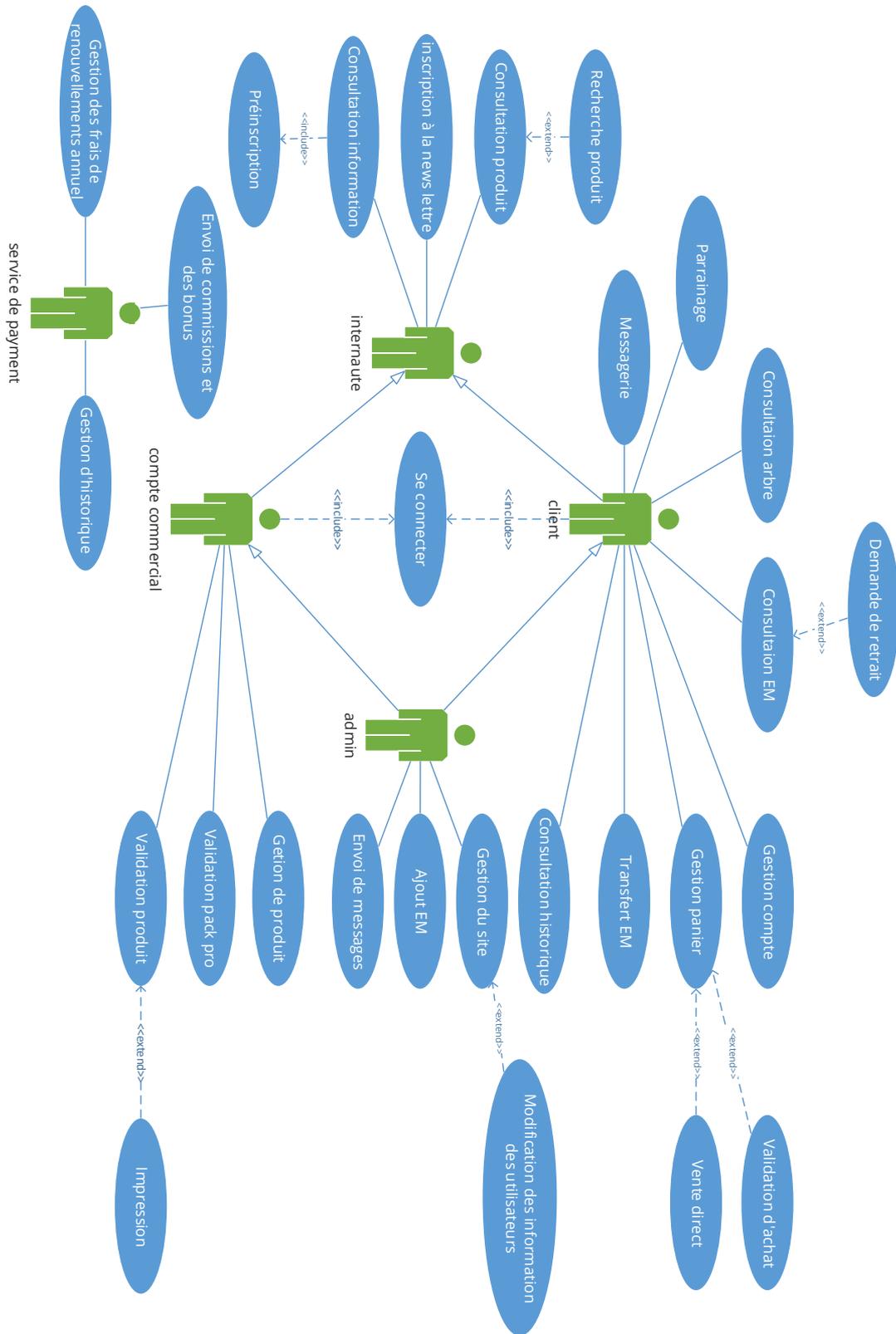


Figure 13 : Diagramme des cas d'utilisations.

Toutes les extensions figurant sur la figure précédente ont des conditions bien définies comme un point d'extension

Ce chapitre, nous a permis de bien comprendre les exigences de l'organisme d'accueil et de bien se familiariser avec le system de « New Life », il nous a permis également d'avoir une vision prématurée sur les acteurs qui interagissent avec le system, les relations entre eux et leurs cas d'utilisations, ces derniers seront plus détailler dans le chapitre suivant.

Chapitre III

Conception

Diagrammes d'états transitions

Diagrammes de séquence détaillés

Le diagramme de classe

Modèle logique des données

Dictionnaires de données

À travers ce chapitre, nous décrivons la conception choisie pour réaliser le travail demandé.

En premier lieu, nous illustrerons les différentes interactions entre les composants du système grâce aux diagrammes de séquence, par la suite, nous détaillerons la conception de notre application aux moyens de diagramme de classe et de modèle logique de données.

III.1. Modélisation dynamique

Dans cette partie, nous allons analyser le comportement de notre système ainsi que les événements et la communication entre les différents objets, afin d'analyser la partie dynamique de notre application web.

III.1.1. Les diagrammes d'états transitions

Pour simplifier la gestion des comptes clients de « New Life », nous les avons subdivisés en états transitions, cela nous permettra de connaître les droits de chaque client et de bien les gérer, afin de garantir un fonctionnement optimal.

Le diagramme d'état de transition de la « **figure 14** », fournit une description des différents Etats du compte client dans notre application web.

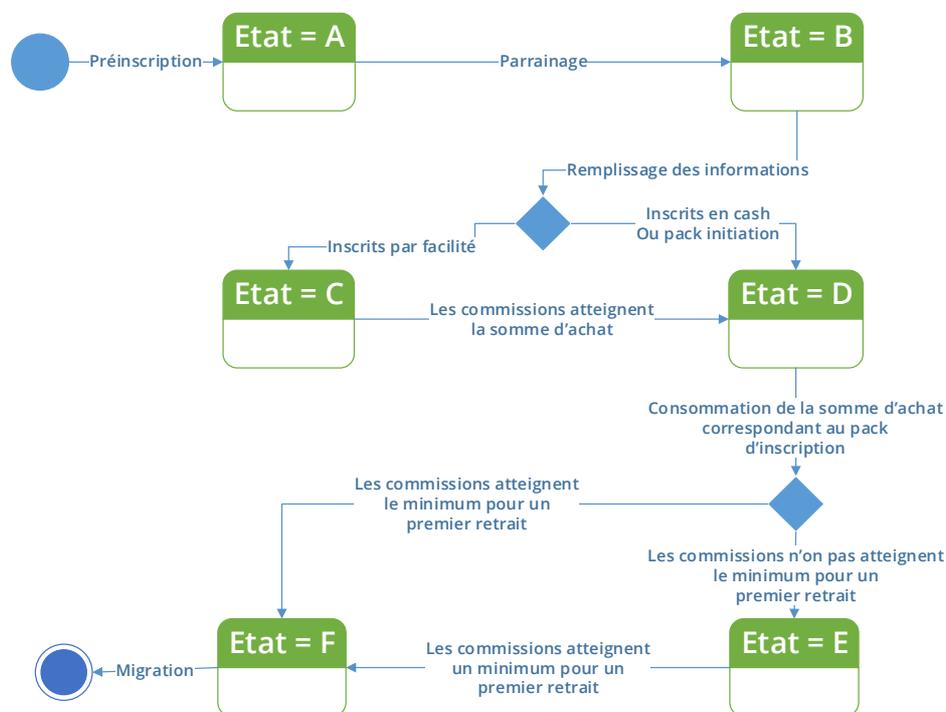


Figure 14 : Diagramme d'états/transitions des états du compte client.

Pour garder toujours une visibilité entre le gérant de « New Life » et ses clients, nous avons divisé le parcours des produits et les demandes de retrait en états transition.

Le diagramme d'état transition de la « **figure 15** », fournit une illustration du parcours des produits lors de l'ajout au panier jusqu'à sa livraison finale.

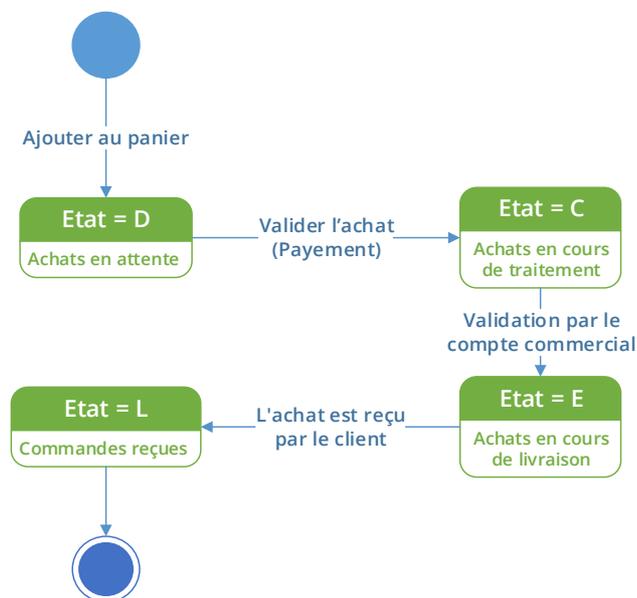


Figure 15 : Diagramme d'états de transitions du parcours des demandes de livraisons.

III.1.2. Les diagrammes de séquence détaillés

Cette partie permet d'illustrer les principaux diagrammes de séquence détaillés pour le point de vue dynamique.

Remarque :

Pour distinguer entre les différentes utilisations des EM, nous les avons subdivisés et renommés comme suit :

CO : les commissions d'équilibre.

CD : les commissions directes.

E-Bank : les EM que les utilisateurs peuvent transférer et payer les frais de parrainage.

FI : frais d'inscription.

FL : frais de livraison.

G-Bonus : la valeur des EM de G-Bonus.

Les diagrammes de séquences suivants, montrent une vue dynamique sur l'interaction de l'utilisateur, avec le système dans chacun des scénarios possibles, à travers notre travail.

III.1.2.1.Cas d'utilisation «Se connecter»

CAS D'UTILISATION N° 1 SE CONNECTER

RESUME	Vérification de l'identité des utilisateurs. Les classes (authentification, utilisateur et compte commercial) nous permettent de communiquer avec l'utilisateur (entrée/sortie). la classe contrôle est une classe que nous avons créé dans le but de gérer toutes les communications entre les objets de notre système, cette dernière diffère selon le cas d'utilisation utilisé et elle est de type « singleton » [14]. La classe utilisateur pour l'instant sert à stocker toutes les informations des utilisateurs.
ACTEURS	Administrateur, client, compte commercial
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> • L'utilisateur envoie une demande d'ouverture de session en spécifiant son login et mot de passe. • La demande va passer de l'interface authentification vers le contrôleur du cas d'utilisation « se connecter ». • Grâce à la classe utilisateur, on peut déterminer si les données (login et mot de passe) existent. <p>[SI (donnée n'existe pas)]</p> <ul style="list-style-type: none"> • Afficher un message d'erreur. <p>[SINON]</p> <ul style="list-style-type: none"> • Récupération de l'état de l'utilisateur et création d'une session. <p>[SI (Etat = W)]</p> <ul style="list-style-type: none"> • Rediriger le compte commercial vers son interface spécifique. <p>[SINON]</p> <p>[SI (Etat <> A)]</p> <ul style="list-style-type: none"> • Récupération de la liste de tous les fils de l'utilisateur. • mis à jour de la session. • Rediriger le compte client vers son interface spécifique. <p>[FINSI]</p> <p>[FINSI]</p> <p>[FINSI]</p> <p>[Fin]</p>

Tableau 13 : Description du cas d'utilisations « Se connecter ».

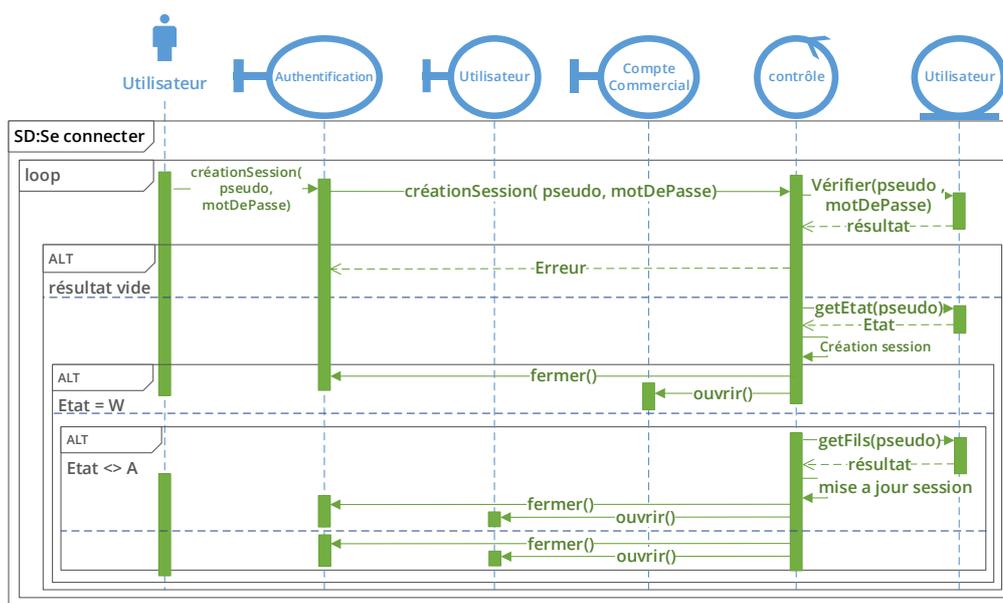


Figure 16 : Diagramme de séquence détaillé du cas d'utilisation «Se connecter».

III.1.2.2.Cas d'utilisation «payements»

CAS D'UTILISATION N° 2
PAYEMENTS

RESUME	C'est le cas le plus important de notre travail. il nous permet l'envoi des commissions d'équilibre par le service de payement, qui se fait chaque dimanche à cinq heures du matin. l'absence des classes « interface » et « contrôle » pour la raison que les données sont déjà vérifiées. Et l'acteur, est le système lui-même, donc il communique directement avec les classes persistantes. La classe compte contient toutes les informations relative au EM des clients. La classe historique nous permet de sauvegarder les transactions de payement.
ACTEURS	service de payement.
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> • Ce qui suit, se répète pour tous les utilisateurs qui disposent d'une ligne, dans la classe compte. • Nous récupérons les variables que nous avons besoin pour le payement. Nous récupérons les points gauches, les points droits, le restDiv (variable qui nous permet de savoir combien il reste de commission d'équilibre pour envoyer un point G-Bonus), le gain général du client pour gérer le premier pack d'inscription (initiation), firstPay qui nous évitera de payer les utilisateurs qui sont parrainés moins d'une semaine. <p>[SI (firstPay = faux)]</p> <ul style="list-style-type: none"> • Nous récupérons le pack d'inscription et l'état transition du compte client.

- Nous calculons ce que nous allons rajouter au compte client. Les commissions dans la variable CO, les point G-bonus dans GB et le nouveau restDiv dans NRestDiv.
[SI (pack = 'N1')]
[SI (CO + gainGénéral >= 660)]
- Si après avoir ajouté les commissions, le client dépassera le gain général autorisé, dans ce cas nous rajoutons seulement la somme manquante.
- Nous calculons et nous mettons à jours la classe compte.
- Nous ajoutons la transaction à l'historique.
[SINON]
- Nous ajoutons directement la somme gagnée sans avoir la calculer.
- Nous ajoutons la transaction à l'historique.
[FINSI]
[SINON]
- Pour le calcul des autres niveaux nous avons besoin du max out du pack d'inscription. Donc nous le récupérons dans la classe utilisateur.
[SI (GB >= maxOut div 10 div 4)] // maxOut on G-Bonus.
- Nous supprimons le reste des gains et nous rajoutons juste le max out lui-même.
- Nous ajoutons la transaction à l'historique.
[SINON]
- Nous rajoutons les points G-Bonus gagnés et nous mettons à jours le restDiv.
- Nous ajoutons la transaction à l'historique.
[FINSI]
[SI (CO >= maxOut)]
- Nous supprimons le reste des gains et nous rajoutons juste le max out lui-même.
- Nous ajoutons la transaction à l'historique.
[SINON]
- Nous rajoutons les commissions d'équilibre gagnées et ont met à jours le restDiv.
- Nous ajoutons la transaction à l'historique.
[FINSI] [FINSI]
- Maintenant, après avoir mis à jour la classe compte nous vérifions l'état de transition du compte, si ce dernier mérite d'aller vers l'état suivant. Pour cela, nous avons besoin des commissions du client et de la valeur d'achat correspondante à son pack d'inscription.
[SI (etat = C et commission >= valeur pack achat)]
- Nous modifions son état à D.
[FINSI]
[SI (etat = E et (pack <> N1 et commission >= 300)ou (pack = N1 et commission >= 1))]
- Nous modifions son état à F.
[FINSI] [SINON]
- Nous mettons à jour la variable firstPay pour que la prochaine fois le compte sera pris en considération.
[FINSI] [Fin]

Tableau 14 : Description du cas d'utilisations «service de payement ».

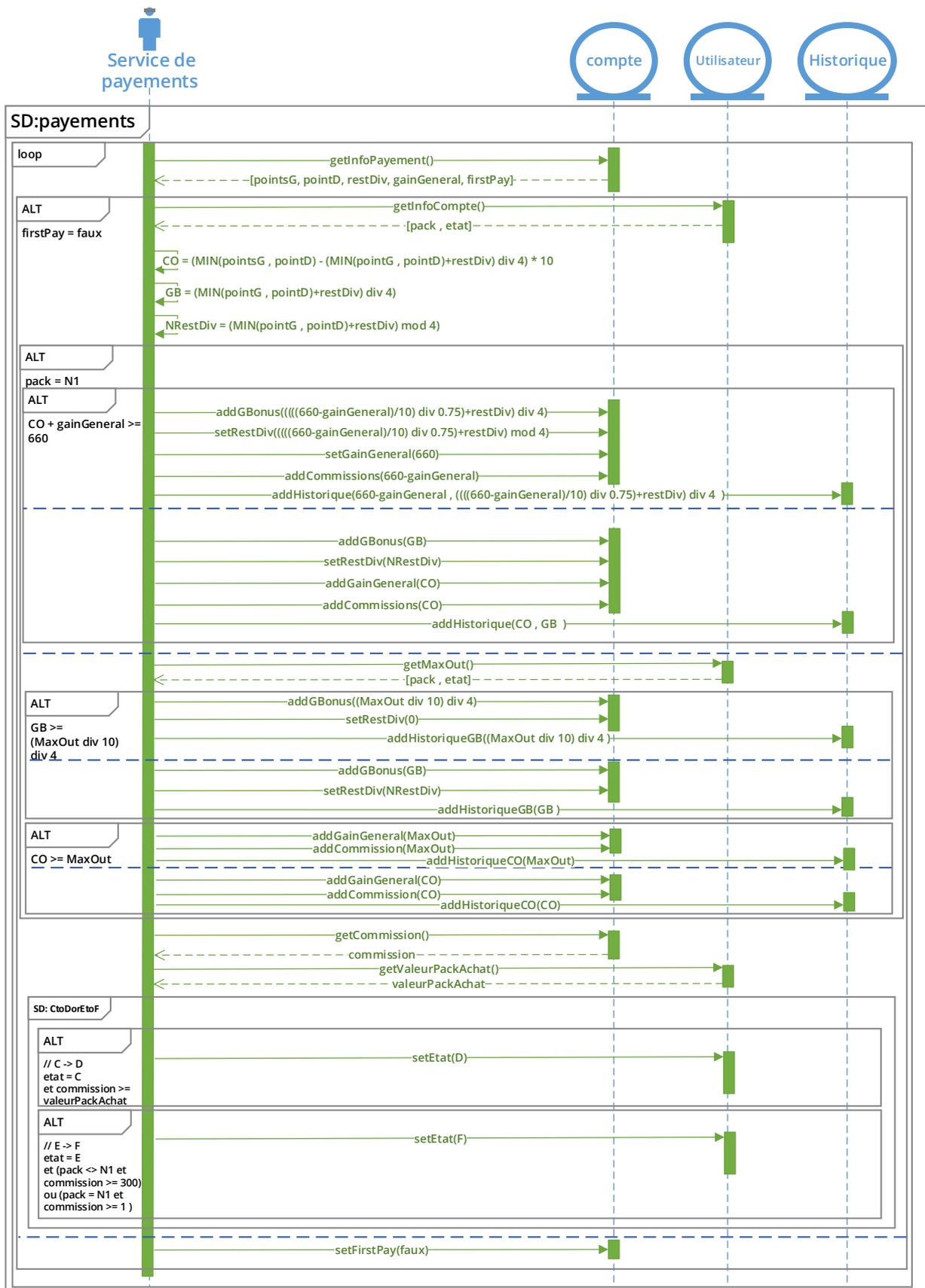


Figure 17 : Diagramme de séquence détaillé du cas d'utilisation « service de paiements».

III.1.2.3. Cas d'utilisation «Consulter les informations»

CAS D'UTILISATION N° 3
CONSULTER LES INFORMATIONS

RESUME	Un cas d'utilisation simple ou un utilisateur consulte ses informations
ACTEURS	Administrateur, client, compte commercial
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> Le cas d'utilisation «se connecter » est ré-exécuté. Un utilisateur demande de consulter ses informations. <p>[SI (type = 1)]</p> <ul style="list-style-type: none"> Nous récupérons les informations sur les EM du client. <p>[FINSI]</p> <ul style="list-style-type: none"> Nous récupérons les informations de l'utilisateur dans la classe utilisateur. <p>[Fin]</p>

Tableau 15 : Description du cas d'utilisation «Consulter les informations».

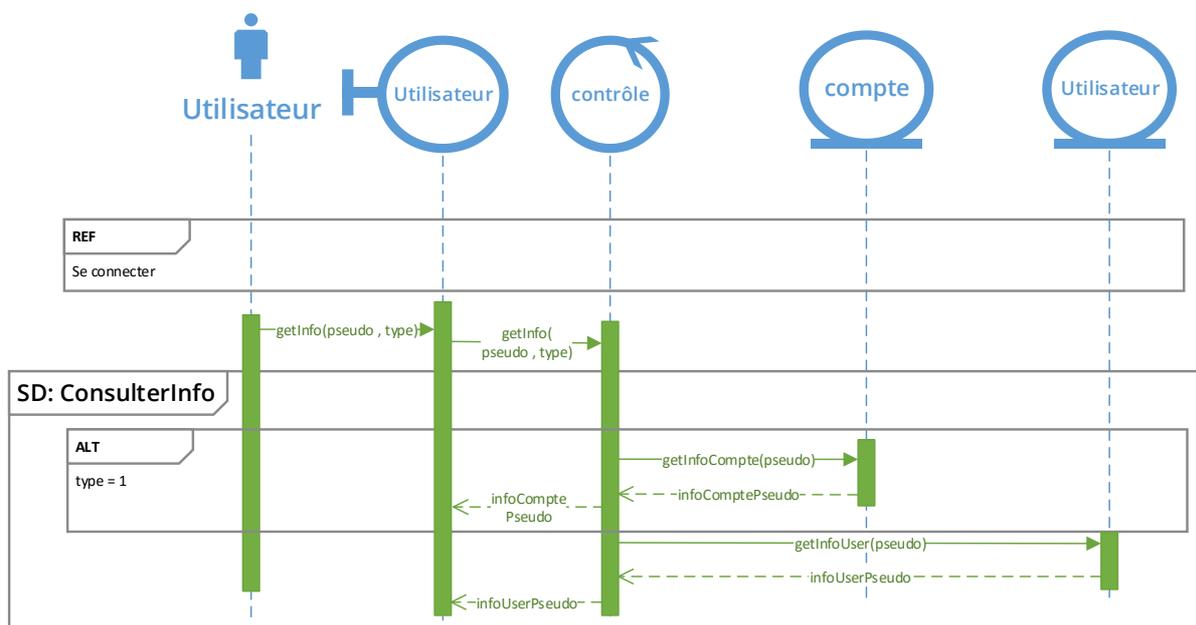


Figure 18 : Diagramme de séquence détaillé du cas d'utilisation « Consulter les informations».

III.1.2.4.Cas d'utilisation «Parrainage»

**CAS
D'UTILISATION
N° 4**

PARRAINAGE

RESUME	Le cas d'utilisation parrainage montre comment le système réagit lorsqu'un nouveau client est rajouté dans l'arbre.
ACTEURS	Administrateur, client
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> • Le cas d'utilisation «se connecter » est ré-exécuté. • L'utilisateur introduit le pseudo du parrain direct (bénéficiaire de la commission directe). • On récupère les informations de l'utilisateur grâce au cas getInfo. <p>[SI (Etat = A)]</p> <ul style="list-style-type: none"> • Nous affichons une erreur. <p>[SINON]</p> <ul style="list-style-type: none"> • Les informations du pseudo direct sont affichées dynamiquement ainsi qu'une zone saisissable. • L'utilisateur introduit le pseudo « pseudoP » du parrain du nouveau client (l'endroit où mettre le nouveau client). • Le serveur vérifie que le pseudo est un fils de l'utilisateur <p>[SI (pseudoP n'est pas inclut dans les fils de l'utilisateur)]</p> <ul style="list-style-type: none"> • Un message d'erreur est affiché dynamiquement sur la page. <p>[SINON]</p> <ul style="list-style-type: none"> • Nous récupérons les informations de l'utilisateur grâce au cas getInfo. <p>[SI (pseudoP contient deux fils au premier niveau)]</p> <ul style="list-style-type: none"> • Nous affichons une erreur. <p>[SINON]</p> <ul style="list-style-type: none"> • Affichage des informations sur le pseudo du parrain et ses fils (nom, prénom), une zone saisissable est affichée. • L'utilisateur introduit le pseudo du nouveau client « pseudoN ». • Nous récupérons les informations de l'utilisateur grâce au cas getInfo. <p>[SI (Etat <> A)]</p> <ul style="list-style-type: none"> • Nous affichons une erreur. <p>[SINON]</p> <ul style="list-style-type: none"> • Nous affichons un arbre avec le nouveau client ainsi que des informations concernant les frais de parrainage (E-Bank, FI), une zone saisissable est affichée. • L'utilisateur introduit son code PIN et confirme le paiement, donc un message est envoyé avec toutes les données précédentes. • Nous récupérons les informations de l'utilisateur connecté grâce au cas getInfo.

```
[SI ((E-Bank ou FI) sont Insuffisants) ou PIN
incorrect ou données incorrectes)]
• Nous affichons une erreur.
  [SINON]
• Nous retirons les frais de parrainage dans le compte de l'utilisateur
connecté
• Nous ajoutons les commissions directes dans le compte du pseudoD
(bénéficiaire de la commission directe).
• Nous demandons de mettre à jour les points d'équilibre de tous les
prédécesseurs du nouveau client.
• Nous ajoutons le nouveau client à l'arbre, et nous mettons à jour
son état de transition, et nous ajoutons les frais d'investissement
dans son compte CO.
• Nous affichons un message de confirmation et nous redirigeons
l'utilisateur vers son arbre pour voir le résultat final du parrainage.
  [FINSI]
[FINSI]

[FINSI]
[FINSI]
[FINSI]
[Fin]
```

Tableau 16 : Description du cas d'utilisation «Parrainage ».

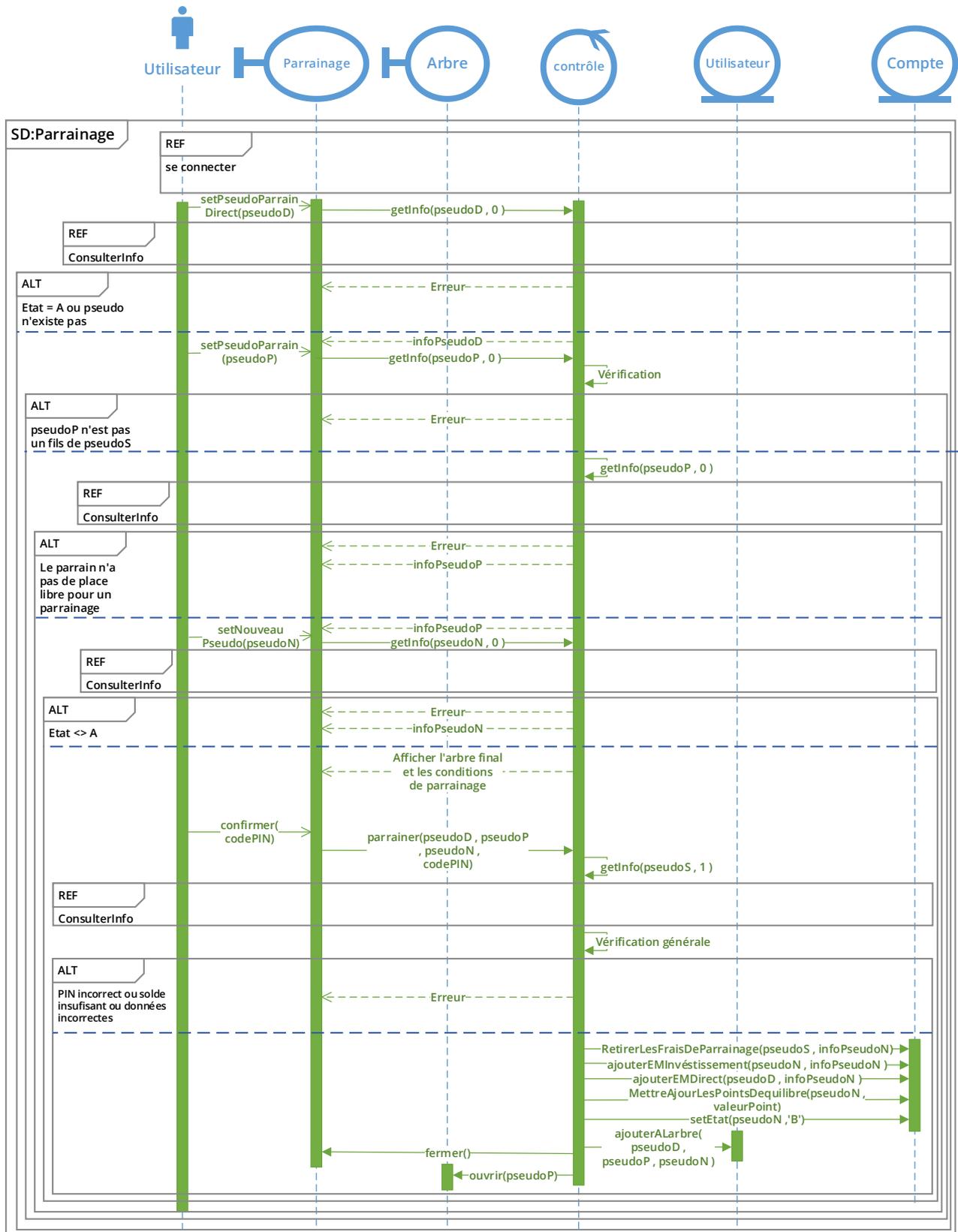


Figure 19 : Diagramme de séquence détaillé du cas d'utilisation « Parrainage ».

III.1.2.5.Cas d'utilisation «Transfert EM»

**CAS
D'UTILISATION
N° 5**

TRANSFERT EM

RESUME	Ce cas nous permet de décrire comment les transferts sont gérés par notre système. La classe transfert EM contient des informations sur l'historique des transferts.
ACTEURS	Administrateur, client
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> Le cas d'utilisation «se connecter » est ré-exécuté. L'utilisateur introduit le pseudo du destinataire ensuite un message asynchrone est envoyé automatiquement au serveur. On récupère les informations du destinataire grâce au cas getInfo. <p>[SI (pseudoD.etat = A)]</p> <ul style="list-style-type: none"> Un message d'erreur est affiché dynamiquement sur la page. <p>[SINON]</p> <ul style="list-style-type: none"> Le client demande le transfert en saisissant le montant à envoyer, le type d'EM à envoyer et le code PIN du client connecté. <p>[SI (données incorrect)]</p> <ul style="list-style-type: none"> Un message d'erreur est affiché dynamiquement sur la page. <p>[SINON]</p> <ul style="list-style-type: none"> Nous récupérons les informations du destinataire grâce au cas getInfo. <p>[SI (PIN incorrect ou solde insuffisant)]</p> <ul style="list-style-type: none"> Un message d'erreur est envoyé vers le client. <p>[SINON]</p> <ul style="list-style-type: none"> Mis à jours des nouveaux soldes dans la classe compte. Nous ajoutons la transaction dans la classe transfert EM. <p>[FINSI]</p> <p>[SI (type = transfert de commissions directes vers les commissions d'équilibre ou type transfert des commissions par l'administrateur)]</p> <ul style="list-style-type: none"> Maintenant après avoir mis à jour la classe compte (les commissions plus précisément) nous vérifions l'état de transition du compte, si le compte mérite d'aller vers l'état suivant. Pour cela nous avons besoin des commissions du client et la valeur d'achat correspondante à son pack d'inscription. Nous ré-exécutons le cas « CtoDorEtoF » <p>[FINSI]</p> <p>[FINSI]</p> <p>[Fin]</p>

Tableau 17 : Description du cas d'utilisations «Transfert EM ».

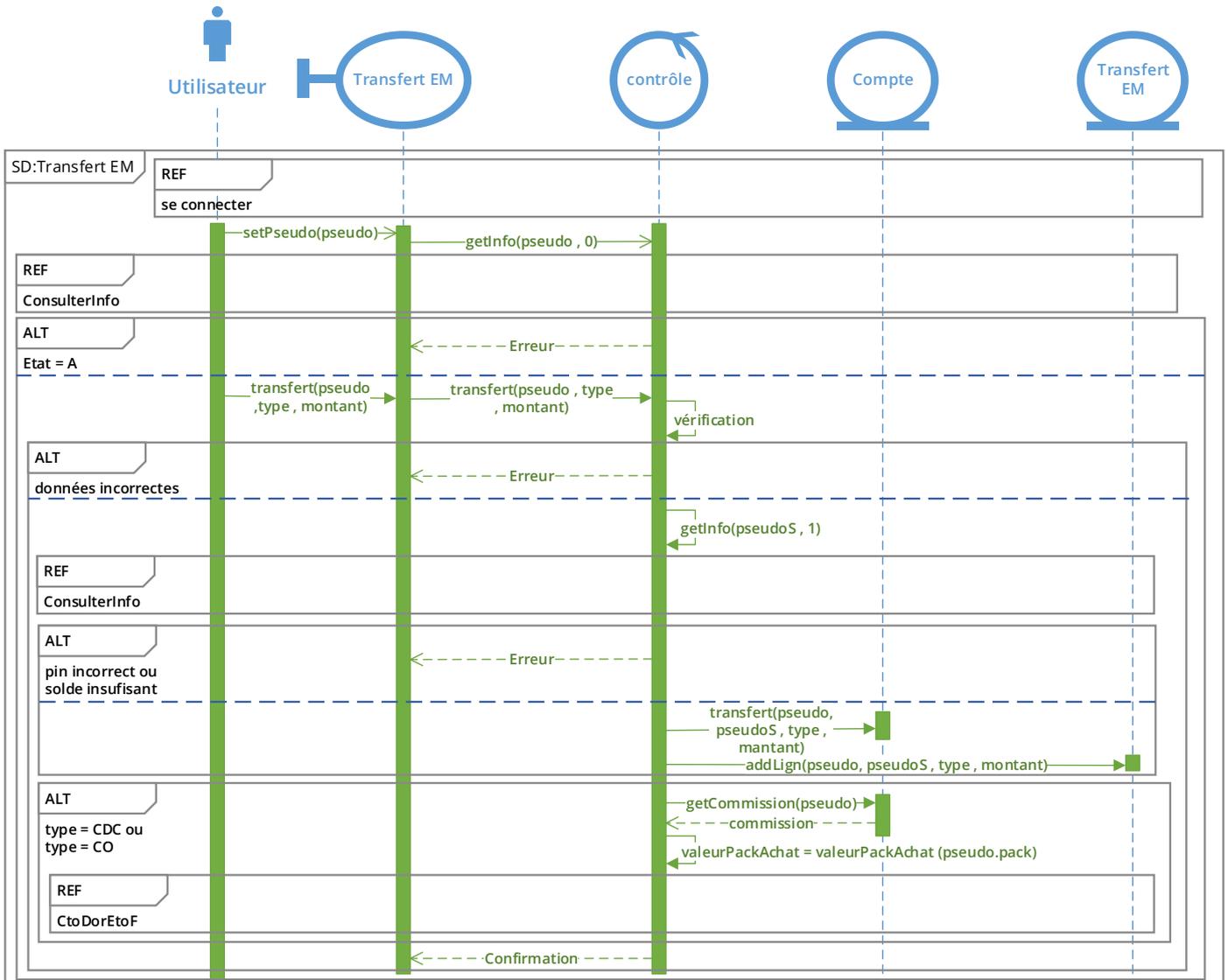


Figure 20 : Diagramme de séquence détaillé du cas d'utilisation «Transfert EM ».

III.1.2.6.Cas d'utilisation «Gestion panier»

**CAS
D'UTILISATION
N° 6**

GESTION PANIER

RESUME	Le cas qui nous permet de gérer le panier des clients, la classe produit contient toutes les informations à propos des produits, la classe commande contient les commandes de livraison, la classe panier contient les paniers des utilisateurs.
ACTEURS	Administrateur, client
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> • Le client demande l'affichage des produits disponibles. il peut ajouter des contraintes « rech » s'il désire faire une recherche. • Nous récupérons les informations des produits dans la classe produits et nous les affichons pour le client. • Le client choisit un produit et clique pour demander des informations à propos d'un produit spécifique. • Nous fermons l'interface liste produit pour ouvrir l'interface produit. • Le cas d'utilisation «se connecter » est ré-exécuté. • Maintenant le client peut ajouter ce produit dans son panier. en demandant l'ajout au panier à la classe produit. • En utilisant le contrôleur, nous recherchons l'identifiant du panier de l'utilisateur et ses informations de livraison dans les classes panier et utilisateur. • Nous affichons les informations de livraison pour l'utilisateur, et là, il peut soit confirmer l'envoi à ses propres coordonnées ou donne des coordonnées d'un destinataire à part. • Nous ajoutons une ligne de commande pour le client. Avec l'identifiant du panier et l'identifiant du produit. • Nous redirigeons le client directement vers son panier et nous lui affichons les produits qu'il a commandés avec leurs états de transitions. • Bien évidemment, le client peut supprimer à tout moment une ligne de commande avec l'état D <p>[Fin]</p>

Tableau 18 : Description du cas d'utilisations «Gestion panier ».

III.1.2.7.Cas d'utilisation «Validation d'achat »

CAS D'UTILISATION VALIDATION D'ACHAT

N° 7

RESUME	La validation d'achat permet au client de demander la livraison d'un produit, contrairement à la vente directe la validation d'achat permet aussi de modifier l'état du compte client.
ACTEURS	Administrateur, client
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> • Le cas d'utilisation «se connecter » est ré-exécuté. • La référence « afficher panier » dans le cas d'utilisation « gestion panier » est exécutée. • Le client demande la validation de l'achat en envoyant l'identifiant de la ligne de commande. • La classe contrôle vérifie les droits du client connecté. <p>[SI ((Etat = A ou Etat = B ou Etat = C) ou jour = vendredi ou jour = samedi)]</p> <ul style="list-style-type: none"> • Un message d'erreur est affiché dynamiquement sur la page. <p>[SINON]</p> <ul style="list-style-type: none"> • Nous vérifions que le client dispose des EM suffisants pour l'achat, donc nous allons prendre le prix du produit et ses frais de livraison dans la classe produit et les commissions de l'utilisateur et ses FL dans la classe compte. <p>[SI (CO insuffisant ou FL insuffisant)]</p> <ul style="list-style-type: none"> • Un message d'erreur est affiché dynamiquement sur la page. <p>[SINON]</p> <ul style="list-style-type: none"> • Nous changeons l'état de la ligne de commande en C pour l'envoyer au compte commercial. • Nous retirons les commissions et les FL dans le compte du client. • Nous mettons à jour le panier du client. • Pour vérifier que le client a acheté la somme correspondante à son pack d'inscription, nous récupérons les données dans la classe utilisateur. <p>[SI (etat = D)]</p> <ul style="list-style-type: none"> • On récupère le total des sommes que le client a acheté. <p>[SI (valeur produit acheter >= valeur produit requise)]</p> <p>[SI ((pack <> N1 et commission >= 300) ou (pack = N1 et commission >= 1))]</p> <ul style="list-style-type: none"> • Nous changeons l'état du compte du client directement à l'état F. <p>[SINON]</p> <ul style="list-style-type: none"> • Nous changeons l'état du compte du client à l'état E. <p>[FINSI]</p> <p>[FINSI]</p> <p>[FINSI]</p> <p>[FINSI]</p> <p>[FINSI] [Fin]</p>

Tableau 19 : Description du cas d'utilisation «Validation d'achat ».

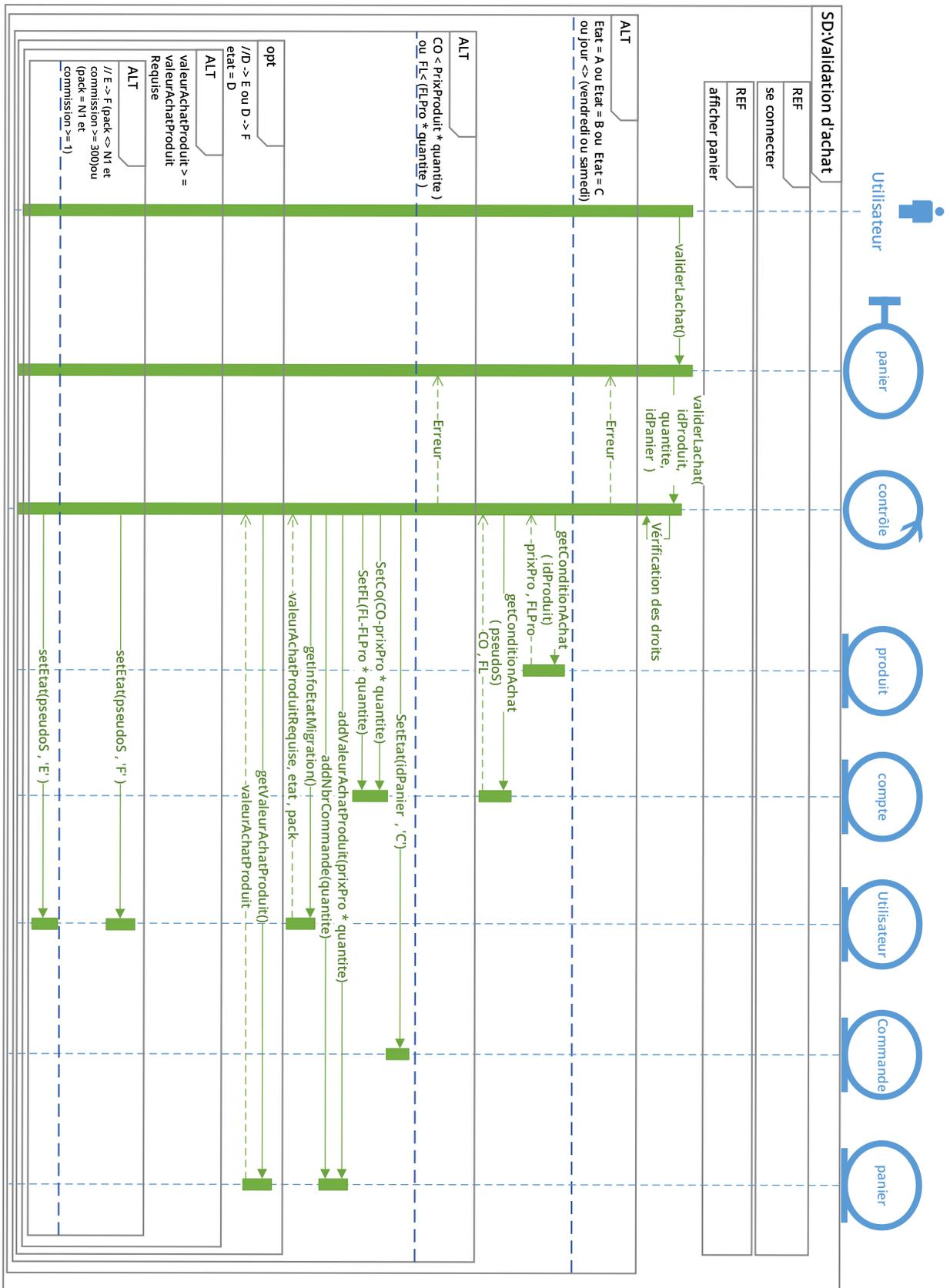


Figure 22 : Diagramme de séquence détaillé du cas d'utilisation «Validation d'achat».

III.1.2.8. Cas d'utilisation « validation produit »

CAS D'UTILISATION N° 8 VALIDATION PRODUIT, IMPRESSION

RESUME	Après validation de l'achat par le client, le compte commercial va recevoir les commandes et valide leur livraison.
ACTEURS	Compte commercial
SCENARIO NOMINAL	<p>[Début]</p> <ul style="list-style-type: none"> Le compte commercial envoie une demande de la liste des commandes de livraison de produits. La classe contrôle va chercher toutes les lignes de commande avec un état C et les affiche pour le compte commercial. Le compte commercial peut changer l'état du produit en envoyant un message contenant l'identifiant de la ligne de commande. Le compte commercial peut demander une version imprimable des étiquettes à coller sur les produits. Nous devons récupérer les informations de l'utilisateur en question et les informations du produit, et cela pour chaque ligne de commande. <p>[Fin]</p>

Tableau 20 : Description du cas d'utilisation «Validation produit ».

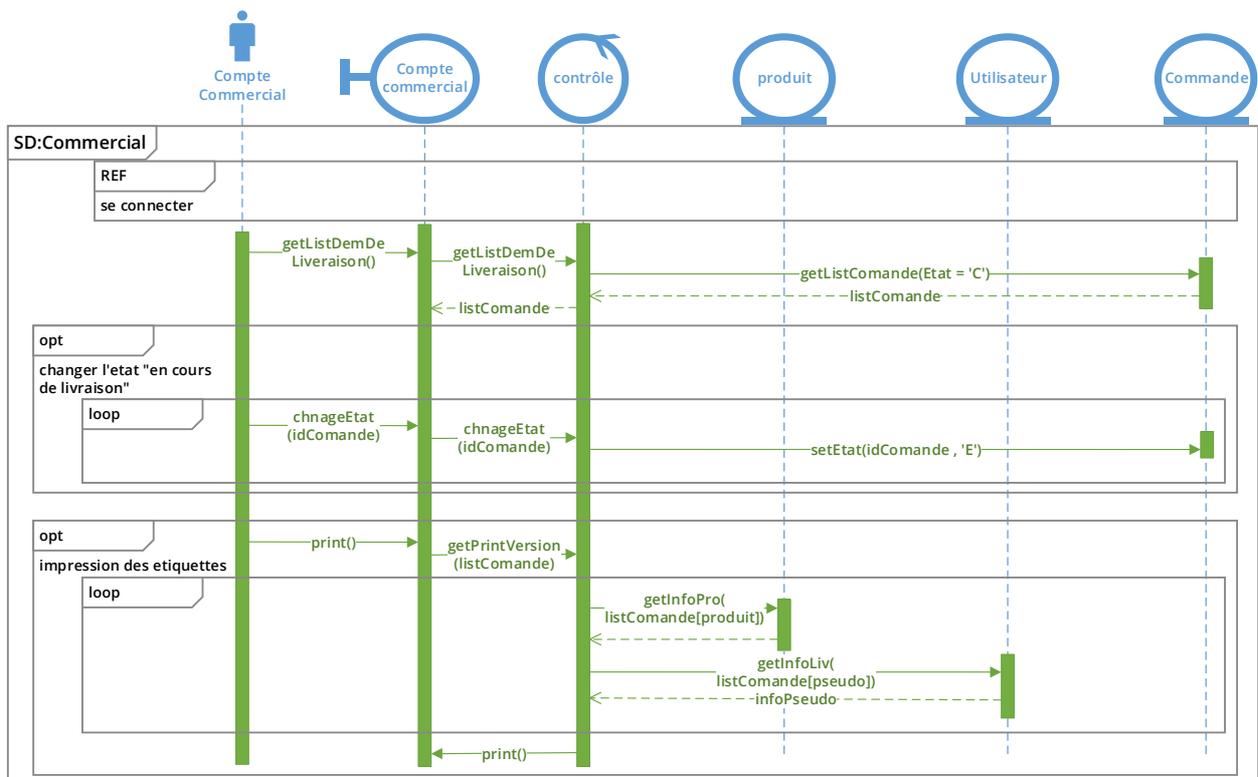


Figure 23 : Diagramme de séquence détaillé du cas d'utilisation «validation produit ».

III.2. Modélisation statique

Pour modéliser la partie statique du système, la première étape est la réalisation du diagramme de classe représenté par la « **figure 24** », suivi par le dictionnaire de données et le modèle logique de données déduit à partir du diagramme de classe conçu.

III.2.1. Le diagramme de classe

Le diagramme de classe exprime d'une manière générale la structure statique d'un système, en termes de classes et de relations entre elles.

Le diagramme de classe de la « **figure 24** » fournit une description bien détaillée des différentes classes de l'application ainsi que les différentes associations qui les relient.

III.2.2. Modèle logique de données

La description conceptuelle a permis de représenter le plus fidèlement possible les réalités de l'univers informatisé. Mais cette représentation ne peut pas être directement manipulée et acceptée par un système informatique, car le diagramme de classe est une représentation des données dans un formalisme compris par les concepteurs et pas par la machine. Il est donc nécessaire de passer du niveau conceptuel à second un niveau plus proche des capacités des systèmes informatiques.

L'application des règles de passage (Voir ANNEXE 2), nous permet d'avoir le modèle logique de données associé au diagramme de classe de la « **figure 24** » :

Utilisateur (Dz, typeDePym, pack, ccp, ccpkey, nomBanque, RIP, Pseudo#, idPers#, DzGauche#, DzDroit#, DzDirect#) ;

CompteCommercial (pseudo, motDePasse, mail, etat, codePIN) ;

ClientPackPro (idCPPro, etat, dateIns, DZ#) ;

Personne (idPers, nom, prenom, adresse, pays, wilaya, sexe, tel, zip, tel2, carte, etatH, DZ#) ;

Compte (idCompte, pointsG, pointsD, commissions, G-bonus, EBank, FraisIns, fraisLiv, resteDiv, MoisDirect, MoisGroupe, gainGeneral, comDirect, firstPay, DZ#) ;

Produits (idProd, nom, prix, prixMembre, stock, details, ProdFav, marque, image1, image2, image3, image4, fraisLivProd, prodGBonus, bonusEnGBonus, idCat#) ;

CategorieProduit (idCat, nomCategorie) ;

Panier (idPnr, TotalDAchat, nbrDeCommande, DZ#) ;

LigneCommande (idLC, date, quantité, etat, idPnr#, idProd#, idPers#) ;

Retrait (idRet, somme, type, date, vu, dateR, DZ#) ;

TransfertEM (idTrns, dateT, type, somme, vu, DZ#, DzDestinataire#) ;

Message (idMsg, message, date, vu, type) ;

Historique (idHis, type, Ems, vu, dateEnv, DZ#) ;

Recevoir (idMsg#, DZ#) ;

Remarque :

Pour faciliter l'implémentation et la rapidité du système nous avons créé une table secondaire appelée « parrainage » pour optimiser le system, elle contient les informations de parrainage « DZGauche », « DZDroit », « DZDirect » de la table utilisateur et un autre attribut nommé « upLine » pour faciliter le parcours des prédécesseurs.

Parrainage (DZ#, DZGauche#, DZDroit #, DZDirect #, DZUpLine#) ;

III.2.3. Dictionnaire de données

Le dictionnaire de données répertorie tous les termes et leurs définitions.

La table « CompteCommercial », nous permet de stocker les informations techniques relatives à l'identification, ainsi que l'état du compte.

	Définition de l'attribut	Attribut	Type
Compte Commercial	L'identifiant de la table compte commercial	pseudo	Varchar(30)
	Le mot de passe haché avec un algorithme personnalisé	motDePasse	Varchar(50)
	E-mail du client	mail	Varchar(326)
	L'état de la transition du client	etat	Varchar(1)
	Le code PIN du client	codePIN	Varchar(6)

Tableau 21 : Le glossaire de la table « CompteCommercial ».

En plus de la table « CompteCommercial », La table « Utilisateur » nous permet de spécialiser le client parrainé et d'enregistrer ses informations de retrait.

	Définition de l'attribut	Attribut	Type
Utilisateur	L'identifiant de l'utilisateur (le DZ des clients de « New Life »)	DZ	Varchar(15)
	La méthode de paiement du client	typeDePym	Varchar(1)
	Le pack d'inscription du client	Pack	Varchar(2)
	Le numéro du compte ccp du client	ccp	Int(20)
	La clé du compte ccp du client	Ccpkey	Int(2)
	Le nom de la banque du client	nomBanque	Varchar(20)
	Le RIP du compte du client	RIP	Varchar(20)

Tableau 22 : Le glossaire de la table « Utilisateur ».

La table « ClientPackPro », permet de spécialiser les clients, pour que le compte commercial gère les demandes de livraisons des packs pro des nouveaux clients parrainés.

	Définition de l'attribut	Attribut	Type
ClientPackPro	L'identifiant de la classe clientPackPro	idCPro	Int(10)
	L'état de la transition de la livraison du pack pro des nouveaux clients	Etat	Varchar(1)
	La date de préinscription et mise à jour pour la date de validation du client	dateIns	DateTime

Tableau 23 : Le glossaire de la table « ClientPackPro».

La table « Personne », nous permet de sauvegarder les informations relatives à l'utilisateur, l'héritier, le destinataire du produit.

	Définition de l'attribut	Attribut	Type
Personne	L'identifiant de la classe personne	idPers	Int(10)
	Nom du client héritier ou destinataire	nom	Varchar(30)
	Prénom du client héritier ou destinataire	prenom	Varchar(30)
	L'adresse exacte du client héritier ou destinataire	adresse	Varchar(300)
	Le pays du client héritier ou destinataire	pays	Varchar(2)
	La wilaya du client héritier ou destinataire	wilaya	Int(2)
	Le numéro de téléphone du client héritier ou destinataire	tel	Int(10)
	Le code postal du client héritier ou destinataire	zip	Varchar(6)
	Le deuxième numéro de téléphone du client héritier ou destinataire	tel2	Int(10)
	Le numéro de la carte d'identité nationale du client héritier ou destinataire	carte	Varchar(15)
	Le sexe du client héritier ou destinataire	sexe	Varchar(1)
	L'état où habite le client héritier ou destinataire en dehors de l'Algérie	etatH	Varchar(30)

Tableau 24 : Le glossaire de la table « Personne».

La table « Compte », nous permet de sauvegarder les informations relatives au contenu du compte « EM » du client (commission, point gauche, point droit ... etc.).

	Définition de l'attribut	Attribut	Type
Compte	L'identifiant de la classe Compte	idCompte	Int(6)
	Le nombre des points disponibles sur le côté gauche du client	pointsG	Int(6)
	Le nombre des points disponibles sur le côté droit du client	pointsD	Int(6)

	La somme des commissions disponibles sur le compte du client	commissions	Int(9)
	La somme des G-Bonus disponibles sur le compte du client	G-Bonus	Int(9)
	La somme des E-Bank disponibles sur le compte du client	EBank	Int(11)
	La somme des FI disponibles sur le compte du client	FraisIns	Int(9)
	La somme des FL disponibles sur le compte du client	FraisLiv	Int(9)
	Une variable primordiale pour le calcul des G-Bonus (valeur de 1 à 4)	resteDiv	Int(1)
	Le nombre d'inscrits direct du client du mois courant	moisDirect	Int(3)
	Le nombre d'inscrits du groupe du client du mois courant	moisGroupe	Int(5)
	La somme des commissions directes du client	comDirect	Int(9)
	La somme de gain général du client	gainGeneral	bigint(14)
	Variable utilisée pour résoudre une faille du paiement du dimanche matin	firstPay	Booléen

Tableau 25 : Le glossaire de la table « Compte ».

La table « Panier » est un ensemble de produits, elle nous permet d'avoir le total d'achat d'un client.

	Définition de l'attribut	Attribut	Type
Panier	L'identifiant du panier	idPnr	Int(11)
	La somme totale des produits achetés par un client	totalDAchat	Int(9)
	Nombres de commandes demandées par l'utilisateur	NbrDe Commande	Int(2)

Tableau 26 : Le glossaire de la table « Panier ».

La table « Produits », contient les informations des produits.

	Définition de l'attribut	Attribut	Type
produits	L'identifiant de la classe produit	idProd	Int(10)
	Le nom ou le titre du produit	nom	Varchar(120)
	Le prix du produit	prix	Int(6)
	Le prix dédié au client inscrit à « New Life »	prixMember	Int(6)
	Le stock du produit	Stock	Int(3)
	Les détails concernant le produit	details	Texte
	Produits qui s'affichent sur la page d'accueil ou non	ProdFav	Booléen

	La marque du produit	marque	Varchar(120)
	L'image qui s'affiche la première du produit	image1	Varchar(100)
	La deuxième image du produit	image2	Varchar(100)
	La troisième image du produit	image3	Varchar(100)
	La quatrième image du produit	image4	Varchar(100)
	La valeur des frais de livraison du produit fixé pour toutes les wilayas	fraisLivProd	Int(5)
	Produit G-Bonus ou pas	prodGBonus	Booléen
	Valeur des bonus d'achat en G-Bonus	bonusEnGBonus	Int(5)

Tableau 27 : Le glossaire de la table « Produits ».

La table « CatégorieProduits », contient les types des produits.

	Définition de l'attribut	Attribut	Type
CategorieProduits	L'identifiant de la catégorie	idCat	Int(11)
	Le nom ou titre de la catégorie	nomCategorie	Varchar(150)

Tableau 28 : Le glossaire de la table « CategorieProduits ».

La table « LigneCommande », contient toutes les informations relatives à la ligne de commande d'un produits ainsi les demandes de livraison des produits.

	Définition de l'attribut	Attribut	Type
LigneCommande	L'identifiant du panier	idLc	Int(11)
	La date d'ajout au panier	date	DateTime
	La quantité du même produit	Quantité	Int(2)
	Etat de transition du panier (la commande)	etat	Varchar(1)

Tableau 29 : Le glossaire de la table « LigneCommande ».

La table « TransfertEM », permet de garder une trace de toutes les transactions des transferts « EM ».

	Définition de l'attribut	Attribut	Type
TransfertEM	L'identifiant de la transaction	idTrns	Int(11)
	La date de la transaction	DateT	DateTime
	Le type de la transaction (FI, FL, E-Bank, CO)	Type	Varchar(1)
	La somme des EM transférés	Somme	Int(11)
	Variable pour déterminer la visibilité de la transaction par le client	Vu	Booléen

Tableau 30 : Le glossaire de la table « Transfert EM ».

La table « Message », nous permet la communication entre l'administrateur et ses clients.

	Définition de l'attribut	Attribut	Type
Message	Identifiant du message	idMsg	Int(11)
	Le message à envoyer	message	Text
	La date d'envoi	date	DateTime
	Variable pour savoir que le client a vu le message	vu	Booléen
	Le type de message (message pour tous, info exclusive, simple message)	type	Varchar(1)

Tableau 31 : Le glossaire de la table « Message ».

La table « Retrait », permet à l'administrateur de recevoir les demandes de retraits.

	Définition de l'attribut	Attribut	Type
Retrait	L'identifiant de la demande de retrait	idRet	Int(10)
	La somme de la demande de retrait	somme	Int(10)
	Type de demande (CCP, BANQUE ...)	type	Varchar(1)
	La date de la demande	date	DateTime
	Variable pour savoir que l'administrateur a vu la demande	vu	Booléen
	La date de la réaction de l'administrateur	dateR	DateTime

Tableau 32 : Le glossaire de la table « Retrait ».

La table « Historique », nous permet de garder une trace des évènements de paiement, bonus mensuel et G-Bonus.

	Définition de l'attribut	Attribut	Type
Historique	L'identifiant de l'évènement	idHis	Int(11)
	Le type d'évènement	type	Varchar(1)
	La somme de « EM » géré par l'évènement	EMs	Int(11)
	Variable pour savoir que le client a vu l'évènement	vu	Booléen
	La date de l'évènement	dateEvn	DateTime

Tableau 33 : Le glossaire de la table « Historique ».

En pratique, Les classes techniques provenant de l'application de « New Life », permettent d'identifier de nouvelles responsabilités techniques telles que stocker toutes les informations relatives au bon fonctionnement du système, ainsi la récupération des mots de passe, la publicité, et les inscriptions à la newsletter.

La table « Accueil », permet à l'administrateur de gérer le diaporama de la page d'accueil.

	Définition de l'attribut	Attribut	Type
accueil	L'identifiant de table accueil	idAcc	Int(1)
	L'image du diaporama de la page d'accueil	image	Varchar(100)
	Grand texte du diaporama de la page d'accueil	GTexte	Text
	Petit texte du diaporama de la page d'accueil	PTexte	Text
	Le lien pour rediriger l'utilisateur lors d'un clic	lien	Text

Tableau 34 : Le glossaire de la table « Accueil».

La table « Bonus », permet à l'administrateur de gérer les bonus mensuel.

	Définition de l'attribut	Attribut	Type
bonus	L'identifiant de la table bonus	IdBon	Int(11)
	Le nombre de personnes inscrites directement pour atteindre le bonus	bonusDirect	Int(5)
	Le nombre total de personnes inscrites par le groupe de l'utilisateur	bonusGroupe	Int(5)
	Le montant de bonus	Somme	Int(6)
	Le pack d'inscription concerné	pack	Varchar(2)

Tableau 35 : Le glossaire de la table « Bonus».

La table « Newsletter », nous permet d'identifier les internautes qui sont inscrits au newsletter.

	Définition de l'attribut	Attribut	Type
newsletter	L'identifiant de la table newsletter	idNI	Int(11)
	e-mail de l'inscrit à la newsletter	mail	Varchar(326)

Tableau 36 : Le glossaire de la table « Newsletter».

La table « pubPageInscription », permet à l'administrateur de gérer la publicité dans la page d'inscription.

	Définition de l'attribut	Attribut	Type
pubPageInscription	L'identifiant de la table publicité pour la page d'inscription	idPubP	Int(3)
	Le nom, titre de la ligne du bonus.	nom	Varchar(100)

	L'image du bonus	image	Varchar(100)
	La valeur du bonus pour le pack Initiation	N1	Varchar(10)
	La valeur du bonus pour le pack classic	N2	Varchar(10)
	La valeur du bonus pour le pack evolution	N3	Varchar(10)
	La valeur du bonus pour le pack premium	N4	Varchar(10)
	Explications du bonus	details	Text

Tableau 37 : Le glossaire de la table « pubPageInscription».

La table « verificationEmail », nous permet la vérification des e-mails et la réinitialisation du mot passe.

	Définition de l'attribut	Attribut	Type
VerificationEmail	L'identifiant de la table vérification Email	idEm	Int(11)
	Le code qui sera envoyé par e-mail	code	Varchar(150)
	Vérification d'e-mail, réinitialisation du mot de passe	type	Varchar(1)

Tableau 38 : Le glossaire de la table « VerificationEmail».

La table « Publicité », permet à l'administrateur de gérer la publicité qui s'affiche sur la page compte de l'utilisateur.

	Définition de l'attribut	Attribut	Type
Publicité	L'identifiant de la table publicité	IdPub	Int(11)
	La photo de la publicité	Photos	Varchar(150)
	Le lien pour rediriger l'utilisateur lors d'un click	lien	Varchar(250)

Tableau 39 : Le glossaire de la table « Publicité».

Ce chapitre a été consacré à la conception de l'application web. En premier lieu nous avons exposé les différentes interactions entre les objets du système grâce aux diagrammes de séquence détaillés. Sur ce, nous sommes arrivés à l'obtention du schéma relationnel en passant par la conception du modèle logique de données déduit à partir du diagramme de classe.

Désormais, la phase conception est terminée, nous pouvons s'étaler alors sur l'activité finale de notre projet qui est la phase de réalisation et mise en œuvre donné dans le chapitre suivant.

Chapitre IV

Réalisation et mise en œuvre

Diagramme de déploiement

Mise en ligne et validation

Structure et IHM

Sécurité de l'application

Maintenance et statistiques

A ce stade du processus, les cas d'utilisation sont bien cernés, le problème a été analysé en profondeur et nous avons défini une conception appropriée aux besoins de l'application. Nous pouvons alors entreprendre l'implémentation.

Premièrement dans ce chapitre, nous illustrerons comment nous aurons pu déployer, mise en ligne et valider notre application, ensuite, nous survolerons quelques langages que nous avons utilisé pour la réalisation, et puisqu'un langage de programmation ne peut se détacher des outils et des logiciels de programmation, chacun de ces derniers se verra attribuer une brève description. Ensuite, nous allons nous intéresser à l'IHM et aux interfaces que nous aurons pu réaliser. Puis, nous allons nous intéresser à la sécurité de l'application, où nous allons décrire les moyens que nous avons utilisés pour garantir une bonne utilisation. Nous clôturons ce chapitre, par la phase de maintenance.

IV.1. Phases de déploiement

Dans cette phase de déploiement, nous mettrons en œuvre le processus de mise en service, le déploiement de notre application met en jeu le processus de mise en ligne de l'application web. En plus de cela, nous verrons comment nous allons testé et validé notre travail avec les dirigeants de « New Life » et comment nous aurons pu les former et les aider à mieux se familiariser avec son utilisation.

IV.1.1. Architectures client/serveur

Les différents types architectures les plus connus sont [15] :

Architecture à 1-tiers : Dans une approche d'application de type 1-tiers, les trois couches sont fortement et intimement liées, et s'exécutent sur la même machine. Dans ce cas, on ne peut pas parler d'architecture client-serveur mais d'informatique centralisée.

Architecture deux tiers : L'architecture deux tiers le mot tier signifiant tierce partie (aussi appelée architecture à deux niveaux), caractérise les systèmes clients/serveurs dans lesquels le client demande une ressource et le serveur la lui fournit directement. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir le service.

Architecture trois tiers : Dans l'architecture 3-tiers (appelée aussi architecture à trois niveaux), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre :

- **Client**, c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur web) chargée de la présentation.
- **Serveur d'application (appelé également middleware)**, chargé de fournir la ressource mais faisant appel à un autre serveur.
- **Serveur de donnée**, fournissant au serveur d'application les données dont il a besoin.

L'architecture n-tiers : L'architecture n-tiers a été pensée pour pallier aux limitations des architectures trois tiers et concevoir des applications puissantes et simples à maintenir. Ce type d'architecture permet de distribuer plus librement la logique applicative, ce qui facilite la répartition de la charge entre tous les niveaux.

Pour notre cas nous avons choisi une architecture à trois tiers pour séparer entre le serveur apache et le serveur de base de donnée MySQL.

IV.1.2. Diagramme de déploiement

Le diagramme de déploiement du système que nous avons réalisé, est illustré par la figure suivante :

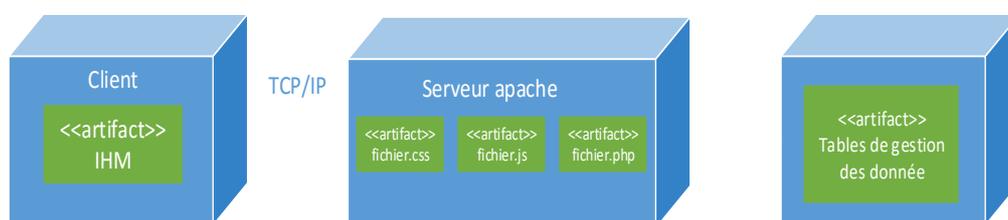


Figure 25 : Diagramme de déploiement de l'application.

Chaque utilisateur a son propre poste client qui est un « PC » connecté au serveur web Apache, qui est lui-même un serveur d'applications sur lequel est déployée en particulier l'application d'authentification. Chacun de ces utilisateurs partagent la même interface homme-machine qui utilise le service d'authentification contenu par le serveur Apache. Toutes les tables sont stockées dans une base de données spécifiques qui est hébergée par le serveur de base de données « MySQL ».

IV.1.3. Mise en ligne

Dans le but d'offrir un service de qualité aux visiteurs, il faut donc avoir recours au service d'un hébergeur de qualité, du côté de « New Life », ils nous ont simplement exigé une rapidité de service et une sécurité optimale.

Vu la complexité du système binaire, notre application devra être capable de surmonter les connexions qui vont être multipliées du fur et à mesure.

Vu que nous avons utilisé les « event scheduler » [16] il nous faut aussi un accès racine au serveur (Root) pour activer le service.

Voici donc les éléments que nous avons pris en compte lors de notre recherche d'hébergements :

- Espace disque acceptable plus de 20 Go.
- Bande passante illimité.

- Un hébergement où le service « event scheduler » et les « procédures stockées » [17] sont activés, ou un serveur VPS [18].
- Notre serveur doit supporter PHP et MySQL.
- Le service doit être accompagné d'un nom de domaine.
- Serveur de messagerie.
- Un certificat SSL [19] pour la sécurisation des transferts de données.

Après avoir cherché des hébergeurs sur le web, nous avons opté pour « planète Hoster », qui offre un service qualité/prix acceptable. Pour leur service d'hébergement mutualisé, ils ne disposent pas du service « event scheduler » donc, il nous a fallu nous documenter sur les serveurs VPS. A la fin, nous avons opté pour un serveur VPS avec 30GO de mémoire extensible, avec un processeur de 3 cœur 3.2GHZ chaque un, et une mémoire vive de 3GO, disposant d'un système Unix, WHM [20] et cPanel [21] préinstallé,

Le seul manque est le certificat SSL qui n'est pas inclu dans le service, c'est pour cela que nous avons créé un certificat auto signé avec openSSL [22].

Pour le nom de domaine nous avons utilisé « **www.newlifeopportunity.com** » pour l'application final et « **www.newlifeopportunity.ml** », et le sous domaine « **www.test.newlifeopportunity.com** » pour les tests.

Côté messagerie, nous avons créé quelques adresses mails pour les gérants de « New Life » « **Contact@newlifeopportunity.com** » pour pouvoir contacter « New Life » concernant des informations générales, « **support@newlifeopportunity.com** » pour contacter « New Life » pour les problèmes techniques, « **commercial@newlifeopportunity.com** » pour tout contact avec le service commercial.

IV.1.4. Validation

La validation de notre système ne peut se faire qu'à travers une phase de tests sur un sous ensemble représentatif des acteurs et des données impliquées. A cette fin, nous avons entrepris une expérimentation avec quelques clients, et les gérants de «New Life». Toutes les fonctionnalités prévues lors de la conception ont été testées. A part des détails techniques d'interfaces et de contrôle de saisie. Ces fonctionnalités répondent correctement aux exigences de «New Life». Ceci dit, il faut préciser qu'une autre phase de validation va être opérée sur le terrain d'exploitation de l'application « New Life ». En effet, c'est cette phase qui va nous permettre de voir comment les utilisateurs effectifs de ce produit vont réagir et comment ils vont l'apprécier.

IV.1.5. Guide d'utilisation

Pour aider les clients de « New Life » à se familiariser avec l'application web, nous avons créé quelques vidéos qui expliquent les services fondamentaux tels que : la préinscription, le parrainage, ainsi que la modification des informations et la consultation de l'arbre, etc...

IV.2. Les langages utilisés

Au cours de la réalisation de l'application web, une multitude de langages de programmation ont été mis en œuvre afin que ce dernier puisse être implémenté comme il a été décrit lors de la phase de conception.

on s'est penché sur PHP/MySQL pour les langages coté serveur vue notre expérience dans ces derniers, JavaScript avec sa bibliothèque XMLHttpRequest (Ajax) qui nous aidera à mieux interagir coté client, CSS3 et HTML5 pour nous garantir une bonne structuration des données et un design à la hauteur des attentes des « New Life » et ses clients, enfin JSON pour faire communiquer PHP avec JavaScript .

Voici une brève description de chacun des langages utilisés :

IV.2.1. HTML (HyperText Markup Language)

Le langage HTML [23] est le langage universel utilisé sur les pages web lisibles par tous les navigateurs web (Internet Explorer, Netscape, Mozilla, etc...). Ce langage fonctionne suivant l'assemblage et la combinaison de balises permettant de structurer et donner l'apparence voulue aux données textes, images et multimédias suivant la mise en page voulue.

C'est un langage qui permet de créer des hyperliens, à savoir des liens d'un document à un autre ou d'un endroit d'un document à un autre endroit du même document (identificateur de fragment).

IV.2.2. JavaScript

JavaScript [24] est un langage de script orienté objet principalement utilisé dans les pages HTML. A l'opposé des langages serveurs (qui s'exécutent sur le site), JavaScript est exécuté sur l'ordinateur de l'internaute par le navigateur lui-même. Ainsi, ce langage permet une interaction avec l'utilisateur en fonction de ses actions (lors du passage de la souris au-dessus d'un élément, du redimensionnement de la page...). La version standardisée de JavaScript est l'ECMAScript.

IV.2.3. AJAX

AJAX (Asynchrones JavaScript And XML) [13] est une méthode de développement web basée sur l'utilisation d'un script JavaScript pour effectuer des requêtes web à l'intérieur d'une page web sans recharger la page. AJAX rend plus interactifs les sites web et offre une meilleure ergonomie ainsi qu'une réactivité améliorée en permettant de modifier interactivement une partie de l'interface web seulement.

AJAX n'est ni une technologie ni un langage de programmation mais une manière de développer des pages web en se basant sur certaines technologies comme HTML et CSS pour la présentation, DOM (Document Object Model) pour la représentation en objets de la page web, JavaScript et en particulier l'objet XMLHttpRequest pour manipuler des requêtes et des réponses.

IV.2.4. PHP (Hypertext Preprocessor)

Le PHP [8] est un langage de programmation, sa principale application se situe au niveau de la gestion des sites web dynamiques, Les capacités de PHP ne s'arrêtent pas à la création de pages web. Il est aussi possible de manipuler des images, de créer des fichiers PDF, de se connecter à des bases de données ou des serveurs, et même d'instancier des objets Java. Un module annexe lui permet également de fournir des interfaces graphiques classiques (client lourd, sans navigateur ou serveur web). PHP est un langage impératif orienté-objet.

IV.2.5. MySQL

MySQL [9] est un serveur de bases de données relationnelles Open Source. Un serveur de bases de données stocke les données dans des tables séparées plutôt que de tout rassembler dans une seule table. Cela améliore la rapidité et la souplesse de l'ensemble. Les tables sont reliées par des relations définies, qui rendent possible la combinaison de données entre plusieurs tables durant une requête. Le SQL dans "MySQL" signifie "Structured Query Language" : le langage standard pour les traitements de bases de données.

IV.2.6. CSS

Les feuilles de styles (en anglais "Cascading Style Sheets", abrégé CSS) [12] sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML.

IV.2.7. JSON

JSON (JavaScript Object Notation – Notation Objet issue de JavaScript) [25] est un format léger d'échange de données. Il est facile à lire ou à écrire pour des humains. Il est aisément analysable ou générable par des machines. Il est basé sur un sous-ensemble du langage de programmation JavaScript.

JSON est un format texte complètement indépendant de tout langage, mais les conventions qu'il utilise seront familières à tout programmeur habitué aux langages descendant du C, comme par exemple : C lui-même, C++, C#, Java, JavaScript, Perl, Python et bien d'autres. Ces propriétés font de JSON un langage d'échange de données idéal.

IV.3. Outils et logiciels utilisés

Les langages de programmations seuls, ne peuvent pas nous permettre de nous avancer rapidement, pour cela nous allons nous servir de quelques logiciels :

On a choisi aptana studio pour une bonne implémentation et une visibilité du code, ainsi que sa gratuité. Comme nous le savons, PHP ne peut pas s'exécuter sans un environnement, donc nous avons opté pour WAMP serveur pour que nous puissions travailler en local. Pour la partie graphique, nous nous sommes appuyé sur quelques éditeurs d'image et vidéo : paint.NET pour sa simplicité, adobe Illustrator pour la création graphique vectorielle, et pleins d'autres logiciels qui nous ont permis de réaliser notre travail. Visio de Microsoft pour les diagrammes, UML, After Effects qui nous ont permis de réaliser quelques vidéos de démonstration pour « New Life » et ses clients. Coté hébergements, nous avons choisi un serveur VPS avec un système d'exploitation Unix et WHM pour la gestion administrateur et cPanel pour la gestion des comptes.

IV.3.1. Aptana Studio

Aptana IDE [26] est un environnement complet pour le développement web multiplateformes (Linux, MacOS, Windows) Open-Source, il facilite le code en fournissant des aides à la saisie pour le JavaScript, l'HTML, les CSS et PHP.

Aptana est basé sur Eclipse, et propose de nombreuses fonctionnalités : auto-complétion, fonction de preview, debugger JavaScript, gestion ftp...etc.

IV.3.2. Environnement WampServer

WampServer (anciennement WAMP5) est une plateforme de développement Web, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que PHPMyAdmin pour l'administration web des bases MySQL.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'un tray icône (icône près de l'horloge de Windows).

Le serveur web « Apache » [27] est l'un des serveurs les plus répons sur internet et il est basé sur le protocole HTTP. L'objectif est de fournir, à un maximum de personnes et d'organisations, une plate-forme solide pour réaliser des tests et des applications de productions, son grand avantage est le fait qu'il fonctionne sur de nombreux systèmes d'exploitation dont Microsoft Windows, Linux,....

MySQL et PHP sont respectivement un Système de Gestion de Bases de Données (SGBD) et un langage impératif orienté objet.

IV.3.3. Paint.NET

Paint.NET [28] est un outil de retouche photo particulièrement simple à prendre en main, il s'est développé comme un outil puissant mais simple pour facilement retoucher, retailer, enlever les défauts ou appliquer des effets aux photos, de nombreuses fonctionnalités sont proposées faisant du produit un logiciel incontournable pour les photographes amateurs.

IV.3.4. Visio

Microsoft Visio (officiellement Microsoft Office Visio) [29] est un logiciel de diagrammes pour Windows qui fait partie de la suite bureautique Microsoft Office.

Les principales fonctionnalités de Visio sont :

- Créer des diagrammes professionnels rapidement avec des formes mises à jour et de nouveaux outils et options de mise en forme.
- Associer des diagrammes à des données dynamiques et interpréter les données complexes.
- Collaborer facilement en équipe sur les diagrammes.

IV.3.5. cPanel

CPanel [21] est un panneau de configuration basé sur Linux conçu pour les hébergeurs web. Constitué d'une interface graphique permettant l'automatisation des paramètres, l'hébergement de l'application web est ainsi simplifié. cPanel est doté de 3 principales fonctions qui permettent d'accéder à différents niveaux d'utilisation tels que l'administration et la revente d'un hébergement, ou la simple configuration de l'application web. Ainsi, tous ces aspects sont contrôlés à partir d'un simple navigateur web.

IV.3.6. VPS (Serveur Privé Virtuel)

Un serveur dédié virtuel (également appelé serveur virtuel), en anglais virtual private server (VPS) ou virtual dedicated server (VDS) [18] est une méthode de partitionnement d'un serveur en plusieurs serveurs virtuels indépendants qui ont chacun les caractéristiques d'un serveur dédié (serveur informatique), en utilisant des techniques de virtualisation. Chaque serveur peut fonctionner avec un système d'exploitation différent et redémarrer indépendamment. Dans le domaine de l'hébergement web, plusieurs dénominations recoupent le même type d'offres et donc de services. Les acronymes VPS (Virtual Private Server) et VDS (Virtual Dedicated Server) désignent le même concept, et leur usage est parfois confus.

IV.3.7. Adobe Illustrator

Adobe Illustrator est un logiciel de création graphique vectorielle. Il fait partie de la gamme Adobe et peut être utilisé indépendamment ou en complément de Photoshop, il offre des outils de dessin vectoriel puissants [30]. Adapté aussi bien à la création de document papier qu'à celle d'illustrations pour Internet (logos, affiches, etc.) ce logiciel est orienté vers le marché professionnel, il intègre de nombreuses options propres à améliorer la productivité.

IV.4. Interface Homme-Machines(IHM)

« New Life » nous a rien exigé coté interface, mais ce n'est pas pour cela que nous allons pas essayer de nous surpasser pour offrir à « New Life » des interfaces conviviale, et à ses clients une facilité d'utilisation et une autre expérience avec le marketing de réseaux.

IV.4.1. Structure de l'application

La structuration de l'application web consiste à bâtir l'architecture globale de l'application web en organisant les différentes informations.

Après avoir interrogé les gérants de « New Life » pour leur souhait concernant la structure, ils nous ont donné une liberté, à condition de ne pas faire perdre le client en surfant sur l'application.

Il s'agit dans un premier temps, de faire l'inventaire des contenus de l'application web, et de les regrouper par thématique afin de constituer des rubriques.

Dès lors que les principales rubriques sont identifiées, il nous a fallu alors déterminer leur organisation et leur découpage en sous-rubriques.

Schématiquement, nous aurons la « **figure 26** » qui représente la structure de notre application qui peut être aussi donnée par un diagramme de navigation.

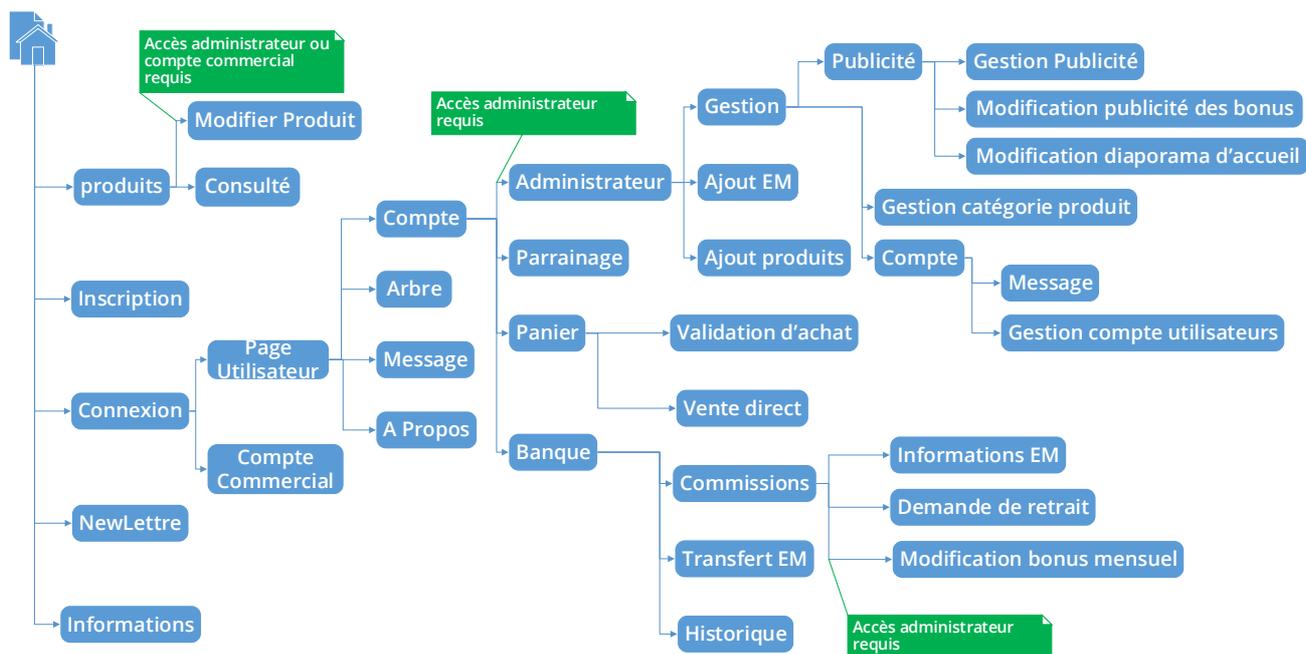


Figure 26 : Structure de l'application web.

Dans la « figure 26 » toutes les pages disposent d'un retour arrière plus un lien vers la page d'accueil et la page utilisateur.

IV.4.2. Présentations des interfaces de l'application

Nous allons présenter, dans ce qui suit, quelques exemples représentatifs de l'IHM de notre application.

IV.4.2.1. IHM du Visiteur

Nous avons choisi de montrer la page d'accueil de notre application Web car elle est très importante. En effet, c'est elle qui va inciter le visiteur à aller plus loin pour découvrir « New Life ».

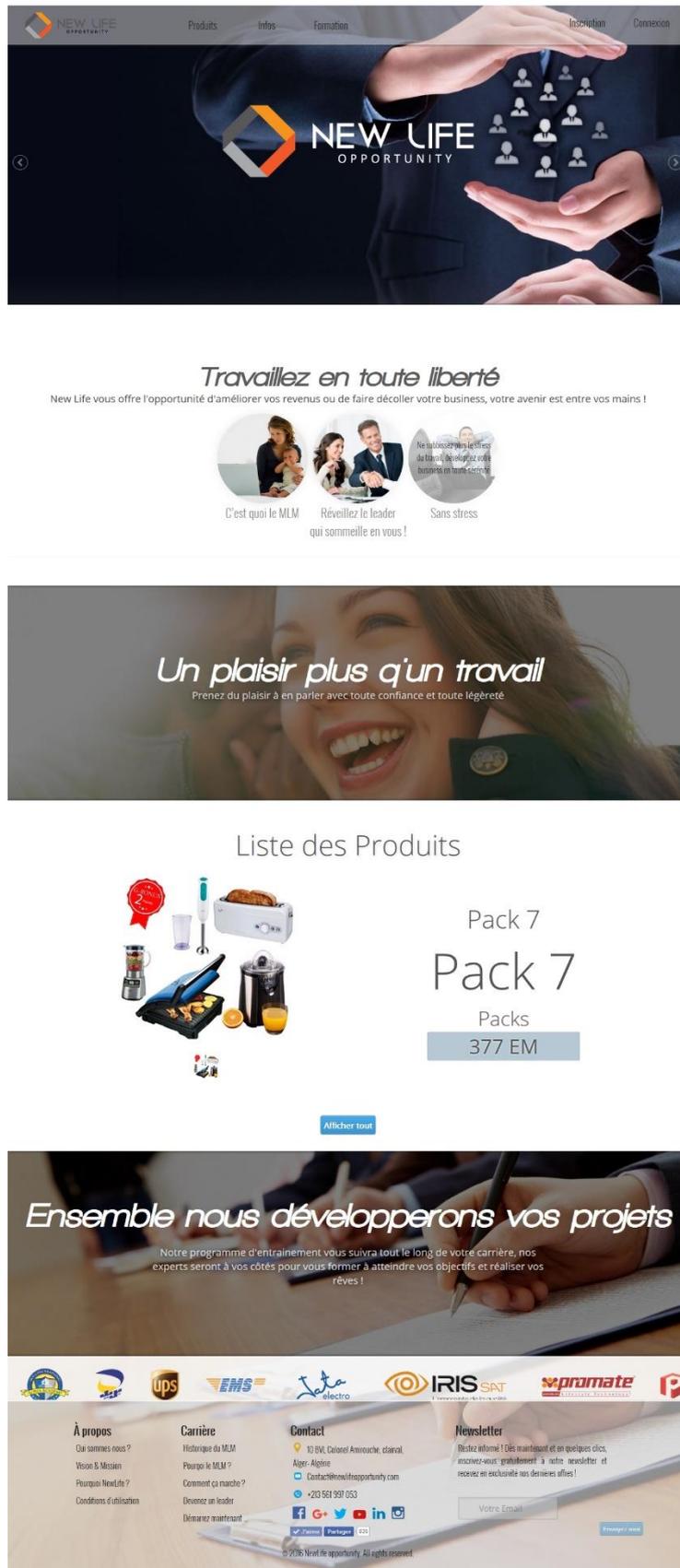


Figure 27 : IHM de la page d'accueil.

La page du formulaire d'inscription 1 : c'est l'interface qui s'affiche lors d'une demande d'inscription, là nous expliquons au visiteur les différents prix et avantages des packs d'inscription.

Initiation	Classic	Evolution	Premium
Cash : 126 Facilité : 46 Points d'équilibre : 2 C.Directes : 10 Premier retrait : 300 Max out : 600 par semaine	Cash : 286 Facilité : 100 Points d'équilibre : 4 C.Directes : 20 Premier retrait : 300 Max out : 1200 par semaine	Cash : 587 Facilité : 200 Points d'équilibre : 8 C.Directes : 30 Premier retrait : 300 Max out : 2000 par semaine	Cash : 986 Facilité : 400 Points d'équilibre : 12 C.Directes : 40 Premier retrait : 300 Max out : 4000 par semaine

Figure 28 : La page « Formulaire d'inscription 1 ».

La page du formulaire d'inscription 2: après avoir choisi le pack d'inscription, un formulaire d'inscription s'affiche, C'est le formulaire qui s'affiche pour tout internaute désirant s'inscrire a «New life » , pour se faire, il doit saisir ses coordonnées (Pseudo, E-mail, Numéro de téléphone et un mot de passe).

Le pack Classic

Remplissez soigneusement le formulaire suivant

rima06 ❌	14782 ✔️
----------	----------

Ce pseudo existe déjà

..... ✔️

Retaper votre mot de passe

badja.riad@gmail.com ❌

Cette adresse mail existe déjà

Retaper votre E-mail

benallaoua ✔️	rima ✔️
---------------	---------

Numéro de télépl	Numéro de télépl
------------------	------------------

Cash ▼	Homme ▼
--------	---------

CONDITIONS GENERALES

Les présentes conditions générales ont pour rôle de déterminer la responsabilité des différents acteurs qui interagissent, ou encore de déterminer les sanctions en cas de non-respect des règles. En d'autres termes, ces conditions fixent un cadre légal à toute transaction et sécurisent la vente.

Les conditions générales sont importantes car elles

Si vous passez à l'étape suivante, vous acceptez nos termes d'utilisation !

Je ne suis pas un robot  reCAPTCHA
Confidentialité - Conditions

s'enregistrer et passer à l'étape suivante

Figure 29 : La page « Formulaire d'inscription 2 ».

IV.4.2.2.IHM du client

Ce sont les pages qui s'affichent lorsque c'est un client qui est connecté, ses pages bien évidemment contiennent des liens, des informations limitées par rapport à l'administrateur ou bien compte commercial.

Voici une illustration de la page de l'espace client :



Figure 30 : IHM du client.

Arbre de l'utilisateur : c'est l'interface qui offre aux utilisateurs une vue sur leur arbre pour qu'ils puissent visualiser leurs réseaux, parrainer des personnes, et les aider à gérer leur équilibre.

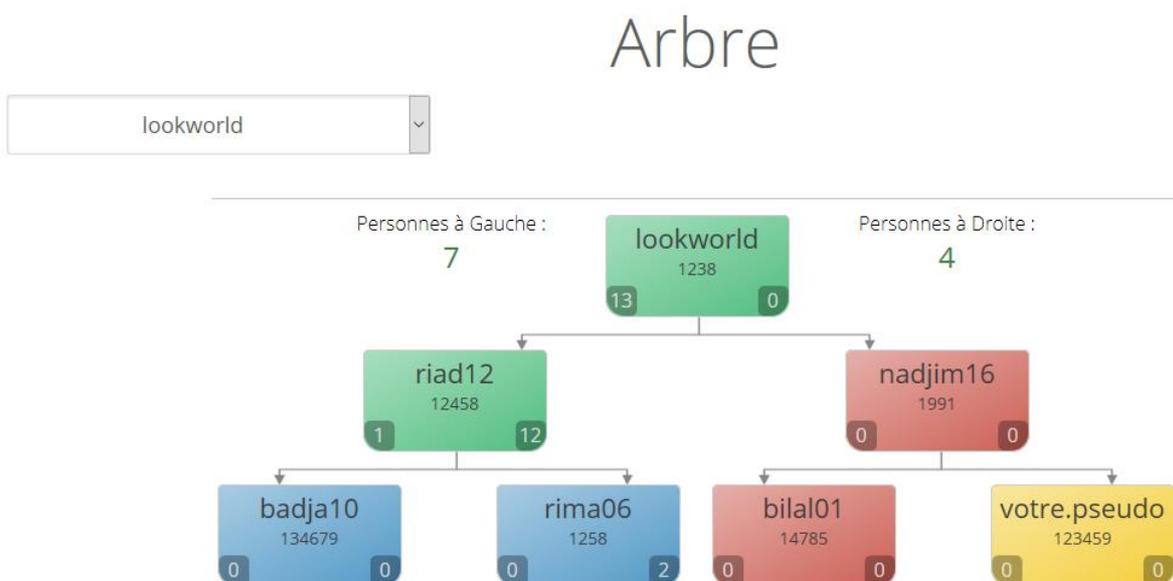


Figure 31 : La page « Arbre de l'utilisateur ».

Page de parrainages : c'est l'interface qui s'affiche lorsque le client demande de parrainer des nouveaux clients :

Parrainage

Le parrain directe :

le parrain directe :DZ134679

Le parent :

le nouveau RI :

Visualisation du parrainage

```

graph TD
    A["badja10  
BADJA Riad"] --> B["Libre"]
    A --> C["badja123  
BADJA Riad"]
  
```

Conditions d'inscription

E-Bank	Frais d'inscription
80	46
<input type="text" value="Votre code PIN"/>	
<input style="background-color: #2980b9; color: white;" type="button" value="Parrainer"/>	

Figure 32 : La page « parrainage de l'utilisateur ».

IV.4.2.3.Espace de l'administrateur

C'est l'administrateur de l'application qui a beaucoup plus de privilèges par rapport aux autres utilisateurs, voir gestion des utilisateurs, gestion de l'application etc...

Voici quelques illustrations des IHM de l'administrateur :

Gestion compte : Dans cette page, l'administrateur peut envoyer des messages pour tous, modifier les comptes utilisateurs et changer le mot de passe du compte commercial :

Message pour tous

Ecrivez le message

Tapez votre texte ici

Envoyer

Gestion des comptes d'utilisateurs

Ecrivez le pseudo, DZ ,nom ou prenom du client

Recherche

Recherche

changer le mot de passe du compte commercial

Cliquer sur le bouton et suivez les instruction

Changer le mot de passe

Figure 33 : IHM de l'administrateur « Gestion compte ».

Gérer les catégories : c'est l'interface qui s'affiche lorsque l'administrateur demande de gérer les catégories des produits :

Gestions des catégories de produits

La liste des catégories de produits

Accessoires de Mode	⚙️	🗑️
Electroménager	⚙️	🗑️
Formations	⚙️	🗑️
Gadgets	⚙️	🗑️
High-Tech	⚙️	🗑️
Packs	⚙️	🗑️
Soins et Beauté	⚙️	🗑️
Voyages	⚙️	🗑️

Ajouter une catégories

Nom de la catégories

Ajouter

Figure 34 : IHM de l'administrateur « Gestions des catégories de produits ».

IV.4.2.4. Espace du service commercial

Ce sont les pages qui s'affichent lorsque c'est un compte commercial qui est connecté à l'application.

Voici les IHM du compte commercial :

Validation des commandes : C'est dans cette interface que le compte commercial gère les commandes de livraison :

Service commercial

[Ajouter des produits](#)

Toutes les commandes ▼

 **TV LED E4600
HD 32'** **Quantité : 1**
livraison : OUI
Le 18/06/2016 à 12h56
1 exemplaire(s)

Produit en cours livraison ...

BADJA riad
kjhkjfhskjhg ksfjghdfjdkh Bejaia DZ | Tel:665652147 | Zip:06000 | CIN:123544354

 **TABLETTE
G8010** **Quantité : 1**
livraison : OUI
Le 18/06/2016 à 12h56
1 exemplaire(s)

[Confirmation](#)

BADJA riad
ighile-ouazougue Bejaia DZ | Tel:661157404 | Zip:06000 | CIN:123544354

[Sélectionner tout](#)

[Imprimer](#)

Figure 35 : IHM de l'espace du service commercial.

Liste des nouveaux inscrits (pack Pro) : C'est l'interface qui permet au compte commercial de gérer les livraisons de pack pro :

Service commercial

Nouveaux membres Nouveau

rima06	BENALLAOUA Ryma DZ1258 0666565654 032659874 date de naissance: 04/03/1993 rima@badja.me DZ: sdfksdljkgkdghdfkjghfdkj Alger 124585 Trans: D Inscrit le: 07/02/2016	Initiation	Vue
badja01	BADJA riad FR12389 032659874 012457852 date de naissance: 15/10/1992 riad@badja.me FR: lsfsksh slkfjdkgid flkjdf paris 12454852 Trans: C Inscrit le: 26/02/2016	Classic	Vue

Figure 36 : La page « Liste des nouveaux inscrits ».

IV.5. Sécurité de l'application

Vu la sensibilité du système de « New Life », nous avons tout mis en œuvre pour garantir une sécurité optimale, et cela pour préserver l'image et les données de « New Life ». Nous avons opté pour les bonnes pratiques suivantes :

- Chaque page de l'application est soumise à une vérification des droits d'accès. Avant même que le serveur exécute des opérations critiques, si le client n'est pas autorisé à les exécuter, la page sera bloquée et le lien est redirigé vers une page d'erreur.
- Les variables sont vérifiées du côté client (navigateur) pour optimiser la relation homme-machine, et elles seront revérifiées côté serveur pour garantir la structure des données souhaitée et éviter les injections SQL.
- On a opté pour l'utilisation du captcha de Google « reCAPTCHA » [31].
- Les mots de passe des clients sont hachés avec une combinaison de deux algorithmes de hachage MD5 [32] et SHA [33], qui est sous la forme suivante :
`sha1 ('?' .md5 ('5' . sha1 ('@.@' . $motDePasse . '? . ? ')))`
- Si une authentification échoue, le client est obligé d'attendre une seconde pour le prochain essai vu qu'elle n'influe pas sur le temps d'attente du client. Le nombre minimum de caractères d'un mot de passe est de 8 caractères, nous avons pris juste 40 cas possible de caractères, donc un attaquant avec « brut force » doit attendre 40^8 seconde, soit plus de six millions d'années.

- La réinitialisation d'un mot de passe est protégée avec un code de 150 caractères envoyé à l'email du client, le code est utilisable juste une fois.
- Les comptes des clients sont déconnectés automatiquement s'il y a une non-activité de dix minutes, et cinq pour l'administrateur.
- Les transactions importantes (ajout EM, modification des compte d'utilisateur, transfert EM, parrainage) sont protégées par code PIN.
- On a veillé à ne pas divulguer les informations des utilisateurs même entre les clients de « New Life » (ils peuvent juste voir les « DZ » de tout le monde, et les noms et prénoms de leurs descendants).

IV.6. Phases de tests et maintenance

Après la phase de validation sur le terrain, nous avons constaté quelques anomalies techniques, quelques bugs, et des problèmes de saisie parce que nous avons été strictes sur les caractères autorisés. Même les gérants de « New Life » ont constaté quelques défauts sur le fonctionnement du système pour des raisons marketing et éthique. Pour cela, nous avons fait quelques modifications sur l'application web pour qu'il s'adapte aux nouvelles exigences de « New Life ».

Voici quelques modifications que nous avons apportées à l'application web :

IV.6.1. Modification des packs d'inscription

« New Life » a suspecté que le premier pack d'inscription pose un problème juridique et éthique donc, elle a préféré de le changer, et de le mettre comme les autres packs. Dans l'autre part, elle a trouvé que les autres niveaux étaient chers pour les clients, que ces derniers n'arrivaient pas à les acquérir, ce qui suit montre les changements apportés :

PACK INSCRIPTION	INITIATION	CLASSIC	EVOLUTION	PREMIUM
INVESTISSEMENT	126	286	587	986
PACK PRO	46	86	127	226
ACHAT	80	200	460	760
COMMISSION DIRECT	10	20	30	40
POINT D'EQUILIBRE	2	4	8	12
BONUS DEMARRAGE	0	0	7	17
BONUS MENSUEL	0	46	107	166
1 ^{ER} RETRAIT	300	300	300	300
FACILITE	46	100	200	400
MAX OUT	600	1200	2000	4000

Tableau 40 : Les nouvelles valeurs du système de « New Life ».

IV.6.2. Méthodes de livraison

« New Life » a trouvé que ces clients payent des frais de livraisons même s'ils habitent près du siège officiel, donc ils nous ont demandé que lors de la commande, le client précise s'il désire recevoir le produit grâce à une entreprise de livraison, ou simplement venir le récupérer chez « New Life » .

IV.6.3. Autres

« New Life » a constaté plusieurs manques dans l'application pour cela nous avons apporté les solutions suivantes :

- Ajouter la date de naissance au formulaire de complément d'information
- Ajouter quelques informations au compte commercial lors des demandes de livraisons des packs pro
- Supprimer la photo des produits lors de l'impression des étiquettes
- Changer le design de la liste des produits
- Désactiver la migration

IV.6.4. Etude statistique

Pour conclure notre rapport, nous avons décidé de faire une étude statistique sur l'évolution de « New Life » et le trafic de l'application web. Pour cela, nous avons analysé premièrement la base de donnée afin de voir l'évolution des inscriptions, ensuite nous expliquons les données qui sont recueillies par le plugin AwStats [34] qui est inclut dans le cPanel.

IV.6.4.1. Evolution des inscriptions

Dans le but de suivre l'évolution des inscriptions finales (parrainage), nous avons décidé de faire un schéma qui les détaille depuis le lancement du premier livrable (février/2016) jusqu'à la date de la rédaction de cette section (juin/2016).

	PACK 1	PACK 2	PACK 3	PACK 4
FÉVR-16	2	183	8	68
MARS-16	3	7	3	0
AVR-16	3	2	1	0
MAI-16	23	19	2	9
JUIN-16	8	4	1	0
TOTALE	39	215	15	77

Tableau 41 : Evolution des inscriptions.

IV.6.4.2. Evolution du trafic

Visite / visiteur : ici, nous s'intéressons aux visiteurs, aux pages visionnée, et à la bande passante consommée donnée par le « **tableau 42** » :

MOIS	VISITEURS DIFFÉRENTS	VISITES	PAGES	BANDE PASSANTE
FÉVR-16	1 298	1 873	52 854	9.61 Go
MARS-16	791	1 075	20 854	6.41 Go
AVR-16	642	894	15 052	6.34 Go
MAI-16	676	931	21 224	4.87 Go
JUIN-16	603	778	14 087	3.92 Go

Tableau 42 : Evolution du trafic (Visite / visiteur).

Heures de visite : nous intéressons à présent au nombre de visiteurs à chaque heure de la journée pour voir si le payement à 5h du matin est une bonne prédiction

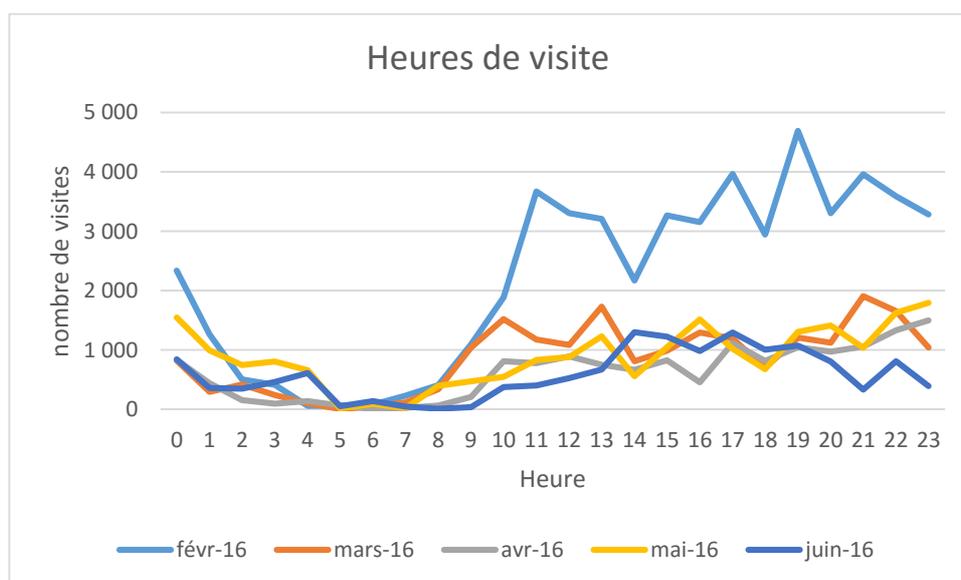


Figure 37 : Diagramme représentant l'évolution du nombre de visites en fonction des heures.

La « **figure 37** » montre que la fréquence des visites la plus basse pour les cinq mois est enregistrée entre 5h et 6h30min, nous déduisons que le payement à 5h du matin est un bon choix.

URL (arbre / connexion) : mettre la fonction qui permet de récupérer les fils dans l'authentification au lieu de l'arbre, nous donnons l'histogramme de la « **figure 38** » :

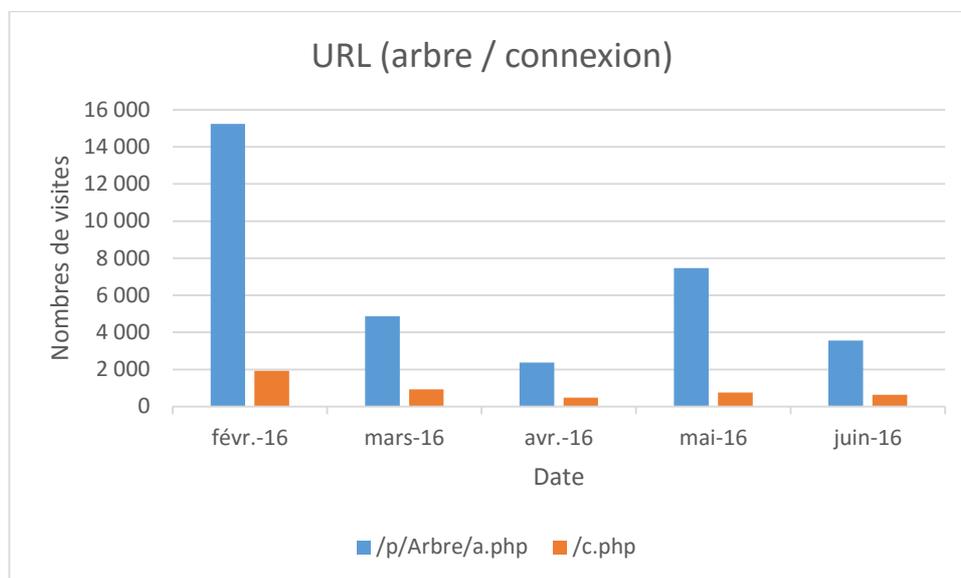


Figure 38 : Diagramme représentant le nombre de visites par rapport aux URL (arbre / connexion).

La comparaison des histogrammes pour chaque mois, confirme la bonne pratique de mettre la fonction qui permet de récupérer les fils dans l'authentification au lieu de l'arbre ;

La phase de réalisation est l'étape la plus importante dans le cycle de vie d'une application web.

Dans ce chapitre, nous avons présenté les aspects pratiques liés à la réalisation de l'application, à savoir les outils de développement nécessaires, et nous avons illustré quelques interfaces homme-Machine associés à notre application. Suivi du diagramme de déploiement associé à notre système. En dernier, nous avons présenté la mise en ligne de notre application ainsi la phase de la maintenance. En clôturant par une étude statistique.

Conclusion générale

Notre travail était dans le but de concevoir une application web pour informatiser une entreprise de marketing de réseau « New Life ». Cette application web permet de bien gérer leur système d'une manière simple et rapide. Pour ce faire, nous avons eu recours à un stage au sein de l'entreprise dans le but de nous impliquer et comprendre d'une manière précise son fonctionnement.

Pour débiter notre rapport, nous avons commencé par recueillir des informations détaillées sur le fonctionnement du système de « New Life », où cela nous a permis de comprendre notre objectif à réaliser, qui était de concevoir une application web pour la gestion des représentants, des produits et des paiement. Ensuite, nous avons élaboré un cahier de charge qui illustre les besoins fonctionnel, et non fonctionnel du travail demandé.

Cela nous a aidé à comprendre comment mettre en pratique nos connaissances théoriques sur les démarches de développement. Nous avons appris aussi, à mieux utiliser UML et les méthodes agiles sur le terrain. Et enfin, d'apprendre à héberger une application web et de la gérer grâce à WHM et cPanel.

Enfin, nous avons pu réaliser notre travail en respectant le cahier de charge, et en ajoutant des fonctionnalités que « New Life » a jugées intéressantes telles que le parrainage interactif et la simplicité du désigne.

En guise de perspective, nous voulons développer une application mobile qui utilisera les services web de notre application, modifier le désigne pour l'adapter aux petits écrans des smartphones, ajouter à l'administrateur des fonctionnalités pour consulter les statistiques de l'entreprise, ainsi que l'intégration du paiement en ligne dès son commencement en Algérie.

Bibliographie.

[8] : PHP 5 avancé [4^{ème} édition], Damien Seguy.

[10] : L'eXtreme programming, Jean-Louis Bénard, Laurent Bossavit, Régis Médina, Dominic Williams, Edition Eyrolles, 2002.

[11] : Conception de bases de données avec UML [1^{ère} édition], GILLES Roy, Presses de l'Université de Québec, 2009.

[15] : Tout sur les réseaux et internet [2^{ème} Edition], Jean-François Pillou, Edition Dunod, 2009

[17] : Administrez vos bases de données avec MySQL [2^{ème} édition], Chantal Gribaumont, OpenClassrooms - ex-Site du Zéro, 15/07/2014.

[23] : HTML et JavaScript, Daniel CHARNAY, Philippe CHALEAT.

Webographie.

[1] : <http://sandrinefourneaux.wikeo.be/le-mlm-cest-quoi.html> **Date accès** : 06/02/2016

[2] : <http://www.ecole-mlm.com/quelle-est-la-difference-entre-marketing-de-reseau-et-affiliation/>
Date accès : 06/02/2016

[3] : <http://netbusinessrating.com/fr/comparatif-6-mlm-1> **Date accès** : 06/02/2016

[4] : <http://www.iris-sat.dz> **Date accès** : 15/02/2016

[5] : <http://www.jata.es/ingles> **Date accès** : 15/02/2016

[6] : <https://www.ups.com> **Date accès** : 15/02/2016

[7] : <http://www.ems.dz> **Date accès** : 15/02/2016

[9] : <http://www.futura-sciences.com/magazines/hightech/infos/dico/d/internet-mysql-4640> **Date accès** : 04/03/2016

[12] : <https://www.w3.org/Style/CSS/> **Date accès** : 04/03/2016

[13] : <http://www.commentcamarche.net/contents/5-ajax-asynchronous-javascript-and-xml> **Date accès** : 04/03/2016

- [14] : http://www.tutorialspoint.com/java/java_using_singleton.htm **Date accès** : 03/04/2016
- [16] : <https://dev.mysql.com/doc/refman/5.7/en/event-scheduler.html> **Date accès** : 20/01/2016
- [18] : https://fr.wikipedia.org/wiki/Serveur_d%C3%A9di%C3%A9_virtuel **Date accès** : 04/03/2016
- [19] : <https://www.ovh.com/fr/ssl/fonctionnement-ssl.xml> **Date accès** : 04/03/2016
- [20] : <https://support.hostgator.com/articles/what-is-whm-web-host-manager> **Date accès** : 01/05/2016
- [21] : https://fr.wikipedia.org/wiki/Serveur_d%C3%A9di%C3%A9_virtuel **Date accès** : 01/05/2016
- [22] : <https://www.openssl.org/> **Date accès** : 01/05/2016
- [24] : <http://www.futura-sciences.com/magazines/high-tech/infos/dico/d/internet-javascript-509/> **Date accès** : 04/03/2016
- [25] : <http://www.json.org/json-en.html> **Date accès** : 04/03/2016
- [26] : <https://fr.wikipedia.org/wiki/Aptana> **Date accès** : 04/03/2016
- [27] : <http://siguillaume.developpez.com> **Date accès** : 04/03/2016
- [28] :
http://www.01net.com/telecharger/windows/Multimedia/photo_numerique/fiches/33029.html
Date accès : 04/03/2016
- [29] : <http://office.microsoft.com/fr-001/visio/> **Date accès** : 04/03/2016
- [30] : <http://forumde.lalogotheque.com/index.php?showtopic=11712> **Date accès** : 04/03/2016
- [31] : <https://fr.wikipedia.org/wiki/ReCAPTCHA> **Date accès** : 06/01/2016
- [32] : <https://fr.m.wikipedia.org/wiki/MD5> **Date accès** : 06/01/2016
- [33] : <https://fr.m.wikipedia.org/wiki/SHA> **Date accès** : 06/01/2016
- [34] : <http://www.awstats.org/> **Date accès** : 16/06/2016

Annexe 1

Méthodes agiles et XP

1. Méthodes agiles

Les méthodes de développement dites « **méthodes agiles** »(en anglais *Agile Modeling*, noté *AG*) visent à réduire le cycle de vie du logiciel (donc accélérer son développement) en développant une version minimale, puis en intégrant les fonctionnalités par un processus itératif basé sur une écoute client et des tests tout au long du cycle de développement.

L'origine des méthodes agiles est liée à l'instabilité de l'environnement technologique et au fait que le client est souvent dans l'incapacité de définir ses besoins de manière exhaustive dès le début du projet. Le terme « agile » fait ainsi référence à la capacité d'adaptation aux changements de contexte et aux modifications de spécifications intervenant pendant le processus de développement. En 2001, 17 personnes mirent ainsi au point le manifeste agile dont la traduction est la suivante :

- individus et interactions plutôt que processus et outils
- développement logiciel plutôt que documentation exhaustive
- collaboration avec le client plutôt que négociation contractuelle
- ouverture au changement plutôt que suivi d'un plan rigide

Une méthode agile est avant tout **itérative** sur la base d'un affinement du besoin mis en œuvre dans des fonctionnalités en cours de réalisation et même déjà réalisées. Cet affinement, indispensable à la mise en œuvre du concept adaptatif, se réalise en matière de génie logiciel sous deux aspects :

- Fonctionnellement, par adaptation systématique du produit aux changements du besoin détecté par l'utilisateur ;
- Techniquement, par remaniement (restructuration) régulier du code déjà produit.

Une méthode agile est ensuite, éventuellement, **incrémentale**, lorsque le projet, quel que soit le nombre de participants, dépasse en durée une dizaine de journées en moyenne, la production de ses fonctionnalités s'effectue en plusieurs incréments.

La notion d'**adaptatif**, quant à elle, nécessite au-delà d'un simple principe, la mise en œuvre de techniques de contrôle de l'évolution du livrable et d'une métrique formelle des

modifications, avant, après et en cours de la production. Il en découle une planification opérationnelle élémentaire, directement visible par le biais de l'affichage mural.

L'acceptation du mode adaptatif, qui permet au client de modifier ses exigences en cours de projet, aura pour conséquence l'éventualité d'un périmètre variable. Grâce aux méthodes agiles, le client est pilote à part entière de son projet et obtient très vite une première mise en production de son logiciel. Ainsi, il est possible d'associer les utilisateurs dès le début du projet

L'agilité comprend plusieurs courants de pensée qui ont conduit à des méthodologies reposant sur les mêmes concepts mais présentant chacune des singularités. Présentant successivement Xtreme Programming, DSDM, ASD, CRYSTAL, SCRUM, FDD

1.1. XP : eXtreme Programming

La méthode XP (pour eXtreme Programming) définit un certain nombre de bonnes pratiques permettant de développer un logiciel dans des conditions optimales en plaçant le client au coeur du processus de développement, en relation étroite avec le client.

1.1.1. Aux origines d'eXtreme Programming

La suppression des risques existants sur un projet représente la philosophie d'XP. . En introduisant cette méthode, ses créateurs veulent en finir avec la dérive de délais, l'annulation des projets, la non-qualité, la mauvaise compréhension du métier du client et l'impossibilité de suivre les changements.

XP a été mis en œuvre pour la première fois en 1996 sur le projet C3, Chrysler Comprehensive Compensation System, qui consistait à mettre en place un nouveau système de gestion de la paie des dix mille salariés du fabricant automobile. Sur ce projet, la méthode XP a été adoptée pour tenter de sauver le projet d'une dérive annoncée.

Les pères de la méthode, Ward Cunningham et Kent Beck définissent eX-treme Programming comme "une méthode basée sur des pratiques qui sont autant de boutons de contrôle poussés au maximum".

1.1.2. Les 4 valeurs d'eXtreme Programming

XP met en avant quatre valeurs prenant en considération à la fois les enjeux commerciaux et les aspects humains des projets de développement d'applications.

Communication

L'absence de communication est certainement l'un des défauts les plus graves qui mettent en péril un projet. Diverses pratiques XP tendent à rendre la communication

omniprésente entre tous les intervenants : entre développeurs (programmation en binôme), entre développeurs et managers (tests, estimations), entre développeurs et clients (tests, spécifications).

Toutes ces pratiques qui forcent à communiquer ont pour but de permettre à chacun de se poser les bonnes questions et de partager l'information.

Simplicité

Cette valeur de simplicité repose sur le pari qu'il coûte moins cher de développer un système simple aujourd'hui quitte à devoir engager de nouveaux frais plus tard pour rajouter des fonctionnalités supplémentaires plutôt que de concevoir dès le départ un système très compliqué dont on risque de n'avoir plus besoin dans un avenir proche. XP encourage donc à toujours s'orienter vers la solution la plus simple qui puisse satisfaire les besoins du client.

Feedback

Le retour est immédiat pour les développeurs grâce aux tests unitaires. Pour les clients, le retour se fait à l'échelle de quelques jours grâce aux tests fonctionnels qui leur permettent d'avoir une vision permanente de l'état du système. Un feedback permanent est positif pour le client qui a une bonne vision du projet, peut détecter tout écart par rapport au planning et à ses attentes de manière à les corriger rapidement. Pour les développeurs, un feedback permanent permet de repérer et de corriger les erreurs beaucoup plus facilement.

Cette notion de feedback est indispensable pour que le projet puisse accueillir le changement.

Courage

Du courage est nécessaire aussi bien chez le client que chez les développeurs. Pour mener à bien un projet XP, le client doit avoir le courage de donner un ordre de priorité à ses exigences, de reconnaître que certains de ses besoins ne sont pas toujours très clairs. De son côté, le développeur doit avoir le courage de modifier l'architecture même si le développement est déjà bien avancé, de jeter du code existant et d'accepter qu'il est parfois plus rapide et efficace de réécrire une portion de code à partir de zéro plutôt que de bricoler du code existant.

1.1.3. Les principes de base

Feedback rapide

L'idée de base est issue de la psychologie de l'apprentissage : plus le retour suit de près une action, plus l'apprentissage qui en résulte n'est important.

C'est pour utiliser au mieux cette caractéristique qu'eXtreme Programming préconise des itérations de courte durée et l'implication forte du client. Un retour rapide pour le développeur permet une correction ou un changement de direction plus rapide et un retour

rapide vers le client lui permet de tester en permanence l'adéquation du système à ses besoins et au marché.

Assumer la simplicité

XP recommande de traiter tous les problèmes par la solution la plus simple possible, partant du principe qu'un travail propre, simple et minimal aujourd'hui est facile à améliorer par la suite.

Changements incrémentaux

Une série de petits changements progressifs est toujours bien plus sûre et bien plus efficace qu'un grand chambardement. Ceci est valable à plusieurs niveaux : dans un projet XP, l'architecture et la conception changent petit à petit, de même que le planning et la composition de l'équipe.

Accueillir le changement à bras ouverts

La meilleure stratégie est celle qui préserve le maximum d'options tout en apportant une solution aux problèmes les plus urgents.

Un travail de qualité

Parmi les quatre variables qui définissent un projet : taille, coûts, délais et qualité, la qualité n'est pas vraiment indépendante. Pour la réussite d'un projet, la qualité ne peut qu'être "excellente" si chacun aime le travail qu'il fait.

Apprendre à apprendre

Plutôt que de s'en remettre à des théories ou des idées reçues pour répondre aux questions comme : quelle quantité de tests est nécessaire ? Combien de temps dois-je passer à retravailler et améliorer le code que j'ai écrit ? Etc., mieux vaut apprendre à chacun à apprendre par soi-même en fonction de chaque projet.

Faible investissement au départ

Allouer peu de ressources à un projet au départ force les clients et les développeurs à aller à l'essentiel et à dégager ce qui est réellement prioritaire et porteur de valeur.

Jouer pour gagner

L'état d'esprit dans lequel doit se placer une équipe XP peut être comparée à celui d'une équipe de football ou de basket : jouer pour gagner et non pour éviter de perdre.

Des expériences concrètes

Toute décision abstraite doit être testée. Dans cet ordre d'idée, le résultat d'une séance de conception ne doit pas être un modèle finalisé un peu parachuté, mais une série de solutions évoquées au cours de la séance.

Communication ouverte et honnête

Il est essentiel que chacun puisse dire sans peur qu'une partie de code n'est pas optimale, qu'il a besoin d'aide, etc.

Travailler avec et non contre les instincts de chacun

Les gens aiment réussir, apprendre, interagir avec d'autres, faire partie d'une équipe, avoir le contrôle des choses. Les gens aiment qu'on leur fasse confiance. Les gens aiment faire un travail de qualité et voir leurs applications fonctionner. XP cherche à offrir à chacun des moyens d'exprimer ces qualités au cours du projet

Adaptation aux conditions locales

Adopter XP ne signifie pas prendre à lettre la méthode mais intégrer ses principes pour l'adapter aux conditions particulières de chaque projet.

Mesures honnêtes

La volonté de contrôler le déroulement d'un projet conduit à évaluer, mesurer certains paramètres. Il faut toutefois savoir rester à un niveau de détail pertinent. Par exemple, il vaut mieux dire "cela prendra plus ou moins deux semaines" que de dire "cela prendra 14 176 heures".

1.1.4. Cycle de vie

Ce paragraphe donne une vision du cycle de vie idéal d'un projet XP. Plusieurs phases composent le cycle de vie d'une application.

Exploration

Au cours de cette phase, les développeurs se penchent sur des questions d'ordre technique destinées à explorer les différentes possibilités d'architecture pour le système et à étudier par exemple les limites au niveau des performances présentées par chacune des solutions possibles.

Le client de son côté s'habitue à exprimer ses besoins sous forme de user stories que les développeurs devront estimer en terme de temps de développement.

User Story : Les User stories sont des « histoires »

Avec :

- Un **acteur**
- qui effectue une **action**
- dans un **objectif** donné.

Une User story doit être développée entièrement pendant une itération.

Planning

Nous ne détaillerons pas ici les procédures entrant dans la phase de planning, Le planning de la première release est fait de telle sorte qu'un système pourvu uniquement des fonctionnalités essentielles soit mis en production dans un temps minimum et soit enrichi par la suite.

Itérations jusqu'à la première release

C'est la phase de développement à proprement parler de la première version de l'application. Chaque itération produit un ensemble de fonctionnalités passant avec succès les tests fonctionnels associés. La première itération est dédiée à la mise en place de l'architecture du système.

Ces itérations courtes permettent de détecter rapidement toute déviation par rapport au planning qui a été fait jusqu'à la sortie de la release. En cas de déviation, quelque chose est à revoir, soit au niveau de la méthode, soit au niveau des spécifications, soit au niveau du planning.

Mise en production

Les itérations de cette phase sont encore plus courtes, ceci pour renforcer le feedback. En général, des tests parallèles sont conduits au cours de cette phase et les développeurs procèdent à des réglages affinés pour améliorer les performances (performance tuning). A la fin de cette phase, le système offre toutes les fonctionnalités indispensables et est parfaitement fonctionnel et peut être mis à disposition des utilisateurs.

Maintenance

Il s'agit dans cette phase de continuer à faire fonctionner le système désormais existant et de lui adjoindre les fonctionnalités secondaires qui avaient volontairement été laissées de côté jusque-là. Le développement de nouvelles fonctionnalités sur un système déjà mis en production requiert une prudence accrue de la part des développeurs et cette phase est donc cruciale. A chaque nouvelle release, une nouvelle phase d'exploration rapide doit avoir lieu.

Mort

La fin d'un projet intervient quand le client n'arrive plus à écrire de user stories supplémentaires ce qui signifie que pour lui, tous ses besoins ont été satisfaits ou quand le système n'est plus capable de recevoir de modifications pour satisfaire de nouveaux besoins tout en restant rentable.

Ce cycle se répète tant que le client peut fournir des scénarios à livrer, comme l'explique la figure 1.

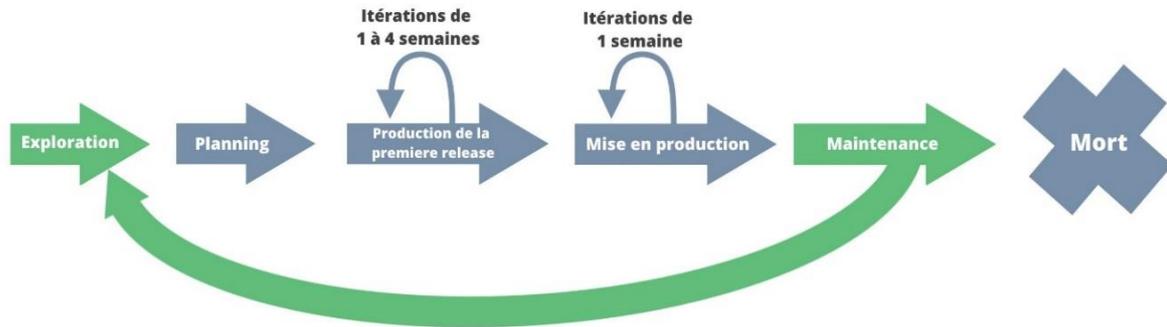


Figure 1 : Les Cycle de vie de « XP ».

L'eXtreme Programming est notamment basé sur les concepts suivants :

- Les équipes de développement travaille directement avec le client sur des cycles très courts.
- Les livraisons de versions du logiciel interviennent très tôt et à une fréquence élevée pour maximiser l'impact des retours utilisateurs.
- L'équipe de développement travaille en collaboration totale sur la base de binômes.
- Le code est testé et nettoyé tout au long du processus de développement.
- Des indicateurs permettent de mesure l'avancement du projet afin de permettre de mettre à jour le plan de développement.

Annexe 2

UML et diagrammes utilisés

1. Présentation générale d'UML

Le génie logiciel et la méthodologie s'efforcent de couvrir tous les aspects de la vie du logiciel. Issus de l'expérience des développeurs, concepteurs et chefs de projets, ils sont en constante évolution, parallèlement à l'évolution des techniques informatiques et du savoir-faire des équipes.

Comme toutes les tentatives de mise à plat d'une expérience et d'un savoir-faire, les méthodologies ont parfois souffert d'une formalisation excessive, imposant aux développeurs des contraintes parfois contre-productives sur leur façon de travailler. Avec la mise en commun de l'expérience et la maturation des savoir-faire, on voit se développer à présent des méthodes de travail à la fois plus proches de la pratique réelle des experts et moins contraignantes.

UML, qui se veut un instrument de capitalisation des savoir-faire puisqu'il propose un langage qui soit commun à tous les experts du logiciel, va dans le sens de cet assouplissement des contraintes méthodologiques.

UML (Unified Modeling Language) signifie «langage unifié de modélisation». C'est un langage formel, normalisé et un support de communication performant qui permet grâce à sa représentation graphique, de concevoir des solutions, de faciliter la comparaison et l'évolution de celles-ci. Son caractère polyvalent et sa souplesse ont en fait un langage de modélisation universel. C'est un langage visuel dédié à la spécification, la construction et la documentation d'un système. L'OMG1 est actuellement responsable de documenter et de faire évoluer ce langage. UML intègre dans un même langage graphique des formalismes et des notations empruntés des méthodes de développement des systèmes qui ont fait leur preuve dont notamment les méthodes dites orientées objet.

1.1. Les diagrammes de l'UML

UML propose treize diagrammes complémentaires qui permettent la modélisation d'un projet tout au long de son cycle de vie, Chaque diagramme étant dédié à la représentation des concepts particuliers d'un système logiciel. Les treize types de diagrammes UML, sont répartis en deux catégories :

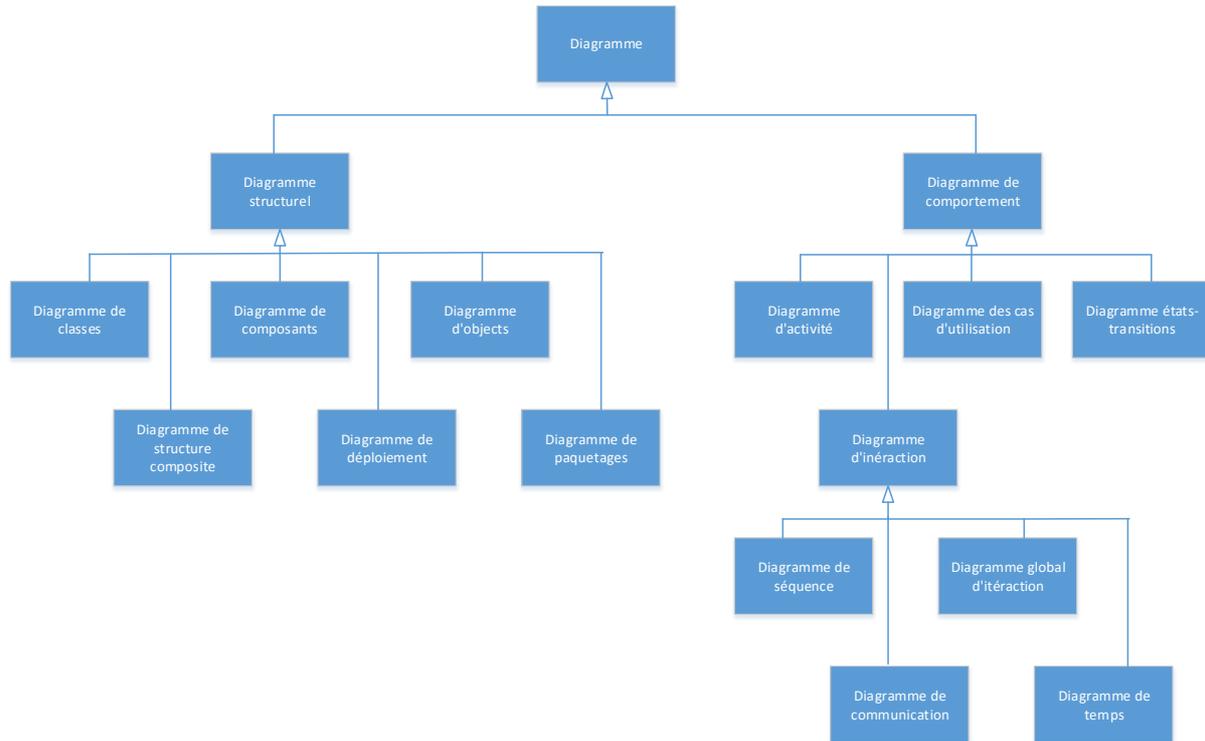


FIGURE 2 : Les 13 types de diagrammes de l'UML 2.0

Diagrammes structurels : Les six diagrammes structurels se présentent comme suit :

- **Diagramme de classes :** identifie la structure des classes d'un système y compris les propriétés et les méthodes de chaque classe.
- **Diagramme d'objets :** montre les instances des éléments structurels et leurs liens à l'exécution.
- **Diagramme de packages :** spécifie l'organisation logique du modèle et les relations entre packages.
- **Diagramme de structure composite :** désigne l'organisation interne d'un élément statique complexe.
- **Diagramme de composants :** indique des structures complexes avec leurs interfaces fournies et requises.
- **Diagramme de déploiement :** définit le déploiement physique des « artefacts » (Objet) sur les ressources matérielles.

Diagrammes comportementaux :

- **Diagramme de cas d'utilisation (uses case diagram) :** permet d'identifier les possibilités d'interactions entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire toutes les fonctionnalités que doit fournir le système.

- **Diagramme de vue d'ensemble des interactions** : fusionne les diagrammes d'activité et de séquence pour combiner des fragments d'interaction avec des décisions et des flots.
- **Diagramme de séquence** : montre la séquence verticale des messages passés entre objets au sein d'une interaction.
- **Diagramme de communication** : désigne la communication entre objets dans le plan au sein d'une interaction.
- **Diagramme de temps** : fusionne les diagrammes d'états et de séquences pour montrer l'évolution de l'état d'un objet au cours du temps.
- **Diagramme d'activité** : montre l'enchaînement des actions et décisions au sein d'une activité.
- **Diagramme d'états-transitions** : permet de décrire sous forme de machines à états finis, le comportement du système ou de ses composants.

1.2. Règles de passage du diagramme de classes au modèle logique

Cette section présente les règles permettant de décrire un schéma logique dans les modèles relationnel et objet-relationnel à partir d'un schéma entité-association ou à partir d'un diagramme de classes UML.

- **Transformation des entités/classes**

Chaque entité devient une relation. L'identifiant de l'entité devient clé primaire pour la relation. Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle d'identifiant. Si aucun attribut ne convient en tant qu'identifiant, il faut en ajouter un de telle sorte que la relation dispose d'une clé primaire (les outils proposent l'ajout de tels attributs). On note cette règle par « R1 ».

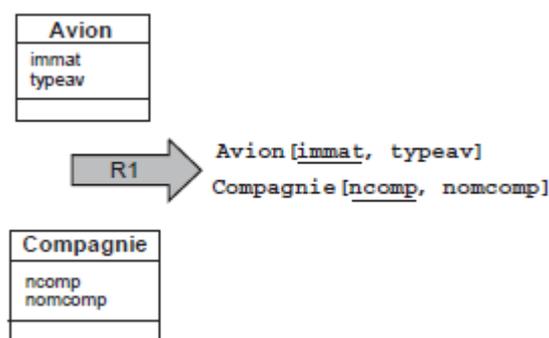


FIGURE 3 : Transformation d'entités/classes.

Remarque :

La règle R1 a été appliquée à l'exemple «FIGURE 3 » de manière à dériver deux relations.

• Transformation des associations

Les règles de transformation que nous allons voir dépendent des cardinalités/multiplicités maximales des associations. Nous distinguons trois familles d'associations :

- un-à-plusieurs.
- plusieurs-à-plusieurs ou classes-associations, et n-aires.
- un-à-un.

1. Associations un-à-plusieurs

Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association. On note cette règle par « R2 ».

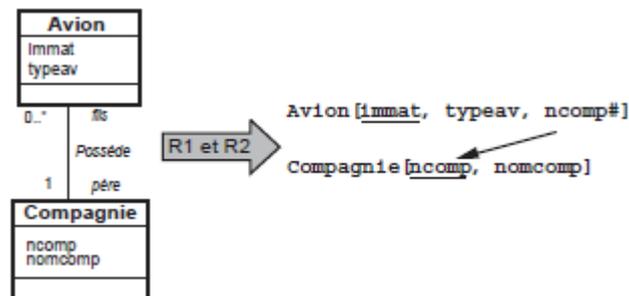


FIGURE 4 : Transformation d'une association un-à-plusieurs.

Remarque :

Les règles R1 et R2 ont été appliquées à l'exemple « FIGURE 4 ». La règle R2 fait apparaître la clé étrangère ncomp dans la relation fils Avion qui a migré de la relation père Compagnie.

2. Associations plusieurs-à-plusieurs

L'association (classe-association) devient une relation dont la clé primaire est composée par la concaténation des identifiants des entités (classes) connectés à l'association. Chaque attribut devient clé étrangère.

Les attributs de l'association (classe-association) doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère. On note cette règle par « R3 ».

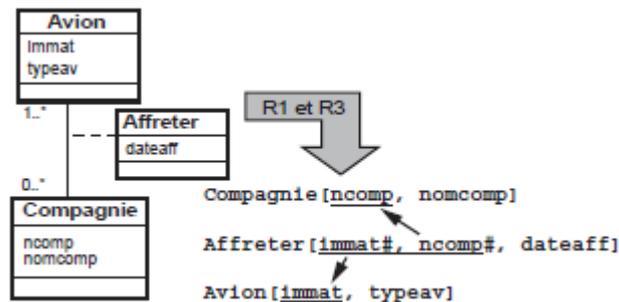


FIGURE 5 : Transformation d'une association plusieurs-à-plusieurs.

Remarque :

Les règles R1 et R3 ont été appliquées à l'exemple « **FIGURE 5** ». La règle R3 crée la relation Affreter dont la clé primaire est composée de deux clés étrangères. L'attribut dateaff de l'association est ajouté à la nouvelle relation.

3. Associations un-à-un

Il faut ajouter un attribut clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas de UML, il faut ajouter un attribut clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de l'entité (classe) connectée à l'association. Si les deux cardinalités (multiplicités) minimales sont à zéro, le choix est donné entre les deux relations dérivées de la règle R1. Si les deux cardinalités minimales sont à un, il est sans doute préférable de fusionner les deux entités (classes) en une seule. On note cette règle par « R4 ».

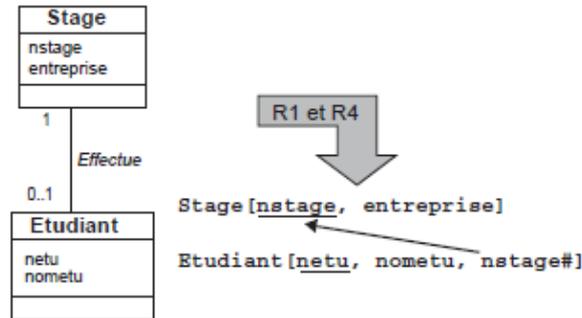


FIGURE 6 : Transformation d'une association un-à-un.

Remarque :

Les règles R1 et R4 sont appliquées à l'exemple « FIGURE 6 ».

- **Transformation de l'héritage**

Trois décompositions sont possibles pour traduire une association d'héritage en fonction des contraintes existantes :

- décomposition par distinction.
- décomposition descendante (push-down). S'il existe une contrainte de totalité ou de partition sur l'association d'héritage, il y aura deux cas possibles de décomposition.
- décomposition ascendante (push-up).

Décomposition par distinction

Il faut transformer chaque sous-classe en une relation. La clé primaire de la sur-classe migre dans la (les) relation(s) issue(s) de la (des) sous-classe(s) et devient à la fois clé primaire et clé étrangère.

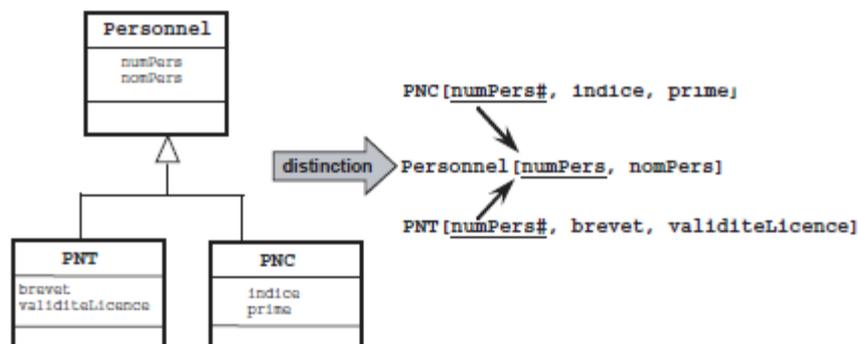


FIGURE 7 : Décomposition par distinction d'une association d'héritage.

Remarque :

L'association d'héritage est traduite en faisant migrer l'identifiant de la sur-classe (Personnel) dans les deux relations déduites des sous-classes. Cet attribut devient aussi clé étrangère.

Décomposition descendante (push-down)

S'il existe une contrainte de totalité ou de partition sur l'association, il est possible de ne pas traduire la relation issue de la sur-classe. Il faut alors faire migrer tous ses attributs dans la (les) relation(s) issue(s) de la (des) sous-classe(s).

Dans le cas contraire, il faut faire migrer tous ses attributs dans la ou les relation(s) issue(s) de la (des) sous-classe(s) dans la (les) relation(s) issue(s) de la (des) sous-classe(s).

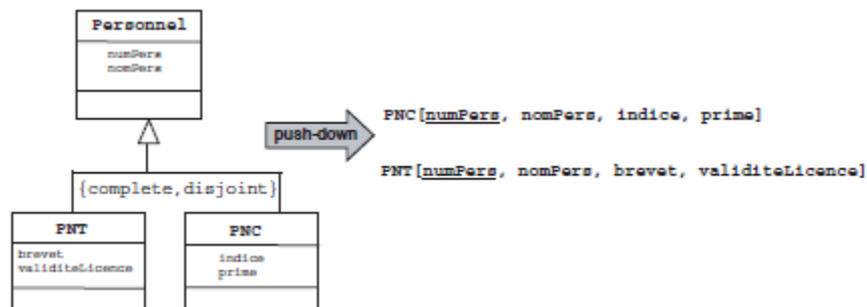


FIGURE 8 : Décomposition descendante (push-down) d'une association d'héritage.

Remarque :

L'exemple « **FIGURE 8** » décrit une contrainte de partition dans l'association d'héritage (aucun personnel ne peut être à la fois PNT et PNC et il n'existe pas un personnel n'étant ni PNT ni PNC). Les deux relations héritent du contenu intégral de la relation issue de la sur-classe (Personnel). La relation Personnel n'apparaît plus au niveau logique et n'est pas nécessaire, car aucun personnel ne peut être ni PNT ni PNC. Ici, on peut dire que Personnel est une entité abstraite (il n'existera pas d'instance de cette entité).

Décomposition ascendante (push-up)

Il faut supprimer la (les) relation(s) issue(s) de la (des) sous-classe(s) et faire migrer les attributs dans la relation issue de la sur-classe.

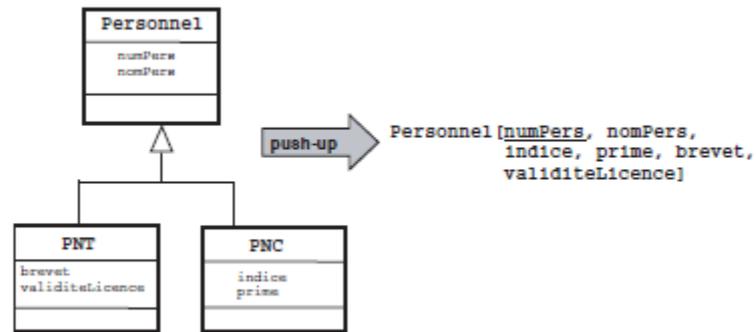


FIGURE 9 : Décomposition ascendante (push-up) d'une association d'héritage.

Remarque :

L'exemple « FIGURE 9 » décrit une association d'héritage UML sans contrainte. En appliquant le principe de décomposition ascendante, nous obtenons une relation issue de la surclasse dans laquelle se trouve le contenu des relations issues des sous-classes.

Le langage UML nous apporte une aide à toutes les étapes de notre projet, comme il nous offre aussi de nombreux avantages pour l'analyse et la conception de notre système.

Résumé

Ce mémoire a pour principal objectif la gestion des représentants, des produits, et le système de paiement d'une entreprise de MLM « New Life ». Pour cela, nous avons défini le marketing de réseaux ainsi que le fonctionnement du système de l'entreprise. Pour concevoir notre application, nous avons utilisé UML comme langage de modélisation et « eXtreme programming » (méthode agile) comme démarche de développement.

Nous avons déployé et hébergé notre application, afin de la valider, et procédé à un ensemble de tests. Nous avons aussi évalué le fonctionnement de l'application pendant cinq mois selon une étude statistique.

Mots clés : Marketing de réseau, Application web, UML2, démarche agile XP, Wampserver.

Abstract

This memory has as main objective the management representatives , products, and the payment system of MLM company " New Life ". For this, we have defined the network marketing and the operation of the company's system. In designing our application , we used UML as modeling language and " eXtreme programming " (agile methods) as a development approach.

We deployed and hosted our application to validate and process a set of tests. We also assessed the application operation for five months in a statistical study.

Tags: Network Marketing, Web application, UML 2, agile approach XP, Wampserver.