

UVT



BULK SWS

Réalisé par :

ZRAN BILEL

Encadré par :

AOUICHAOUI MOEZ

Année Universitaire : 2010/2011

Dédicaces

Je dédie ce modeste travail

A dieu tout puissant de m'avoir éclairé

A tous ceux qui m'ont soutenu et encouragé

Particulièrement à:

*Mon cher père qui m'a soutenue tout au long de mes études avec beaucoup
de patience, d'amour et de prière.*

*Ma chère mère qui n'a cessé de m'entourer avec son amour, ses conseils et
son soutien tout au long de mon parcours.*

Mes chers collègues pour leurs soutiens tout au long de cette année.

A Mes chers amis.

Remerciement

Je remercie pour le bon déroulement de ce stage et l'élaboration de ce rapport tous le staff du centre de service à valeur ajoutée et particulièrement Mr **AOUICHAOUI MOEZ** pour leur sens de l'encadrement et l'intérêt qu'ils ont porté à ce stage.

C'est ainsi qu'à cette occasion je tiens à exprimer ma profonde gratitude à Mr **MANNAI MAKREM** et Mlle **KHABOUCHE BOUTHEINA**, pour toute l'attention qu'ils m'ont bien voulu accorder tout au long de la période de stage et de présenter mes remerciements les plus sincères à tous les agents et les employés de Tunisie Telecom d'EL- KASBA pour leur volonté, la pertinence de leur remarques et la subtilité de leur conseils qui ont été pour moi des atouts importants dans l'élaboration de mon stage dans des bonnes conditions , ainsi, dans la présentation de mon rapport de stage .

Finalement je tiens a remercier l'Université Virtuelle de Tunis qui m'a donnée l'occasion de s'intégrer dans la vie professionnelle durant trois mois. Et j'espère que ce travail sera convainquant et satisfaisant.

Introduction

Générale

Il n'y a aucun doute que l'informatique représente la révolution la plus importante dans le siècle passé et surtout par les technologies de l'information et de la communication (TIC) qui entraînent un changement fondamental dans notre vie quotidienne avec des nouvelles façons de penser, de vivre, de communiquer et d'échanger avec les autres .

Ces technologies deviennent un élément courant dans la vie , comme ils touchent tous les domaines sans exceptions (social , économique, culturel...) grâce à son développement accéléré, à son rôle d'améliorer la productivité et la rentabilité , son pouvoir de perfectionner les échanges concernant l'organisation du travail , et le gain de temps offert, de telle sorte qu'il est devenu difficile de concevoir une procédure administrative au niveau des sociétés et des administrations, sans penser aux méthodes informatisées les plus élaborées capables de répondre à leur besoins .

En effet, dans ce cadre, la société Tunisie Telecom plus précisément le centre de service à valeur ajoutée (centre SVA), m'offre l'opportunité de faire développer un portail SMS destiné au service commercial durant un stage de trois mois de projet de fin d'étude, afin de faciliter le travail et minimiser le temps.

Ce présent rapport s'articule autour de cinq chapitres .Le premier s'intéresse de la **présentation général** du projet qui décrit l'organisme d'accueil ainsi une courte description du projet, le deuxième chapitre présente la partie **étude de l'existant** qui critique l'existant et décrit les différentes solutions proposées, le troisième chapitre est consacré à **l'Analyse et la Spécification des besoins** qui décrit les acteurs et les fonctionnalités de l'application et s'intéresse au spécification fonctionnel et non fonctionnel aussi la spécification semi formelle. Le quatrième chapitre qui est destiné à la **conception détaillée** dont on décrit les spécifications de l'application d'une manière plus formelle par les diagrammes du langage UML et la conception graphique qui nous donne une idée sur les formes et le design d'interfaces de mon application. Le dernier chapitre s'intéresse aux **environnements de la réalisation** .Ce chapitre démontre les environnements matériels et logiciels de l'application et les différentes interfaces accomplies tout au long de cette application.

Chapitre I :

Cadre général

Du projet

Introduction

Dans ce chapitre, je vais donner une brève description de la société d'accueil qui est Tunisie Telecom et présenter son organigramme. Ensuite, je vais expliquer le sujet de mon projet et ses objectifs. Enfin, je vais poser la problématique et les solutions envisagées.

I. Présentation de l'organisme d'accueil

1. Historique de Tunisie Telecom



Figure 1 : logo de Tunisie Telecom

Tunisie Télécom (اتصالات تونس) est le nom commercial de l'opérateur historique de télécommunications en Tunisie.

La loi portant création de l'Office national des télécommunications, dont le nom commercial est *Tunisie Télécom*, est promulguée le 17 avril 1995 et entre en vigueur le 1^{er} janvier 1996. Devenu société anonyme de droit public fin 2002, il change de statut juridique, par un décret du 5 avril 2004, pour devenir une société anonyme dénommée

« Tunisie Télécom ». Elle connaît une privatisation partielle en juillet 2006 avec l'entrée dans son capital, à hauteur de 35 %, de l'émirati Tec om-DIG.

Tunisie Télécom propose des services dans le domaine des télécommunications fixes et mobiles. En juin 2006, il est fort de 1 259 000 abonnés au réseau fixe (RTCP), dont il détient le monopole, et de 3 265 000 abonnés au réseau GSM (la première ligne est inaugurée le 20 mars 1998, faisant de lui le leader sur ce marché devant l'opérateur privé Tunisiana.

Depuis 2008, Tunisie Télécom offre la possibilité aux détenteurs de cartes bancaires nationales d'alimenter le solde de leurs lignes prépayées via les distributeurs automatiques de billets de l'Arab **Tunisien** Bank (service Mobilink).

Le 21 mars 2009, Tunisie Télécom lance une nouvelle marque, Elissa, avec des offres spécifiquement conçues pour les jeunes de moins de 25 ans. Il est également un fournisseur d'accès à Internet (Frame Relay, ADSL, X.25, LS, RNIS et WLL pour la téléphonie rurale).

Tunisie Télécom met en place, exploite et commercialise le premier réseau GSM en Mauritanie (Mattel) à partir de mai 2000. Elle conclut également une convention de coopération technique avec Djibouti Télécom pour le développement de ses réseaux de télécommunications.

Tunisie Telecom El Kasbah est organisé en plusieurs directions, comme exemple le centre de service à valeur ajoutée où j'ai réalisé ce projet. Le centre de service à valeur ajoutée présente plusieurs travailleurs occupants les postes des Techniciens et des ingénieurs compétents. Aussi, il comprend plusieurs types de serveurs normés, des serveurs de type SUN (SPARC, NITRA....) installer par des systèmes d'exploitation Solaris(UNIX), Redhat (LUNIX) et Windows Server, et de type IBM installer par Windows Server (2003, 2008), AIX (UNIX) et Ubuntu, en plus le type Siemens.

2. Organigramme de l'organisme

Pour concrétiser la mise en place de l'ONT, le conseil d'administration s'est réuni en vue d'établir un organigramme transitoire modélisant la structure interne de l'office.

Cet organigramme fait l'objet de plusieurs reformes et restructurations afin de s'adapter aux nouvelles exigences d'efficience et d'efficacité. En fait, l'organigramme de Tunisie Telecom se présente comme suit :

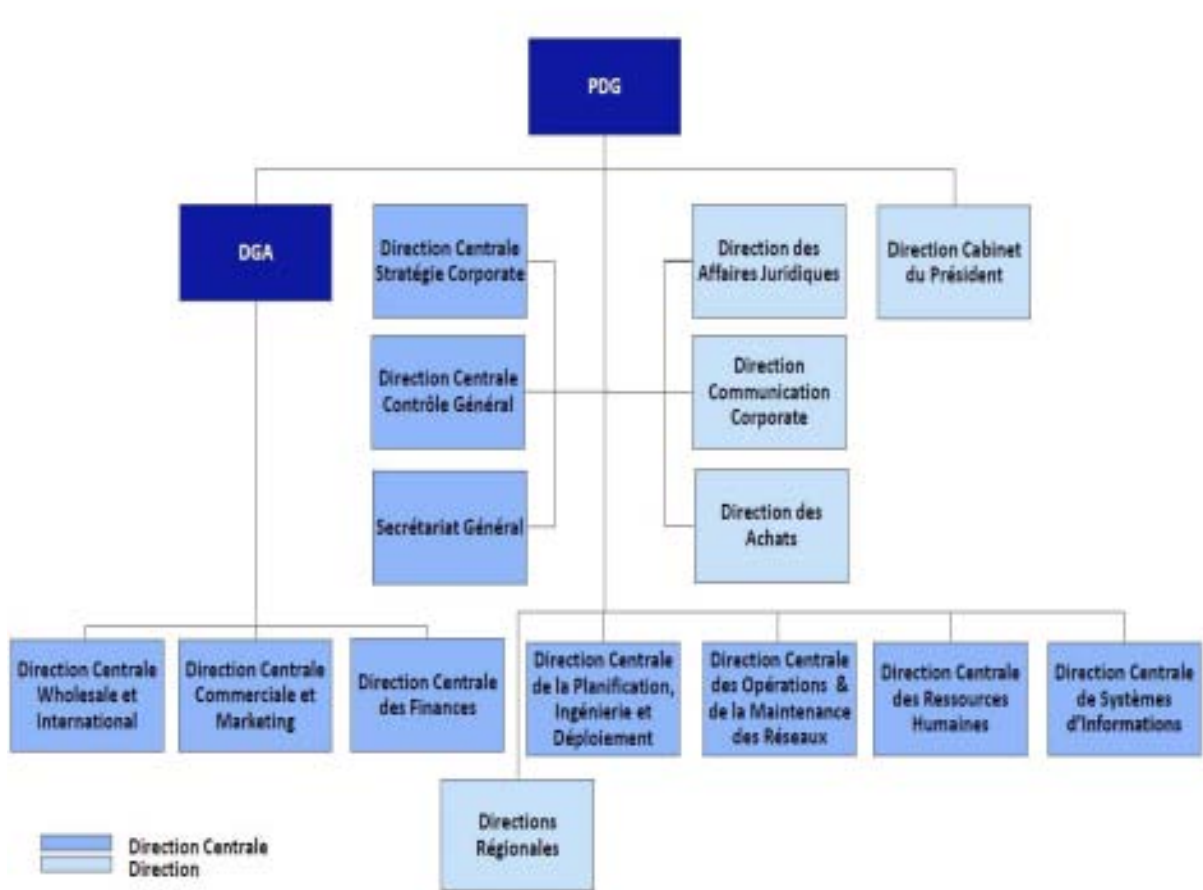


Figure2 : L'organisation Fonctionnelle de Tunisie Telecom

II. Présentation du sujet

Tunisie Telecom est une société qui comporte un nombre important de directions et de services déployés sur l'ensemble du territoire Tunisien .Mais, j'ai consacré à un seul service en l'occurrence le service commercial.

Tunisie Telecom a comme but de gérer les réseaux et les services publics de Télécommunications. Dans ce contexte, elle m'a confié le soin de développer une application qui s'agit de BULK SMS.

Cette application permet d'envoyer des messages de masse promotionnelle ou informationnelle pour la Tunisie Telecom, aussi à travers cette application je pourrais gérer le trafic des messages. Ce système est géré de façon manuelle par la direction informatique représenté par l'ingénieur responsable de la passerelle.

L'envoi des messages manuellement présente plusieurs insuffisances et nécessite une mise à Niveau de ces insuffisances.

1. Problématique

La société Tunisie Telecom opère dans le marché tunisien depuis 1^{er} Janvier 1996 comme Office Nationale de Télécommunications (ONT).Elle est mise en activité en tant qu'opérateur de télécommunication doté de sa propre autonomie financière .Alors, elle a pour principal objectif la modernisation de son infrastructure de télécommunication, la diversification et l'amélioration des services offertes aux abonnés. Donc, elle occupe une place importante sur le marché. Elle utilise ainsi les nouvelles technologies, mais, il existe encore quelques tâches tel que l'envoi de messages en masse qui est encore manuelle, elle manque de simplicité ce qui rend le travail plus lent. Dans ce contexte, j'ai essayé de développer une application pour automatiser cette tâche.

Alors, ce projet est né dans l'objectif de développer une application permettant l'envoi des SMS en bloc. Donc, le but principal à accomplir est de faciliter la tâche au service commercial de développer et gérer leurs campagnes de messages selon leurs obligations commerciales et par la suite, je n'aurais pas besoin d'un ingénieur qu'il doit maîtriser certains langages et avoir certaines compétences pour communiquer l'envoi des messages.

2. Les objectifs

Les objectifs de mon projet sont :

- Administrer l'application d'une manière rapide et facile.
- Gérer les employés d'une façon simple et rapide.
- Envoyer les messages d'une manière plus fiable et mesurable.
- Communiquer instantanément avec les clients d'une manière plus évolutive et puissante.
- La possibilité d'emmètré et de gérer des compagnes de messages sans la peine d'avoir des connaissances aux réseaux informatiques
- La réduction de pourcentage d'erreur car la manipulation et la création des compagnes de messages sera informatisé et sera vérifié dans plusieurs étapes et pas manuel et direct dans la passerelle
- Mon application permettra d'extraire les statistiques

Conclusion

Dans ce chapitre, j'ai étudié le contexte général de mon projet, ainsi que l'ensemble des tâches à réaliser. Pour les accomplir, quelques notions relatives à ce travail doivent être définies. J'ai consacré le chapitre suivant pour illustrer les notions de base nécessaires à l'élaboration de mon projet.

Chapitre II :

Etude de

L'existant

Introduction

Dans cette partie, j'ai détaillé les points faibles dans la société. Puis, j'ai présenté les meilleures solutions de travail et de sécurité proposées.

I. Etude de l'existant

La gestion manuelle des tâches rencontre souvent des difficultés. Pour gérer cet inconvénient, l'outil informatique est devenu absolument nécessaire pour faciliter le travail et surmonter les difficultés affrontées au sein de cette société. Ainsi, cette société a la possibilité de travailler avec d'autres entreprises comme une agence de publicité. Elle contribue également à la promotion de l'usage des TIC et au développement des sociétés innovantes dans le domaine des télécommunications. C'est pour cette raison, on va présenter une application intitulé BULK SMS.

1. Critique de l'existant

L'envoi de messages manuellement présente plusieurs insuffisances et nécessite une mise à niveau de ces insuffisances. Ils se résument dans :

- L'envoi manuel qui nécessite des compétences et des connaissances
- L'indisponibilité à l'employé de gérer et modifier ses campagnes de messages sans la présence de la responsable
- Pas de vérification lors de lancement de campagnes par l'administrateur
- L'envoi des messages fait perdre beaucoup de temps.
- Difficulté d'atteindre beaucoup de gens à un moment unique.
- L'absence de la sécurité rend l'accès facile aux données concernant les campagnes de messages.

2. Solution proposée

Dans le but de faciliter l'envoi des messages, j'ai réalisé une application qui présente une technique de publicité parfait qui peut tirer des nombreux avantages pour l'administrateur ainsi pour les employés .Alors, l'objectif de mon projet est d'essayer de pallier ces insuffisances on offrant des solutions SMS rapides de diffuser, évolutives, mesurables, faciles et puissantes.

Elle contient les fonctionnalités suivantes:

- Accès rapide aux informations concernant les messages et les listes de numéros ; c'est-à-dire que tout utilisateur autorisé peut consulter les informations concernant les campagnes de message à envoyer.
- L'enregistrement des données se fait d'une manière simple.
- La mise à jour de la base de données (Ajout, modification, suppression) : c'est une tâche confiée essentiellement pour l'administrateur. Chaque administrateur doit s'authentifier avant d'accéder pour faire la mise à jour.
- Cette dernière tâche peut être réalisée ainsi par l'employé lorsqu'il a le droit.
- Facilité le développement et la gestion de campagne de message : je n'aurais pas besoin d'un ingénieur ou un agent professionnel qui doit maitriser certains langages et avoir certaines compétences pour communiquer et envoyer des messages.
- Permet au service commercial d'émettre les messages de masse (informationnelle ou promotionnelle) aussi de gérer les paramètres (texte, date début et date fin, source de diffusion et la cible).
- L'envoi des messages à une grande base de données de numéros de téléphone des abonnés : Mon application joue le rôle de passerelle entre réseau IP et réseau GSM pour l'envoi des SMS en gros.

II. La partie sécurité de l'application

Chaque système s'occupe de la partie sécurité ce qui le rend plus sécurisé, l'administrateur a protégé son application à travers un identifiant et un mot de passe et c'est le même pour l'employé. Alors, les données personnelles doivent être sécurisées, par conséquent n'importe quel utilisateur ne peut y accéder sans permission préalable. Cette sécurisation permet d'éliminer l'apparition des sérieux problèmes à tout utilisateur et la protection de toutes les pages de l'application ce qui rend mon application n'est pas exposé au danger.

Conclusion

L'étude de l'existant permet de mieux présenter les différents problèmes au sein de la société Tunisie Telecom ainsi l'étude de la solution proposée.

Chapitre III :

Analyse et

Spécification des

besoins

Introduction

Ce chapitre présente l'analyse du système d'information à développer. J'ai présenté les besoins fonctionnels et non fonctionnels de l'application. Enfin, on trouve la représentation de la spécification formelle des besoins à travers des diagrammes de cas d'utilisation qui seront d'une grande utilité dans le processus de développement de l'application.

I. Spécification des besoins fonctionnels et non fonctionnels

L'analyse du sujet m'a permis de dégager les fonctionnalités qui seront mise à la disposition de l'utilisateur. Dans cette partie, je vais recenser les fonctionnalités que l'application doit offrir à ses différents utilisateurs.

Mon application doit fournir à la fois des fonctionnalités concernant l'administrateur et l'employé.

1. Spécification des besoins fonctionnels

✓ Coté employé

C'est un agent qui suit la transmission des informations qui concerne la clientèle et l'émission des nouvelles promotions.

Le logiciel doit permettre :

- Accès et Identification :
 - Authentification à travers son identifiant et son mot de passe

Il assure :

- Ajout de campagne :
 - Saisir le texte de message,
 - Indiquer la cible clientèle,
 - Indiquer l'adresse cible.

- Tester l'envoi par un numéro choisi,
- Recherche des compagnes validées,
- Gestion des messages :
 - Ajout du message,
 - Modification du message,
 - Suppression du message.
- Importation des listes des adresses
 - > L'employé réalise les taches dédiées à lui.

✓ **Coté administrateur**

L'administrateur est un technicien ou ingénieur responsable au bon fonctionnement de l'application, ainsi il est responsable de la direction commerciale, de tous les ordres issus de cette direction et à tous les agents.

Le logiciel doit permettre l':

- Accès et Identification :
 - Authentification à travers son identifiant et son mot de passe.

Il assure :

- Validation d'envoi,
- Recherche des compagnes validées,
- Consulter les statistiques des compagnes,
- Maintenance de l'application,
- Gestion des comptes des utilisateurs :
 - Ajout, suppression et modification.
- Gestion des groupes :
 - Ajout, suppression et modification des privilèges des groupes enregistrés.

2. Spécification des besoins non fonctionnels

Mon application doit être facile à utiliser, avec une bonne ergonomie. Il doit garantir un temps de réponse court. En plus, il faut que :

- Le système doit fournir une interface accueillante et conviviale,
- Le système doit fournir une interface simple à manipuler,
- Le système doit faciliter la saisie des données,

- Le système doit faciliter la gestion de sauvegarde, des restaurations et la manipulation des données.

De plus l'application doit assurer :

- **La sécurité de fonctionnement** : les données de la société devront être sécurisées pour éviter toute sorte de fraude,
- **Evolution du système** :
 - **Evolution matérielle** : Le système doit satisfaire les besoins de l'utilisateur quelle que soit l'évolution matérielle,
 - **Evolution fonctionnelle** : L'évolution fonctionnelle de l'application peut viser l'architecture globale de l'application en ajoutant des rubriques qui s'adapte à l'application afin de simplifier à l'employé et à l'administrateur la consultation des informations.
- **Maintenance** : On distingue deux sortes de maintenance :
 - **Adaptive** : faire adapter l'application à des changements matériels et logiciels,
 - **Evolutive** : qui améliore la performance de l'application comme par exemple l'ajout d'autres rubriques à l'application.

II. Spécification semi-formelle des besoins

1. Méthode de conception

a. UML

✓ Définition

UML n'est pas une méthode (C'est une description normative des étapes de la modélisation) : ses auteurs ont en effet estimé qu'il n'était pas opportun de définir une méthode en raison de la diversité des cas particuliers. Ils ont préféré se borner à définir un langage graphique qui permet de représenter et de communiquer les divers aspects d'un système d'information aux graphiques qui sont bien sûr associés des textes qui expliquent leur

contenu. UML est donc un métalangage car il fournit les éléments permettant de construire le modèle qui, lui, sera le langage du projet.¹

Pierre-Alain Muller ajoute :

« UML est dans le domaine public, soutenue par le marché : Microsoft, HP, IBM, Oracle... Successeur naturel des méthodes de Booch, OMT et OOSE, UML est le fruit de l'expérience et des besoins de la communauté des utilisateurs. »

✓ Les avantages de l'UML

UML est un langage formel et normalisé. Il permet ainsi : Un gain de précision, un gage de stabilité et l'utilisation d'outils.

UML est un support de communication performant : Il cadre l'analyse et facilite la compréhension de représentations abstraites complexes. Son caractère polyvalent et sa souplesse lui font un langage universel.

Objectif :

- Construire des modèles de systèmes,
- Organiser le travail,
- Gérer le cycle de vie de A à Z,
- Gérer le risque,
- Obtenir de manière répétitive des produits de qualité constante.

En conclusion, j'ai choisi de travailler avec UML parce qu'il exprime mieux la vue statique et dynamique du système d'information et pour mon application web, il est nécessaire de faire une analyse très approfondie pour pouvoir dégager les nécessités de développement ainsi que quelques scénarios d'exécution.

Maintenant dans une conception d'une application web, les diagrammes de classes obtenus doivent être encore modifiés selon la technologie (.net, java/j2ee) ou langage (PHP) utilisée ce qui rend le passage à la génération de code moins évidente...

III. Diagrammes de cas d'utilisation

1. Identification des acteurs et des cas d'utilisation

Dans cette partie, je vais identifier les principaux acteurs du système ainsi que leurs cas d'utilisation.

a. Les acteurs

Un acteur représente l'abstraction d'un rôle joué par des entités externes (Utilisateurs, dispositif matériel ou autre système) qui interagissent directement avec le système étudié.

Un acteur peut consulter et/ou modifier l'état du système, et en émettant et/ou en recevant des messages éventuellement porteurs de données.

Dans cette application je présenterais deux catégories d'acteurs : l'employé et l'administrateur. Dans la suite, on va décrire chaque acteur :

- **L'employé** : il a la possibilité d'accéder aux données enregistrées concernant son groupe. Mais, il ne peut faire que les tâches dédiées à lui.
- **L'administrateur** : il intervient dans la mise à jour de la base de données : il a la possibilité de consulter, modifier et supprimer les données concernant la compagnie à envoyer. Il gère aussi les comptes des utilisateurs. Chaque administrateur possède un identifiant et un mot de passe.

b. Les cas d'utilisation

Un cas d'utilisation (en anglais *use case*) permet de mettre en évidence les relations fonctionnelles entre les acteurs et le système étudié. Le format de représentation d'un cas d'utilisation est un complètement libre, mais UML propose un formalisme et des concepts issus de bonnes pratiques.

Le diagramme de cas d'utilisation permet de représenter visuellement une séquence d'actions réalisées par un système, représenté par une boîte rectangulaire, produisant un résultat sur un acteur, appelé *acteur principal*, et ceci indépendamment de son fonctionnement interne.

A chaque acteur dégagé, correspond une multitude d'actions avec le système, ce qui exige des diagrammes de cas d'utilisation bien détaillés limitant ainsi les actions de chaque acteur.

✓ ***Gestion de la campagne :***

- ***La création de la campagne :***

L'utilisateur peut créer une campagne des messages promotionnels ou informationnels à envoyer dont il saisit le texte du message, la liste des clientèles cibles, l'adresse source, la date début /fin et l'heure début/fin d'envoi de la campagne.

- ***La modification de la campagne :***

Cette tâche concerne la modification dans l'une ou toute les informations saisies lors de la création de la campagne (texte, liste d'adresses cible, adresse source, date début, date fin, heure début, heure fin).

- ***La suppression de la campagne :***

Cette tâche consiste à la suppression de la campagne. Cette tâche se fait par l'administrateur, il est le premier responsable des messages à diffuser.

✓ ***Gestion des messages :***

Cette tâche est assurée par l'employé, il peut ajouter, supprimer ou modifier un message.

✓ **Maintenance :**

Cette tâche est dédiée à l'administrateur dont il gère la planification, l'organisation et la réalisation de l'application.

✓ **La validation d'envoi de campagne :**

Cette tâche est conçue seulement à l'administrateur. Elle consiste à la validation de la campagne saisie et l'annulation des campagnes en cours d'envoi.

✓ **La recherche des campagnes validées:**

Tous les utilisateurs ont le droit de consulter les campagnes enregistrées.

✓ **Consulter les statistiques :**

Tâche conçue pour l'administrateur afin de récupérer les taux qui concernent les nombres des messages envoyés selon une date début et une date fin.

✓ **La gestion des comptes de tous les utilisateurs de l'application :**

Ajouter, modifier ou bloquer les droits et les champs conçus à chaque utilisateur.

✓ **La gestion des groupes :**

Ajouter, modifier ou bloquer les droits d'accès d'un groupe précis.

✓ **L'importation des listes des adresses :**

L'administrateur doit importer une liste des numéros de téléphone comme étant les adresses destination de la campagne.

c. Éléments constitutifs des cas d'utilisation

Six éléments composant ce diagramme : les systèmes, les acteurs, les cas d'utilisation, les associations, les dépendances et les généralisations

- **Système** : cet élément fixe les limites du système en relations avec les acteurs qui utilisent (en dehors du système) et les fonctions qu'il doit fournir (à l'intérieur du système).
- **Acteur** : un acteur est un rôle joué par une personne un système ou un élément qui intervient dans le fonctionnement normal du système.
- **Cas d'utilisation** identifie une fonction clé du système, sans cette fonction, le système ne satisfait pas les besoins des utilisateurs /acteurs. Chaque cas d'utilisation représente une fonction que le système doit être capable d'accomplir.
- **Association** : identifie une interaction entre les acteurs et les cas d'utilisation. Chaque association devient un dialogue qui doit être expliqué dans une description narrative. Chaque description fournit un ensemble de scénarios qui jouent le rôle de tests lors de l'évolution de l'analyse, de la conception et de l'implémentation du cas d'utilisation.
- **Dépendance** : identifie une relation de communication entre deux cas d'utilisation.
- **Généralisation** : définit une relation entre acteurs ou deux cas d'utilisation lorsqu'un des cas hérite de l'autre et modifie ses propriétés ou en ajoute de nouvelle.

2. Les diagrammes des cas d'utilisation

a. Diagramme des cas d'utilisation de l'administrateur

Ce diagramme donne à l'administrateur la possibilité de créer des compagnes et de les envoyer par la suite après l'importation d'une liste des adresses. En plus, il gère les comptes utilisateurs ainsi que la recherche des compagnes validées. Chaque administrateur doit s'authentifier avant d'accéder à l'application.



Figure 3 : Diagramme des cas d'utilisation de l'administrateur

b. Diagramme des cas d'utilisation de l'employé

Ce diagramme donne à chaque employé la possibilité de réaliser les tâches concernant son groupe et de les enregistrer et plus précisément la gestion des compagnes et la gestion des messages. En plus, il a la possibilité de chercher les compagnes validées. Chaque employé doit s'authentifier avant d'accéder à son groupe.

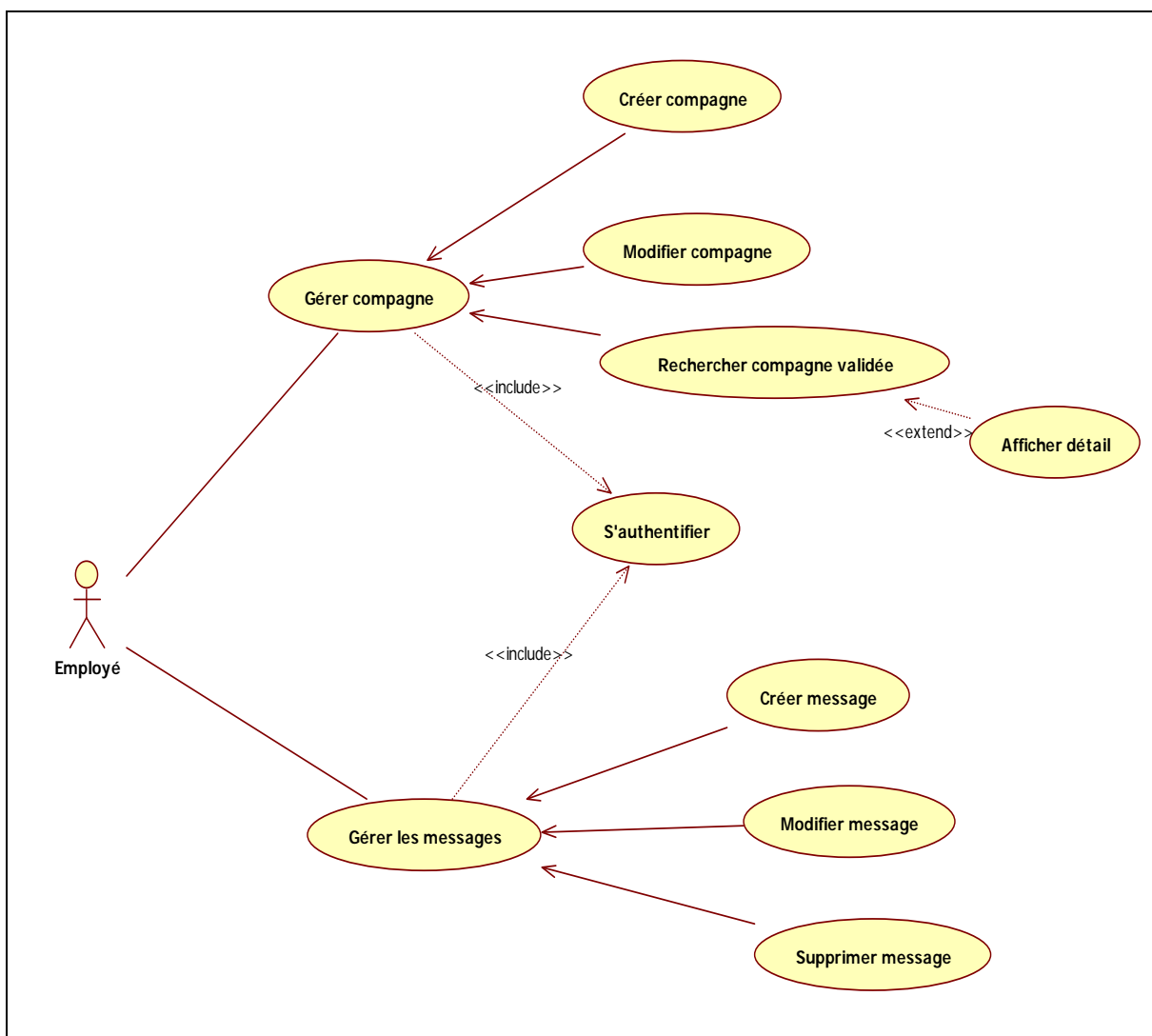


Figure 4 : Diagramme des cas d'utilisation d'employé

c. Diagramme des cas d'utilisation global de système

Un cas d'utilisation est une manière spécifique d'utiliser un système. C'est l'image d'une fonctionnalité du système, déclenchée en réponse à la stimulation d'un acteur externe. Les cas d'utilisation apportent une solution au problème de la détermination et de la compréhension des besoins.



Figure 5 : Diagramme des cas d'utilisation global

Dans ce diagramme, j'ai décrit les différentes fonctions effectuées par l'administrateur et l'employé.

L'administrateur tout d'abord s'identifie, et par la suite, il a le droit de faire la gestion des employés (ajouter un employé, modifier un employé, supprimer un employé), la gestion des messages (ajouter un message, modifier un message, supprimer un message) et enfin la tâche la plus importante c'est l'envoi des campagnes. Il s'occupe aussi de la partie maintenance.

L'employé a le droit de saisir et enregistrer les données, ainsi que les tâches consacrées à son groupe.

Conclusion

Ce chapitre précise les besoins fonctionnels que l'application développée doit offrir aux utilisateurs et les besoins non fonctionnels après l'identification des acteurs ainsi que les cas d'utilisations de chaque acteur.

Chapitre IV :

Conception

Introduction

Ce chapitre présente la partie conception de mon projet. Vu que mon application se présente à la fois comme un système gérant des fonctionnalités complexes d'une part, et comme un produit multimédia d'autre part, je vais présenter la conception en suivant trois grandes parties à savoir : conception générale, conception graphique et la conception détaillée de l'application.

I. Conception de l'application

La conception est un processus créatif, c'est la phase la plus importante dans le cycle de développement d'un projet. Le support de cette phase par des techniques et des outils appropriés est important pour produire une application de haute qualité et pour minimiser l'effort de développement.

1. Modélisation conceptuelle

Dans cette partie, je vais présenter le résultat de l'étude que j'ai fait. On va étudier les besoins de mon application qui sont déjà précisés dans le chapitre précédent en se basant sur les diagrammes d'UML : Je vais présenter en premier lieu la conception de la base de données sous forme d'un diagramme de classes et en deuxième lieu la conception des traitements sous forme des diagrammes de séquence.

a. Conception de la base de données en se basant sur le langage UML

Dans cette partie, je vais modéliser la base de données sous forme d'un diagramme de classe.

- **Description des différentes classes**

Le diagramme de classe contient plusieurs classes :

- **Classe Personne** : contient les coordonnées de l'utilisateur .Il est caractérisé par Matricule, nom et prénom. Il a le droit de consulter l'application,
- **Classe Doc** : présente l'identifiant de document et nom de document de la liste d'adresses des abonnés à parcourir,
- **Classe Groupe** : il est caractérisé par l'identifiant de groupe et nom de groupe, il permet de classer les groupes des utilisateurs de l'application,
- **Classe Message** : il est caractérisé par le titre de message et un texte,
- **Classe Compagne** : il est caractérisé par l'identifiant de compagne, nom de compagne, l'adresse source et l'adresse destination,
- **Classe Authentification**: il contient l'identifiant et le mot de passe de l'utilisateur,
- **Classe Periode** : il est caractérisé par l'heure de début, l'heure de fin, la date de début et la date de fin d'envoi de la compagne.

- Le diagramme de classe global de système

Le diagramme de classes constitue un élément très important de la modélisation : il permet de définir les composantes du système final. Néanmoins, on constate souvent qu'un diagramme de classes proprement réalisé permet de structurer le travail de développement de manière très efficace; il permet aussi, dans le cas des travaux réalisés en groupe (ce qui est pratiquement toujours le cas dans les milieux industriels), de séparer les composantes de manière à pouvoir répartir le travail de développement entre les membres du groupe

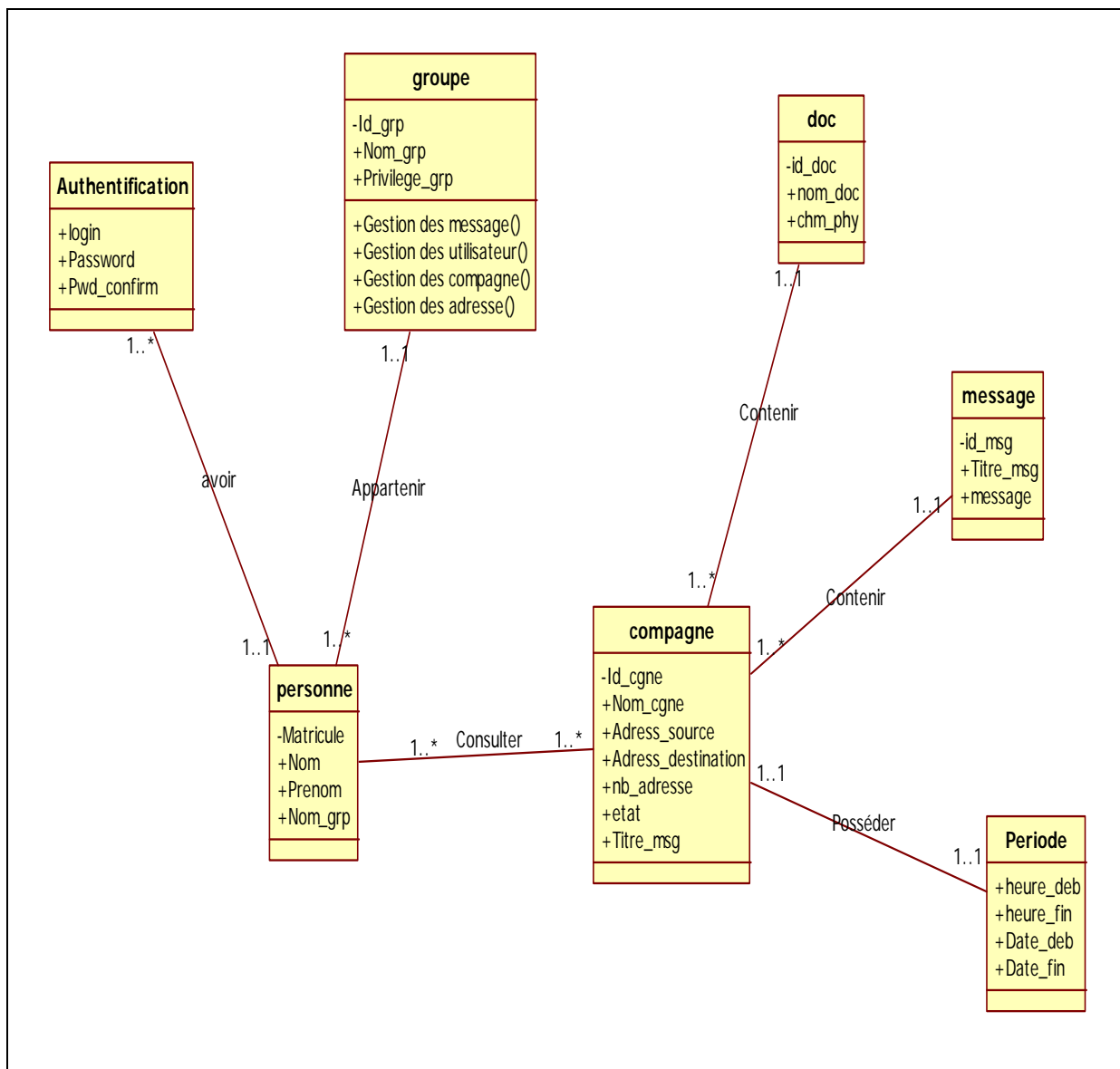


Figure 6 : Diagramme de classe global

- **Les règles de passage du diagramme de classe vers le modèle relationnel**

Le passage du diagramme de classe à la base de données nécessite le suivi de ces règles :

1. classe A <-> classe B avec cardinalité 1, 1 de deux coté

Dans ce cas, a priori, il vaut mieux regrouper le contenu de A et B dans une même table

2. classe A <-> classe B avec cardinalité 1, n de deux coté

C'est-à-dire qu'une entité 'a' peut avoir plusieurs entités 'b' qui lui sont attachées. Dans ce cas, vous allez créer 2 tables --> table A et table B. Dans la table B, tu prévois un attribut supplémentaire contenant l'ID de l'élément A auquel il est rattaché...

3. classe A <-> classe B avec cardinalité n, n de deux coté

Dans ce cas, il faut créer 3 tables :

Une table A, une table B, et une troisième table qui donne les "liens" entre les éléments a et les éléments b : cette table est constituée de 2 colonnes : l'une présente l'ID de la table A, et l'autre l'ID de la table B.

Tout ceci est très global, il y a des nuances en pratique.²

- Le diagramme de classe final de système

Après la suivie des règles de passage du diagramme de classe vers le modèle relationnel, j'ai obtenu ce diagramme de classe :

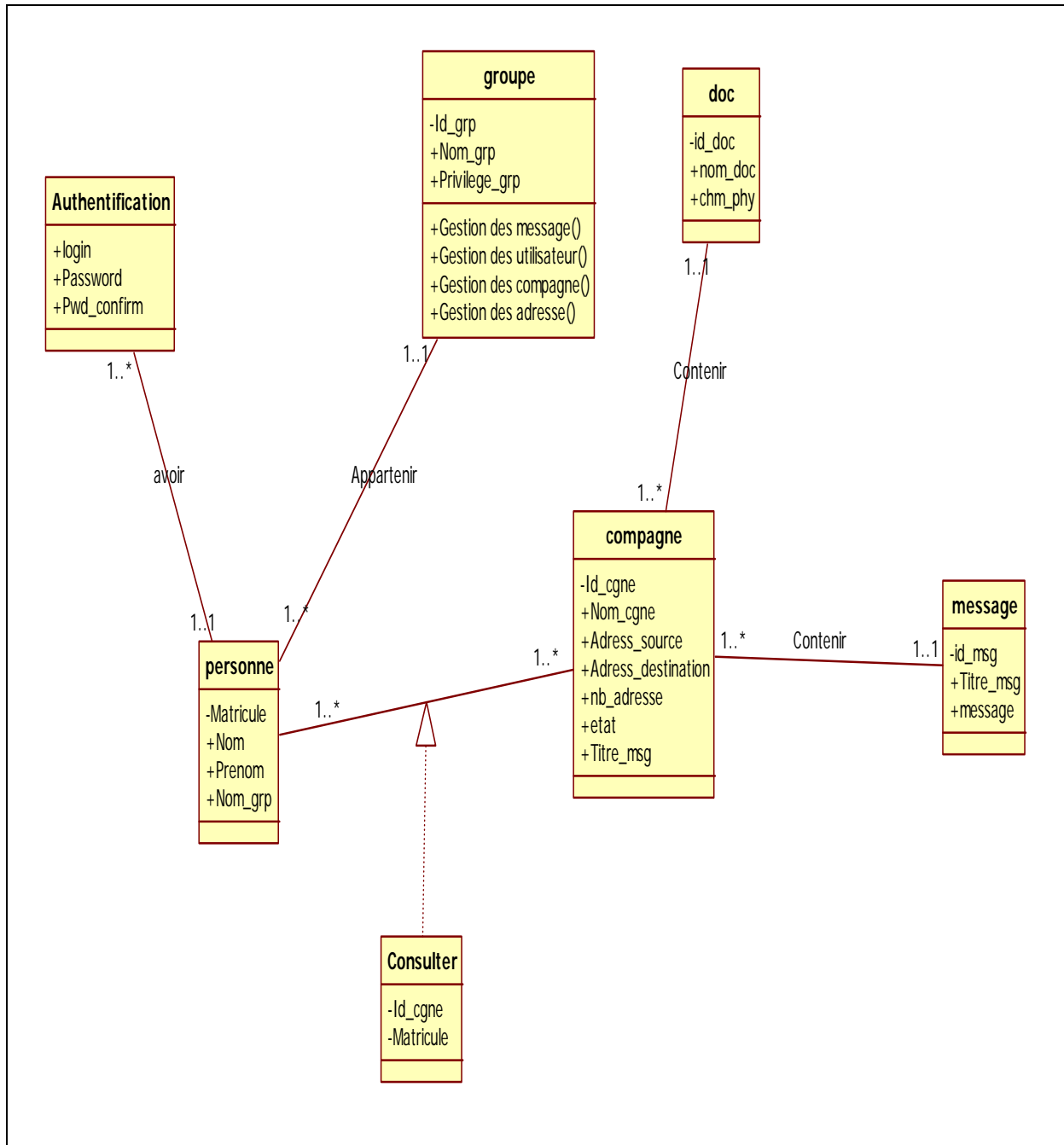


Figure 7 : Diagramme de classe final

- **Le modèle relationnel**

Le modèle relationnel s'agit de la traduction des classes et des associations du diagramme de classe en tables :

Message (id_msg, Titre_msg, message)

Compagne (id_cgne, nom_cgne, Adress_source, Adress destination, Date_deb, Date_fin, Heure_deb, Heure_fin, nb_adresse, etat, Titre_msg#)

Groupe (Id_grp, Nom_grp, Privilege_grp)

Personne (Matricule, Nom, Prenom, Login, Password, Pwd_confirm, Nom_grp#)

Doc (id_doc, nom_doc, chm_phy)

Consulter (Id_cgne#, Matricule#)

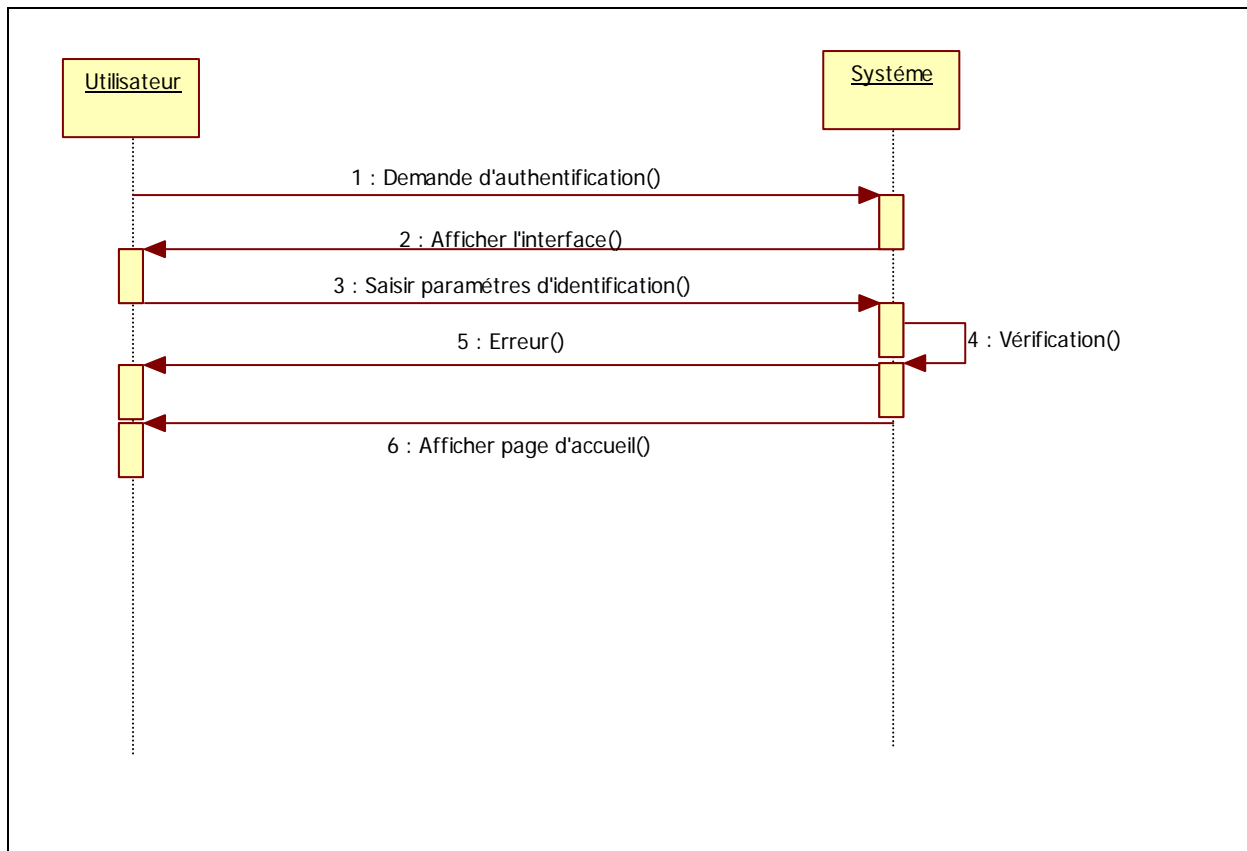
b. Conception des traitements

Dans cette partie, je vais parler des différents diagrammes de séquence de mon application.

- **Le diagramme de séquence**

Un diagramme de séquence est une forme de diagramme comportemental qui me permet de spécifier les interactions qui existent entre un groupe d'objets. Même si d'autres diagrammes comportementaux peuvent convenir, les diagrammes de séquence sont les plus utilisés, principalement parce qu'ils permettent de voir comment les objets s'utilisent mutuellement. Grâce à ces informations, vous pouvez déterminer plus précisément pourquoi deux objets sont liés. Comme les diagrammes de séquence sont toujours lus du haut vers le bas, ils illustrent l'ordre dans lequel les messages sont envoyés entre les objets. Il est tout à fait possible de supprimer certains messages qui participent au présent flux d'événements, si ces messages ne sont pas pertinents dans le contexte du diagramme de séquence. Bien que les diagrammes complexes puissent impressionner, ils apportent peu en matière de communication. Les diagrammes de séquence doivent rester aussi simples que possible et seuls les messages pertinents doivent être représentés.

✓ Diagramme de séquence d'authentification

**Figure 8:** Diagramme de séquence de l'authentification

✓ Diagramme de séquence de l'administrateur

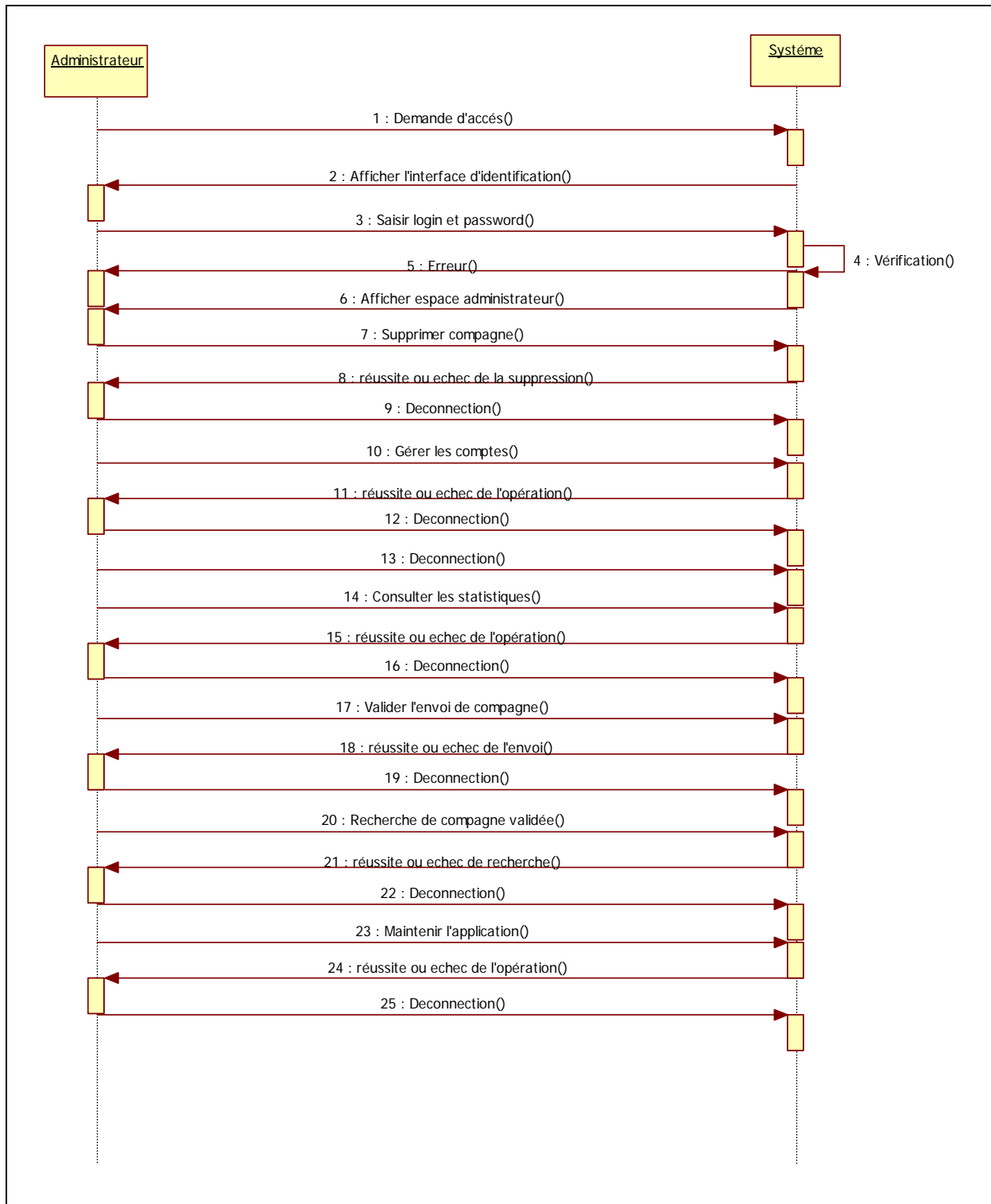


Figure 9: Diagramme de séquence de l'administrateur

✓ Diagramme de séquence de l'employé

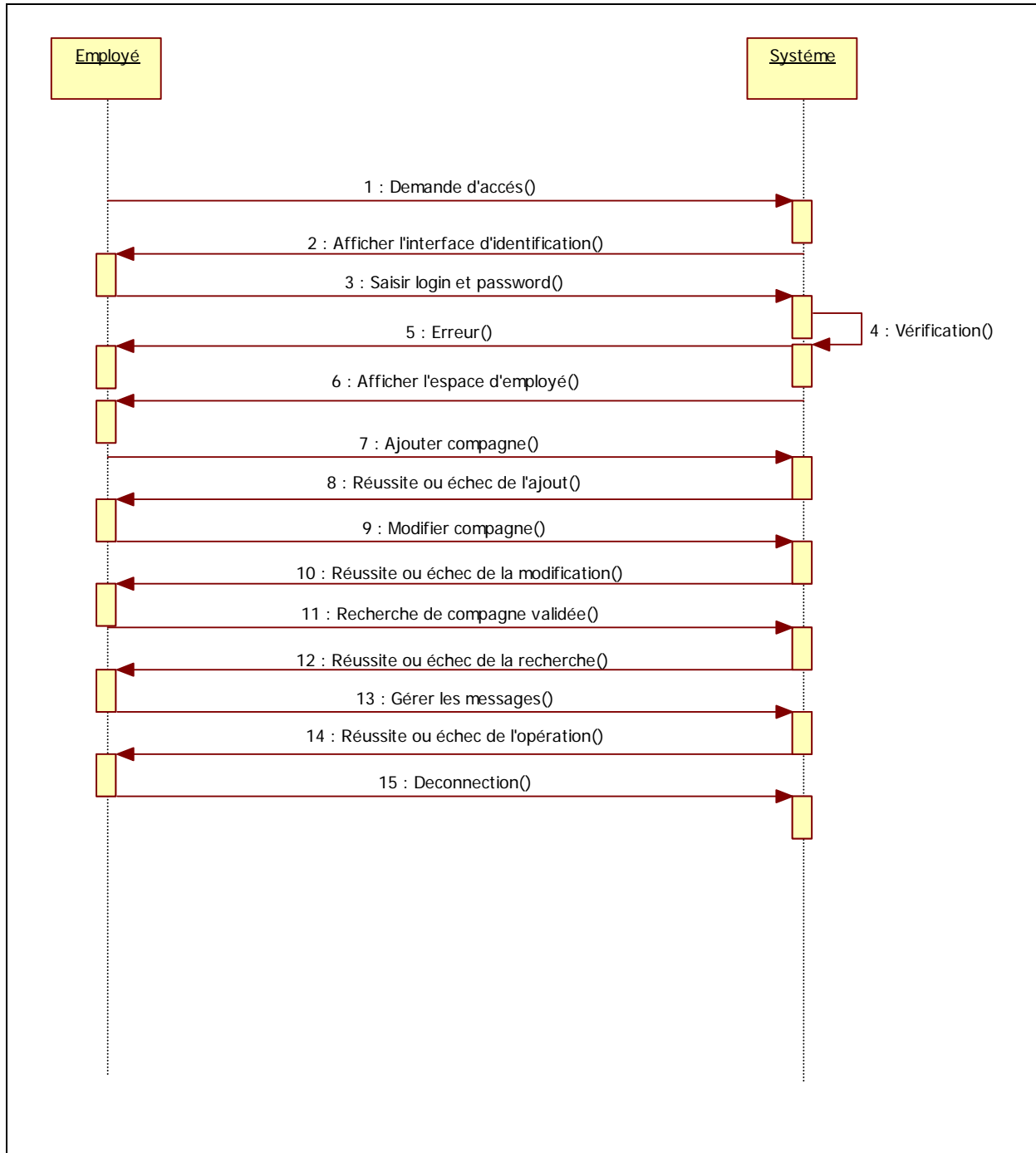


Figure 10: Diagramme de séquence de l'employé

✓ Diagramme de séquence global

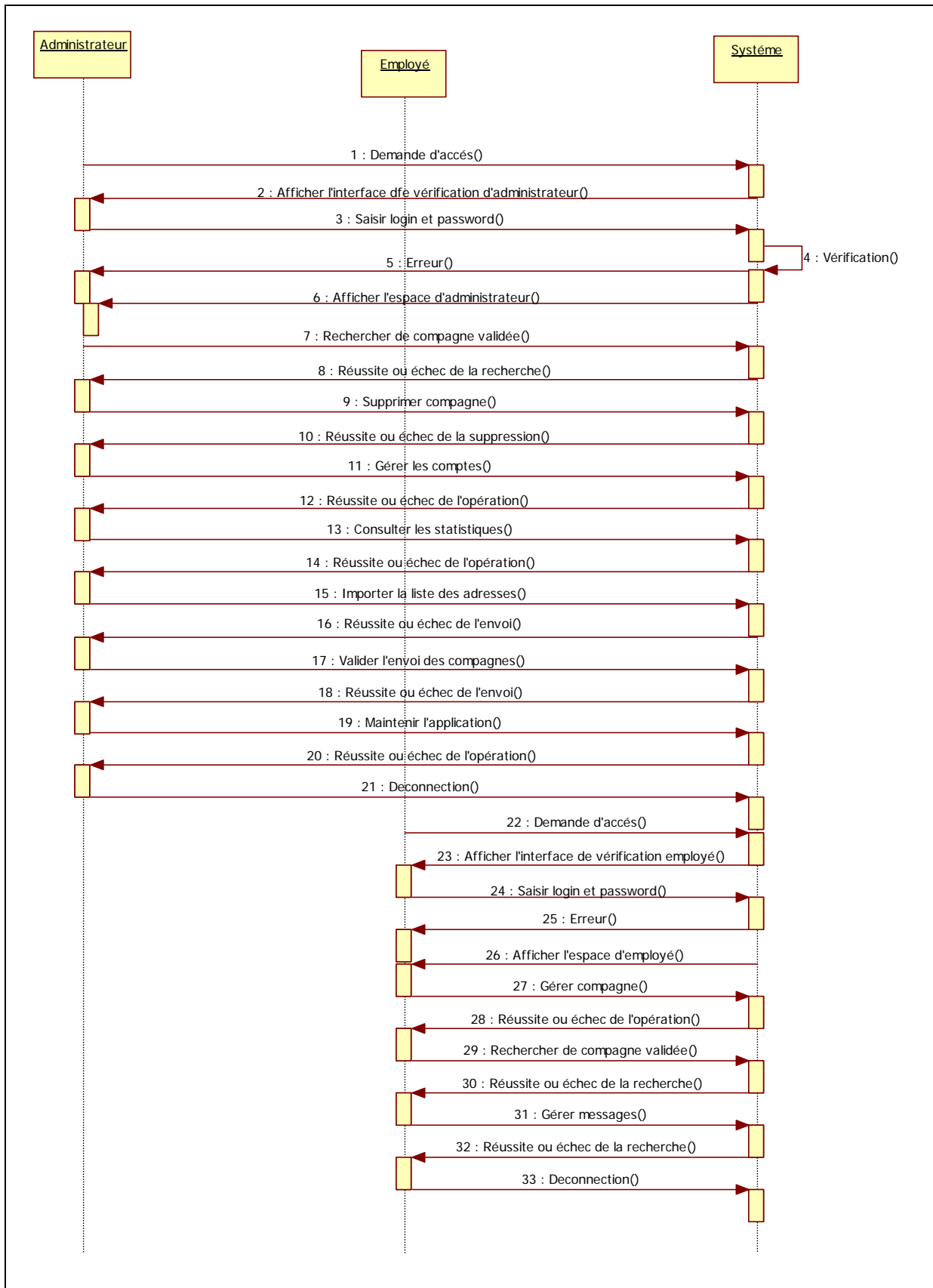


Figure 11: Diagramme de séquence global

Conclusion

Après avoir accomplir la conception de mon application, je vais entamer la partie "Réalisation". Dans le chapitre suivant, je vais présenter l'environnement de travail, les outils de développement utilisés, les tables de ma base de données ainsi que quelques imprimés écran des tests faits pour vérifier que mon système répond bien à son cahier de charge.

Chapitre V :

Réalisation

Introduction

Dans ce chapitre, je présenterai tout d'abord les différents environnements logiciels et matériels utilisés dans l'application, ensuite je présenterai les différentes interfaces accomplies tout au long de l'application.

I. Environnement de la réalisation

1. Environnement matériel

L'environnement matériel utilisé dans l'application est : **Ordinateur**

- Système d'exploitation : Microsoft Windows XP professionnel
- Disque dur : 300 GB

- **Environnement logiciel**

L'environnement logiciel utilisé dans cette application est le suivant :

- **Macromedia Dream Weaver MX 8:**

Macro media Flash MX 2004 est un outil de programmation qui permet de créer une application web graphique et interactive, vous pouvez enrichir vos applications Flash en y ajoutant des images, du son et de la vidéo. Flash inclue de nombreuses fonctionnalités et outils nécessaires pour la création de tel projet sur différentes plates-formes et dans différents environnements.

- **HTML** ("*HyperText Mark up Language*") est un langage dit de "marquage" chargé de formaliser l'écriture d'un document en indiquant la façon dont doit être présenté ce dernier et les liens qu'il établit avec d'autres documents. Il permet notamment la lecture de documents sur Internet à partir des machines différentes grâce au protocole HTTP, permettant d'accéder via le réseau à des documents repérés par une adresse unique, appelée URL. En effet, le HTML n'est pas un langage de programmation, Une page HTML est donc un simple fichier texte contenant des balises (appelées *marqueurs* ou *repères*) permettant de mettre en forme le texte, les images, etc.

- **Java script :**

JavaScript 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 de passage de la souris au-dessus d'un élément, du redimensionnement de la page...). La version standardisée de JavaScript est l'ECMAScript.

Il existe plusieurs façons d'inclure Java Script dans une page HTML:

- **Grâce à la balise <SCRIPT> :** Le code Java script peut être inséré dans une page Web. Pour utiliser les différentes versions de Java Script tout en conservant une certaine compatibilité, il suffit de déclarer plusieurs balises **SCRIPT** ayant chacune comme paramètre la version du Java Script correspondante.
- **En mettant le code dans un fichier :** Il est possible de mettre les codes de Java Script en annexe dans un fichier.
- **Grâce aux événements :** c'est-à-dire une action de l'utilisateur, comme le clic d'un des boutons de la souris.

- **Serveur Web Apache :**

Apache est le serveur web le plus répandu sur Internet permettant à des clients d'accéder à des pages web, c'est-à-dire en réalité des fichiers au format HTML à partir d'un navigateur (aussi appelé browser) installé sur leur ordinateur distant.

Il s'agit d'une application fonctionnant à la base sur les systèmes d'exploitation de type Unix, mais il a désormais été porté sur des nombreux systèmes, dont Microsoft Windows.

➤ **Easy PHP :**

Easy PHP est la solution, qui permet d'installer et de configurer un environnement de travail complet sous Windows, il est en fait un "package" qui contient tous les programmes nécessaires pour pouvoir traiter du PHP, il regroupe ainsi, un serveur Apache, une base de donnée MySQL, une interface utilisateur PHPMYAdmin et aussi bien le langage PHP.

➤ **MySQL :**

Les Système de Gestion de Bases de Données tels que MySQL permettent de manipuler facilement et avec beaucoup de souplesse un très important volume de données. Toutefois, aussi robuste soit MySQL, il peut être intéressant de récupérer l'ensemble des données que contient ma base de données, pour faire une sauvegarde (backup) ou bien tout simplement pour passer à une autre base de données.

MySQL est un serveur de. 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 " base de données relationnelle Open Source qui 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 Structured Query Language" : le langage standard pour les traitements de bases de données.

➤ **Star UML :**

Star UML est un logiciel de modélisation UML, cédé comme open source par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL.

Star UML est bien fait, facile à prendre en main, puisque il gère la plupart des diagrammes spécifiés dans la norme UML 2.0.

Ce logiciel open source est simple d'utilisation et intuitif pour un débutant. «**Star UML**» est simple d'installation et consomme peu de ressources système (mémoire ~200 Mo et utilisation CPU faible).

I. Structure de la base de données

1. L'architecture de la base de données

Les figures suivantes présentent les différentes tables qui existent dans la base de données « bulk ».

a. Table « compagne » :

Cette figure décrit la table « compagne » avec ses différents champs :

	Champ	Type	Attributs	Null	Défaut	Extra	Action					
<input type="checkbox"/>	<u>Id_cgne</u>	int(11)		Non		auto_increment						
<input type="checkbox"/>	Nom_cgne	varchar(30)		Non								
<input type="checkbox"/>	Titre_msg	varchar(30)		Non								
<input type="checkbox"/>	Adress_source	varchar(30)		Non								
<input type="checkbox"/>	Adress_destination	varchar(30)		Non								
<input type="checkbox"/>	Date_deb	date		Non	0000-00-00							
<input type="checkbox"/>	Date_fin	date		Non	0000-00-00							
<input type="checkbox"/>	Heure_deb	time		Non	00:00:00							
<input type="checkbox"/>	Heure_fin	time		Non	00:00:00							
<input type="checkbox"/>	nb_adresse	int(11)		Non	0							
<input type="checkbox"/>	etat	tinyint(4)		Non	0							

Figure 12 : Capture d'écran table « compagne »

b. Table « personne » :

Cette figure décrit la table « personne » avec ses différents champs :

	Champ	Type	Attributs	Null	Défaut	Extra	Action					
<input type="checkbox"/>	<u>id_pers</u>	int(11)		Non		auto_increment						
<input type="checkbox"/>	Matricule	varchar(30)		Non								
<input type="checkbox"/>	Nom	varchar(20)		Non								
<input type="checkbox"/>	Prenom	varchar(30)		Non								
<input type="checkbox"/>	Nom_grp	varchar(30)		Non								
<input type="checkbox"/>	Login	varchar(10)		Non								
<input type="checkbox"/>	Password	varchar(10)		Non								
<input type="checkbox"/>	Pwd_confirm	varchar(10)		Non								

Figure 13 : Capture d'écran table« personne »

c. Table « groupe » :

Cette figure décrit la table « groupe » avec ses différents champs :

	Champ	Type	Attributs	Null	Défaut	Extra	Action					
<input type="checkbox"/>	<u>Id_grp</u>	int(20)		Non		auto_increment						
<input type="checkbox"/>	Nom_grp	varchar(30)		Non								
<input type="checkbox"/>	Privilege_grp	varchar(30)		Non								
<input type="checkbox"/>	heure_deb	time		Non	00:00:00							
<input type="checkbox"/>	heure_fin	time		Non	00:00:00							

Figure 14 : Capture d'écran table« groupe »

d. Table « message » :

Cette figure décrit la table « message » avec ses différents champs :

	Champ	Type	Attributs	Null	Défaut	Extra	Action					
<input type="checkbox"/>	<u>id_msg</u>	int(12)		Non		auto_increment						
<input type="checkbox"/>	Titre_msg	varchar(20)		Non								
<input type="checkbox"/>	message	text		Non								

Figure 15 : Capture d'écran table« message »

e. Table « doc » :

Cette figure décrit la table « doc » avec ses différents champs :

	Champ	Type	Attributs	Null	Défaut	Extra	Action					
<input type="checkbox"/>	<u>id_doc</u>	int(20)		Non		auto_increment						
<input type="checkbox"/>	nom_doc	varchar(20)		Non								
<input type="checkbox"/>	chm_phy	varchar(255)		Oui	NULL							

Figure 16 : Capture d'écran table« doc »

II. La réalisation des interfaces de l'application

1. L'architecture graphique de l'application

L'architecture graphique est la structure qui me permet d'évoluer le design visuel d'un système d'information, on essaye d'harmoniser les couleurs, les lignes et les formes d'une telle application. Dans cette partie, on trouve la charte graphique qui concerne l'impact des couleurs et des formes pour l'utilisateur.

a. La charte graphique

La charte graphique est un document comprenant les caractéristiques des différents éléments graphiques qui peuvent être utilisés sur les différents supports et les interfaces de communication de l'entreprise. La charte graphique permet de garantir l'harmonisation de la communication visuelle dans l'entreprise. Certes, Tunisie Telecom a déjà sa propre charte graphique, cependant je suis obligée de la respecter.

b. Choix des couleurs

J'ai vu que je dois respecter la charte graphique imposée par TUNISIE TELECOM et utiliser les mêmes couleurs utilisés dans la plupart des interfaces qui sont le bleu et le blanc.

c. Choix de maquette

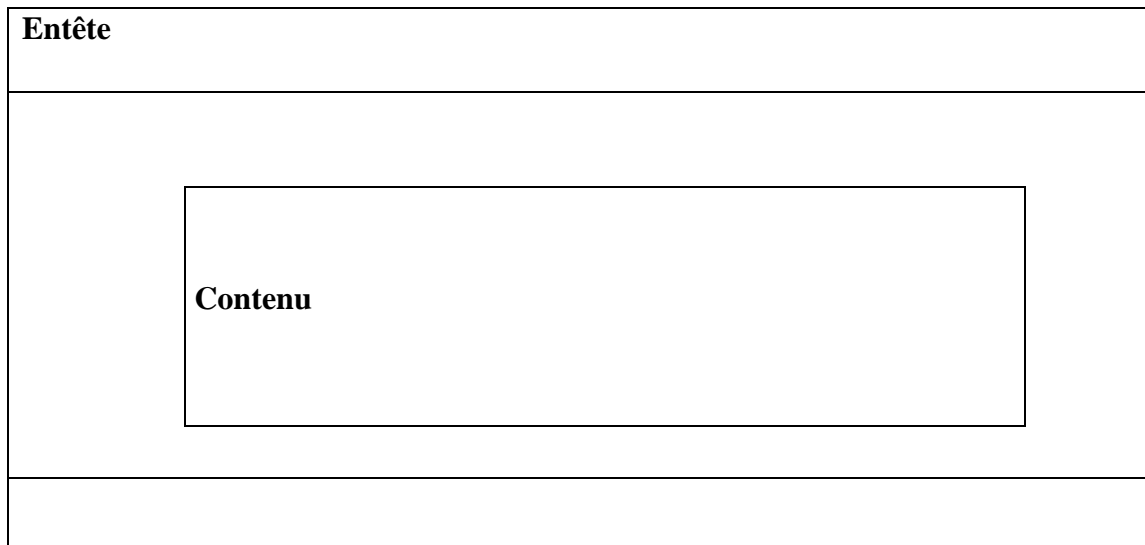


Figure 17 : 1^{ère} maquette

La 1^{ère} maquette est utilisée pour la page Index.

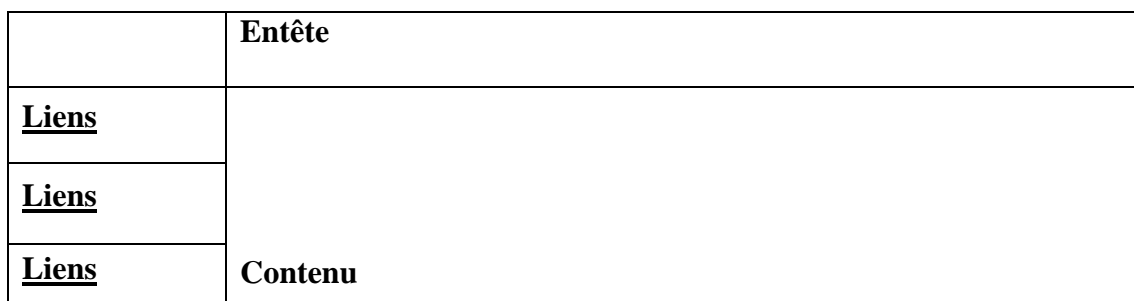


Figure 18 : 2^{ème} maquette

J'ai choisi la 2^{ème} maquette pour les autres pages de mon application dont la forme est rectangle, le contenu est centré et avec des liens à gauche du contenu. Ce choix est pris au sein du Tunisie Telecom comme la plupart des applications existantes qui ont un gabarit similaire à cette maquette.

d. Choix de la typographie

La typographie « Time new roman » qui est disponible sur tous types d'ordinateurs. Le choix de la typographie est fait en accord avec la tonalité graphique de l'application, Dans un souci de simplicité, universalité et de lisibilité, j'ai choisi de l'utiliser.

2. Les différentes interfaces de l'application

a. Page Index

C'est la page index de mon application. Pour les applications à structure hiérarchisée, cette page est tout naturellement celle qui est au sommet de la hiérarchie et vers laquelle, on pourra revenir à partir de toutes les pages de l'application.



Figure19 : Page Index

b. Page d'accueil administrateur

Après l'authentification de l'administrateur, cette page s'affiche qui contient les opérations effectuées par l'administrateur. A partir de laquelle, on pourra revenir à toutes les pages de l'application.



Figure20 : Page d'accueil administrateur

1. Page Statistique des campagnes

A partir de cette page, l'administrateur peut extraire les statistiques des campagnes. En plus, à partir de cette page, il peut accéder à toutes les autres pages.



Figure21 : Page Statistique des campagnes

c. Page parcourir liste d'adresse

Cette page affiche le formulaire qui me permet de parcourir la liste des adresses destinataires.



Figure22 : Page Parcourir liste d'adresse

d. Page Liste des utilisateurs

Cette page présente la liste des utilisateurs déjà créée .Et à partir de cette page, on peut accéder aux pages Ajouter et modifier utilisateur.



Figure23: Page Liste des utilisateurs

Conclusion

Ce chapitre m'a permis de présenter la structure finale de mon application, de décrire les outils utilisés pour la réalisation et la présentation des interfaces de l'application. La réalisation de cette application m'a permis d'acquérir une expérience concernant le développement d'un projet depuis l'étape de spécification jusqu'à la réalisation finale.

Conclusion Générale

Après trois mois du projet de fin d'études, j'arrivais à terme de mon travail, en partant de la réflexion, de la recherche continue, des réunions successives et des études approfondies sur le sujet de mon application, j'ai pu répondre aux exigences demandées par TUNISIE TELECOM. Le projet s'est déroulé selon trois axes principaux afin de passer par les étapes essentielles de tout projet : l'analyse, la conception et la réalisation. Ces derniers m'ont permis de déterminer et d'imaginer l'architecture globale de l'application ainsi que le langage avec lequel j'ai codé et construire la base de données. L'application BULK SMS a été développée avec le langage PHP et une base de données relationnelle MySQL.

L'intérêt principal que j'ai tiré de cette étude est que j'ai bien affronté la vie professionnelle de mon domaine. Cette application m'a été enrichissante puisqu'elle m'a intégré dans les ambiances du travail professionnel et m'a permis de vivre de plus près les étapes de la mise en œuvre d'une application. En effet, le développement d'une application ne doit pas dépendre exclusivement de la facilité d'utilisation du produit, mais aussi, doit tenir compte de la difficulté du problème à résoudre ainsi que l'enchaînement nécessaire à sa réalisation : documentation, spécifications, programmation, tests et validation.

Donc, j'ai pu voir la complexité de la mise en route d'un nouveau projet et de sa rapide évolution qui m'a appris à me mieux organiser afin d'être capable de finaliser mon travail.

Finalement, je suis conscient que ce projet n'est qu'un aperçu de la vie professionnelle et ce que j'ai réalisé ne constitue qu'une graine d'un monde de connaissance très vaste à découvrir.

Liste des figures

Figure 1 : Logo de Tunisie Telecom.....	4
Figure 2 : Organisation fonctionnelle de Tunisie Telecom	6
Figure 3 : Diagramme des cas d'utilisation de l'administrateur.....	22
Figure 4 : Diagramme des cas d'utilisation d'employé.....	23
Figure 5 : Diagramme des cas d'utilisation global	24
Figure 6 : Diagramme de classe global.....	28
Figure 7 : Diagramme de classe final.....	30
Figure 8: Diagramme de séquence de l'authentification.....	33
Figure 9: Diagramme de séquence de l'administrateur.....	34
Figure 10: Diagramme de séquence de l'employé.....	35
Figure 11: Diagramme de séquence global.....	36
Figure 12: Capture d'écran table « compagne ».....	43
Figure 13 : Page Capture d'écran table « personne ».....	44
Figure 14 : Page Capture d'écran table « groupe ».....	44
Figure 15 : Page Capture d'écran table « message ».....	45
Figure 16 : Page Capture d'écran table « doc ».....	45
Figure 17 : 1^{ère} maquette.....	47

Figure 18 : 2^{ème} maquette.....47

Figure 19 : Page Index.....48

Figure 20 : Page d'accueil administrateur.....49

Figure 21 : Page liste des compagnes.....50

Figure 22 : Page Parcourir liste d'adresse.....51

Figure 23: Page Liste des utilisateurs.....52

Annexe

I. Présentation de l'environnement de simulation

1. Passerelle kannel

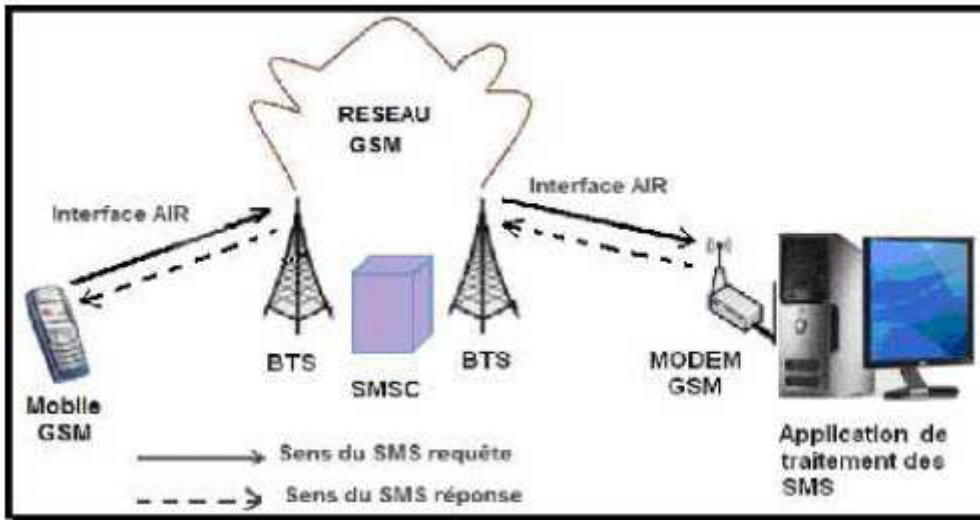
➤ Présentation

Kannel représente l'environnement simulateur du MMG, un projet Open Source, mis sous licence **Free BSD** lancé par la compagnie finlandaise **Wapit Ltd** en juin 1999. Elle est écrite en langage C, et fournit une passerelle mixte SMS et WAP (Wireless application Protocol). La licence Free BSD est une licence libre utilisée pour la distribution de logiciels. Elle permet de réutiliser une partie ou tout le logiciel sans restriction, qu'il soit intégré dans un logiciel libre ou propriétaire. En outre, elle permet également de gérer le push, le pull ou pull-push SMS c'est-à-dire respectivement, l'envoi des messages aux entités extérieures, la réception d'un SMS ou la réception de requête, puis l'envoi de la réponse après traitement à partir de la plateforme. Au niveau de l'application se fait le traitement de la requête, puis l'envoi de la réponse via le SMSC de rattachement à l'abonné demandeur. Dans la mesure où la passerelle est reliée à plusieurs SMSCs, une configuration appropriée se chargera du routage du SMS-MT vers le SMSC concerné ; l'éditeur de service n'a donc pas l'obligation de maîtriser avec ses contours le protocole d'interfaçage au SMSC.

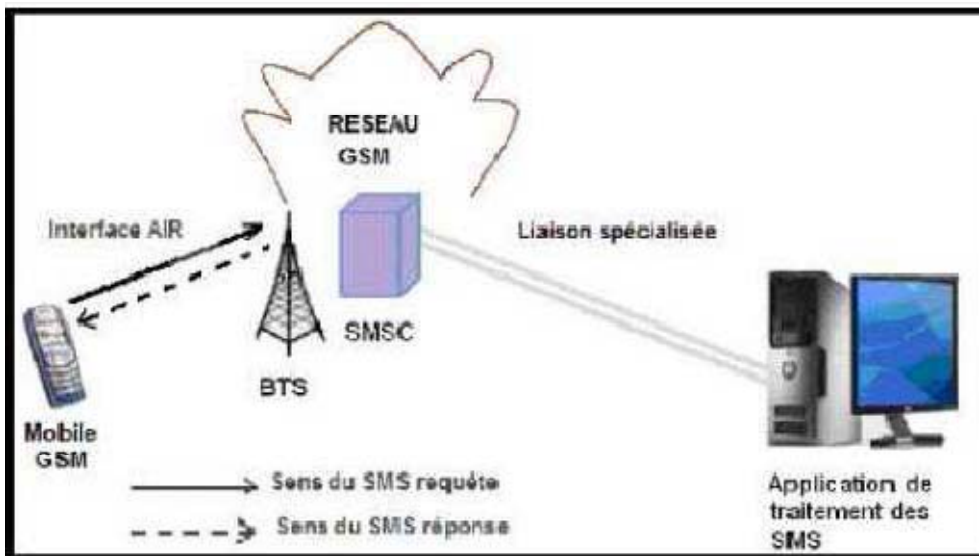
➤ Principe de fonctionnement de kannel

Le SMS est acheminé par le réseau de téléphonie mobile vers le serveur. Le serveur autorise à part la liaison au SMSC via une liaison spécialisée, une liaison à travers un simple modem GSM. L'acheminement des SMS requête-réponse diffère selon les deux modes de liaison :

- Une mode de liaison du serveur au SMSC de l'opérateur, par le truchement d'un modem GSM (SMSC virtuel),



- Pour le mode de liaison, à travers une liaison spécialisée du serveur au SMSC de l'opérateur (SMSC réel)



Remarque : Il faut noter que la liaison via un modem est plus facile sur le côté technique et financière alors que la liaison spécialisé est destiné pour l'établissement qui procède un serveur SMSC dans le taux de trafic de message sera plus élevé qu'avec un modem GSM.

➤ Exigences

Kannel est surtout développée sur les systèmes Linux, et devrait être assez facile par rapport à d'autres-comme les systèmes Unix. Cependant, je ne suis pas encore en charge d'autres plateformes, en raison du manque de temps.

Kannel peut être développé sur les systèmes Windows, par l'utilisation d'un environnement linux pour Windows dans notre cas c'est Cygwin.

➤ **Présentation Cygwin**

Cygwin est un environnement de type Linux pour Windows. Il se compose de deux parties:

- Une DLL (cygwin1.dll) qui agit comme une couche d'émulation Linux API fournit d'importants Linux API fonctionnalité.
- Une collection d'outils qui fournissent Linux look and feel.
- Cygwin n'est pas un moyen d'exécuter des applications Linux natif sur Windows. Vous devez reconstruire votre application à partir des sources si vous voulez qu'il fonctionne sur Windows.
- Cygwin n'est pas un moyen de faire magie natif des applications Windows connaissance d'UNIX ® fonctionnalités, comme les signaux, ptys, etc. Encore une fois, vous avez besoin pour construire vos applications à partir des sources si vous voulez profiter de la fonctionnalité Cygwin.
- **Cygwin** est une collection de logiciels libres à l'origine développés par Cygnus Solutions permettant à différentes versions de Windows de Microsoft d'émuler un système Unix. Il vise principalement l'adaptation à Windows de logiciels qui fonctionnent sur des systèmes POSIX (tels que les systèmes GNU/Linux, BSD, et Unix). Cygwin tente de créer un environnement Unix sous Windows, rendant possible l'exécution de ces logiciels après une simple compilation. Les programmes ainsi portés sur Cygwin, fonctionnent mieux sur Windows NT, Windows 2000 et Windows XP que sur les versions antérieures de Windows, mais certains peuvent s'exécuter de façon tout à fait acceptable sur Windows 95 et Windows 98. La version 1.5.24-2 du 31 janvier 2007 est la dernière à supporter officiellement Windows 95/98/Me.
- Cygwin est actuellement maintenu par les employés de Red Hat et d'autres programmeurs.

Neto graphie

www-lipn.univ-paris13.fr

www.code-sources.com

www.Developpez.com

www.javascript.com

www.codes_php.com

Sommaire

INTRODUCTION GENERALE..... 1

Chapitre I : Cadre générale du projet

I. Présentation de l'organisme d'accueil	4
1. Historique de Tunisie Telecom	4
2. Organigramme de l'organisme	6
II. Présentation du sujet	7
1. Problématique	7
2. Les objectifs	8
Conclusion	8

Chapitre II : Etude de l'existant

Introduction	10
I. Etude de l'existant	10
1. Critique de l'existant	10
2. Solution proposée	11
3. La partie sécurité de l'application	12
Conclusion	12

Chapitre III : Analyse et spécification des besoins

Introduction	14
I. Etude de l'existant	14
1. Spécification des besoins fonctionnels	14
2. Spécification des besoins non fonctionnelles	15
II. Spécification semi-formelle des besoins	16
1. Méthode de conception	16
a. UML	16
III. Diagramme des cas d'utilisation	18
1. Identification des acteurs et des cas d'utilisation	18
a. Les acteurs	18
b. Les cas d'utilisation	18

c. Eléments constitutifs des cas d'utilisation	21
2. Diagrammes des cas d'utilisation	21
a. Diagramme des cas d'utilisation de l'administrateur	22
b. Diagramme des cas d'utilisation de l'employé.....	23
c. Diagramme des cas d'utilisation global de système	24
Conclusion	25

Chapitre IV : Conception

Introduction	27
I. Conception de l'application	27
1. Modélisation conceptuelle	27
a. Conception de la base de données en se basant sur le langage UML	27
b. Conception des traitements	33
Conclusion	38

Chapitre V : Réalisation

Introduction	40
I. Environnement de la réalisation	40
1. Environnement matériel	40
2. Environnement logiciel	40
II. Structure de la base des données	43
1. Règles de passage du diagramme de classe à la base de données	36
1. L'architecture de la base de données	36
a. Table « compagne».....	37
b. Table « personne».....	37
c. Table « groupe».....	38
d. Table « message»	38
e. Table « doc».....	38
III. La réalisation des interfaces de l'application	46
1. L'architecture graphique de l'application.....	46
a. La charte graphique	46

b. Choix des couleurs	46
c. Choix de maquette	47
d. Choix de la typographie	48
2. Les différentes interfaces de l'application.....	48
a. Page index	48
b. Page d'accueil administrateur	49
c. Page Liste des compagnes	50
d. Page parcourir liste d'adresse	51
e. Page Liste des utilisateurs.....	52
Conclusion	52
Conclusion générale	54
Neto Graphie	55
Annexe	57
1. Passerelle kannel	58