

**ENSEIGNEMENT SUPERIEUR ET UNIVERSITAIRE  
INSTITUT SUPERIEUR DE PASTORALE**

**FAMILIALE DE BUKAVU**

**ISPF/BUKAVU**



**B.P.162 BUKAVU**

**Débout face aux défis de notre temps**

**LA MISE EN PLACE D'UN SITE WEB DE  
RESERVATION DES BILLETS EN LIGNE AU SEIN  
DE L'ENTREPRISE LWAMURHULO**

Présenté par : **ATULINDE MWEMA Blaise**

Travail de Fin d'Etudes

Présenté en vue de l'obtention d'un Diplôme de licence en  
Informatique et Gestion des Ressources Humaines

Directeur : **Ass. MUGARUKA BUDUGE Guillain**

**ANNEE ACADEMIQUE 2018-2019**

## **Dédicace**

*A toi mère admirable, Célestine NILIPATA, femme pleine de courage*

*A vous chers frères et sœurs, toujours prêt à vous sacrifier pour moi*

*A vous chers amis, Eli KABERA, BALUME Antoine, Arnold, Gracia L, wani, hope  
NABIMTU, Elesham, Junior AGANZE, Jonathan, Ashah...pour la confiance que vous placer  
en moi*

## Remerciements

Pareil à une femme enceinte, mais qui finit par mettre au monde un bébé.

Ce présent travail marque notre point d'aboutissement de nos études universitaires. Très sûr, abandonnés à nous-mêmes, il nous serait impossible de parachever ce long parcours.

Si l'on cherchait à citer nominativement tous ceux qui méritent notre mot de remerciement, on remplirait tout un gros livre des noms destinés rien qu'au remerciement. Pour que cela n'ait pas lieu, nous remercions en premier lieu tous, tous ceux qui étaient qui sont et qui seront les notre dans toutes notre vie.

Nous ne pouvons rien faire sans la volonté de notre Seigneur Jésus Christ, père céleste nous te rendons encore gloire et honneur ...

Nos remerciements s'adressent spécialement à nos très chers parents MWEMA R. & Célestine NILIPATA, qui n'ont pas fermé les yeux pour que je sois aujourd'hui ce que je suis.

Nous remercions notre cher encadreur, l'Assistant Guillain BUDUGE d'avoir, malgré ses multiples occupations, accepté de diriger ce travail jusqu'à sa fin.

Ainsi, nous remercions également tous les agents de l'entreprise Lwamarhulo et toute personne ayant de près ou de loin apporté son soutien d'une manière ou d'une autre, pour cet heureux aboutissement.

## Abbreviations

<b>CSS:</b>	Cascading Style Sheet
<b>HTML:</b>	Hyper Text Markup Language.
<b>HTTP :</b>	Hyper Text Transfer Protocol
<b>MVC :</b>	model vue contrôleur
<b>PHP :</b>	HyperText PreProcessor
<b>ISPF :</b>	Institut Supérieur pastorale Familiale
<b>UML :</b>	Unified Modeling Language
<b>BDD :</b>	Base de données
<b>CU :</b>	cas d'utilisation

**LISTE DES TABLEAUX**

<b>Tableau 1 : Flux d'informations.....</b>	<b>13</b>
<b>Tableau 2 : Dictionnaire des données.....</b>	<b>27</b>
<b>Tableau 3 : Association et multiplicités.....</b>	<b>30</b>
<b>Tableau 4 : langages de programmation utilises dans notre projet de réalisation d'un site web.....</b>	<b>34</b>
<b>Tableau 5 : formule de la méthode COCOMO.....</b>	<b>43</b>

## LISTES DES FIGURES

<i>FIGURE 1: L'ARBRE DE PROBLEME</i> .....	- 2 -
<i>FIGURE 2: ARBRE A SOLUTION</i> .....	- 6 -
<i>FIGURE 3: DIAGRAMME DES CAS D'UTILISATION</i> .....	- 16 -
<i>FIGURE 4: DIAGRAMME DE SEQUENCE CONNEXION</i> .....	- 19 -
FIGURE 6: DIAGRAMME DE SEQUENCE MODIFICATION .....	- 20 -
FIGURE 7: DIAGRAMME DE SEQUENCE RECHERCHER .....	- 21 -
FIGURE 8: DIAGRAMME D'ACTIVITE CONNEXION .....	- 22 -
FIGURE 9: DIAGRAMME D'ACTIVITE ENREGISTREMENT .....	- 22 -
FIGURE 10: DIAGRAMME D'ACTIVITE MODIFICATION .....	- 23 -
FIGURE 11: DIAGRAMME D'ACTIVITE RECHERCHE .....	- 23 -
FIGURE 12: DIAGRAMME DES CLASSES .....	- 24 -
FIGURE 13: DIAGRAMME DES CLASSES MODELE RELATIONNEL .....	- 26 -
<i>FIGURE 14: DIAGRAMME DE DEPLOIEMENT</i> .....	- 32 -
<i>FIGURE 15 : DIAGRAMME DE DEPLOIEMENT</i> .....	- 35 -
FIGURE 16 : PAGE D'ACCUEIL .....	- 36 -
FIGURE 18 : PROGRAMME DES BATEAUX .....	- 36 -
FIGURE 19 : CREATION COMPTE .....	- 37 -
FIGURE 20 : CREATION COMPTE EN CAS D'ERREUR .....	- 38 -
FIGURE 22 : RESERVATION BILLET.....	- 39 -
FIGURE 24 : ESPACE ADMINISTRATEUR .....	- 40 -
FIGURE 25 : GESTION DES BATEAUX .....	- 40 -
FIGURE 26 : CONSULTER LES RESERVATIONS .....	- 41 -
FIGURE 27 : GESTION PROGRAMME .....	- 41 -
FIGURE 28 : GESTION DESTINATION .....	- 42 -

## **0. Introduction générale**

### **0.1. Problématique**

Depuis la découverte de l'informatique, de nombreuses activités de la vie courante ont été simplifiées. Actuellement les individus peuvent facilement traiter des informations en se servant des logiciels et des réseaux informatiques.

Compte tenu de son évolution, ce système caractérise la majorité des grandes entreprises quel que soit le secteur d'activité.

Actuellement le monde connaît une avancée technologique considérable dans tous les secteurs et cela grâce à l'informatique. L'informatique joue un rôle prépondérant dans le développement des entreprises et d'autres établissements.

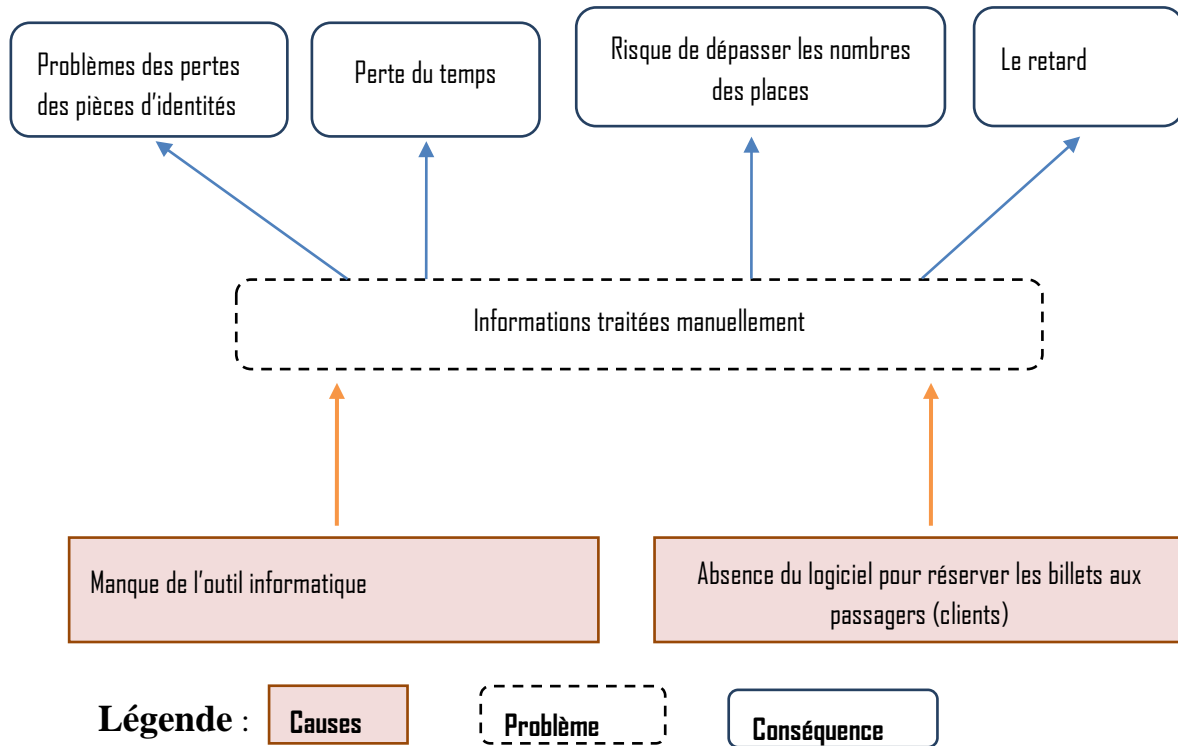
L'informatique s'est avéré être par conséquent une science dont la courbe d'évolution ne cesse de saisir des aspects d'évolution exponentiels dans plus d'un domaine, celui de transport lacustre y compris. Toutefois, cette évolution n'a pas été et ne continue pas à être équitable dans tous les pays. Les pays développés sont sur ce point de l'évolution informatique depuis longtemps et l'emporte sur les pays en voie de développement, parmi lesquels : notre pays, la République Démocratique du Congo. [Dreyfus, 2006], *développement et exploitation durable, l'homme à la recherche du développement, 2006*.

L'informatique a une grande part dans la science et même dans la société, par contre il se trouve qu'il existe encore des services non informatisés dans la ville de Bukavu notamment dans le domaine de transport lacustre. Les passagers (clients) sont obligés de passer dans des points de ventes pour connaître les programmes des bateaux, les prix par classe et la qualité de services qu'offre le bateau lors de voyages de Bukavu pour Goma et vice versa. Par ailleurs, pour la plupart des voyageurs de cette entreprise, ils arrivent le jour et à l'heure même du départ de bateaux soit par manque de temps, soit par leur vouloir pour effectuer toutes ses opérations, ce qui est à la base de ces différents problèmes précités.

Le fait d'attendre pour acheter le billet de voyage ou obtenir des informations le même jour et à la même heure de voyage cause beaucoup des problèmes tel que : avant le voyage il y a un engouement des passagers sur les guichets et cela occasionne un retard, des pertes de pièces d'identités etc. Actuellement, la ville de Bukavu fait face à ces problèmes liés la non informatisation des beaucoup de services notamment ceux du secteur lacustre.

Nous présentons ci-dessous un schéma que nous allons nommer « arbre de problèmes » qui synthétise mieux les problèmes rencontrés à l'entreprise Lwamarhulo (bateau Aganze), mais les causes et conséquences.

**Arbre de problèmes :**



*Figure 1: l'Arbre de problème*

Eu égard aux problèmes ci-dessus, le constant ci-après nous a fort préoccupé et nous a amené à réfléchir autour de les questions maitresses de notre travail que voici :

- Est-il possible de mettre en place une application web qui permettra de réserver et de connaitre le programme de voyage au sein de cette entreprise ?
- Est-il possible de diminuer l'engouement des passagers sur les guichets avant le départ du bateau ?
- Le développement d'une application web de réservation des billets et d'affichage des programmes ne serait-elle pas une solution aux problèmes que connait cette institution ?



## **0.2. Hypothèses**

Se penchant à cette question afin de produire un résultat pouvant venir en aide les passagers de cette entreprise, nous émettons cette hypothèse selon laquelle : la mise en place d'une application web accessible par tous et permettant de réserver un billet de voyage serait judicieux. Grace à cette mise en place du logiciel, le problème de faire la queue ou l'engouement pour acheter les billets avant le départ d'un bateau sera résolu.

## **0.3. Etat de la question**

En ce qui concerne la conception et la réalisation d'un site web de réservation de billet à l'entreprise Lwamarhulo, dans le monde actuel, il est fortement difficile qu'un scientifique fasse un travail tout en n'ayant aucun trait avec les travaux préexistants. Nous avons consulté les travaux de nos prédécesseurs sur cette question pour savoir ce qu'ils ont fait et qu'est-ce que nous, nous allons faire de notre côté.

C'est ainsi que, nous avons décelé après consultation, qu'il y a toute une infinité de travaux qui ne sont pas loin du notre :

- BRAHMI SARA (2015-2016) sujet de mémoire «Proposition d'un module de gestion d'hôtel et d'un site de réservation en ligne, cas Sogesi en Algérie»Ce travail de fin d'études pour l'obtention du diplôme de master en informatique n'était qu'une amélioration des modules de gestion d'hôtel et des restaurations déjà existant au lieu de créer un nouveau.il était question d'ajouter des nouvelles fonctionnalités proposée par l'entreprise, le but principal de l'entreprise était de créer un module qui gère la gestion des hôtels. Ce module était capable de gérer : réservations des clients via le site web, chambres et leurs disponibilités, personnel, les options offertes par l'hôtel. L'utilisateur peut aussi contacter l'hôtel pour des renseignements ou pour consulter la situation géographique de l'hôtel sur la carte Google Mapp.
- MULUMEODERWA (2014 – 2015) sujet de mémoire « La conception d'un system de réservation de vol en ligne, cas C.A.A » son travail portait sur la conception d'un logiciel informatique en utilisant UML comme méthode de conception et PHP comme langage d'implémentation. Il a essayé de résoudre le problème de réservation de vol en mettant en place un système de réservation de

vol en ligne qui est estimé comme un Model dynamique pour cette compagnie cette réservation va permettre au Gestionnaire de service d'agir en connaissance de cause de prévoir en temps opportun les mesures à prendre en fonction de circonstance.

Quant à notre travail, trouve son originalité en proposant un site web où les voyageurs peuvent réserver un billet de voyage et donne un espace à cette entreprises de publier leurs informations (programmes et autres), et là facilement un passager pourra se connecter et voir le jour et l'heure du départ de tel ou tel autre bateau.

Nous utiliserons le même langage d'implémentation PHP et méthode UP que nos prédécesseurs.

#### **0.4. Choix et intérêt du sujet**

##### **0.4.1. Choix du sujet**

Le sujet ne nous est pas choisi, nous l'avons choisi car nous sommes animé par le souci de lutter contre l'engouement du passager lors de l'achat d'un billet de voyage au port

##### **0.4.2. Intérêt du sujet**

###### ***4.2.1. Intérêt personnel***

Entant qu'un étudiant en informatique et gestion des ressources humaines, ce sujet nous permettra de développer nos connaissances acquises jusqu'à ce niveau, ainsi de concevoir un site web qui parviendra à résoudre les problèmes précités.

###### ***4.2.2. Intérêt pour la société***

Les voyageurs de cette entreprise Lwamarhulo ne perdront plus le temps au port pour s'acheter le billet de voyage, par contre ils pourront réserver leurs billets en ligne avant le temps.

###### ***0.4.2.3. Intérêt scientifique***

Les chercheurs scientifiques auront une piste pour effectuer leur recherche dans la conception et la réalisation d'un site web de réservation de billet de voyage.

## **0.5. Méthodes et techniques**

### **0.5.1. Méthodes**

Une méthode est la procédure d'une science, c'est l'ensemble de pratiques qu'elle met en œuvre pour le cheminement de ses démonstrations de ses théorisations soit claire [Raymond Q 2011], Raymond Quivy, *méthode d'enquête et audit*, paris.dunod.2011

Dans ce présent travail, nous avons fait recours à la méthode UP (UnifiedProcess). **UP est une méthode générique de développement de logiciel.** Générique signifie qu'il est nécessaire d'adapter UP au contexte du projet, de l'équipe, du domaine et/ou de l'organisation (*exemple :R.UP ou X.UP*). L'objectif principal d'un système logiciel est de rendre service à ses utilisateurs ; il faut par conséquent bien comprendre les désirs et les besoins des futurs utilisateurs. Le processus de développement sera donc centré sur l'utilisateur. Le terme utilisateur ne désigne pas seulement les utilisateurs humains mais également les autres systèmes. L'utilisateur représente donc une personne dialoguant avec le système en cours de développement.

### **0.5.2. Technique**

C'est un moyen précis pour atteindre un résultat partiel, à un niveau et à un moment précis de la recherche [Raymond Q 2011], Raymond Quivy, *méthode d'enquête et audit*, paris.dunod.2011

#### **5.2.1. Technique d'observation**

Elle constitue le guide du travail, c'est la principale technique qui nous a aidés à collectionner les données de ce travail, après avoir observé certains faits au sein d'agence de transport lacustre de l'entreprise en question.

#### **5.2.2. Interview**

Elle nous a permis de recueillir les informations relatives à notre travail auprès des personnels de cette entreprise.

#### **5.2.3. Documentaire**

C'est une technique de recherche scientifique, elle nous a permis de parcourir les documents pour obtenir les informations nécessaires à la recherche que nous avons menée, nous a également permis d'exploiter certains ouvrages, et le moteur de recherche : Google, les travaux

de fin de cycle pour essayer de voir ce qu'ont fait nos prédécesseurs et ce que nous allons faire à présent.

## **0.6. Objectif du travail**

Nous nous sommes fixé l'objectif principal, développer un site web de réservation des billets de voyage en ligne et publication des programmes pour l'entreprise Lwamarhulo.

### **0.6.1. Objectifs spécifiques**

Nous permet de partir, partie par partie, jusqu'atteindre notre objectif global

Nous nous sommes fixé les objectifs spécifiques ci-dessous :

- Tenter de réduire l'engouement sur les guichets avant le départ d'un bateau ;
- Eviter la perte de temps pour connaître le programme d'un bateau à cette entreprise en question;
- Mettre en place un site web qui sera accessible par tous et facile à utiliser ;
- Connaître les places qui nous restent avant de réserver une place pour une quelconque classe.

Nous présentons notre arbre (schéma) de solution qui est le contraire de l'arbre des problèmes que nous avons présenté ci-haut dont le voici :

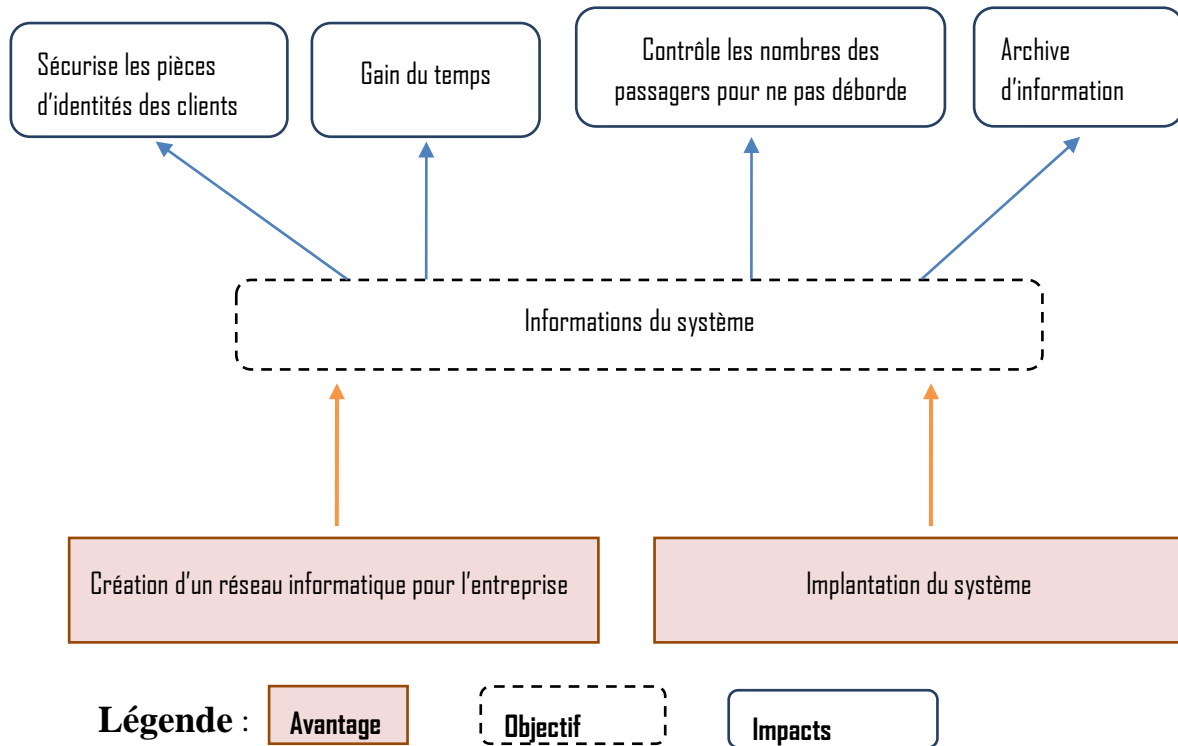


Figure 2: Arbre à solution

Pour arriver à notre objectif, nous aurons besoin des outils informatiques, une bonne connexion internet et un hébergeur du site.

### 0.7. Délimitation du sujet

Spatialement, nos analyses en vue d'implémenter un site web ont porté essentiellement sur l'entreprise Lwamarhulo (dans la ville de Bukavu) située dans la partie orientale de la RDC. Chronologiquement, ce présent travail est conçu et réalisé au cours de l'année académique 2018-2019, soit à partir du mois d'Octobre 2018 jusqu'au mois de Juillet 2019. Signalons aussi que les recherches ont été effectuées (serons effectuées) au cours de cette même période. Analytiquement, nous nous sommes intéressés uniquement à la réservation des billets et la publication de programmes du bateau(Aganze) de cette entreprise.

## **Chapitre I : CONSIDERATIONS THEORIQUES**

Ce présent chapitre, en premier lieu, permet au lecteur d'avoir la possibilité de se familiariser avec certains termes employés dans ce présent travail pour que nous ayons une même compréhension de ces derniers ; ensuite, il présente le cadre d'étude de notre travail.

### **I.1. DEFINITION ET APPROPRIATION DES TERMES**

Comme nous le savons qu'un mot peut avoir plus d'un sens selon le domaine ou le contexte dans lequel il est employé ; c'est une raison pour laquelle nous donnons la définition des quelques mots faisant partie du présent travail que nous avons fait recours.

**Un billet** : est un bien immatériel qui assure à son détenteur un accès personnel voyage ou autres place pour régulent une entrée. Sur un billet nous trouvons un nom du bateau, num matricule, num RCCM (c'est un registre comptable, un numéro d'autorisation du marché), type de classe des passagers, l'heure de départ, l'heure d'arriver, date de voyage, trajectoire, prix du billet, contact, signature du guichetier en fin une description du bateaux ou devise.

1. **Un acteur**: est un utilisateur type qui a toujours le même comportement vis-à-vis d'un cas d'utilisation. Ainsi les utilisateurs d'un système appartiennent à une ou plusieurs classes d'acteurs selon les rôles qu'ils tiennent par rapport au système. [Joseph G, UML 2] Joseph Gabay, *Analyse et conception, notion des acteurs Paris 2008 p.16*

**Un programme** : c'est un support contenant les programmes des bateaux, cette fiche set constituer de/du : prix, nom de bateau, les heures, destination.

**UML** : Le langage de modélisation unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet. Ce langage possède au moins treize diagrammes mais sont regroupés en deux catégories qui sont :

- **Diagrammes statiques** (structurels) : diagramme de classes, diagramme d'objets, diagramme de composants, diagramme de déploiement, diagramme de paquetages et diagramme de structures composites.
- **Diagrammes dynamiques** (comportementaux) : diagramme de cas d'utilisation, diagramme d'activités, diagramme d'états-transitions, diagrammes d'interaction, diagramme de séquence, diagramme de communication, diagramme global d'interaction, diagramme de temps (Michael B 2005)

## **I.2. Présentation du cadre d'étude**

Etant donné que la ville de Bukavu est bien entendu notre cadre spatial, c'est ainsi que dans cette partie nous tacherons de présenter cette dernière.

### **I.2.1. Brève historique de notre milieu d'étude**

C'est depuis plusieurs années que les habitants de la ville de Bukavu avaient réfléchies sur la mise en place de l'outil pouvant servir le transport de l'homme par voie maritime (voie d'eau), suite aux rebellions de 1969 et ses abus qui avaient rendu la ville de Bukavu isolée des autres provinces. A l'issu de cela, les états des voies de communication routiers accentuaient la coupure des relation entre Bukavu et les autres coins du pays, mais du coup la mixtion de 4 réseaux ferroviaires ( KDF,CVC,CFL,CFML) s'était effectuée afin de résoudre ce problème et donna naissance à la SNCC. Cette dernière crée son entreprise dans la province du sud Kivu et débute ses projets le 02 /12 /1974 par ordonnance loi n° 74-027 et fonctionne comme étant un établissement public et commercial, et s'assigne comme objectif :

- D'étudier, de construire et d'exploiter tous le chemin de fer et le service de transport -

De réaliser et exploiter les ports et tous services connexes ou accessoires.

C'est ainsi que l'entreprise Lwamarhulo a commencé ses activités commerciales dans la ville de Bukavu, province du Sud-Kivu en RDC dans les années 2014. Premièrement c'était dans le marché de kadutu où tout a commencé jusqu'à son épanouissement, qui a abouti à la construction du Bateau M/V Aganze. Ainsi, cette entreprise trouva sa place dans le transport lacustre. La construction de son premier bateau à débute en 2017 et prend fin en aout 2018. Il fut construit par l'ingénieur MAKYAMBE en collaboration avec BABI

Le nom « Lwamarhhulo » désigne le nom du village natal du promoteur, qui est situé sur l'île d'Idjwi, groupement Mugote, chefferie NTAMBUKA da la province du sud kivu.

### **I.2.2. Situation Géographique**

Elle est située sur l'avenue Michel micombero, en bas de la clinique universitaire de Bukavu, Quartier Nkafu, commune de kadutu, sur la route principale qui mène vers l'aéroport de Kavumu.

### **I.2.3. Le port attache**

Le principal domaine portuaire du bief moyen est celui de Bukavu qui est composé de 21 ports dont l'entreprise LWAMARHULO est l'un parmi ces 21 ports qui est situé sur la même adressée ci -haut citée et le port de Goma lui appartient par location. Ce port est surveillé par l'inspection de la navigation, commissariat lacustre et commandement du port public de Bukavu.

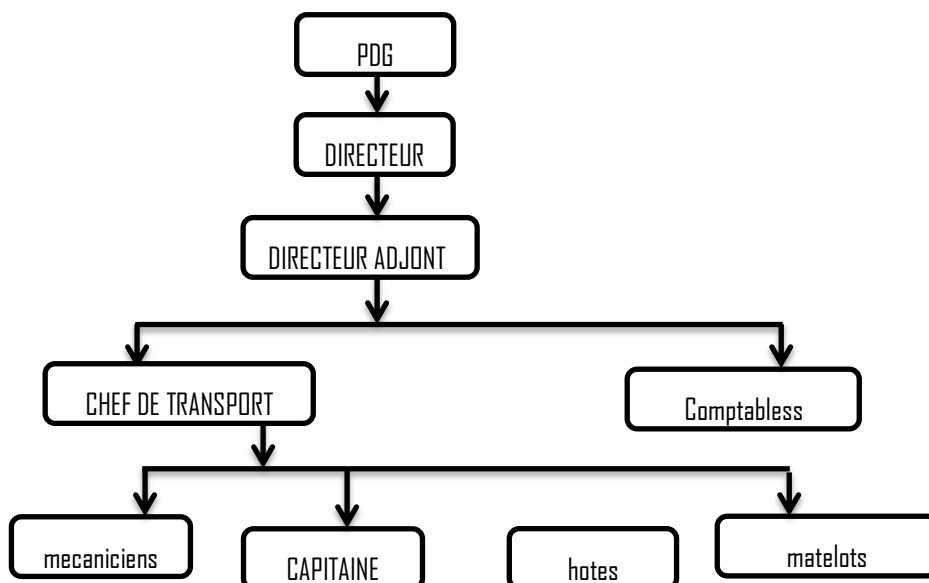
### **I.2.4. Statut juridique**

La forme juridique étant qu'une personne morale, nous y trouvons des avocats pour les conseils, mais aussi pour défendre les intérêts de cette entreprise.

## **Organigramme**

Voici la manière dont se présente l'organigramme au sein de cette entreprise :



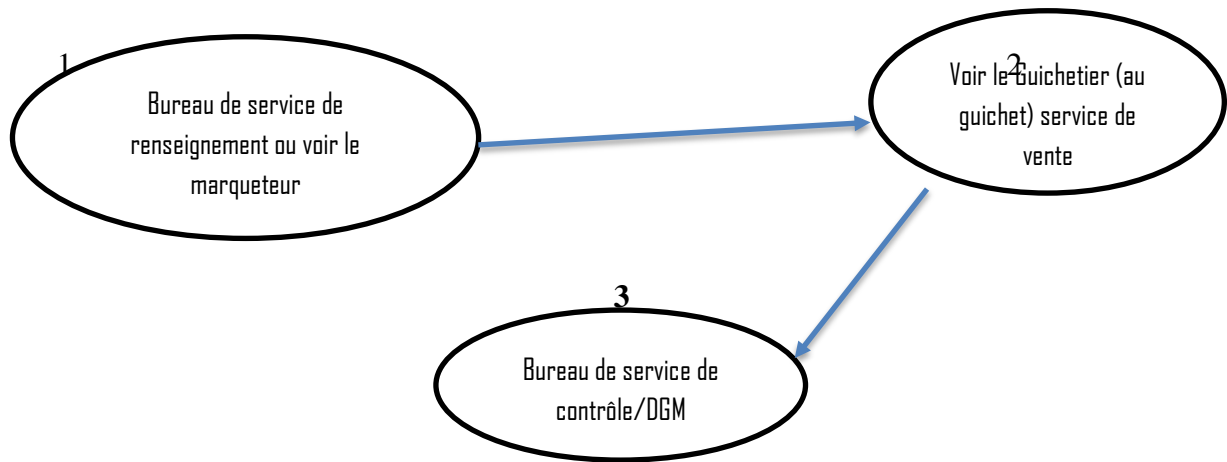


### I.2.3. Les activités (Goma, Bukavu)

Les échanges entre ces deux villes sont basés plus aux produits alimentaires qui viennent du nord Kivu et sont consommés localement ici dans la ville de Bukavu. C'est que l'entreprise LWAMARHULO joue un rôle à travers son Bateaux AGANZE 1 qui permet qu'il y ait l'échange des passagers, mais aussi des produits entre ces deux villes (Goma, Bukavu).

### Procédure à suivre (par le client)

Au sein du port LWAMARHULO, trois grand différents services qui sont à la disposition de voyageurs (le bureau de renseignement et autres, guichet et le bureau de contrôle). La procédure est aussi très simple et est reprise à la démonstration ci-dessous :



## Chapitre II

### ANALYSE DE L'EXISTANT ET CONCEPTION DU LOGICILE 2.1.

#### Introduction :

Dans ce chapitre, il est question de faire la représentation des réalités au sein de cette entreprise.

Ce chapitre a un double point (objectif) : comprendre comment les habitants de la ville de

Bukavu s'achetaient les billets de voyage et voir comment nous allons effectuer l'analyse critique ainsi que la modélisation du système en générale.

## 2.2. Analyse et critique de l'existant

Départ nos observations, sur notre entreprise œuvrant dans le domaine du transport lacustre et maritime dans la ville de Bukavu, nous avons constaté une faiblesse : du côté publication des programmes et du coté vente des billets (la lenteur), lorsqu'un client veut acheter un billet ou savoir un programme à deux options en face de lui : soit il part consulter le marqueteur soit il part directement vers le guichetier. Leurs enregistrements de pièce d'identité ou autres chose sont fait à la main, les programmes sont affichés sur un papier ou publier par des marqueteurs via des mégaphones...

### 2.2.1. Critique de l'existant

Eu égard de ce qui précède, nous avons constaté que si un voyageur veut acheter un billet ou acquérir quelques informations sur le programme du bateau il se déplace. Ceci a causé : le fait de faire la queue au port, le retard, manque de bonne places dans le bateau voulu, pertes (des pièces d'identité, d'énergie physique ...) et couts (de transport ...)

### 2.2.2. Etude des documents et leur description

Nous avons eu à utiliser un seul document parmi les documents qui rassurent le voyage par voie maritime (le billet) et un autre document qui permet aux gérants et aux clients de savoir le programme.

#### a) Schéma du circulation d'information

Acteurs	Rôles
1. Guichetier et Marqueteur	Marqueteur demande les programmes de la journée soit de la semaine auprès du guichetier tous les matins avant de se mettre au travail.
	Le guichetier livre tous les programmes de la journée ou de la semaine au marqueteur

2. Client Marqueteur	et	Client se renseigne auprès du Marqueteur
		Le marqueteur lui fournit les informations dont il a besoin
3. Client guichetier	et	Le client achète son billet auprès du guichetier (ce mouvement est fait cash )
		Le guichetier reçoit l'argent et donne le billet au client selon sa demande.
4. Agent DGM client	et	L'agent DGM contrôle et met un sceau billet et pièce d'identité (opération fait avant le départ)
		Le client présente son billet et sa pièce d'identité pour la vérification et l'imposition du sceau.
5.		Après le control si tous est en ordre, le client gagne sa place dans le bateau.

**Tableau1 : Flux d'informations**

Dans ce travail, nous, nous intéressons seulement à nos deux acteurs (le client et le gérant), car tous le programmes seront bien publiés en ligne on n'aura pas besoin des marqueteurs.

### **2.2.3. Piste de solution**

En voyant tous ces problèmes en face de nous, la mise en place d'une application web accessible par tous et permettant de réserver un billet de voyage serait judicieux. Aussi, les technologies Web actuelles, nous permettraient d'y arriver et corriger par ricochet ces multiples problèmes. Le système n'est pas l'outil à utiliser par tous, l'accès est limité aux personnes bien définies. Pour ce faire nous allons utiliser le langage UML et la méthode UP.

### **2. 3. Modélisation du système**

La modélisation s'appuie sur le concept modèle qui peut être défini comme étant une image de la réalité, une abstraction de quelque chose de réel. Modéliser un système consiste à créer une représentation standard dans le but de prévoir son évolution. Comme nous venons de l'annoncer dans le précédent point, nous allons utiliser UML pour la modélisation de notre futur système. UML : Le langage de modélisation unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet. Ce langage possède au moins treize diagrammes mais sont regroupés en deux catégories qui sont :

- **Diagrammes statiques** (structurels) : diagramme de classes, diagramme d'objets, diagramme de composants, diagramme de déploiement, diagramme de paquets et diagramme de structures composites.
- **Diagrammes dynamiques** (comportementaux) : diagramme de cas d'utilisation, diagramme d'activités, diagramme d'états-transitions, diagrammes d'interaction, diagramme de séquence, diagramme de communication, diagramme global d'interaction, diagramme de temps (Michael B 2005)

Dans le cadre de ce mémoire nous allons essayer de cadrer notre projet en soutirant quelques diagrammes qui nous seront nécessaires. En outre, nous allons présenter les diagrammes ciaprès :

- Le Diagramme des Cas d'utilisation
- Le Diagramme des Classes
- Le Diagramme des Séquences
- Le Diagramme d'activité
- Le Diagramme du déploiement

### **2.3.1. Réalisation des diagrammes**

Nous essayons d'étendre la représentation des diagrammes effectués au niveau de l'analyse en y intégrant les aspects techniques plus proches de nos préoccupations physiques. UML propose de représenter les cas d'utilisation sous une forme graphique nommée diagramme de cas d'utilisation appartenant au modèle des besoins. Les rôles sont définis pour chaque acteur. Une relation entre acteurs et cas représente une communication entre l'acteur et le cas. Le cas (d'utilisation) est représenté par une ellipse qui porte son nom (à l'intérieur ou en dessous). Un diagramme de cas d'utilisation montre acteurs et cas d'utilisation ensemble avec leurs relations. La relation entre un acteur et un cas d'utilisation est appelée association et correspond au fait que l'acteur participe à un cas d'utilisation.

Un cas d'utilisation, comme tout diagramme UML, permet de décrire une réalité selon différents niveaux de raffinement. Il convient, entre autres, de signaler le "niveau d'abstraction" de la vue afin de permettre au lecteur une meilleure interprétation de ce qui est et n'est pas montré.

### **2.3.2. Spécification des exigences d'après le diagramme de cas d'utilisation**

Un cas d'utilisation (<<use case>>) la partie qu'il ne faut pas négliger car sa joue un rôle fondamental dans le cycle de vie d'un projet de développement logiciel. En phase initial, ils

permettent d'identifier les utilisateurs et de comprendre leurs attentes. Le cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel. Présente une unité secrète d'interaction entre l'homme et le système [Bertrand M 1997]

Les besoins sont définis par la mise en place d'un site web permettant de voir les programmes du bateau ainsi que de réserver son billet de voyage. Le site devra donc centraliser tous les programmes du bateau en ligne sans fournir trop d'effort. Un acteur a le droit d'ouvrir une page, de voir le programme, de créer un compte, s'authentifier. Les cas que nous relève ce diagramme de cas d'utilisation concernent les acteurs.

Un acteur est un utilisateur type qui a toujours le même comportement vis-à-vis d'un cas d'utilisation. Ainsi les utilisateurs d'un système appartiennent à une ou plusieurs classes d'acteurs selon les rôles qu'ils tiennent par rapport au système. [Joseph Gabay, 2008]

Les acteurs identifiés pour ces différentes fonctionnalités sont : (i) le client : visite le site pour se renseigner sur les services qu'offre le bateau et peut probablement créer un compte pour lui permettre d'effectuer les opérations importantes dans ce cas-là il devient un client car il est enregistré sur le site, et enfin, (ii) un gérant : quand à lui peut enregistrer, modifier un programme peut aussi voir les clients ayant réserve pour un voyage.

Les exigences étant spécifiées et les acteurs étant identifiés, il devient alors possible de représenter le diagramme de cas d'utilisation de cette étude, notamment à travers la figure suivante :

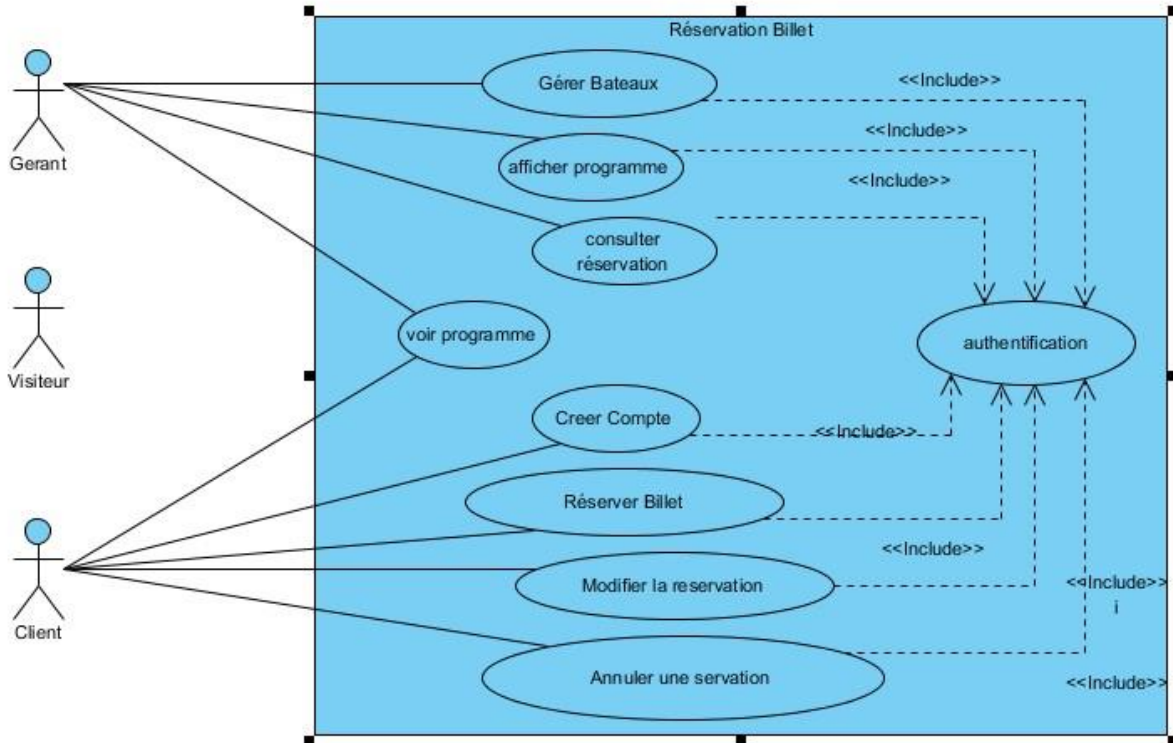


Figure 2: Diagramme des Cas d'utilisation

### 2.3.3. La représentation textuelle des cas d'utilisation

UML n'impose ni ne préconise aucun format particulier de description textuelle des cas d'utilisation. Elle est cependant couramment utilisée, c'est pour cette raison que nous l'avons introduit aussi dans ce présent travail. Cette représentation sous forme textuelle des cas d'utilisation donne une description de leurs comportements, de leurs actions et réactions. Le contenu de cette représentation textuelle est la suivante :

#### □ Le nom du cas d'utilisation ;

- **Les acteurs**, ce sont les acteurs déclencheurs du cas.
- **Le niveau du C.U**, nous donne une indication de lecture Un objectif utilisateur ;  
Ou une sous fonction ;
- **Les pré conditions**, sont les conditions à remplir pour que le cas d'utilisation puisse être exécuté ;
- **Les post-conditions**, déterminent ce qui est vrai après l'exécution du scénario nominal réalisé avec succès.
- **Scénario principal** (normal) est un scénario typique de succès. Il raconte, étape par étape, l'histoire des interactions entre acteur et système, dans le meilleur des cas au plus précisément dans le meilleur des scénarios

- **Les extensions** : correspondent aux autres scénarios possibles, avec branchements et incidents de parcours, conduisant aussi bien au succès qu'à l'échec. Ils sont de la forme « condition : étapes numérotées »

Ainsi, pour notre cas d'utilisation ci : gérer le Bateau, gérer programme, leurs représentations textuelles est la suivante : Acteur principal : gérant ;

- Acteur principal : gérant ou administrateur ;
- Objectif : fournir les informations sur les programmes de voyage, sur le Bateau et consulter les réservations.
- Préconditions : Introduire le log et le mot de passe
- Post condition : la mise à jour d'un programme, □ Scénario nominal :

1 : gérant clique sur le bouton gérer programme voyage (enregistrer, ajouter, modifier...) programme ;

2 : le système lui donne la possibilité de au bouton ( ) les programmes ;

3: le gérant valide une modification apporté sur un programme ;

4 : Le système enregistre les informations ou modifications apporté sur un programme ;

- Exigences supplémentaires : le administrateur doit s'authentifier

Pour le cas d'utilisation consulté la réservation, la représentation textuelle se présente comme suit :

- Acteur principal : Administrateur ;
- Objectif : voir la réservation d'un billet de la part du client et voir un client qui a annulé sa réservations.
- Préconditions : Introduire le login et le mot de passe □ Post condition : voir la réservation pour un voyage.
- Exigences supplémentaire : le gérant doit s'authentifie

Pour le cas d'utilisation voir programme, la représentation textuelle se présente comme suit :



- Acteur principal : gérant ;
- Objectif : consulter la mise à jour d'un programme voir s'il y a quelque chose a modifié ou non sur les programmes precedents.
- Préconditions : aucune □ Post condition : aucune □ Scénario nominal :  
1 : l'utilisateur démarre la page ;  
2 : voit tous les programmes
- Exigences supplémentaire : aucun

Pour le cas d'utilisation créer compte, la représentation textuelle se présente comme suit :

- Acteur principal : client ;
- Objectif : le client ouvre son propre compte pour pouvoir réserver un billet de voyage ;
- Précondition : avoir un espace dans le système ;
- Post condition : le compte déjà enregistré dans la base de donnée ; □ Scénario nominal :  
1 : le client complète tous les champs requis pour la création d'un compte client puis va cliquer sur le bouton créer compte ;      2 : soit le système accepte ses informations soit non
- Extension :  
1-2a : le client ignore de compléter un ou plusieurs champs ;  
3: le système signale après saisie à l'administrateur qu'il y a des champs vides, le cas d'utilisation lui donne une seconde chance de saisir ses informations.  
1 : le système enregistre les informations complètes saisies par le client.
- Exigences supplémentaires : le bouton vers le formulaire de création du compte doit être bien visible à la page d'accueil.

Pour le cas d'utilisation voir programme, la représentation textuelle se présente comme suit :

- Acteur principal : client ;
- Objectif : voir tous les détails,
- Préconditions : aucune,
- Post condition : aucune, □ Scénario nominal :  
1 : démarrer la page  
2 : voir programmes du bateau

- Exigences supplémentaire : aucun

### 2.3.4. Diagramme de séquence

1. Ce diagramme est une représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans UML. Les diagrammes de séquence sont organisés en fonction du temps, Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence de messages [Pascal Roques, 2008] : *UML2 Modéliser une application Web, Editions Eyrolles, 4e Edition, 2008 p.22*

#### a) Diagramme de séquence connexion

Ce diagramme de séquence connexion montre comment l'utilisateur peut arriver à se connecter dans notre application, l'utilisateur commande tout d'abord la page d'authentification le système lui affiche la page, saisie du login et de mot de passe, vérification de son login et mot de passe puis le système lui affiche la page d'authentification

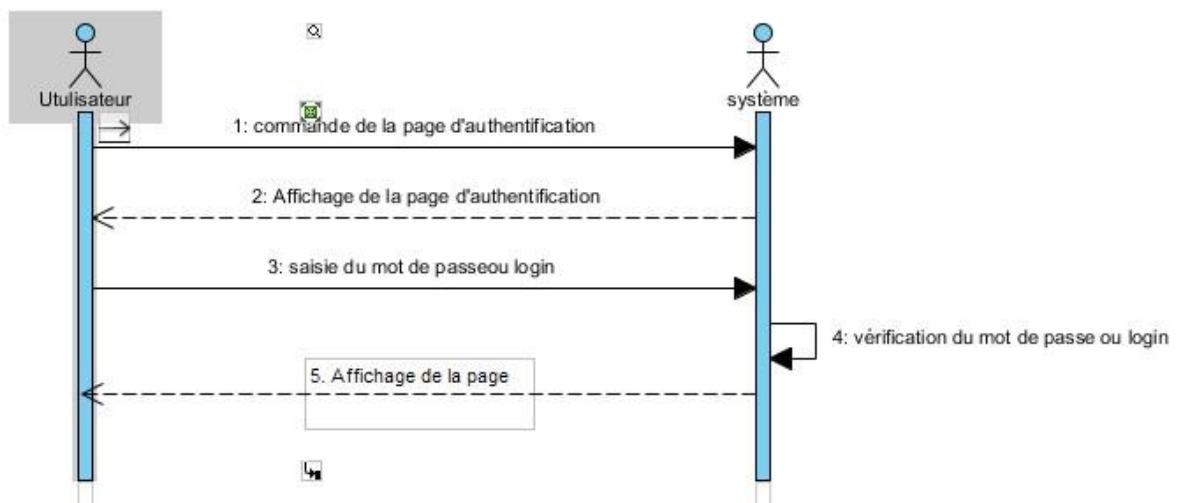


Figure 3: Diagramme de séquence Connexion

#### b) Diagramme de séquence enregistrement

Ce diagramme de séquence enregistre montre comment l'utilisateur peut arriver à l'enregistrement dans notre application en accédant tout d'abord à l'authentification et l'affichage, en suite à l'affichage de la page personnelle et remplit le formulaire puis l'envoi des données saisie à la base de données puis l'étape du traitement de la requête de l'utilisateur après que les données sont enregistrées le système envoie à l'utilisateur un message de confirmation.

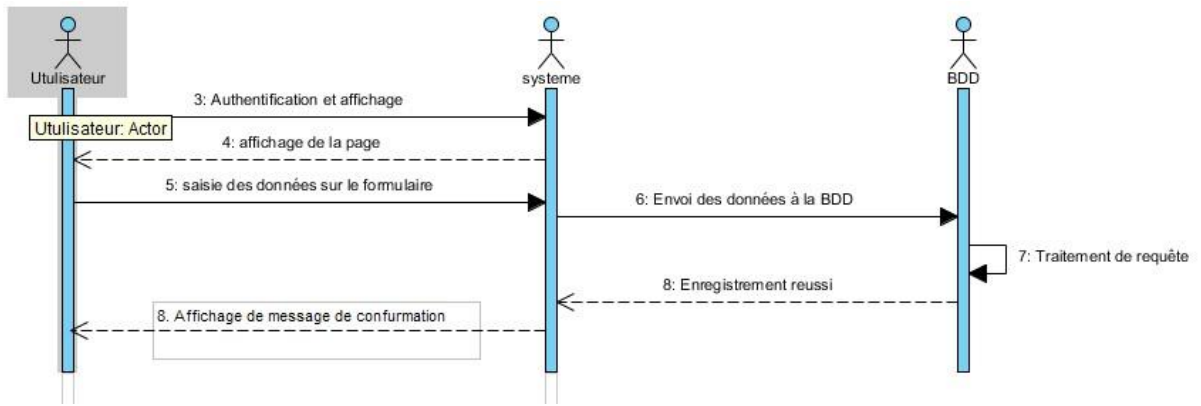


Figure 4: Diagramme de Séquence Enregistrement

### c) Diagramme de séquence modification

Ce diagramme de séquence modification montre comment l'utilisateur peut arriver à modifier les données entrés dans notre application en accédant tout d'abord à l'authentification et l'affichage, puis le système affiche la page personnelle et l'utilisateur saisie l'élément souhaité puis il sa requête a la base de données ,l'étape du traitement de la requête de l'utilisateur pendant 10 seconde max après le chargement des données puis l'affichage des données, l'utilisateur saisie en suite les nouveaux éléments sur un formulaire, envoi de ces élément a la base de données sa enregistre puis revoie un message de confirmation

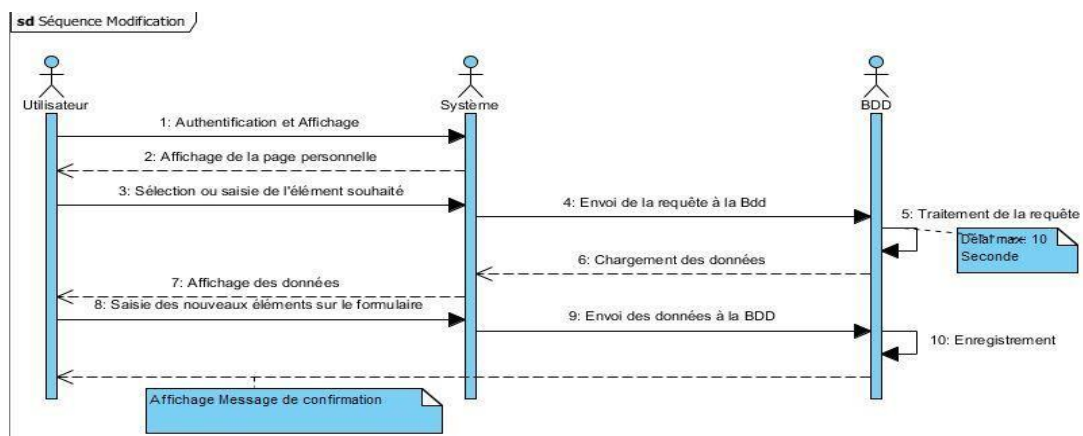


Figure 5: Diagramme de séquence Modification

### d) Diagramme de séquence recherche

Ce diagramme de séquence recherche montre comment l'utilisateur peut arriver à effectuer sa recherche dans notre application, en accédant tout d'abord à l'authentification et l'affichage, en suite à l'affichage de la page personnelle et on peut soit sélectionner soit saisir l'élément souhaité puis l'envoi des données saisie à la base de données puis l'étape du traitement de la

requête de l'utilisateur qui prendra 10 seconde puis on passe à la phase chargement de données, les données que cherche l'utilisateur s'affichent.

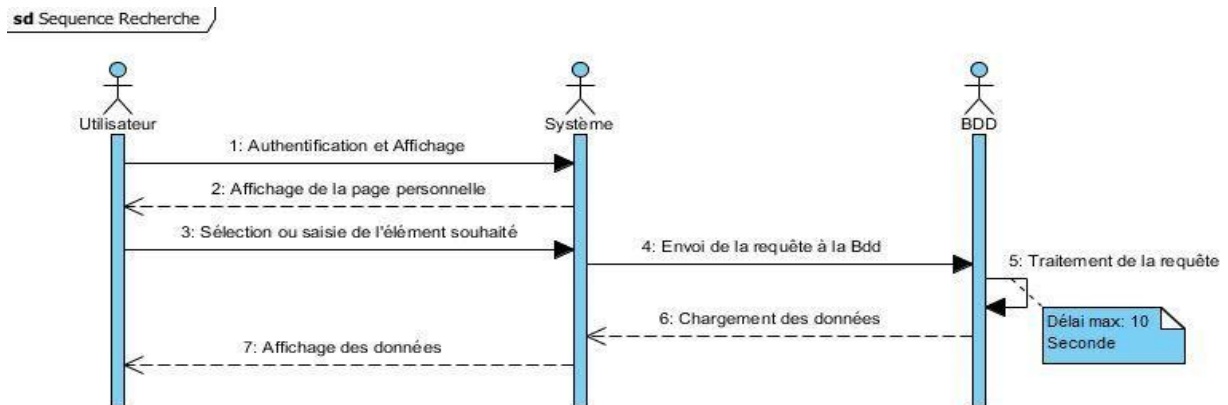


Figure 6: Diagramme de séquence rechercher

### 2.3.5. Le diagramme d'activité

1. Le diagramme d'activité est un diagramme comportemental d'UML, permettant de modéliser le comportement du système dont la séquence des activités et leurs exécutions, ce diagramme permet de déclencher les événements en fonction des états du système. Le diagramme d'activité est également utilisé pour décrire un flux de travail (workflow) [Joseph G, UML 2] Joseph Gabay, *Analyse et conception, notion des acteurs Paris 2008 p.25*

#### Activité authentification

Ici l'utilisateur lance l'application puis l'application s'ouvre affiche l'interface ou l'utilisateur va saisir le login et mot de passe, le système vérifie les éléments fournis par l'utilisateur alors, si le login et le mot de passe sont correct le système te donne l'espace du travail fin sinon il envoie un message d'erreur.

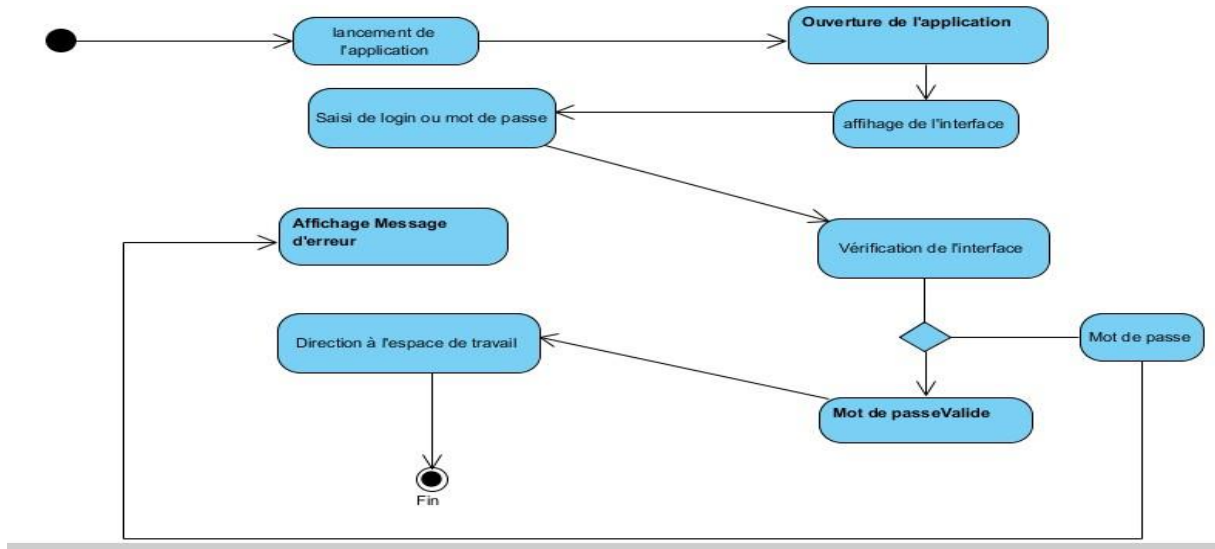


Figure 8: Diagramme d'Activité Connexion

### b) Activité Enregistrement

Pour y accéder on lance l'application nous verrons un formulaire dit ajout de X éléments, on saisit une nouvelle information en l'envoi dans la BDD le système vérifie la validation de données une fois que ses données sont vraie les données restent stocker dans la BDD et on confirme par un message fin, dans le cas contraire o t'affiche un message d'erreur et tu rentres la phase saisie de nouvelles infos

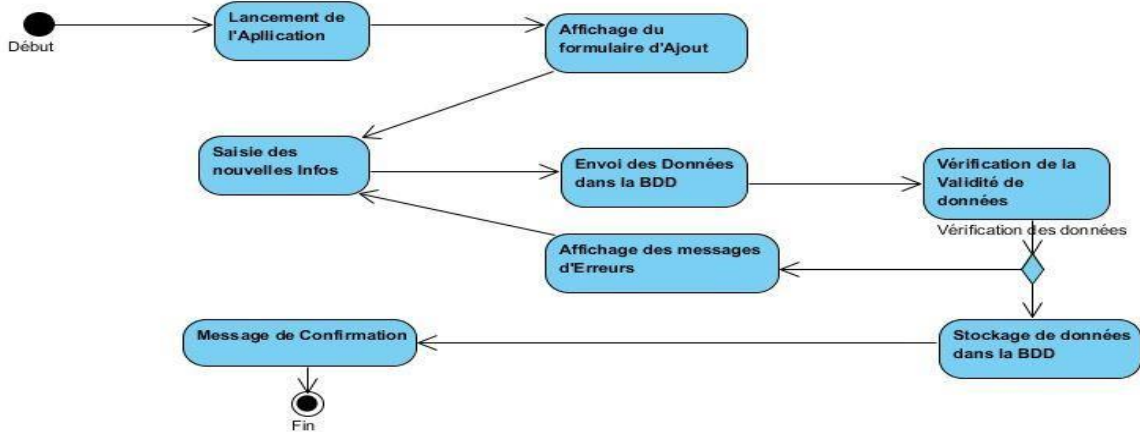


Figure 9: Diagramme d'Activité Enregistrement

### c) Activité Modification

On lance l'application et la page de recherche pour voir ce que l'on veut modifier la page s'affiche on sélectionne un bateau et ses information possible on envoie notre requête à la BDD, phase traitement de requête une fois trouver on l'affiche, puis l'utilisateur saisie des nouvelles

infos puis il envoi ça dans la BDD les données sont vérifiées restent stocker et on renvoi un message de confirmation de cette modification à l'utilisateur, sinon retour à la phase saisie ou sectionner

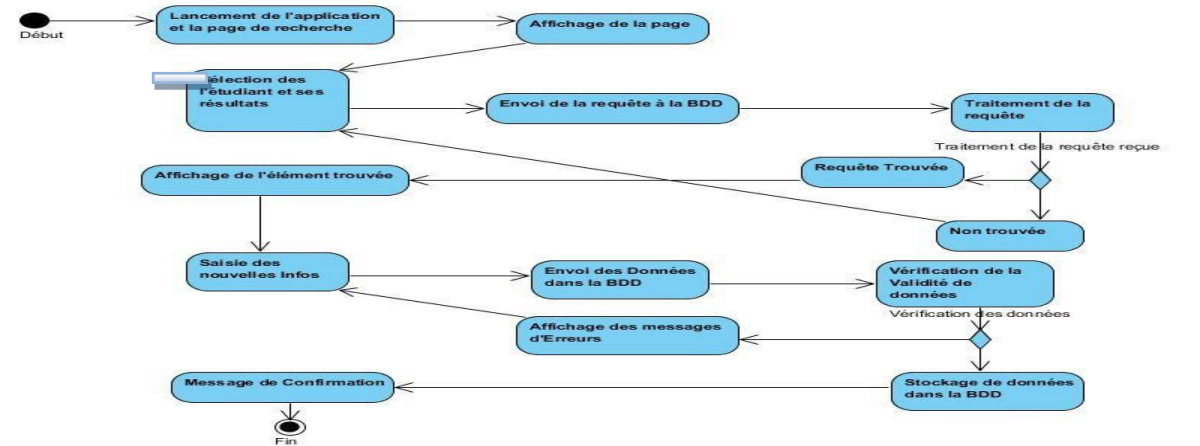


Figure 10: Diagramme d'Activité Modification

#### d) Activité Recherche

Ici on lance le l'application et la page de recherche, la page s'affiche en suite on sélectionne l'élément rechercher on envoi la demande à la BDD, une fois le traitement fait et qu'on trouve l'élément le système nous l'affiche fin, si non il nous ramène à la phase saisie ou sélectionner élément rechercher

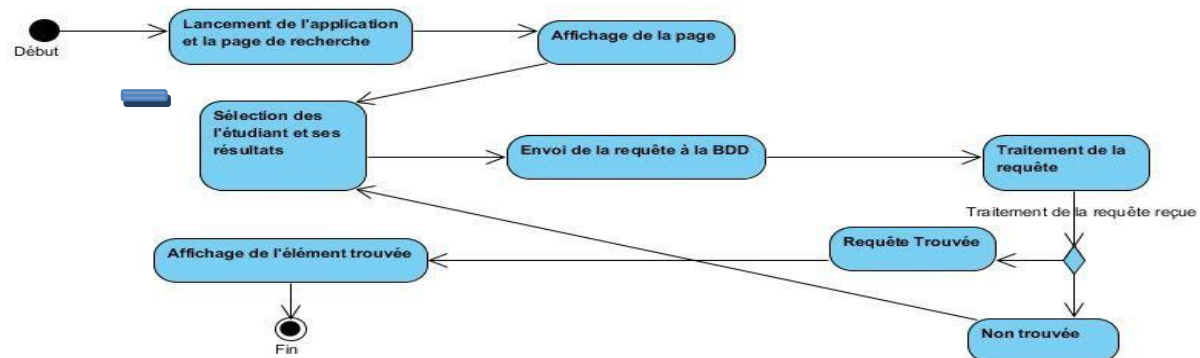


Figure 11: Diagramme d'Activité recherche

#### 2.3.6. Diagramme de classe

Les objets du système sont décrits par des classes dont une forme simplifiée de la représentation en UML. Cette représentation est constituée de trois parties.

- La première partie contient le nom de la classe.
- La deuxième partie contient les attributs.

- La troisième partie contient les méthodes.
2. Le diagramme de classe décrit la structure du système en particulier (décrit la structure des entités manipulées par les utilisateurs) en modélisant ses classes, attributs, opérations et relations entre objets, En conception, le diagramme de classes représente la structure d'un code orienté objet, Tandis qu'un diagramme de C.U représente le système du point de vue des acteurs, le diagramme de classes en décrit sa structure interne. Il permet ainsi de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation. Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet. Une classe représente un ensemble d'objets ayant les mêmes caractéristiques (attributs, méthodes, relations). [Joseph G, UML 2] Joseph Gabay, *Analyse et conception, notion des acteurs Paris 2008 p. 28*

Ainsi dans le cas qui est le nôtre, le diagramme de classe suivant peut-être établi :

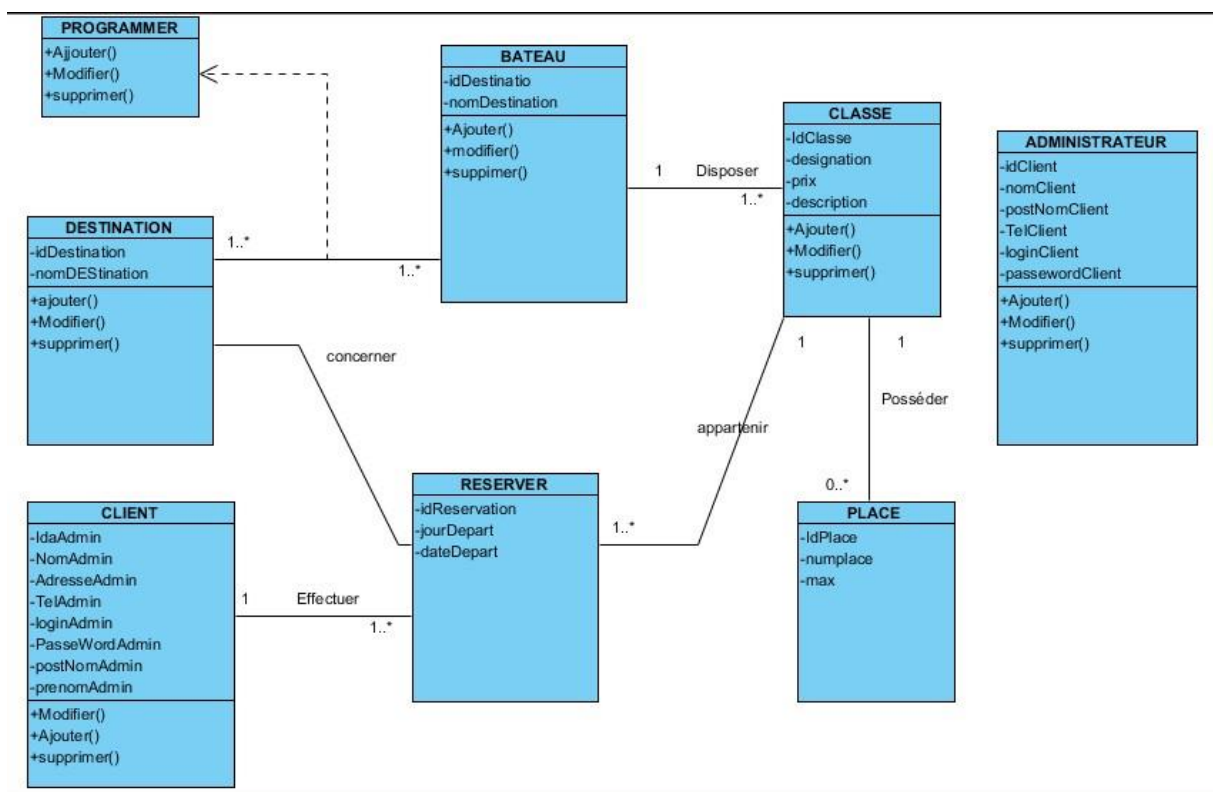


Figure 12: Diagramme des classes

## 2.4. Passage du modèle conceptuel au modèle relationnel

Une base de données est constituée d'un ensemble de tables, chacune de ces tables est composée de champs des données et des enregistrements. Nous savons que, un diagramme de classe ne

connaît pas la notion de table, et qu'une base des données ne connaît pas aussi le concept des classes reliées entre-elles via des relations portant des signes des cardinalités. Pour cela, il existe un autre modèle relationnel de données, qui utilise essentiellement le formalisme des tables logiques.

#### **2.4.1. Règles de passage du modèle conceptuel au modèle relationnel**

1. Transformation des classes : chaque classe du diagramme UML devient une relation, il faut choisir une instance de la classe pouvant jouer le rôle de clé.
2. Transformation des associations : Une association est une relation entre deux classes (association binaire), qui décrit les connexions structurelles entre leurs instances. Une association indique donc qu'il peut y avoir des liens entre des instances des classes associées. Nous distinguons trois familles d'associations.
  - a) Association 1...\* : on ajoute 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.
  - b) Association 0...\* : association binaire, la classe-association devient une relation. La clé primaire de cette relation est la concaténation des identifiants des classes connectées à l'association.
  - c) Association 1...1 : on ajoute un attribut de type 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 la classe connectée à l'association. Si les deux multiplicités minimales sont à un, il est préférable de fusionner les deux classes en une seule. [Pierkot 2010] En appliquant ces règles du passage d'un diagramme de classe vers un modèle relationnel, nous avons eu le schéma relationnel suivant :



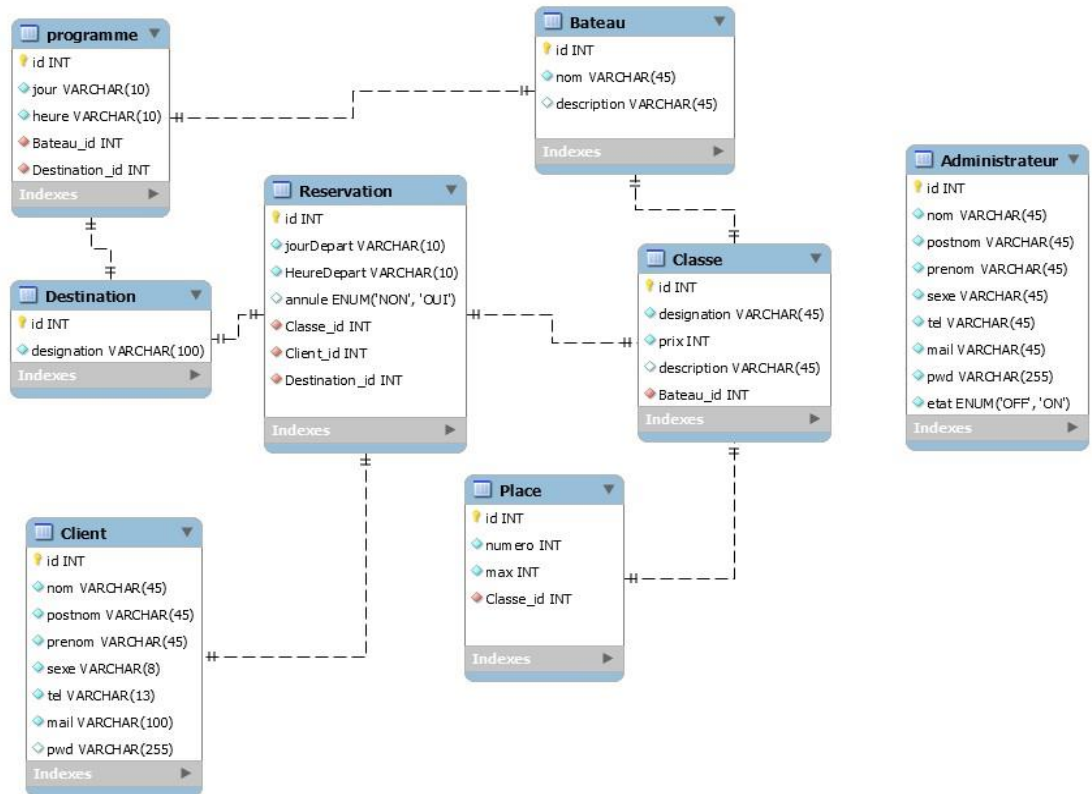


Figure 13: Diagramme des classes modèle relationnel

## 2.5. Dictionnaire de données

Nous procéderons par une collecte et analyse des informations recueillit dans différente sources et documents que nous avons enterrés tout au début de ce chapitre. Ces informations vont nous permettre d'établir le dictionnaire de données ci-dessous :

## bateau

Colonne	Type	Null	Valeur par défaut	Commentaires
id ( <i>Primaire</i> )	int(11)	Non		
nom	varchar(45)	Non		
description	varchar(45)	Oui	NULL	

### Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	0	A	Non	

## classe

Colonne	Type	Null	Valeur par défaut	Commentaires
id ( <i>Primaire</i> )	int(11)	Non		
designation	varchar(45)	Non		
prix	int(11)	Non		
description	varchar(45)	Oui	NULL	
Bateau_id	int(11)	Non		

### Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	4	A	Non	
fk_Classe_Bateau1_idx	BTREE	Non	Non	Bateau_id	1	A	Non	

## client

Colonne	Type	Null	Valeur par défaut	Commentaires
id ( <i>Primaire</i> )	int(11)	Non		
nom	varchar(45)	Non		
postnom	varchar(45)	Non		
prenom	varchar(45)	Non		
sexe	varchar(8)	Non		
tel	varchar(13)	Non		
mail	varchar(100)	Non		
pwd	varchar(255)	Oui	NULL	

### Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	3	A	Non	

## destination

Colonne	Type	Null	Valeur par défaut	Commentaires
id (Primaire)	int(11)	Non		
designation	varchar(100)	Non		

### Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	1	A	Non	

## place

Colonne	Type	Null	Valeur par défaut	Commentaires
id (Primaire)	int(11)	Non		
numero	int(11)	Non		
max	int(11)	Non		
Classe_id	int(11)	Non		

### Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	0	A	Non	
max_UNIQUE	BTREE	Oui	Non	max	0	A	Non	
fk_Place_Classe1_idx	BTREE	Non	Non	Classe_id	0	A	Non	

## programme

Colonne	Type	Null	Valeur par défaut	Commentaires
id (Primaire)	int(11)	Non		
jour	varchar(10)	Non		
heure	varchar(10)	Non		
Bateau_id	int(11)	Non		
Destination_id	int(11)	Non		

### Index

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	7	A	Non	
fk_programme_Bateau1_idx	BTREE	Non	Non	Bateau_id	1	A	Non	
fk_programme_Destination1_idx	BTREE	Non	Non	Destination_id	2	A	Non	

## reservation

Colonne	Type	Null	Valeur par défaut	Commentaires
id (Primaire)	int(11)	Non		
jourDepart	varchar(10)	Non		
heureDepart	varchar(10)	Non		
annule	enum('NON', 'OUI')	Oui	NON	
Classe_id	int(11)	Non		
Client_id	int(11)	Non		

Destination_id	int(11)	Non		
----------------	---------	-----	--	--

**Index**

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	3	A	Non	
fk_Reservation_Client1_idx	BTREE	Non	Non	Client_id	3	A	Non	
fk_Reservation_Classe1_idx	BTREE	Non	Non	Classe_id	2	A	Non	
fk_Reservation_Destination1_idx	BTREE	Non	Non	Destination_id	2	A	Non	

**administrateur**

Colonne	Type	Null	Valeur par défaut	Commentaires
id (Primaire)	int(11)	Non		
nom	varchar(45)	Non		
postnom	varchar(45)	Non		
prenom	varchar(45)	Non		
sexe	varchar(45)	Non		
tel	varchar(45)	Non		
mail	varchar(45)	Non		
pwd	varchar(255)	Non		
etat	enum('OFF', 'ON')	Non		

**Index**

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
PRIMARY	BTREE	Oui	Non	id	2	A	Non	

**Tableau2 : Dictionnaire des données**

**2.6. Les règles de gestion**

Elle précise les contraintes qui doivent être respectées par le modèle, elle sert à définir l'ensemble des normes à respecter par les acteurs ; la définition de règle de gestion mène parallèlement au recueil des données permet la mise en place des relations entre les objets :(Jules M 2006)

Le diagramme de classe pour notre système est basé sur les règles de gestion, ces règles de gestions sont dans un tableau, description des associations suivantes :

Association	Désignation	Classe	Multiplicités
<b>programmer</b>	Un bateau peut être programmé pour une ou plusieurs destinations	Bateau	1..*
		Destination	1..*
<b>Disposer</b>	Un bateau peut disposer une ou plusieurs classes	Bateau	1..*
		classes	1
<b>Effectuer</b>	Un client peut effectuer une Ou plusieurs réservations	Client	1
		Réservation	1..*
<b>concerner</b>	Une Classe peut concerner une ou plusieurs réservations	Classe	1..*
		réservation	1
<b>Posséder</b>	Une Classe peut posséder une ou plusieurs places	classe	1
		place	1..*

*Tableau3 : Association et multiplicités*

Cette phase nous a permis d'identifier le processus de fonctionnement de l'application, les principaux cas d'utilisation, les diagrammes de séquence, diagramme d'activité le diagramme de classe et règles de gestion. Le chapitre suivant sera consacré à la présentation de la partie développement de notre application.

## CHAPITRE III : DEVELOPPEMENT ET IMPLEMENTATION DU SITE WEB

Au cours de ce chapitre, il nous est donné l'occasion d'expliquer le fonctionnement de notre logiciel pour une bonne utilisation. Il est donc question de montrer les bases de son fonctionnement, son schéma architectural sa compatibilité et présenter les langages utilisés et principaux écrans de notre application. Ce chapitre va permettre à l'utilisateur de ce système a mieux manipule l'application ainsi que connaitre son environnement de travail.

### 3.1. Environnement informatique

L'environnement système est une configuration matérielle et logicielle propre à un type d'ordinateur. Les caractéristiques matérielles et logicielles prises en compte pour la réalisation de l'application sont : un ordinateur avec un processeur de 2.16 GHz de fréquence et une mémoire RAM de 4.00 GB et un système d'exploitation WINDOWS10 de 64bit.

### 3.2. Diagramme de déploiement

Le diagramme de déploiement permet de représenter l'architecture physique supportant l'exploitation du système. Cette architecture comprend des nœuds correspondant aux supports physiques (serveurs, routeurs, switch etc.) ainsi que la répartition des artefacts logiciels (bibliothèques, exécutable, etc.) sur ces nœuds. C'est un véritable réseau constitué de nœuds et de connexions entre ces nœuds qui modélise cette architecture. [Joseph Gabay, 2008]

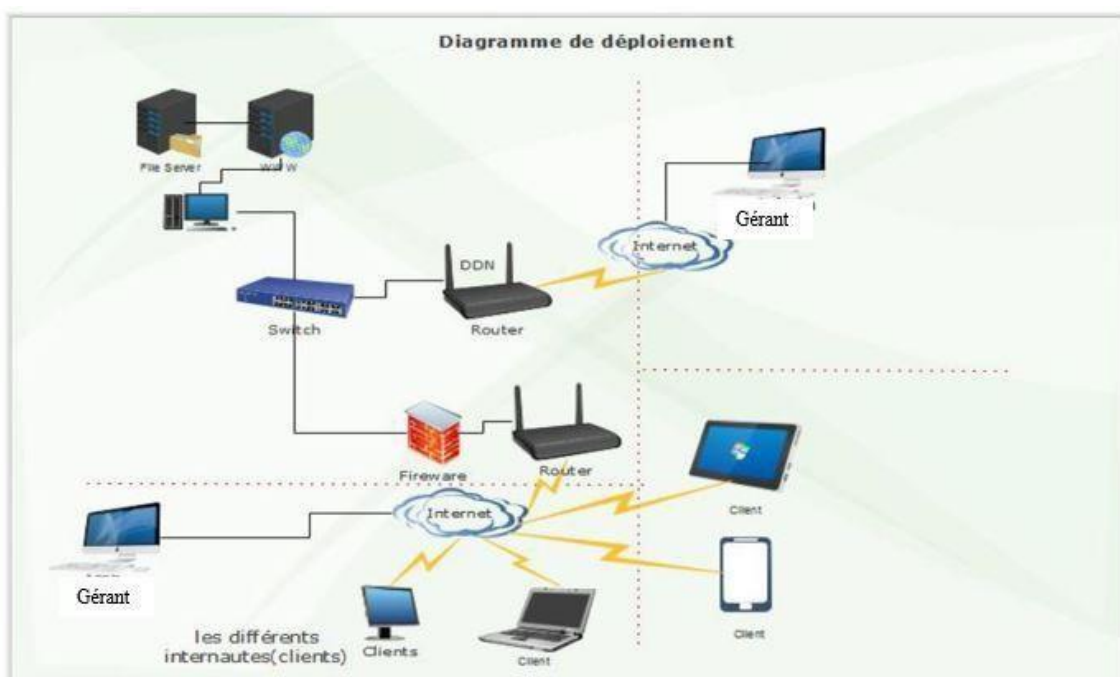


Figure 14: Diagramme de déploiement

### 3.3. Langages de programmation utilisés

Il existe plusieurs langages de programmation utilisés pour créer des sites Internet, mais parmi eux nous avons utilisés quelques-uns :

HTML	<i>(HyperText MarkupLanguage)</i> : permet de gérer et d'organiser un texte C'est en HTML qu'on peut écrire ce qui doit être affiché sur la page : du texte, des photos, liens et autres [Francis 2008]	C'est grâce à ce langage que nous avons construit nos pages et que nous avons fait apparaître des écrits, des photos...
css	<i>CascadingStyle Sheets</i> , « cascade » est la combinaison des différentes sources de styles appliqués à un même document dit feuille de style. Cette feuille de style fournit la mise en forme des contenus d'une page HTML, Elle s'applique à une ou plusieurs pages du site elle donne un bon aperçu de la page. [Francis 2008] cette feuille est en d'autre terme dit la mise en forme de nos page web	Grace à cette feuille nous avons fait le design des nos pages HTML
MySQL	C'est un système de gestion de base de données. Son rôle est de stocker et de gérer une grande quantité des données en les organisant sous forme de tables, et de permettre la manipulation de ces données à travers les langages de requête SQL. [Philippe R 2009]	sert à créer et gérer la Bdd.
PHP	<i>(PHP HypertextPreProcessor)</i> c'est un langage de script. Ça nous permet de créer un sites web dynamique. Nous aide à créer une de pages HTML, des fichiers PDF, PHP intègre des possibilités de connexion à la majorité des bases de données (Oracle, SQL Serveur, MySQL, dBaseetc.), [piere 2008]	PHP joue le rôle d'intermédiaire entre notre base de données (MYSQL) et notre page (HTML).




JavaScript	JavaScript est un langage de script côté client, ce qui signifie qu'il s'exécute côté client, dans un navigateur Web. JavaScript peut aussi être employé côté serveur et en dehors d'un navigateur. Si le navigateur est compatible, JavaScript donne accès à la page en cours et permet au script de déterminer les propriétés du client, de rediriger l'utilisateur vers une autre page, d'accéder aux cookies, etc. [Christian soutou 2009]	Nous a servi à afficher quelques messages, principalement les messages de confirmations et d'erreurs.
------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------

Tableau4 : langages de programmation utilisés dans notre projet de réalisation d'un site web

### 3.4. Principaux logiciels utilisés

Dans la réalisation de ce projet de mémoire nous avons fait recours a quelques logiciels :

- ✚ MS Office Word 2010 : c'est un logiciel de traitement de texte, ça nous a servi à saisir le travail, faire sa mise en forme et la mise en page.
- ✚ Notepad++ version 7.5. : Notepad++ est un éditeur de texte générique codé en C++, nous a servi à taper nos différentes lignes de codes (PHP, SQL, CSS, HTML, JavaScript).
- ✚ Visual paradigm version 10.2. : qui est un logiciel permettant aux programmeurs de mettre en place des diagrammes UML, nous a permis à produire les différents modèles que nous avons présenté dans le chapitre précédent.
- ✚ MySQL WorkBench version 6.0. : qui est un logiciel de gestion et d'administration de bases de données MySQL et nous permis à créer notre base de données et à produire un Script SQL.
- ✚ Wamp server version 3.0.6. : WampServer est une plateforme de développement Web permettant de faire fonctionner localement des scripts PHP. WampServer est un environnement de travail comprenant deux serveurs (Apache et MySQL) est un interpréteur (exécute) de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL. Il nous a servi comme un serveur local pour l'exécution et le test de nos Scripts. Avant l'utiliser WampServer, il faut s'assurer que celui-ci avait été bien installé en suite bien lancé. Il faut que l'icône prenne la couleur verte, comme indiqué cette image .
- ✚ Mozilla Firefox version 60.2: qui est un navigateur web libre et gratuit, nous a servi à faire le test de notre site web en local.



### 3.5. Le schéma architectural de l'application ; design Pattern : Modèle-VueContrôleur (MVC)

**MVC signifie Modèle / Vue / Contrôleur.** C'est un découpage couramment utilisé pour développer des applications web. Pour générer une page, un contrôleur réalise systématiquement des opérations basiques telles que lire des données, et les afficher. Avec un peu de capacité d'abstraction, on peut voir deux autres couches qui apparaissent : une pour gérer les données (notre **modèle**) et une autre pour gérer l'affichage des pages (notre **vue**). [Savageman 2010] en outre, le contrôleur est considéré comme le « **chef d'orchestre** » : il récupère la demande de l'utilisateur à travers la vue (la page HTML) par un href, un formulaire ou un header. Il échange des données avec le modèle, fait les calculs (qui peuvent être complexes) puis choisit une vue à afficher en lui fournissant les variables.

**Modèle** : c'est une couche qui gère les données du site, sa récupère les informations dans la base de données, les organise pour permettre au contrôleur de faire un bon traitement [Savageman 2010].

**Contrôleur** : c'est la couche qui gère la logique du code qui prend des décisions, sa joue l'intermédiaire entre **Modèle** et **Vue**, il récupère les informations venant dans **Modèle** il les analyse puis il renvoie le texte à afficher sur la vue

**Vue** : c'est une couche d'affichage d'une page, sa récupère la variable pour savoir ce qu'il va afficher [Savageman 2010].

Nous sommes motivés à utiliser cette technologie (MVC) car elle nous a permis de bien organiser nos codes sources. Son but, est de séparer la logique du code en trois parties

Une simple image qui explique notre modèle MVC ci haut détaillé :

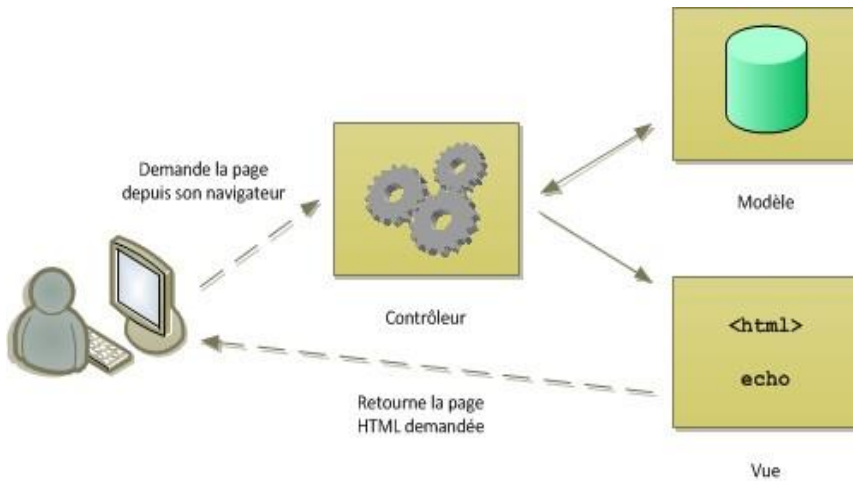


Figure 15 : Diagramme de déploiement

### 3.6. Capture des principaux écrans de l'application

Dans cette partie, nous allons présenter certaines pages de notre application face auxquelles se trouvera l'utilisateur. Cette implémentée est simple et souple à utiliser. Tout commence d'abord par la page d'accueil qui se présente ainsi :

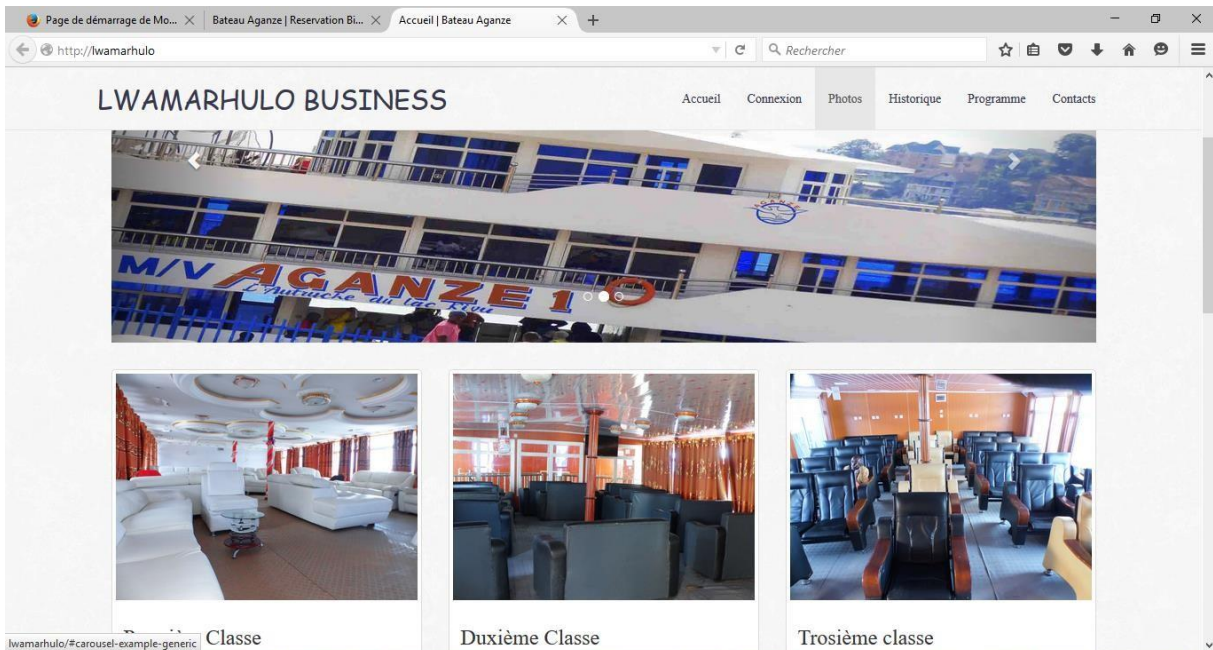


Figure 16 : Page d'accueil

Sur cette page d'accueil nous voyons juste au-dessus, cinq petits boutons (programme, connexion, photos, historique et contact), sur la page cliquer sur le bouton « PROGRAMME » tous les programmes des bateaux inscrits sur ce site vont s'afficher comme suit :

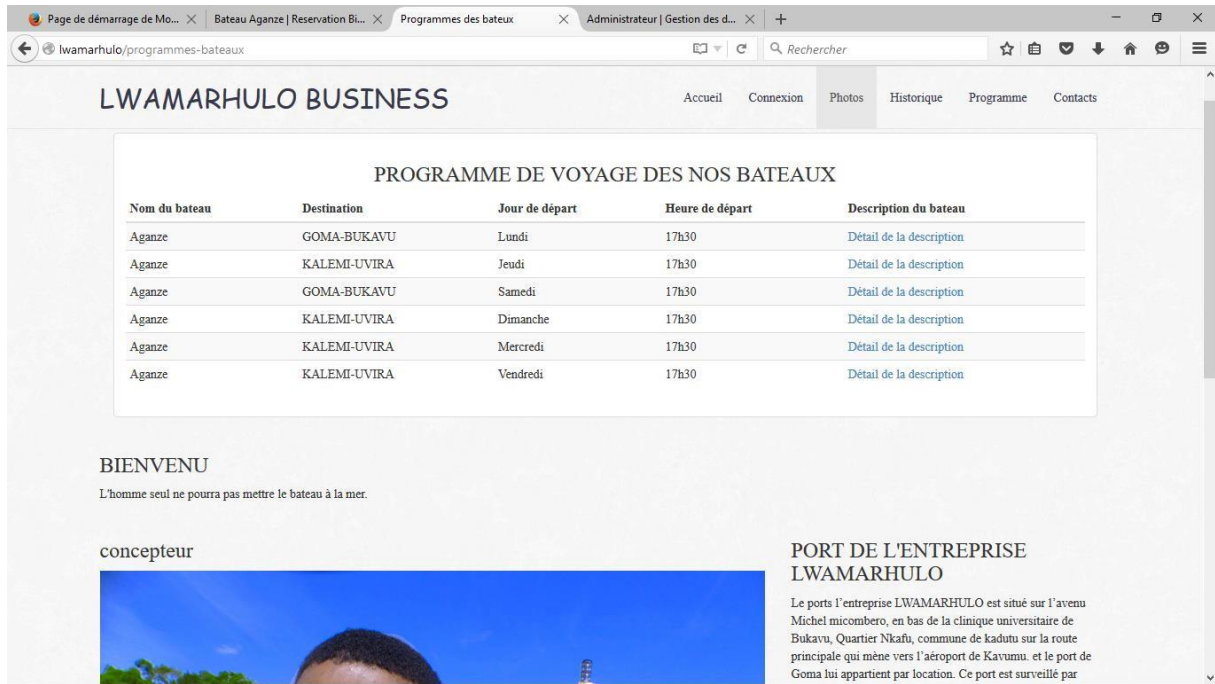


Figure 18 : programme des bateaux

Toujours sur page d'accueil cliquer sur le bouton du milieu « CONNEXION » Ce bouton contient des options proposées selon le besoin de l'utilisateur comme : (**connexion client, connexion administrateur, créer compte client, créer compte administrateur**) et la page s'affiche de la manière suivante :

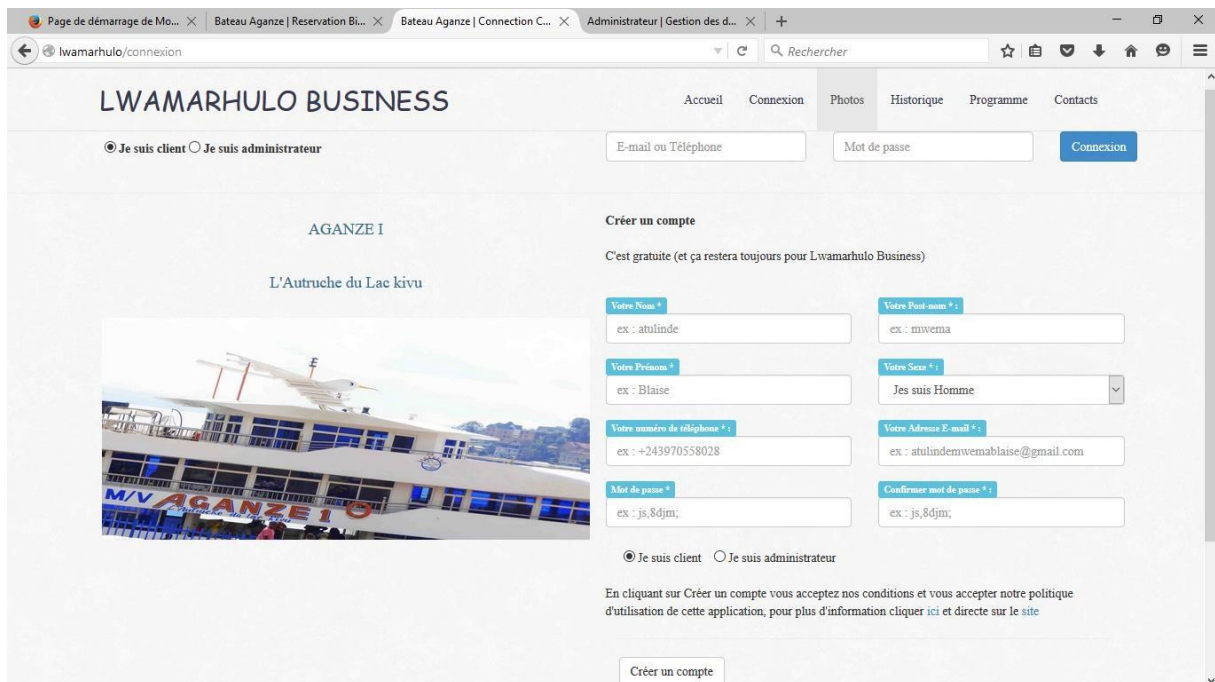


Figure 19 : création compte

Si vous êtes un nouveau client, vous allez pouvoir compléter tous ces champs et en suite vous cliquez sur le bouton <<CREER COMPTE>>, si tout se passe bien, un message vous sera envoyé :



Si non, donc il y a erreur en complétant l'un ou plusieurs champs. Et le champ ayant un problème va s'afficher de cette façon :

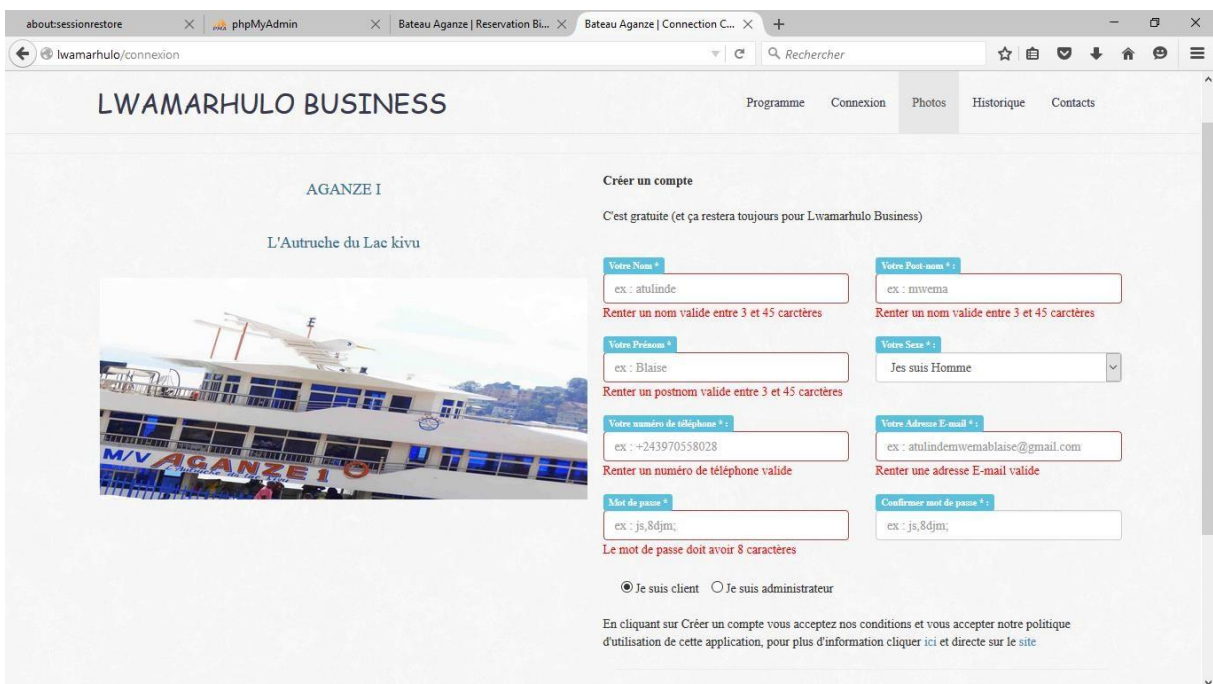
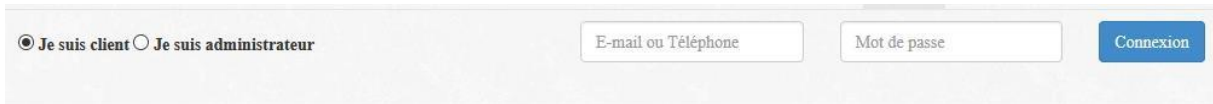


Figure 20 : création compte en cas d'erreur

Votre compte est déjà créé, vous aller directement à haut de la page pour compléter ceci :



Vous précisez d'abord si vous êtes un client ou un administrateur, en suite vous complète la case login par votre numéro de téléphone ou votre adresse mail que vous saisissez lors de la création du compte et complétez la case du mot de passe. Et pour terminer, vous cliquez sur le bouton <<connexion>>. S'il y a eu erreur, un message pareil se présente:



Si non, vous entrez dans votre espace propos au sein du système ou vous pouvez effectuer votre réservation qui se présente de la manière suivante :

ASHAH Ahah

RESERVER VOTRE BILLET ICI

votre réservation est prise avec succès

Jour de départ : ex : 15/05/2019

Heure de départ : 17h30

Quelle destination : GOMA-BUKAVU

Dans quel bateau et quelle classe ?

Quelle classe voulez-vous au sein de bateau Aganze ?

Classe 1 Prix = 20

Réserver votre billets Voir les programmes des bateaux

Figure 22 : réservation billet

Ici, le client peut maintenant effectuer sa réservation en remplissant tous ces champs, entre temps : la date départ, l'heure de départ, la destination mais aussi en choisissant le bateau et la classe de votre choix. Et pour terminer, il clique sur le bouton <<RESERVER VOTRE BOUTON>> et le message lui sera envoyé <<votre réservation est prise avec succès>>

Toujours dans son espace, l'utilisateur peut aussi annuler ou modifier sa réservation. Il suffit qu'il clique sur le bouton <<VOS RESERVATIONS>> et une page s'ouvre de la sorte :

ANTOINE Antoine

VOS RESERVATIONS

Jour	Heure	Bateau	Classe	Destination	Annuler	Annuler ?
24/07/2019		AGANZE 1	Classe 2	GOMA-BUKAVU	NON	Modifier

Figure 23 : annulation réservation

L'utilisateur peut annuler en cliquant sur le bouton <<Annuler>>. Après avoir effectué tout ce qu'il faut dans sa page, il peut maintenant se déconnecter en cliquant sur le bouton <<DECONNEXION>> pour revenir à la page d'accueil.

Et du compte administrateur, il peut aussi se connecter en suivant la même logique que celui du client mais en précisant qu'il est administrateur. En cliquant sur le bouton <<connexion>> qui va lui diriger dans son espace, sa page s'ouvre de cette manière :



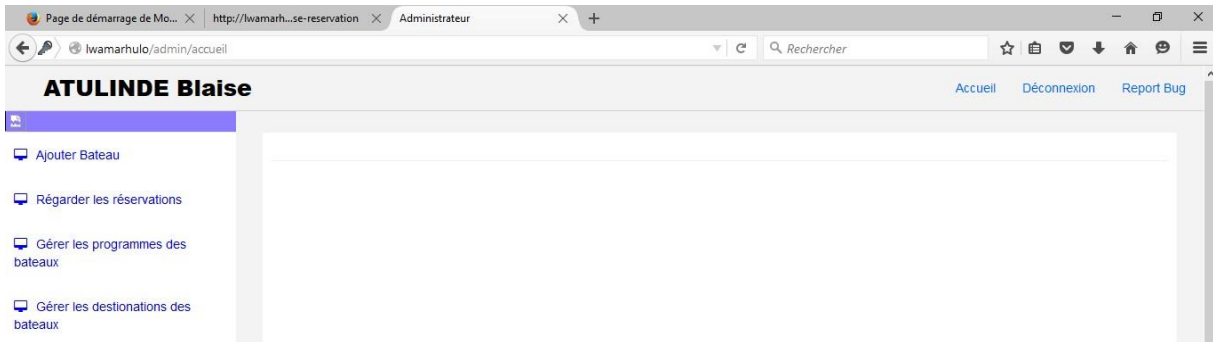


Figure 24 : Espace administrateur

Dans espace, il peut ajouter le bateau en cliquant sur le bouton <<AJOUTER BATEAU>>. Et la page apparait de cette façon :

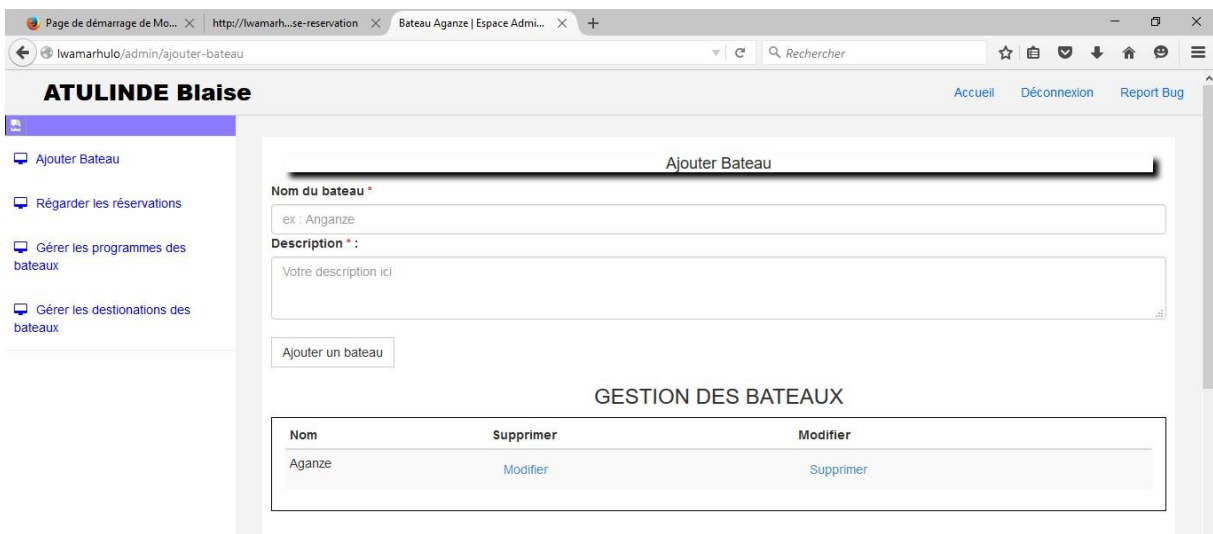


Figure 25 : gestion des bateaux

La page est déjà affichée, il peut maintenant compléter les champs dont le nom du bateau et la description du bateau (nombre de classes, la taille du bateau,...). Et pour terminer, il clique sur le bouton <<AJOUTER BATEAU>>, mais il peut modifier le bateau ou le supprimer. Pour le faire il suffit qu'il clique sur les boutons <<MODIFIER OU SUPPRIMER>> et le message suivant se présente:



si il clique sur oui, le bateau est supprimé et la même procédure pour modifier un bateau, puisque il peut y arriver un moment que le bateau peut changer son nom, tel est le cas du bateau MIS RAFIKI, aujourd'hui ETOILE DU KUVU.

Toujours dans son espace, l'administrateur peut consulter tous les clients ayant effectué leurs réservations lors d'un voyage et ceux qui les ont annulées. Il suffit qu'il clique sur le bouton <<Regarder les réservations>> et la page s'ouvre de cette manière :

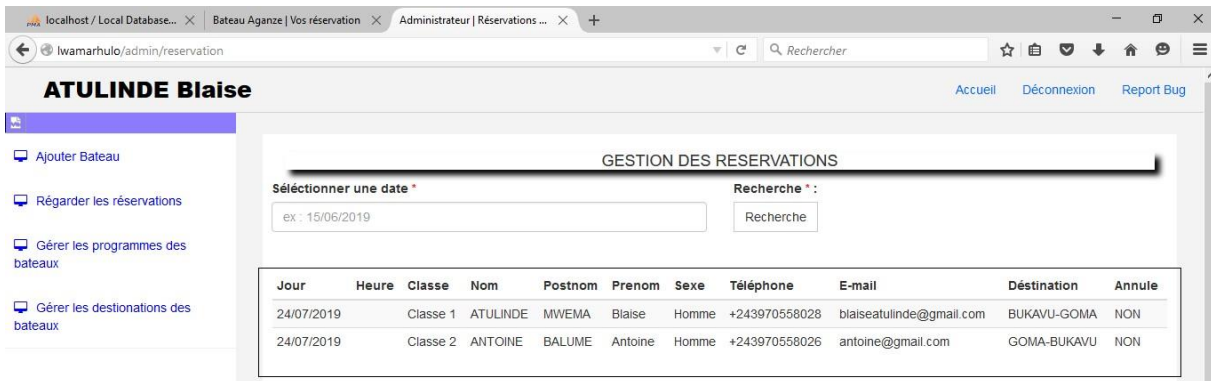


Figure 26 : consulter les réservations

Egalement, l'administrateur peut gérer les programmes des bateaux en cliquant sur << gérer les programmes des bateaux>> et la fenêtre suivante s'ouvre de la sorte :

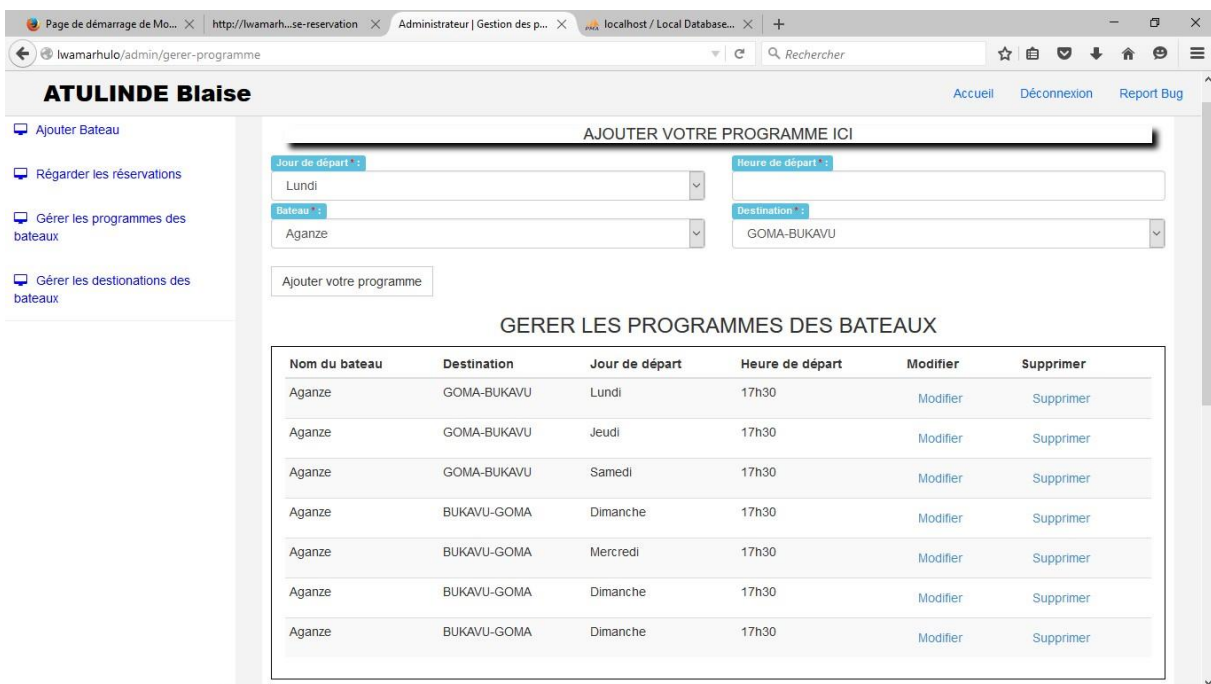


Figure 27 : gestion programme

Ici il peut ajouter, modifier et supprimer le programme d'un bateau. Pour ajouter un programme, l'administrateur doit pouvoir compléter tous les champs et en suite il clique sur le bouton <<AJOUTER VOTRE PROGRAMME>>.

Pour modifier un programme, l'utilisateur doit cliquer sur le bouton <<modifier>> d'un programme de son choix et il modifie le champ qu'il veut.

Et en fin pour supprimer, l'utilisateur doit cliquer sur le bouton supprimer un programme de son choix et modifier le champ qu'il veut. Mais avant qu'il supprime un programme, un message doit d'abord s'afficher de la sorte :



Sans oublier aussi qu'un utilisateur peut ajouter, modifier et supprimer la destination d'un bateau. Pour y arriver, dans son espace il doit cliquer sur le <<GERER LES DESTINATIONS DES BATEAUX>> et la page suivante s'ouvre :

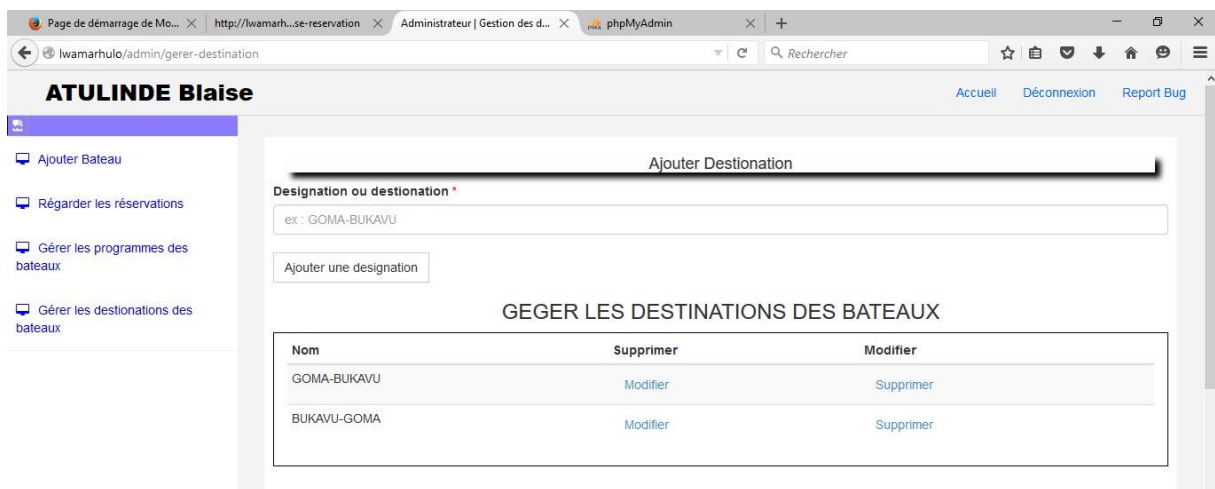


Figure 28 : gestion destination

Pour ajouter, il doit compléter le champ destination et puis cliquer sur le bouton <<AJOUTER UNE DESTINATION>> et pour supprimer ou modifier, il doit supprimer ou modifier la destination de son choix comme l'image le montre ci-haut.

### 3.7. Estimation du Coût de l'application

Pour estimer le coût de notre application, nous avons fait recours à la méthode COCOMO. Ce dernier est un acronyme pour COConstructiveCOst Model. C'est une méthode pour estimer le coût d'un projet logiciel dans le but d'éviter les erreurs de budget et les retards de



livraison, qui sont malheureusement habituels dans l'industrie de développement logiciel [AUBONNET 2017]. La méthode COCOMO est issue du modèle en Spirale pour la planification des projets qui définit quatre cadrans dans chaque spire dont un seul pour le développement et trois pour la gestion du projet. Ce dernier, à l'avantage d'être ouvert. Les données de calibrage, les formules et tous les détails des définitions sont disponibles. La participation à son développement est encouragée [AUBONNET 2017]

$$\text{Charge en mois/Homme} = a (\text{Kisl})^b$$

Kisl = kilo instruction source livrée (lignes de programme source testées, sans tenir compte des commentaires dans les lignes de code)

$$\text{Durée normale en mois} = c(\text{charge en mois/Homme})^d$$

Les paramètres a, b, c et d dépendent de la catégorie du projet. Soit I la taille du logiciel. Projet simple si  $I < 50$  Kisl, spécifications stables, petite équipe. Projet moyen logiciel comporte entre 50 Kisl et 300 Kisl (spécifications stables, petite équipe). Projet complexe si  $I > 300$  Kisl, grande équipe [AUBONNET 2017]. Ce tableau illustre mieux cette théorie :

**Tableau n°2 : Formule de la Méthode COCOMO [AUBONNET 2017]**

Type de Projet	Charge en mois homme	Durée en mois
Simple	$C = 3,2(\text{Kisl})^{1.05}$	$D = 2.5(C)^{0.38}$
Moyen	$C = 3(\text{Kisl})^{1.12}$	$D = 2.5(C)^{0.35}$
Complexe	$C = 2,8(\text{Kisl})^{1.2}$	$D = 2.5(C)^{0.32}$

*Tableau 5 : formule de la méthode COCOMO*

NB : Selon la complexité et la taille (en Kisl) de l'application, la charge et le temps de développement varient. **Ce projet est de type Simple, et nous estimons Kisl = 9** d'où :

$$C = 3.2(\text{Kisl})^{1.05} \text{ et } D = 2.5(C)^{0.38}$$

$$C = 3.2(9)^{1.05} = 32,14 \text{ mois/homme}$$

$$D = 2.5(32,14)^{0.38} = 9,34 \text{ mois}$$

$$\frac{C}{D} = \frac{32,14}{9,34} = 3,57 = 4 \text{ personnes comme taille de l'équipe}$$

Donc la taille de l'équipe de développement de l'application devrait être de 4 personnes mais il a été développé par 2 personnes (Concepteur et Directeur).

On sait que : Un homme/mois (HM) représente l'équivalent du travail d'une personne pendant un mois, généralement 20 jours ; Un homme/mois (HM)=152 heures de travail par mois. Nous estimons la rémunération à 5\$/heure, 1 jour = 8 heures et 1 mois= 20 jours.

$$4 * 8 * 20 * 9,34 = 5\,997,6 \$$$

***Nous estimons le Coût de notre application à 5 997,6 \$***

## **Conclusion générale**

Nous voici au terme de notre présent travail portant sur la conception et la réalisation d'un site web de réservation des billets de voyage au sein de l'entreprise Lwamarhulo. Une bonne condition de voyage étant le droit d'un passager, ce dernier a le plein droit de se sentir à l'aise, il a le devoir de faire tout son possible de prendre la première place et a le temps de connaître les programmes de bateaux et a l'obligation d'être ponctuel aux heures d'embarquement, mais aussi l'entreprise Lwamarhulo doit devoir publier les programmes pour qu'ils soient vus par tous les clients et de respecter les heures et les ordres d'embarquement.

En effet, nous avons collectionné ces idées, pas par notre propre intention mais par les lamentations des passagers en particulier, ceux qui sont dans l'impossibilité de faire de tours pour savoir plus sur les programmes d'un quelconque bateau de cette entreprise et pour ceux qui ne veulent pas faire la queue au port, ceux qui arrivent en retard et qui ratent leurs voyages. Face à cela, il s'est avéré d'une importance très capitale de penser aux moyens nous offert par la nouvelle technologie et c'est ce qui nous a conduit à estimer que la mise en place d'un site web de réservation des billets de voyage au sein de l'entreprise Lwamarhulo pourrait palier une telle situation, par sa publication en ligne via le réseau internet des programmes et la réservation des billets de voyage.

Pour y parvenir, nous sommes parti de la problématique, des hypothèses, des objectifs, des méthodes et techniques, et dont l'ensemble nous a conduit aux résultats.

Le travail en soi est constitué de trois chapitres.

Dans le premier chapitre, il était question de présenter le cadre de notre étude, le milieu d'étude de notre travail, la situation du transport lacustre au sein de cette entreprise. Dans le second chapitre, nous avons d'abord introduit la notion des analyses et de critique des existants, ensuite nous avons modélisé la mise en place de l'application et dans le troisième chapitre, nous avons présenté les différents écrans de notre application en question.

Les résultats auxquels nous avons abouti dans cette analyse, permettront d'éviter les problèmes liés à la réservation des billets de voyage ainsi qu'à la publication des programmes de l'entreprise. Nos analyses vont nous permettre de réduire le retard et le manque de bateau, cela permettra aux personnes n'ayant pas le temps mort de se procurer de places dans le bateau sans beaucoup de peine.

Ce travail est censé résoudre le problème de publications des programmes et de réservation des billets de voyage, plus loin encore il résoudra d'autres problèmes qui jusque-là sillonnent dans nos pensées et dont des solutions ne sont que dans nos imaginations.

Les scientifiques disent souvent « en trouvant une résolution à un problème, on crée d'autres problèmes et ce qui rend les recherches de plus en plus multiples, du jour le jour » Tout comme on dit encore « Je n'ai pas échoué, j'ai trouvé dix mille moyens qui ne fonctionnent pas », sa signifie, qu'il ne faut jamais s'arrêter il faut apprendre à foncer jusqu'à trouver la solution.

Ce travail est loin d'être le meilleur des autres et aussi même loin d'être le dernier à être réalisé, car le monde actuel connaît une avancée extraordinaire de l'informatique. Il est vrai que ce travail a des limites, en cela, nous laissons la place à nos successeurs, de bien vouloir nous compléter de toucher d'autres aspects qui n'ont pas fait l'objet du présent travail, de se servir de ce dernier pour ensuite faire face aux mille solutions qui se présenteront, et nous restons captifs vis-à-vis de toutes observations, ajouts, correction et remarques qui seront dégager à notre intention.

## Bibliographie

### A. Ouvrages

3. [Dreyfus k 2006], *developpement et exploitation durable , l'homme a la recherche du développement,2006.*
4. [Olivier S 2001] *Introduction à la modélisation orientée objet avecUML2,p15*
5. [Joseph G, UML 2] Joseph Gabay, *Analyse et conception, notion des acteurs Paris 2008 p.16,25,28,32*
6. [Pascal Roques, 2008] : *UML2 Modéliser une application Web, Editions Eyrolles, 4e Edition, 2008 p.22*
7. [Raymond Q 2011], *Raymond Quivy, méthode d'enquête et audit, paris.dunod.2011*
8. [Pierkot 2010] : *Pierkot Modélisation et stockage des données géographiques ,structure et modèle relationnel, 2010, Eyrolles, p.30*

9. [F D08] : *Francis Drailard, Premier pas en CSS ( pour de butant)*, Paris cedex 05, 2008, 2<sup>e</sup> édition, Eyrolles, p.33
10. [P09] : *Philippe Rigaux, le GUIDE MySQL et PHP*, Paris, 2009, p. 34
11. [ Pierre C 2008] : *Cyril Pierre de Geye, PHP 5 avancé, Paris, 2008, 4<sup>e</sup> édition, Eyrolles, p.34*
12. *Christian Soutou, UML 2 pour les bases de données, Eyrolles 2002 p.34*
13. [Savageman 2010] : *présenter un site complet, architecture MVC, OpenClassroom, p.35,36*

## **B. Articles, mémoire et TFC**

- 1.[Brahamira Sara 2015, 2016]. *Proposition d'un module de gestion d'hôtel et d'un site de réservation des chambres. Cas Sogesi en Algérie.p.2*
- 2.[Mulumeoderwa 2014 – 2015] *sujet de mémoire « La conception d'un system de réservation de vol en ligne, cas C.A.A.p.3*
3. [ouvrage de A.LEDERER ] *rapport annuelle sur la situation des bateaux*

## **C. Webographie**

1. <http://www.developpez.com/cour UML> par Laurent Audibert ;consulté samedi le 26 mai à 22h
2. <http://www.larousse/dictionnaires/fancais> consulté Lundi le 24 Mai, 2019
3. <https://openclassrooms.com/forum/sujet/message-d-erreur-sur-quot-localhostquot-94700> consulté Lundi le 24 décembre, 2019
4. <https://forum.alsacreations.com/topic-1-83399-1-Resolu-PHP-recuperer-les-valeursdune-table-SQL-dans-un-menu-dero.html> consulté Lundi le 24 décembre, 2018

## **TABLE DE MATIERE**

<b>DEDICACE .....</b>	<b>I</b>
<b>REMERCIEMENTS .....</b>	<b>II</b>
<b>ABBREVIATIONS .....</b>	<b>III</b>
<b>LISTE DES TABLEAUX .....</b>	<b>ERREUR ! SIGNET NON DEFINI.</b>
<b>0. INTRODUCTION GENERALE .....</b>	<b>- 1 -</b>
0.1. PROBLEMATIQUE .....	- 1 -
0.2. HYPOTHESES .....	- 3 -
0.3. ETAT DE LA QUESTION .....	- 3 -
0.4. CHOIX ET INTERET DU SUJET .....	- 4 -
0.4.1. Choix du sujet .....	.....

0.4.2. <i>Intérêt du sujet</i> .....	- 4 -
4.2.1. Intérêt personnel .....	- 4 -
4.2.2. Intérêt pour la société .....	- 4 -
0.4.2.3. Intérêt scientifique .....	- 4 -
0.5. METHODES ET TECHNIQUES .....	- 5 -
0.5.1. <i>Méthodes</i> .....	- 5 -
0.5.2. <i>Technique</i> .....	- 5 -
5.2.1. Technique d'observation .....	- 5 -
5.2.2. Interview .....	- 5 -
5.2.3. Documentaire .....	- 5 -
0.6. OBJECTIF DU TRAVAIL .....	- 6 -
0.6.1. <i>Objectifs spécifiques</i> .....	- 6 -
0.7. DELIMITATION DU SUJET .....	- 7 -
<b>CHAPITRE I : CONSIDERATIONS THEORIQUES</b> .....	<b>- 8 -</b>
I.1. DEFINITION ET APPROPRIATION DES TERMES .....	- 8 -
I.2. PRESENTATION DU CADRE D'ETUDE .....	- 9 -
1.2.1. <i>Brève historique de notre milieu d'étude</i> .....	- 9 -
1.2.2. <i>Situation Géographique</i> .....	- 9 -
1.2.3. <i>Le port attache</i> .....	- 10 -
1.2.4. <i>Statut juridique</i> .....	- 10 -
<b>CHAPITRE II ANALYSE DE L'EXISTANT ET CONCEPTION DU LOGICILE</b> .....	<b>- 12 -</b>
2.1. INTRODUCTION : .....	- 12 -
2.2. ANALYSE ET CRITIQUE DE L'EXISTANT .....	- 12 -
2.2.1. <i>Critique de l'existant</i> .....	- 12 -
2.2.2. <i>Etude des documents et leur description</i> .....	- 12 -
a) <i>Schéma de la circulation d'information</i> .....	- 13 -
2.2.3. <i>Piste de solution</i> .....	- 13 -
2.3. MODELISATION DU SYSTEME .....	- 13 -
2.3.1. <i>Réalisation des diagrammes</i> .....	- 14 -
2.3.2. <i>Spécification des exigences d'après le diagramme de cas d'utilisation</i> .....	- 15 -
2.3.3. <i>La représentation textuelle des cas d'utilisation</i> .....	- 16 -
2.3.4. <i>Diagramme de séquence</i> .....	- 19 -
2.3.5. <i>Le diagramme d'activité</i> .....	- 21 -
2.4. PASSAGE DU MODELE CONCEPTUEL AU MODELE RELATIONNEL .....	- 25 -
2.4.1. <i>Règles de passage du modèle conceptuel au modèle relationnel</i> .....	- 25 -
2.5. DICTIONNAIRE DE DONNEES .....	- 26 -
2.6. LES REGLES DE GESTION .....	- 29 -
<b>CHAPITRE III : DEVELOPPEMENT ET IMPLEMENTATION DU SITE WEB</b> .....	<b>- 31 -</b>
3.1. ENVIRONNEMENT INFORMATIQUE .....	- 31 -
3.2. DIAGRAMME DE DEPLOIEMENT .....	- 31 -
3.3. LANGAGES DE PROGRAMMATION UTILISES .....	- 32 -
3.4. PRINCIPAUX LOGICIELS UTILISES.....	- 33 -

3.5. LE SCHEMA ARCHITECTURAL DE L'APPLICATION ; DESIGN PATTERN : MODELE-VUE-CONTROLEUR (MVC) ..... - 34 -  
3.6. CAPTURE DES PRINCIPAUX ECRANS DE L'APPLICATION ..... - 35 -  
3.7. ESTIMATION DU COUT DE L'APPLICATION ..... - 43 -

**CONCLUSION GENERALE ..... - 45 -**

**BIBLIOGRAPHIE ..... - 47 -**

A. OUVRAGES ..... - 47 -  
B. ARTICLES, MEMOIRE ET TFC ..... - 47 -  
C. WEBOGRAPHIE ..... - 47 -