



Institut Africain d'Informatique
Représentation du Togo
(IAI-TOGO)

Rue de la Kozah, Nyekonakpoè
Tél : (+228) 22 20 47 00
07 BP : 12456 Lomé 07 (TOGO)
Email : iaitogo@iai-togo.tg
Site web : www.iai-togo.tg



Compagnie Aérienne ASKY

Immeuble BIDC-CEDEAO
BP : 2988 Lomé-TOGO
Tél : (228) 22 23 05 10
Fax : (228) 22 20 89 00
Email: headoffice@flyasky.com
Site web : www.flyasky.com

RAPPORT DE STAGE PRATIQUE

Type de stage : Génie Logiciel

THEME DE STAGE :

CONCEPTION ET REALISATION D'UNE APPLICATION WEB DE RESERVATION DE VOL ET VENTE DE BILLET

Période : 08 Juin au 05 Août 2022

Elaboré et présenté par :

JOHNSON Amissamba Kékéli Marie-Sharon

Etudiante en 2^{ème} année : Tronc Commun
Année Académique : 2021-2022

Superviseur

M. GNOFAME Sina,
Chargé de cours à IAI-TOGO

Maitre de stage

M. SAMA Manzimatè,
IT Manager ASKY

REMERCIEMENTS

En premier lieu, nous tenons à remercier Dieu le Tout Puissant et miséricordieux qui nous a donné la vie et la force d'accomplir ce travail.

En second nous tenons à remercier :

- M. AGBETI Kodjo Constant, Directeur Général du CENETI et Représentant Résident à l'IAI-TOGO, qui par sa foi en cet institut réussit à mener à bout la lourde et difficile mission qui lui est confiée ;
- M. AMEYIKPO Kossi Nicolas, Directeur des Affaires Académiques et de la Scolarité, pour tous les efforts multiformes qu'il déploie pour la réussite des étudiants de l'IAI-TOGO ;
- M. SAMA Manzimatè, notre maître de stage à ASKY Airlines pour sa grande disponibilité, son oreille attentive, nos discussions fructueuses, ses remarques et son soutien pendant notre stage ;
- M. GNOFAME Sina, Chargé de cours à IAI-TOGO, notre superviseur pour son entière disponibilité, son aide inestimable, son suivi continu et ses conseils ;
- M. EKPAYE Taas Sadikou, Développeur d'applications à ASKY Airlines ;
- Tout le personnel de l'IAI-TOGO, pour la qualité de la formation.

SOMMAIRE

SOMMAIRE.....	iii
GLOSSAIRE.....	iv
LISTE DES FIGURES.....	v
LISTE DES TABLEAUX.....	vi
INTRODUCTION	1
PARTIE 1. CAHIER DES CHARGES.....	2
1.1 Présentation du sujet.....	3
1.2 Problématique du sujet	3
1.3 Intérêt du sujet	3
1.3.1 Objectifs.....	3
1.3.2 Résultats.....	4
PARTIE 2. PRE-PROGRAMMATION.....	5
2.1 Etude de l'existant	6
2.2 Critique de l'existant	6
2.3 Planning prévisionnel de réalisation	6
2.4 Etude détaillée de la solution	7
PARTIE 3. REALISATION ET MISE EN OEUVRE	21
3.1 Matériels et logiciels utilisés.....	22
3.1.1 Matériels	22
3.1.2 Logiciels.....	22
3.2 Sécurité de l'application	26
3.3 Evaluation financière de la solution.....	26
3.4 Présentation de l'application	27
3.4.1 Mise en place de la base de données	27
3.4.2 Plan de navigation	30
3.4.3 Quelques masques de saisies.....	31
3.4.5 Quelques états et statistiques.....	34
CONCLUSION.....	35
BIBLIOGRAPHIE INDICATIVE	36
WEBOGRAPHIE INDICATIVE.....	36
DOCUMENTS ANNEXES	36
TABLE DES MATIERES.....	37

GLOSSAIRE

Abréviations	Significations
UML	Unified Modeling Language
XML	eXtensible Markup Language
SGBD	Système de Gestion de Base de Données
API	Application Programming Interface
IAI	Institut Africain d'Informatique
SQL	Structured Query Language
WEB	World Wide Web
GPL	General Public License
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
DB2/UDB	Universal Database
ASP	Active Server Pages

LISTE DES FIGURES

Figure 1 : Logo de PowerAMC	7
Figure 2 : Cas d'utilisation du client	10
Figure 3 : Cas d'utilisation de l'administrateur	11
Figure 4 : Diagramme des cas d'utilisation du client et de l'administrateur.....	12
Figure 5 : Diagramme des classes	16
Figure 6 : Diagramme d'activités	18
Figure 7 : Diagramme de séquence de Réservation de vol	19
Figure 8 : Diagramme de séquence d'Acheter un billet	20
Figure 9 : Logo de Visual Studio 2022	23
Figure 10 : Logo de Swagger	23
Figure 11 : Logo de MySql	24
Figure 12 : Logo de ASP.NET CORE	24
Figure 13 : Logo de JavaScript,Html,Css et Bootstrap.....	25
Figure 14 : Logo de JQuery	26
Figure 15 : Page d'authentification	31
Figure 16 : Enregistrement d'un vol	32
Figure 17 : Enregistrement d'une réservation	32
Figure 18 : Enregistrement d'un paiement	33
Figure 19 : Récupération des commentaires	33
Figure 20 : Swagger de l'API	33
Figure 21 : Liste de vols	34

LISTE DES TABLEAUX

Tableau 1 : Planning prévisionnel de réalisation	6
Tableau 2 : Liste des cas d'utilisation par acteur	9
Tableau 3 : Description textuelle du cas d'utilisation « Réserver vol »	13
Tableau 4 : Description textuelle du cas d'utilisation « Payer un billet »	13
Tableau 5 : Description textuelle du cas d'utilisation « S'authentifier »	14
Tableau 6 : Description des caractéristiques du matériel utilisé	22

INTRODUCTION

L'informatique permet de simplifier l'exécution des tâches quotidiennes dans tous les domaines par sa rapidité dans le traitement des données, sa fiabilité et la possibilité qu'elle donne d'automatiser certaines tâches.

Cette discipline est inscrite au programme de **l'Institut Africain d'Informatique - Représentation du Togo (IAI-Togo)**, école supérieure qui forme des informaticiens afin de doter notre pays le Togo de cadres informaticiens qualifiés. Cette formation se déroule sur une durée de trois (3) années après lesquelles, les dits étudiants obtiennent un diplôme d'Ingénieurs des Travaux Informatiques ou de Licence Professionnelle en fonction du parcours choisi.

Afin de valider cette formation, chaque étudiant, à la fin de sa deuxième année, est soumis à un stage pratique d'une durée de huit (8) **semaines** dans une entreprise, où il devra prouver toutes les compétences acquises par la conception et le développement d'une solution informatique.

C'est ce qui nous a amené dans les locaux de la compagnie aérienne **ASKY AIRLINES** où nous avons eu à étudier le thème qui suit: «**CONCEPTION ET REALISATION D'UNE APPLICATION WEB DE RESERVATION DE VOL ET VENTE DE BILLET**».

Notre document sera réparti en cinq (05) grandes parties qui sont : le **cahier des charges, la pré-programmation, la réalisation et mise en œuvre**.

PARTIE 1. CAHIER DES CHARGES

1.1 Présentation du sujet

Notre projet consiste à mettre en place une plateforme web qui permettra à un utilisateur de pouvoir réserver un vol disponible et de payer pour le vol, son billet ; d'où le thème: «**CONCEPTION ET RÉALISATION D'UNE APPLICATION WEB DE RÉSERVATION DE VOL ET VENTE DE BILLET**».

1.2 Problématique du sujet

Il n'y a encore pas si longtemps, il était beaucoup plus difficile voire même impossible d'effectuer une réservation de vol ou encore moins d'effectuer un achat en ligne qu'aujourd'hui. Il existait deux possibilités : téléphoner ou se déplacer vers un point de vente. Les agences de voyages ou les guichets des compagnies aériennes étaient les seuls endroits où l'on pouvait obtenir des informations pour réserver un vol. Dans le souci de faciliter la vie à sa clientèle et lui permettre un gain considérable de temps et une certaine aisance, l'entreprise choisit de mettre en place ce système. Aider la société à atteindre ses objectifs est l'une de nos priorités. Nous tenterons de répondre aux questions suivantes :

- ***Comment arriver à faire une réservation quand on est distant ?***
- ***Cette compagnie pourrait-elle mettre à la portée du public toutes ces informations sans contrainte de distance et de temps ?***

1.3 Intérêt du sujet

1.3.1 Objectifs

L'objectif de ce thème est de mettre en place un site qui permettra :

- de suivre et gérer les réservations ;
- d'enregistrer une réservation et effectuer le paiement de billet ;
- de programmer un voyage ;
- d'assurer une sécurité au niveau du paiement ;
- de permettre aux clients d'effectuer eux-mêmes des réservations quand ils le souhaitent ;
- de consulter la traçabilité des réservations.

1.3.2 Résultats

Les résultats attendus pour ce projet sont notamment :

- Réserver un vol ;
- Faciliter l'accès aux réservations pour les clients ;
- Mettre fin aux appels téléphoniques ou aux échanges par e-mail pour confirmer une réservation ;
- Payer sa réservation depuis son emplacement.

PARTIE 2. PRE-PROGRAMMATION

2.1 Etude de l'existant

Le but de l'analyse de l'existant est la recherche des points forts et des points faibles du système existant. Ainsi, l'étude de l'existant fait l'état de lieux du système actuel.

La compagnie dispose d'un site web et d'une application mobile permettant de spécifier les paramètres de la réservation. Ces paramètres, bien que variant en fonction de la ressource à réserver, comporte à chaque fois, la date de début de la réservation ainsi que la date de fin de même que les lieux de départ et d'arrivée. Le processus de réservation de vol dans la compagnie constitue une collecte de données des clients, qui permet une meilleure traçabilité des réservations. Le client qui recherche un vol devra donc spécifier ses besoins, quand il trouve le vol désiré, il réserve en payant le montant escompté.

2.2 Critique de l'existant

Dans le processus de réservation, l'utilisation d'un support téléphonique comme moyen de communication entre les différents acteurs entraîne une lourdeur dans l'échange de l'information, échange qui est nécessaire pour une correcte coordination des données. Cette caractéristique peut entraîner une saturation du réseau téléphonique.

Le coût des billets étant proportionnel au départ et la destination du vol, celui-ci se retrouve quelques fois élevé sans des offres de réduction. L'opération de réservation devient donc impossible à réaliser dans le cas où il ne peut fournir les fonds suffisants.

2.3 Planning prévisionnel de réalisation

Tableau 1 : Planning prévisionnel de réalisation

ACTIVITES	DATE DEBUT	DATE FIN	DUREE (Jours)
Prise de contact avec le centre d'accueil	08/06/2022	08/06/2022	1
Prise de connaissance du thème	27/06/2022	27/06/2022	1
Elaboration et validation du cahier des charges	05/07/2022	27/07/2022	22
Apprentissage des outils logiciels et technologies à utiliser	12/07/2022	17/07/2022	5
Programmation	12/07/2022	09/08/2022	28

Tests fonctionnels, corrections et amélioration	01/08/2022	09/08/2022	8
Rédaction du rapport de stage	05/07/2022	06/08/2022	31
Dépôt du rapport de stage	09/08/2022	09/08/2022	1

2.4 Etude détaillée de la solution

Pour réaliser ce projet, nous avons utilisé un outil de modélisation : PowerAMC

PowerAMC est l'un des premiers outils qui permet d'élaborer des modèles de données que cela soit MERISE, UML ou autre, de manière graphique et de les implémenter quel que soit le SGBD et ce de manière automatique. De même, l'outil permet de modéliser les processus métiers.

✧ Avantages

- PowerAMC est un outil simple à utiliser. Le déploiement d'un poste suffit à rendre l'outil efficient.
- L'outil fonctionne nativement avec tous les SGBD courants du marché (ORACLE, SQL SERVER, DB2/UDB).
- L'outil permet une documentation des développements.
- L'outil permet une rétro-documentation de l'existant.
- L'outil génère des graphiques exportables et importables facilement via un format XML.



Figure 1 : Logo de PowerAMC

Une façon de mettre en œuvre UML est de considérer différentes vues qui peuvent se superposer pour collaborer à la définition du système. Bien souvent, les utilisateurs ne sont pas des professionnels de l'informatique. Il leur faut donc un moyen simple pour exprimer leurs besoins. C'est précisément le rôle des diagrammes de cas d'utilisation

qui permettent de recueillir, d'analyser et d'organiser les besoins et de recenser les grandes fonctionnalités d'un système.

✓ **Diagramme de cas d'utilisation**

Un diagramme des cas d'utilisation capture le comportement d'un système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes (les cas d'utilisations) ayant un sens pour les acteurs.

Les cas d'utilisations permettent d'exprimer le besoin des utilisateurs d'un système.

Les objectifs principaux des diagrammes des cas d'utilisations sont :

- de fournir une vue de haut-niveau de ce que fait le système ;
- d'identifier les utilisateurs (acteurs) du système ;
- de déterminer les secteurs nécessaires des interfaces homme-machine(IHM).

Les cas d'utilisation se prolongent au-delà des diagrammes imagés. En fait, des descriptions textuelles des cas sont souvent employées pour compléter ces derniers et donner plus de détails sur ceux-ci. C'est aussi pour permettre aux utilisateurs finaux de valider le fonctionnement du système à concevoir. Le schéma suivant représente une modélisation du diagramme de cas d'utilisation de notre projet.

Tableau 2 : Liste des cas d'utilisation par acteur

Cas d'utilisation	Acteurs
S'authentifier	Administrateur
Consulter les vols	Client
Choisir vol	Client
Remplir un formulaire	Client
Réserver vol	Client
Payer billet	Client
Annuler une réservation	Client
Valider une réservation	Administrateur
Créer réservation	Administrateur
Modifier réservation	Administrateur
Visualiser le site web	Client
Demander modification de vol	Client
Supprimer un vol	Administrateur
Ajouter un vol	Administrateur
Modifier un vol	Administrateur
Voir les commentaires	Administrateur

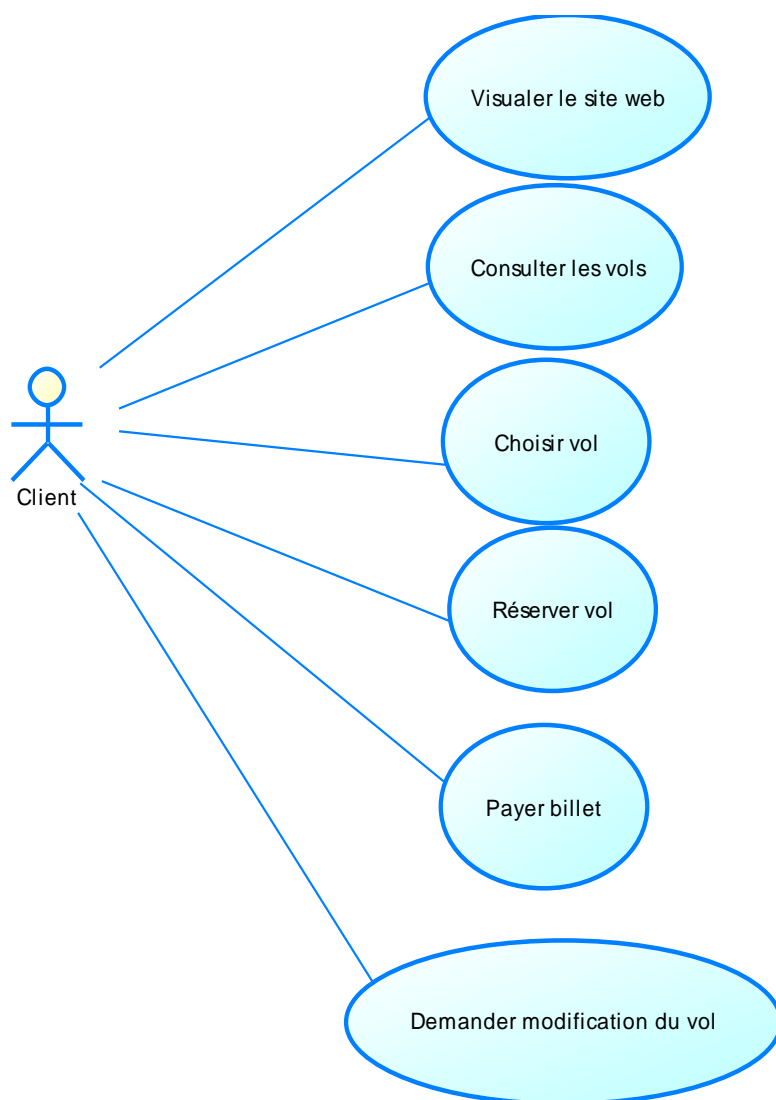


Figure 2 : Cas d'utilisation du client

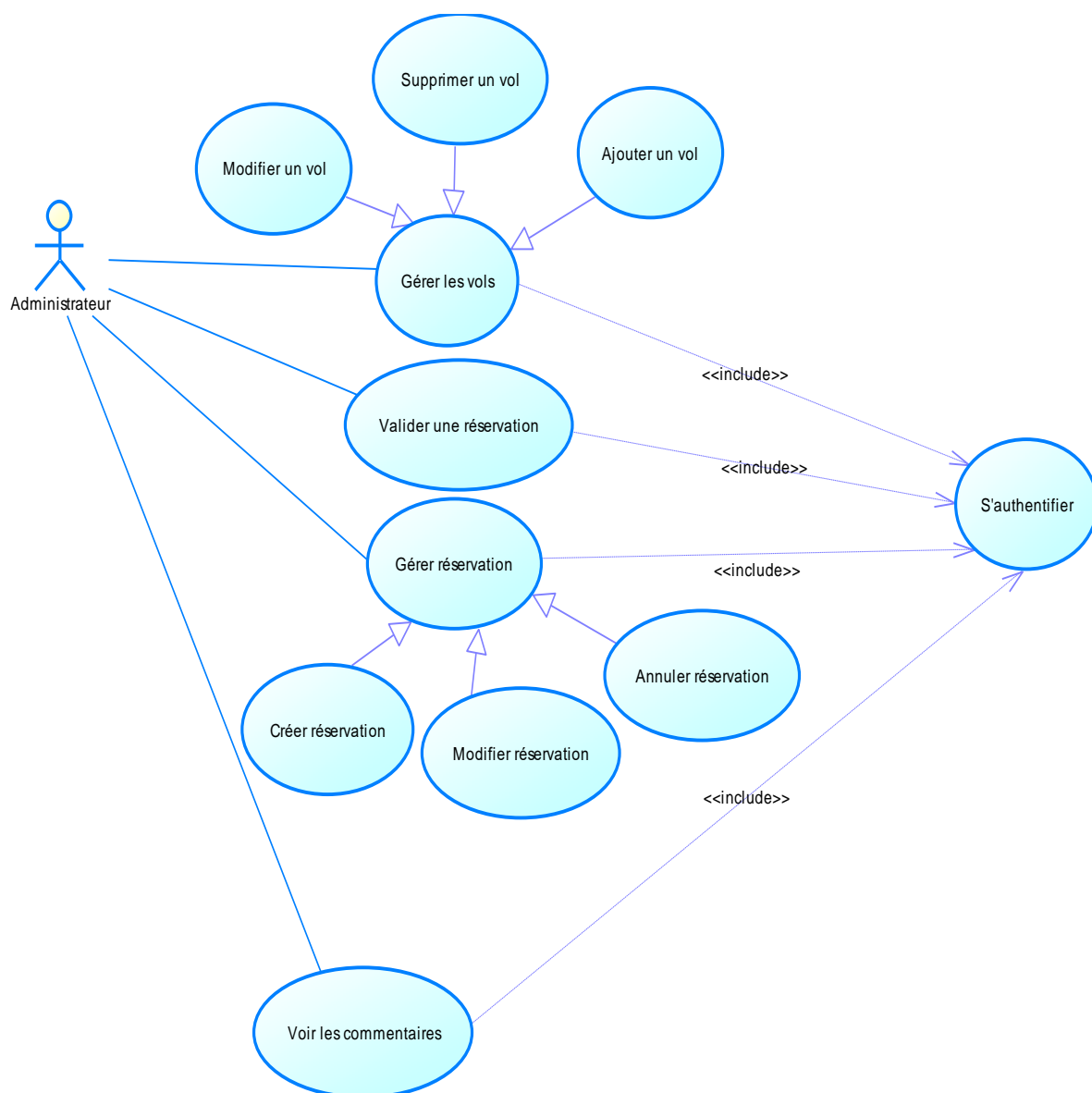


Figure 3 : Cas d'utilisation de l'administrateur

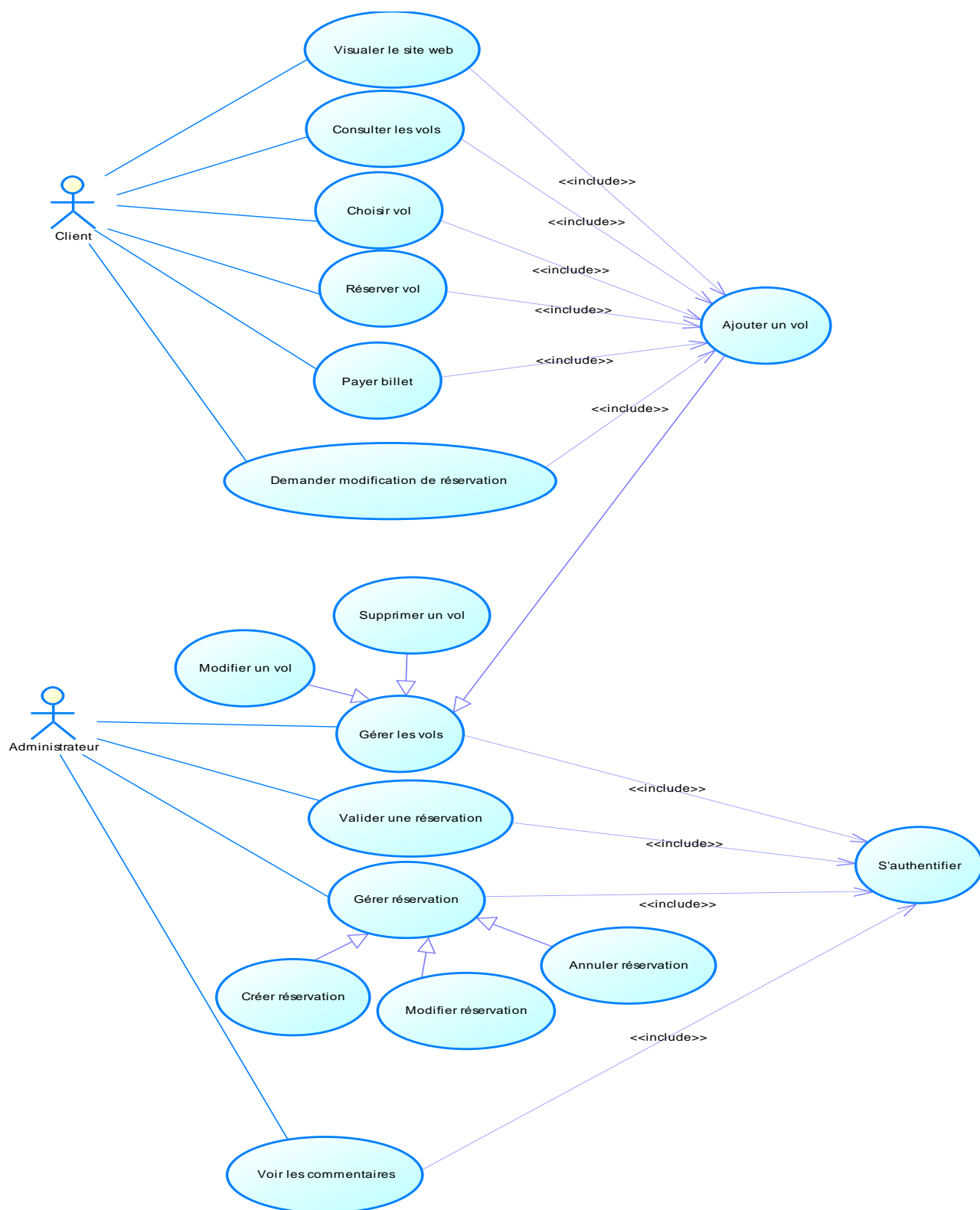


Figure 4 : Diagramme des cas d'utilisation du client et de l'administrateur

❖ Description de certains cas d'utilisation

Pour décrire la dynamique du cas d'utilisation, le plus naturel consiste à recenser toutes les interactions de façon textuelle. Cette description permet de clarifier le déroulement de la fonctionnalité et de décrire la chronologie des actions qui devront être réalisées. Nous ferons les descriptions textuelles de : réserver un vol, payer un billet, s'authentifier, car ils constituent l'essence même de notre projet et il est capital que tous ses contours soient compris.

- **Réserver vol**

Tableau 3 : Description textuelle du cas d'utilisation « Réserver vol »

Titre	Réserver vol
Résumé	Ce cas d'utilisation permet aux clients de réserver vol
Acteur principal	Client
Date	25/07/2022
Responsable	JOHNSON Amissamba Kékéli Marie-Sharon
Version	1.0
Préconditions	Le site dispose de vols pour la clientèle
Scénario nominal	1) Le client visualise le site 2) Le système affiche la liste des vols disponibles 3) Le client consulte la liste des vols disponibles 4) Le client choisit le vol désiré 5) Le client fait une demande de réservation 6) Le client remplit le formulaire et valide 7) Le système enregistre et valide.
Scénario alternatif	Le système affiche un message d'erreur lorsque l'utilisateur fournit des données incomplètes ou erronées
Post-conditions	La base données est mise à jour

- **Payer un billet**

Tableau 4 : Description textuelle du cas d'utilisation « Payer un billet »

Titre	Payer un billet
-------	-----------------

Résumé	Ce cas d'utilisation permet aux clients de payer un billet
Acteur principal	Client
Date	25/07/2022
Responsable	JOHNSON Amissamba Kékéli Marie-Sharon
Version	1.0
Préconditions	Le client confirme sa réservation
Scénario nominal	1) Le client cliquer sur « Payer un billet » 2) Le système le redirige vers le formulaire de paiement 3) Il verse le montant du billet 4) Le système vérifie et valide le paiement
Scénario alternatif	
Scénario d'exception	E1 : Le paiement n'est pas validé -Ce scénario se produit lorsque le montant versé est insuffisant
Post-conditions	Avoir le billet pour le vol

- **S'authentifier**

Tableau 5 : Description textuelle du cas d'utilisation « S'authentifier »

Titre	S'authentifier
Résumé	Ce cas d'utilisation permet à l'administrateur de s'authentifier
Acteur principal	Administrateur
Date	25/07/2022
Responsable	JOHNSON Amissamba Kékéli Marie-Sharon
Version	1.0
Préconditions	L'administrateur possède un compte administrateur
Scénario nominal	1) Le système affiche le formulaire d'authentification 2) L'administrateur saisit son nom d'utilisateur et son mot de passe 3) Le système vérifie la validité des informations fournies 4) Le système donne l'accès à l'interface administrateur
Scénario alternatif	Dans le cas où l'utilisateur fournit un nom d'utilisateur et/ou mot de passe erroné, le système affiche un message d'erreur dans la boîte

	d'outil en authentification et attend que l'utilisateur ressaisisse ses informations
Post-conditions	Base de données mise à jour Le client obtient sa réservation valide

✓ Diagramme des classes

Le diagramme de classe exprime la structure statique du système en termes de classes et de relations entre ces classes. L'intérêt du diagramme de classe est de modéliser les entités du système d'informations. Le diagramme de classe permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. Ces informations sont structurées, c'est-à-dire qu'elles sont regroupées dans des classes. Le diagramme des classes met en évidence d'éventuelles relations entre ces classes. Le diagramme des classes est le diagramme le plus répandu dans les spécifications d'UML. Le diagramme de classes comporte 6 concepts qui sont :

- Classe
- Attribut
- Identification
- Relation
- Opération
- Généralisation/Spécialisation

En somme, une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les instances de la classe.

Ci-dessous, est représenté le Diagramme de Classe :

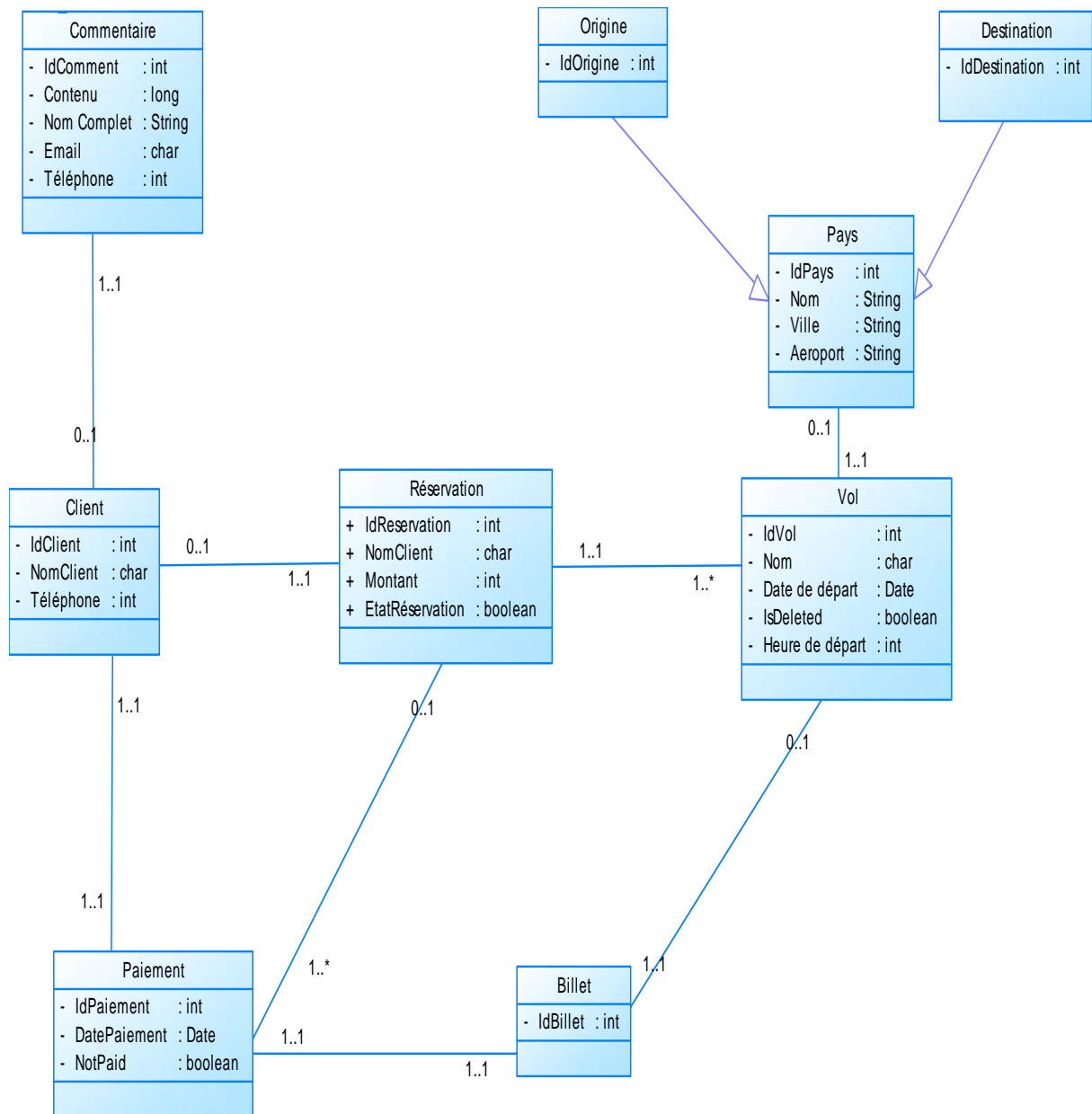


Figure 5 : Diagramme des classes

✓ Diagramme d'activités

Le diagramme d'activités se présente comme un organisme qui décrit les différents enchainements ou étapes du déroulement d'un cas d'utilisation ou groupe de cas d'utilisation. Le diagramme d'activités est une représentation proche de l'organigramme ; la description d'un cas d'utilisation par un diagramme d'activité correspond à sa traduction algorithmique. Une activité est l'exécution d'une partie du cas d'utilisation, elle est représentée par un rectangle aux bords arrondis.

Certains des composants les plus courants d'un diagramme d'activités sont :

- Actions : étape dans l'activité où les utilisateurs ou le logiciel exécutent une tâche donnée.
- Nœud de décision : embranchement conditionnel dans le flux qui est représenté par un losange. Il comporte une seule entrée et deux sorties ou plus.
- Flux de contrôle : autre nom des connecteurs qui illustrent le flux entre les étapes du diagramme.
- Nœud de départ : symbolise le début de l'activité. Cet élément est représenté par un cercle noir.
- Nœud de fin : représente l'étape finale de l'activité. Il est représenté par un cercle noir avec un contour.

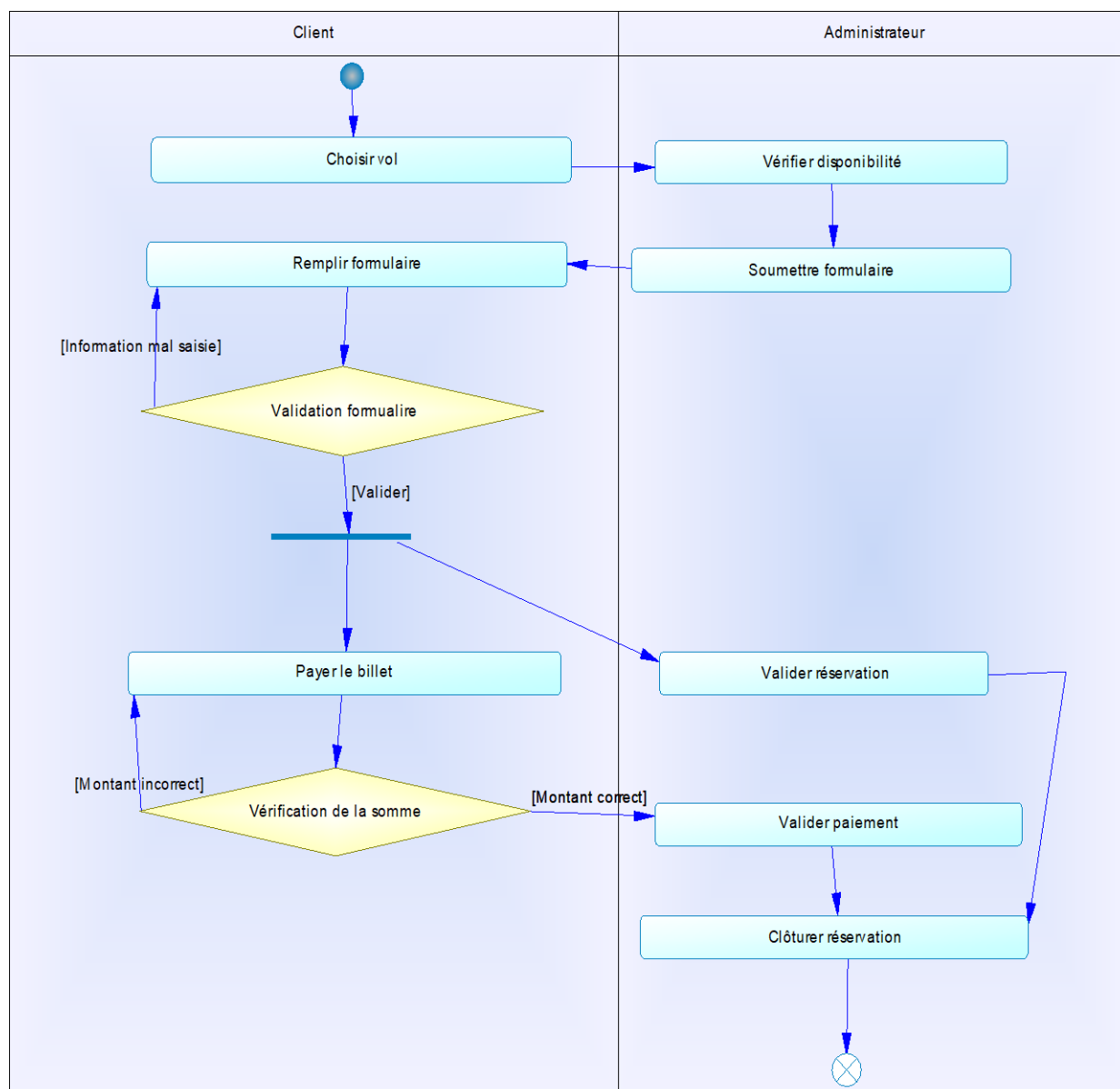


Figure 6 : Diagramme d'activités de « Réserver vol »

✓ Diagramme de séquences

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML. On montre ces interactions dans le cadre du scénario nominal des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système.

❖ Diagramme de séquence des « Réserver un vol »

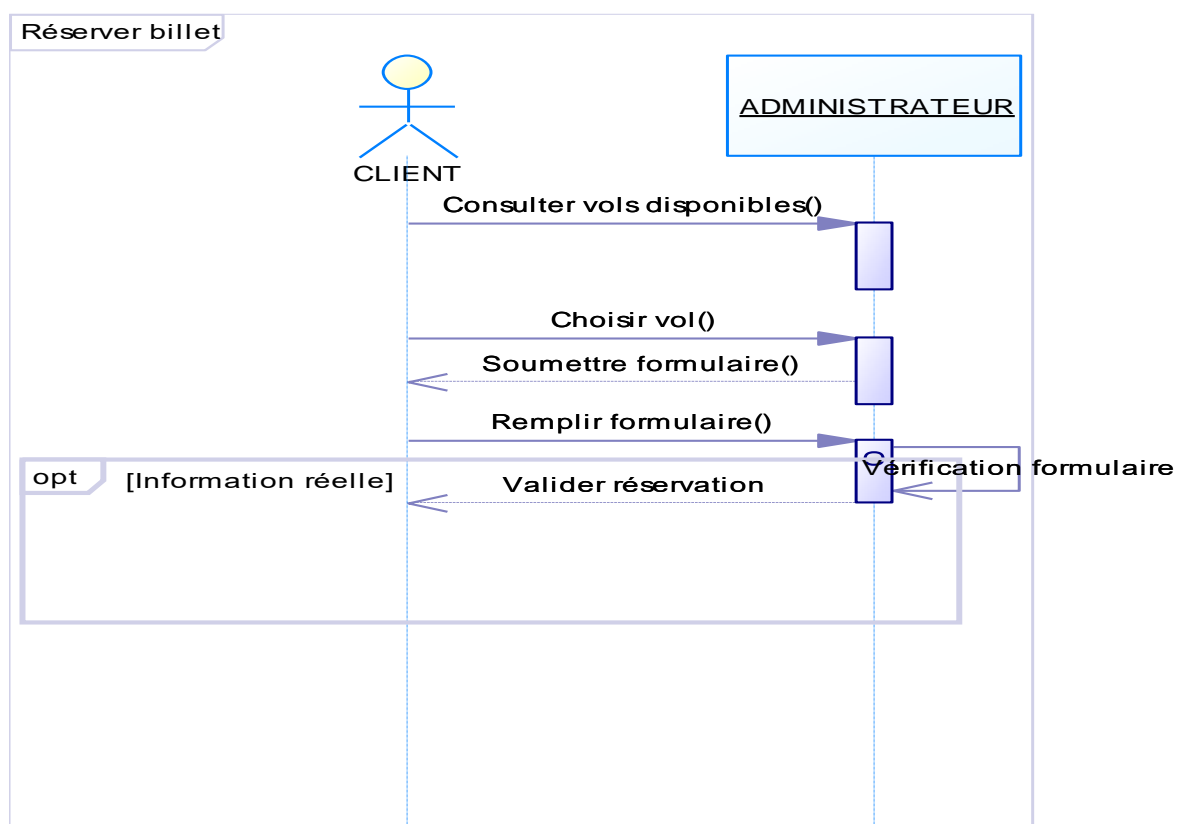


Figure 7 : Diagramme de séquence de
Réservation de vol

❖ Diagramme de séquence de Acheter un billet

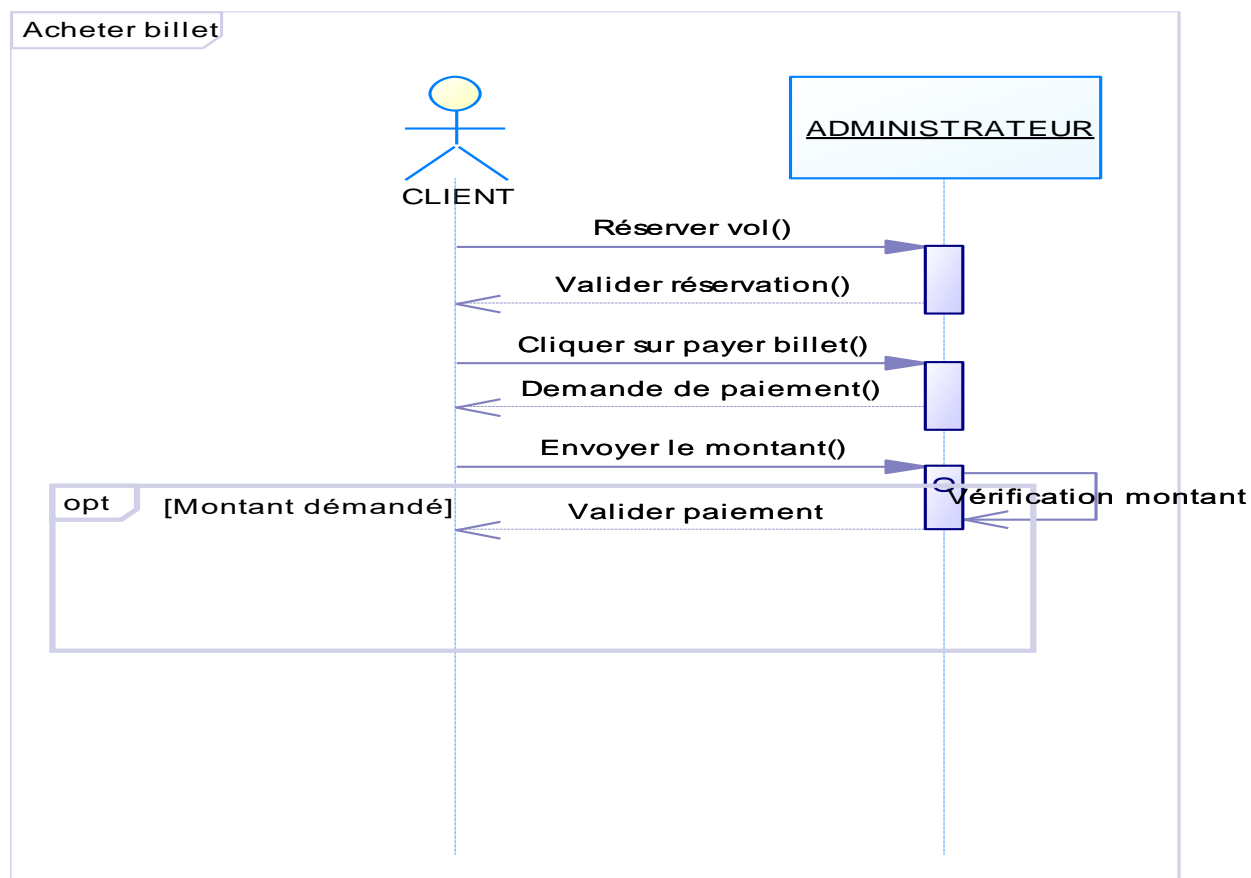


Figure 8 : Diagramme de séquence d'Acheter un billet

PARTIE 3. REALISATION ET MISE EN OEUVRE

3.1 Matériels et logiciels utilisés

3.1.1 Matériels

Pour la réalisation notre projet nous avons utilisé un ordinateur portable. Dans le tableau ci-dessous nous verrons les différentes caractéristiques.

Tableau 6 : Description des caractéristiques du matériel utilisé

Marque	Processeur	Mémoire RAM	Stockage	Système d'exploitation
<i>DELL Inc. LATITUDE E5450</i>	Intel(R) Core(TM) i5-5300U CPU @ 2.30GHz 2.29 GHz	8,00 Go	256 Go	Windows 11 Professionnel 64 bits, processeur x64

3.1.2 Logiciels

Pour la réalisation de notre projet nous avons utilisé diverse technologie (Langage de programmation, outils et environnement de développement) que nous citons ci-dessous.

Visual Studio

Microsoft Visual Studio est une suite de logiciels de développement pour Windows et mac OS conçue par Microsoft. La dernière version s'appelle Visual Studio 2022.

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



Figure 9 : Logo de Visual Studio 2022

Swagger

Swagger simplifie le processus de développement d'API pour les entreprises, les équipes et les utilisateurs grâce à son ensemble d'outils professionnels et open source. Il vous aide à concevoir et à documenter les API à grande échelle et à rationaliser votre flux de travail avec les spécifications OpenAPI.



Figure 10 : Logo de Swagger

MySQL

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, PostgreSQL et Microsoft SQL Server. SQL fait référence au Structured Query Language, le langage de requête utilisé. MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que

vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateur.

C'est un logiciel libre, open source, développé sous double licence selon qu'il est distribué avec un produit libre ou avec un produit propriétaire. Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique. Un logiciel qui intègre du code MySQL ou intègre MySQL lors de son installation devra donc être libre ou acquérir une licence payante. Cependant, si la base de données est séparée du logiciel propriétaire qui ne fait qu'utiliser des API tierces (par exemple en C# ou php), alors il n'y a pas besoin d'acquérir une licence payante MySQL. Ce type de licence double est utilisé par d'autres produits comme le framework de développement de logiciels



Figure 11 : Logo de MySql

ASP.NET CORE

ASP .NET Core est un framework du langage C Sharp; .NET Core est un framework open source indépendant de la plateforme qui offre des performances et une évolutivité élevée ainsi qu'une gestion parallèle des versions. C'est avec ce framework que nous avons réalisé le côté « Back-end » de notre logiciel.



Figure 12 : Logo de ASP.NET CORE

Html, Css , JavaScript, Bootstrap

Créé par Twitter, **Bootstrap** est un framework open source de développement web orienté interface graphique. Proposé en open source (sous licence MIT), ce framework utilisant les langages **HTML**, **CSS** et **JavaScript** fournit aux développeurs des outils pour créer un site facilement. **Bootstrap** est pensé pour développer des sites avec un design responsif, qui s'adapte à tout type d'écran, et en priorité pour les smartphones. Il fournit des outils avec des styles déjà en place pour des typographies, des boutons, des interfaces de navigation et bien d'autres encore. On les appelle les langages "Front-End". C'est avec cela, que nous avons réalisé toute les vues coté web.



Figure 13 : Logo de JavaScript,Html,Css et Bootstrap

jQuery

jQuery est une bibliothèque JavaScript rapide, petite et riche en fonctionnalités. Il rend les choses comme la traversée et la manipulation de documents HTML, la gestion des événements, l'animation et Ajax beaucoup plus simples avec une API facile à utiliser qui fonctionne sur une multitude de navigateurs.

Le but de la bibliothèque étant le parcours et la modification du DOM (y compris le support des sélecteurs CSS et un support basique de XPath), elle contient de nombreuses fonctionnalités ; notamment des animations, la manipulation des feuilles de style en cascade (accessibilité des classes et attributs), la gestion des événements, etc. L'utilisation d'Ajax est facilitée et de nombreux plugins sont présents.



Figure 14 : Logo de JQuery

3.2 Sécurité de l'application

Pour une meilleure sécurité de notre application, l'accès à la plate-forme est régi par un système d'authentification au travers d'un nom d'utilisateur et d'un mot de passe. L'accent sera mis sur la confidentialité de données traitées suivant les attributions accordées. En plus de ces règles essentielles nous préconisons de mettre en place les règles suivantes :

- Identifier les vulnérabilités et menaces éventuelles du réseau grâce à un Firewall ;
- Hachage des mots de passe ;
- Le masquage de la page administrateur ;
- L'authentification : Avant d'effectuer une action quelconque sur la plateforme, l'utilisateur doit obligatoirement s'authentifier.

3.3 Evaluation financière de la solution

Désignatio n	Description	Quantité	Prix unitaire (Fcfa)	Total	Observatio ns
Matériel	DELL Inc. LATITUDE E5450	1	250.000	250.000	Existe déjà
Logiciel	Configuration des outils	2	5.000	10.000	Existe déjà

	(xampp,C#,...)				
Fibre optique, Bande passante : 50mb/s	Forfait internet	1	15.000	15.000	Existe déjà
Conception	Montant pour la conception de l'application	2.000/heure	2.000x40 0	800.000	Existe déjà
Total		5		1.075.000	Existe déjà

3.4 Présentation de l'application

Notre application est une application web qui permet de réserver et de payer un billet. Elle se décompose en plusieurs modules suivant le client qui consulte les vols disponibles puis ensuite réserve. Ainsi le système permet aux clients de pouvoir réserver un vol et de payer un billet.

3.4.1 Mise en place de la base de données

➤ Table Comments

```
'CREATE TABLE `Comments` (  
  `IdCom` int(11) NOT NULL AUTO_INCREMENT,  
  `Contenu` longtext NOT NULL,  
  `NomComple` longtext NOT NULL,  
  `Email` longtext NOT NULL,
```

```

`Telephone` int(11) NOT NULL,

PRIMARY KEY (`IdCom`)

) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8mb4'

```

➤ Table Logins

```

'CREATE TABLE `Logins` (

`UserId` int(11) NOT NULL AUTO_INCREMENT,

`UserName` longtext NOT NULL,

`Password` longtext NOT NULL,

PRIMARY KEY (`UserId`)

) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8mb4'

```

➤ Table Reservations

```

'CREATE TABLE `Reservations` (

`IdReservation` int(11) NOT NULL AUTO_INCREMENT,

`NomClient` longtext NOT NULL,

`Montant` int(11) NOT NULL,

`EtatReservation` tinyint(1) NOT NULL,

`IdVol` int(11) NOT NULL,

PRIMARY KEY (`IdReservation`),

KEY `IX_Reservations_IdVol` (`IdVol`),

CONSTRAINT `FK_Reservations_Vols_IdVol` FOREIGN KEY (`IdVol`)

REFERENCES `Vols` (`IdVol`) ON DELETE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4'

```

➤ Table Paiements

```
'CREATE TABLE `Paiements` (
  `IdPaiement` int(11) NOT NULL AUTO_INCREMENT,
  `DatePaiement` datetime(6) NOT NULL,
  `NotPaid` tinyint(1) NOT NULL,
  `IdReservation` int(11) NOT NULL,
  PRIMARY KEY (`IdPaiement`),
  KEY `IX_Paiements_IdReservation` (`IdReservation`),
  CONSTRAINT `FK_Paiements_Reservations_IdReservation` FOREIGN KEY
  (`IdReservation`) REFERENCES `Reservations` (`IdReservation`) ON DELETE
  CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4'
```

➤ Table Vols

```
'CREATE TABLE `Vols` (
  `IdVol` int(11) NOT NULL AUTO_INCREMENT,
  `Nom` longtext NOT NULL,
  `Origine` longtext NOT NULL,
  `Destination` longtext NOT NULL,
  `DateDepart` datetime(6) NOT NULL,
  `IsDeleted` tinyint(1) NOT NULL,
  `UserId` int(11) NOT NULL,
  PRIMARY KEY (`IdVol`),
```

```

KEY `IX_Vols_UserId` (`UserId`),

CONSTRAINT `FK_Vols_Logins_UserId` FOREIGN KEY (`UserId`) REFERENCES
`Logins` (`UserId`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=utf8mb4'

```

3.4.2 Plan de navigation

Les requêtes supportées sont de type « JSON » et l'outil swagger peut être utilisé pour les requêtes en direction de la base de données. L'API est donc lancée et fonctionnelle. L'application web consomme l'API. On accède à l'application après déploiement sur le serveur IIS Express du VPS. La première page à laquelle on accède est la page de d'accueil. Cette page accessible à tout le monde. L'administrateur à accès à un compte qu'il a créé comme nous l'avons signifié plus haut. Lorsque l'administrateur est authentifié il a accès à la page de de l'administrateur. Pour ne pas paraître trop vague dans la suite nous présenterons les différentes interfaces de nos différentes applications.

3.4.3 Quelques masques de saisies

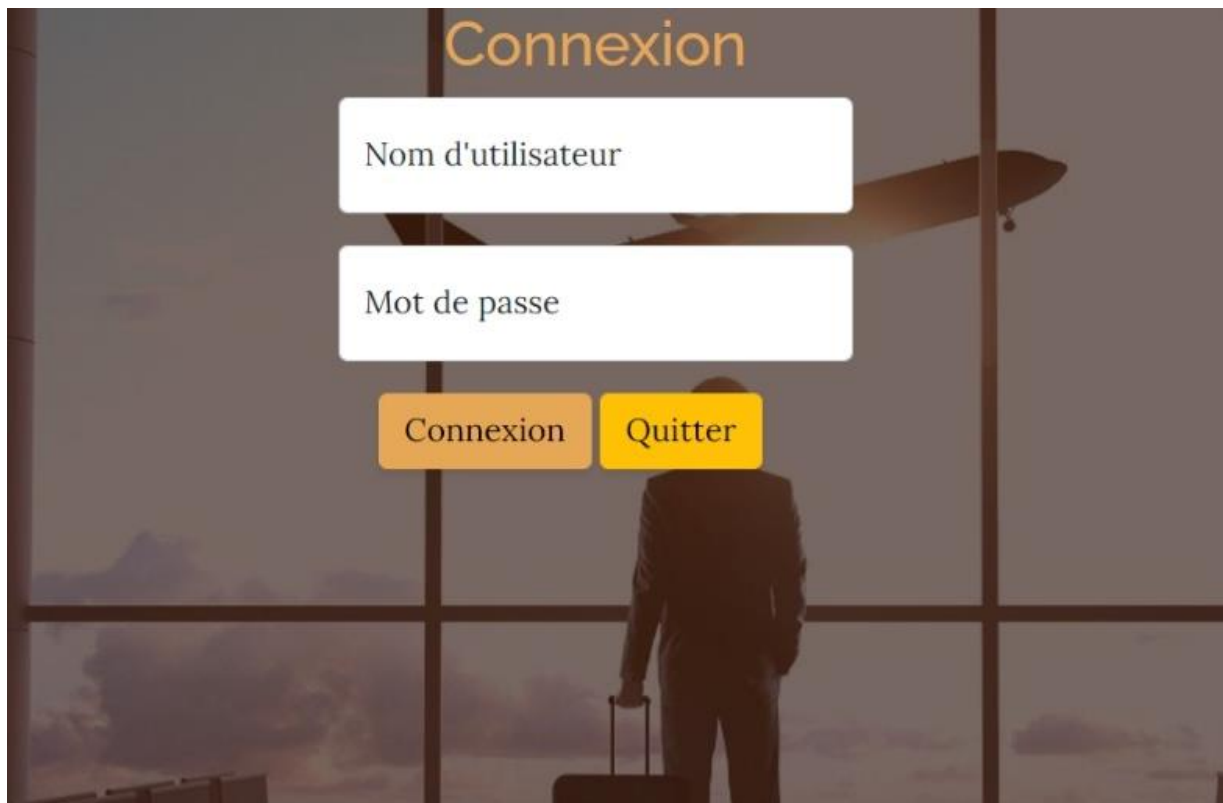


Figure 15 : Page d'authentification

Créer un nouveau vol

Nom

Origine

Destination

Date de départ

jj/mm/aaaa

Créer

Supprimer

Figure 16 : Enregistrement d'un vol

Créer une réservation

Nom

Montant

Etat de la réservation ☐ Oui ☒ Non

Créer

Supprimer

Figure 17 : Enregistrement d'une réservation



Créer un paiement

Paieement

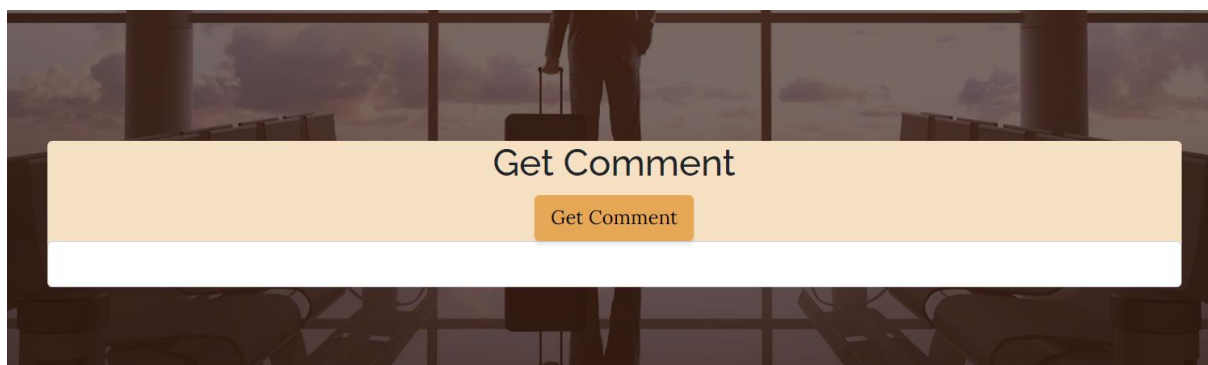
jj/mm/aaaa

Etat paiement ☐ Oui ☒ Non

Créer

Supprimer

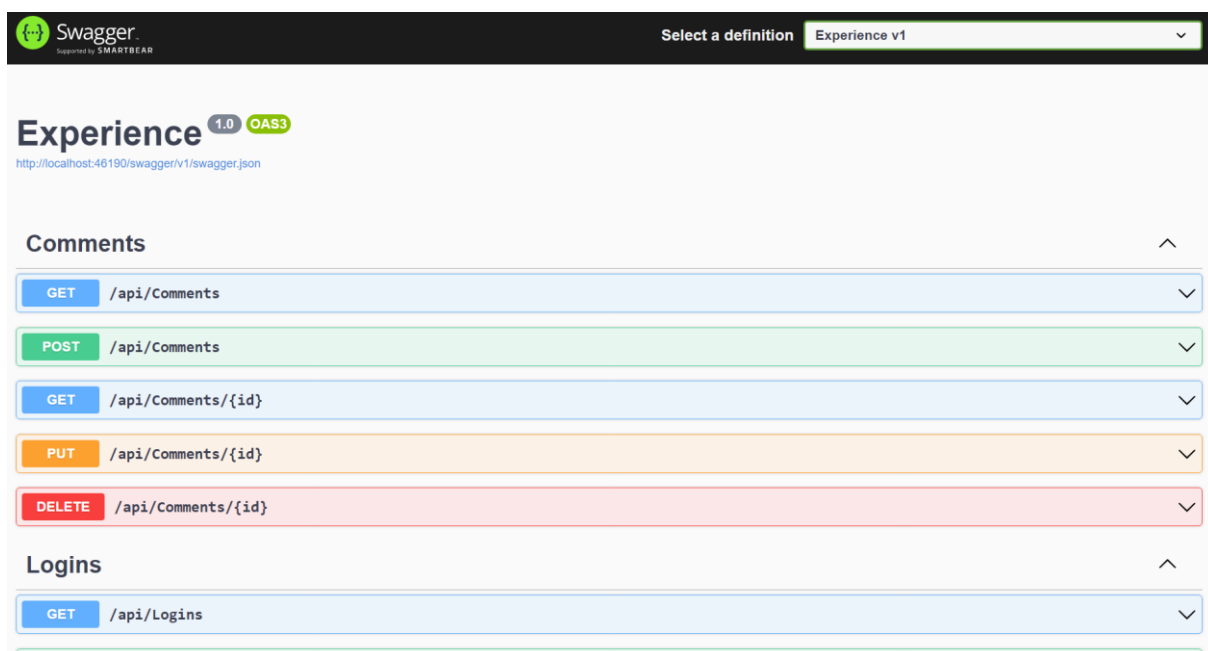
Figure 18 : Enregistrement d'un paiement



Get Comment

Get Comment

Figure 19 : Récupération des commentaires



Swagger

Select a definition Experience v1

Experience 1.0 OAS3

http://localhost:48190/swagger/v1/swagger.json

Comments

GET /api/Comments

POST /api/Comments

GET /api/Comments/{id}

PUT /api/Comments/{id}

DELETE /api/Comments/{id}

Logins

GET /api/Logins

Figure 20 : Swagger de l'API

3.4.5 Quelques états et statistiques



	Nom	Origine	Destination	Date de départ	
1	V1	Lomé	Accra	27/08/2022	Réserver
2	V3	Abidjan	Dakar	09/08/2022	Réserver
3	V4	Lomé	Cotonou	07/08/2022	Réserver

Figure 21 : Liste de vols

CONCLUSION

Notre projet avait pour objectif de mettre en place une application permettant de gérer les réservations de vol et du paiement de billet. Ce projet a été une occasion de sortir du cadre théorique et d'appliquer les connaissances acquises lors de notre formation à IAI-Togo. Sur le plan technique ce projet nous a permis d'acquérir de nouvelles connaissances. Dans la première partie nous avons fait l'étude descriptive de notre projet et étudié la solution ordonnée vis-à-vis de la problématique dégagée. Dans la deuxième partie, celle de l'analyse et de la conception nous avons effectué une étude détaillée de notre projet en utilisant la méthode UML. Ensuite, la troisième partie nous a permis de décrire l'environnement technique et les outils de développement de notre application et aussi de présenter la structure de la base de données, ainsi que quelques interfaces utilisateurs et codes sources de notre application. La première version de cette application ne saurait combler totalement les attentes des utilisateurs. Ainsi, lorsque de nouveaux besoins seront demandés, de nouvelles fonctionnalités viendront s'ajouter à notre application de ce fait à l'améliorer.

BIBLIOGRAPHIE INDICATIVE

➤ Note de cours

- Cours d'UML : M. KETOGLO Alfred ;
- Cours de Programmation C# : M. GBODUI Joseph-Roland ;
- Rédaction scientifique : M. GBODUI Joseph-Roland ;
- Conception des bases de données : M. TCHANTCHO Leri ;

WEBOGRAPHIE INDICATIVE

- <https://www.google.com/>, Tout le long du projet ;
- <https://openclassrooms.com/fr/>, Tout le long du projet ;
- <https://stackoverflow.com/>, Tout le long du projet ;
- <https://docs.microsoft.com/>, tout le long du projet ;
- <https://developpez.net/>, tout le long du projet ;
- <https://www.youtube.com/>, tout le long du projet ;
- <https://www.w3.org/>, tout le long du projet ;

DOCUMENTS ANNEXES

- WOAGOU Yendouboame Aimé : **e-data : plateforme de collecte, de gestion et de contrôle de données en temps réel pour la sécurité nationale** (2020-2021)

TABLE DES MATIERES

REMERCIEMENTS.....	ii
SOMMAIRE.....	iii
GLOSSAIRE.....	iv
LISTE DES FIGURES.....	v
LISTE DES TABLEAUX.....	vi
INTRODUCTION.....	1
PARTIE 1. CAHIER DES CHARGES.....	2
1.1 Présentation du sujet.....	3
1.2 Problématique du sujet.....	3
1.3 Intérêt du sujet.....	3
1.3.1 Objectifs.....	3
1.3.2 Résultats.....	4
PARTIE 2. PRE-PROGRAMMATION.....	5
2.1 Etude de l'existant.....	6
2.2 Critique de l'existant.....	6
2.3 Planning prévisionnel de réalisation.....	6
2.4 Etude détaillée de la solution.....	7
PARTIE 3. REALISATION ET MISE EN OEUVRE.....	21
3.1 Matériels et logiciels utilisés.....	22
3.1.1 Matériels.....	22
3.1.2 Logiciels.....	22
3.2 Sécurité de l'application.....	26
3.3 Evaluation financière de la solution.....	26
3.4 Présentation de l'application.....	27
3.4.1 Mise en place de la base de données.....	27
3.4.2 Plan de navigation.....	30
3.4.3 Quelques masques de saisies.....	31
3.4.5 Quelques états et statistiques.....	34
CONCLUSION.....	35
BIBLIOGRAPHIE INDICATIVE.....	36
WEBOGRAPHIE INDICATIVE.....	36
DOCUMENTS ANNEXES.....	36
TABLE DES MATIERES.....	37