MINISTERE DE LA PLANIFICATION DU DEVELOPPEMENT ET DE LA COOPERATION

REPUBLIQUE TOGOLAISE

Travail-Liberté-Patrie



Institut Africain d'Informatique, Représentation du TOGO (IAI-TOGO) 07 BP 12456 Lomé 07, TOGO

TEL: (+228) 22 20 47 00
E-mail: iaitogo@iai-togo.tg
Site web: www.iai-togo.tg



CLIN SARLU

<u>TEL</u>: (+228) 92 92 92 85

(+228) 22 51 39 8

E-mail: contact@clin.tg **Site web**: www.clin.tg

RAPPORT DE STAGE PRATIQUE EN ENTREPRISE TYPE DE STAGE : STAGE DE PROGRAMMATION

MISE EN PLACE D'UNE APPLICATION WEB DE GESTION DE STOCK

Période: Du 07 juin au 05 Août 2022

Rédigé et présenté par :

AWESSO Lidaou Espoir

Etudiant en Deuxième année Tronc commun

Année Académique : 2021-2022

SUPERVISEUR

M. AMEVOR Kossi A.

Enseignant à l'IAI-TOGO

MAITRE DE STAGE

M. N'SOUGAN Folly Woèdè

Chef projet à CLIN SARL U

REMERCIEMENTS

J'adresse mes sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé mes réflexions et ont accepté de répondre à mes questions durant mes recherches. Mes remerciements vont :

- À M. AGBETI Kodjo, Directeur Général du CENETI et Représentant National de l'IAI-TOGO ;
- À M. AMEYIKPO Kossi Nicolas, Directeur des Affaires Académiques et de la Scolarité de l'IAI-TOGO ;
- Aux membres de l'administration de l'IAI-TOGO, pour l'encadrement rigoureux ;
- À mon maitre de stage, M. N'SOUGAN Folly Woèdè, Chef Projet à CLIN SARL U;
- À M. ADRAKE Emile, Développeur à CLIN SARL U ;
- À mon superviseur, M. AMEVOR kossi Jean, enseignant à l'IAI-TOGO;
- À mes parents pour leur soutien, et leur sacrifice.

SOMMAIRE

REME	RCIEMENTS	i
SOMM	AIRE	ii
GLOS	SAIRE	iii
INTRO	DUCTION	1
PARTI	E1 : CAHIER DES CHARGES	3
1.1	Présentation du sujet	3
1.2	Problématique du sujet	3
1.3	Intérêt du sujet	3
PARTI	E 2 : ANALYSE ET CONCEPTION	6
2.1	ETUDE DE L'EXISTANT	6
2.2	CRTITIQUE DE L'EXISTANT	
2.3	Planning prévisionnel de réalisation	7
2.4	ETUDE DETAILLEE DE LA SOLUTION	7
PARTI	E3 : REALISATION ET MISE EN ŒUVRE	. 23
3.1	Matériels et logiciel utilisé	. 23
3.2	Sécurité de l'application	. 25
3.3	Evaluation financière de la solution	. 26
3.4	Présentation de l'application	. 26
CONC	LUSION	. 34
BIBLIC	GRAPHIE INDICATIVE	. 35
WEBO	GRAPHIE INDICATIVE	. 36
DOCU	MENTS ANNEXES	. 37

GLOSSAIRE

ACRONYME	SIGNIFICATION
CLIN	Centre Local D'incubation Numérique
CU	cas d'utilisation
DCU	Diagramme de cas d'utilisation
IAI	Institut Africain d'Informatique
IDE	Environnement de Développement Intégré
SGBD	Système de Gestion des Bases de Données
SQL	Structured Query Language
UML	Unified Modeling Language

LISTE DES FIGURES

Figure 1: Logo de UML	8
Figure 2:Logo de poweramc	9
Figure 3:Diagramme de cas d'utilisation de l'administrateur et du personnel	11
Figure 4:Diagramme d'activité de « s'authentifier »	16
Figure 5:Diagramme d'activité de « Ajouter une demande »	16
Figure 6:Diagramme d'activité de « Valider un demande »	17
Figure 7:Diagramme d'activité de « Ajouter une affectation »	18
Figure 8:Diagramme de séquence de « S'authentifier »	19
Figure 9:Diagramme de séquence de « Ajouter une demande »	19
Figure 10:Diagramme de séquence de « Valider une demande »	20
Figure 11:Diagramme de séquence de « Ajouter une affectation »	20
Figure 12:Diagramme des classes	21
Figure 13:MySQL	
Figure 14:Visual Studio Code	24
Figure 15:Yii2	25
Figure 16:Plan de navigation	30
Figure 17:Formulaire d'authentification des agents	30
Figure 18:Formulaire d'ajout d'une demande	31
Figure 19:Formulaire d'ajout d'une affectation	31
Figure 20:Historique des affectations	32
Figure 21:Répartition des matériels par catégorie	32
Figure 22:Répartition des matériels selon leurs états et répartition des affectations	
des matériels	33

LISTE DES TABLEAUX

Tableau	1:Planning prévisionnel de réalisation	7
Tableau	2:Cas d'utilisation	10
Tableau	3:Evaluation financière de la solution	26

INTRODUCTION

L'informatique, science du traitement automatisé de l'information s'impose de plus en plus comme un vecteur de développement des entreprises. Le monde actuel est attaché à l'outil informatique et ne peux s'en débarrasser. Le monde est connecté et la technologie ne cesse d'accroitre. Par conséquent, L'IAI-TOGO (Institut Africain d'informatique Représentation du Togo) insère dans son cursus d'ingénieurs de travaux informatiques un stage pratique à la fin de la deuxième année pour permettre aux étudiants de mettre en pratiques les connaissances acquises, d'être plus compétant et d'intégrer le monde professionnel sans difficulté. Au cours du stage, déroulé du 07 juin au 05 Août, au centre CLIN SARLU, et pour thème de stage "GESTION DE STOCK", nous avons constaté que chaque demande de sortie de matériel est faite sur du papier ; il est presqu'impossible de retrouver l'historique de certaines demandes et affectations, et aussi de faires des statistiques sur l'ensemble des matériels en stock. Ce présent rapport comporte trois parties : le cahier des charges, la pré programmation, la réalisation et la mise en œuvre.

PARTIE 1 : CAHIER DES CHARGES

PARTIE1: CAHIER DES CHARGES

Dans la première partie de notre document, ou cahier de charge, il est question de la formalisation et de la clarification des besoins. Le cahier de charge est le résultat de la phase d'analyse des besoins et contient obligatoirement les résultats attendus en termes de spécifications fonctionnelles et non fonctionnelles. Il comporte la description du cadre de stage, les besoins et exigences des utilisateurs futurs, les fonctionnalités attendues ainsi que les contraintes techniques. En grandes lignes de ce document, nous aurons la présentation du thème de stage, la problématique du sujet, l'intérêt du sujet, les objectifs et résultats attendus.

1.1 Présentation du sujet

Le projet qui nous a été confié au « **Centre Local D'incubation Numérique** (CLIN) » consiste à concevoir et implémenter une application web qui permettra de gérer les sorties et retours en magasin des matériels non consommables de la société.

1.2 Problématique du sujet

CLIN Sarl U est un centre qui créer des applications innovantes et donne des formations chaque année à plusieurs apprenants dans biens de domaines. Les séances d'apprentissages nécessitent parfois des matériels spécifiques présents au magasin.

En effet, toute sortie de matériel est précédée d'une demande faite chez la secrétaire. Les demandes et affectations sont donc faites par écrit. Il sera donc très difficile de faire des rappels aux personnels détenant des matériels, de faire les statistiques et d'analyser l'historiques des affectations de matériels.

Il est alors besoin de mettre en place un système de gestion de stock, pour avoir une accélération des affectations de matériels avec les statistiques parfaites. Afin d'atteindre ses objectifs, nous tenterons de répondre aux questions suivantes :

- ✓ Comment notre projet pourra répondre aux besoins ?
- ✓ Comment gérer les affectations de matériel plus efficacement ?
- ✓ Comment assurer les retours de matériel en stock ?
- ✓ Comment réduire les risques de perte en mettant en lieu sur une sauvegarde des historiques ?
- ✓ Comment réduire efficacement la lourdeur administrative ?

1.3 Intérêt du sujet

1.3.1 Objectifs

L'objectif visé par ce projet est la mise en place d'un système qui permettra de :

- ✓ Gérer toutes les informations relatives sur un matériel;
- ✓ Faire une demande de sortie de matériel ;
- ✓ Faire un suivi depuis le traitement de la demande jusqu'à son retour ;

- ✓ Sortir les historiques des affectations;
- ✓ Avoir en temps réel la situation en stock.

1.3.2 Résultats

Une fois le système en place, un utilisateur doit pouvoir ressortir les affirmations suivantes :

- ✓ Les informations relatives à un matériel sont enregistrées et peuvent être consultées;
- ✓ La demande de sortie de matériel peut être faite ;
- ✓ Le traitement des affectations est optimisé ;
- ✓ Les historiques des affectations sont disponibles ;
- ✓ La situation en stock est disponible en temps réel.

PARTIE 2: PRE-PROGRAMMATION

PARTIE 2: ANALYSE ET CONCEPTION

Avant toute conception de l'application proprement site, il est nécessaire de comprendre et de mieux cerner le sujet. La phase d'analyse étant déterminante dans la réalisation de tout projet informatique, cette partie nous permettra d'étudier le sujet et ainsi proposer les solutions.

2.1 ETUDE DE L'EXISTANT

Les matériels en stock au magasin sont des matériels non consommables. Lors de la demande de sortie d'un matériel l'intéressé rempli une fiche au niveau du secrétariat grâce à laquelle il obtient son bon de stock. Sur cette fiche sont renseignés : les dates (sotie, entrée), les informations personnelles de l'intéressé (nom, prénom, département) et les informations relatives au matériel (désignation, référence, quantité, utilité). Le délai de retour est fixé en fonction de l'utilité pour laquelle le matériel est demandé. Après utilisation du matériel sur le terrain, ceux-ci sont retournés au magasin tout en vérifiant les états ou manquements.

Le matériel ne peut être cédé qu'aux internes sauf pour cas de location.

Un inventaire est fait à une certaine périodicité pour remarquer les différents états des matériels du magasin.

2.2 CRTITIQUE DE L'EXISTANT

Les opérations étant principalement manuelles, la garantie de sécurité des données n'est pas élevée. Ceci peut engendrer la consultation de ces informations par tout individu qui en rentre en possession. Aussi, les risques de pertes d'informations sont accrus étant donné qu'une bonne partie de ses informations est stockée sur support papier. Les demandes et affectations sont faites par écrit. Nous observons une lenteur et des erreurs de saisie lors de l'enregistrement des informations.

2.3 Planning prévisionnel de réalisation

Tableau 1:Planning prévisionnel de réalisation

N°	Tâche	Début	Fin	Durée
1	Insertion	07-06-2022	12-06-2022	6 jours
2	Choix d'outil & Analyse	13-06-2022	19-06-2022	7 jours
3	Conception	20-06-2022	29-06-2022	10 jours
4	Implémentation	30-06-2022	24-07-2022	25 jours
5	Période de test	25-07-2022	31-07-2022	7 jours
6	Finalisation du document	01-08-2022	05-08-2022	5 jours

2.4 ETUDE DETAILLEE DE LA SOLUTION

2.4.1 Choix de l'outils d'analyse

UML

L'outils d'analyse utilisé est UML. Le langage de modélisation unifié, en anglais Unified Modeling Langage (UML) est un langage de modélisation graphique à base de pictogrammes (dessins figuratifs stylisés ayant fonction de signe) 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. UML est le résultat de la fusion de précédents langages de modélisation objet : Booch, OMT, OOSE. Principalement issu des travaux de Grady Booch, James Rumbaugh et Ivar Jacobson, UML est à présent un standard adopté par l'Object Management Group (OMG).

UML est un langage formel, défini par un méta modèle. Le méta modèle d'UML décrit de manière très précise tous les éléments de modélisation (les concepts véhiculés et manipulés par le langage) et la sémantique de ces éléments (leur définition et le sens de leur utilisation). C'est en cela qu'on dit qu'UML normalise les concepts objet. Un

méta modèle permet de limiter les ambiguïtés et encourage la construction d'outils. Il permet aussi de classer les différents concepts du langage (selon leur niveau d'abstraction ou leur domaine d'application) et expose ainsi clairement sa structure.



Figure 1: Logo de UML

❖ 2TUP

Pour notre projet nous avons utilisés le processus 2TUP pour associer à l'outil d'analyse choisi.

2TUP (2 track unified process, prononcez "toutiyoupi") est un processus de développement logiciel qui implémente le Processus Unifié.

Le **2TUP** propose un cycle de développement en Y, qui dissocie les aspects techniques des aspects fonctionnels. Il commence par une étude préliminaire qui consiste essentiellement à identifier les acteurs qui vont interagir avec le système à construire, les messages qu'échangent les acteurs et le système, à produire le cahier des charges et à modéliser le contexte (le système est une boîte noire, les acteurs l'entourent et sont reliés à lui, sur l'axe qui lie un acteur au système nous mettons les messages que les deux s'échangent avec le sens). Le processus s'articule ensuite autour de 3 phases essentielles:

- Une branche technique;
- Une branche fonctionnelle ;
- Une phase de réalisation.

2.4.2 Présentation de l'outils de modélisation

Il existe une multitude d'outils de modélisation parmi lesquelles nous pouvons citer : DBDesigner, StarUML...

Pour réaliser ce projet, nous avons utilisé Power AMC POWER AMC qui 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

• Power AMC 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 SERVEUR, 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 2:Logo de poweramc

2.4.3 Diagramme de cas d'utilisation

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. Il s'agit donc de la première étape UML d'analyse d'un système. 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 ;

 ↓ Identifier les utilisateurs (" acteurs ") du système ;

🖶 Déterminer les secteurs nécessaires des interfaces homme-machine(IHM).

Les cas d'utilisations 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.

Dans notre projet, nous avons décelé uniquement des acteurs qui sont :

Le Personnel

C'est un interne à l'entreprise, il effectue une demande.

La Secrétaire

Elle hérite de tout ce que le personnel fait. Elle est celle qui :

- Traite les demandes ;
- Gère le matériel ;
- Gère les états du matériel :
- Gère les catégories du matériel ;

○ Le Directeur

- C'est le chef de l'entreprise. Il hérite de tout ce que la secrétaire fait.

❖ Les cas d'utilisations

Un cas d'utilisation exprime le comportement du système en terme d'action et réaction face à un besoin d'un utilisateur. Dans notre projet nous avons identifié les cas d'utilisations suivants :

CAS D'UTILISATION	ACTEURS	
	Ajouter un matériel	
	Supprimer un matériel	Secrétaire
Gestion du matériel	Modifier un matériel	&
	Consulter un matériel	Directeur
	Ajouter une demande	Directeur
	Modifier une demande	&
	Supprimer une demande	Secrétaire
Gestion des demandes	Consulter une demande	&
Gestion des demandes		Personnel
	Valider une demande	Directeur
	Rejeter une demande	&
		Secrétaire
	Ajouter une affectation	Directeur
Gestion des affectations	Supprimer une affectation	&
de matériel	Consulter une affection	Secrétaire
	Modifier une affection	D
	S'authentifier	Directeur
Gestion des utilisateurs	Modifier les droits des	& Convétaire
	utilisateurs	Secrétaire
	Consulter les statistiques	Directeur
	Imprimer liste des matériels	& Secrétaire
Gestion des états et	materiels	&
statistiques		Personnel
Statistiques	Imprimer l'historique des	Directeur
	affectations	&
	anodations	Secrétaire
	Ajouter une catégorie	
Gestion des catégories	Modifier une catégorie	Directeur
de matériel	Supprimer une catégorie	& Coarátaira
	Consulter une catégorie	Secrétaire
	Ajouter un état	Directour
Gestion des états	Modifier un état	Directeur
d'usage	Supprimer un état	& Secrétaire
Tableau 2:Con d'utilisation	Consulter un état	Secretaire

Tableau 2:Cas d'utilisation

❖ Le diagramme

Le schéma suivant représente une modélisation du diagramme de cas d'utilisation de notre projet.

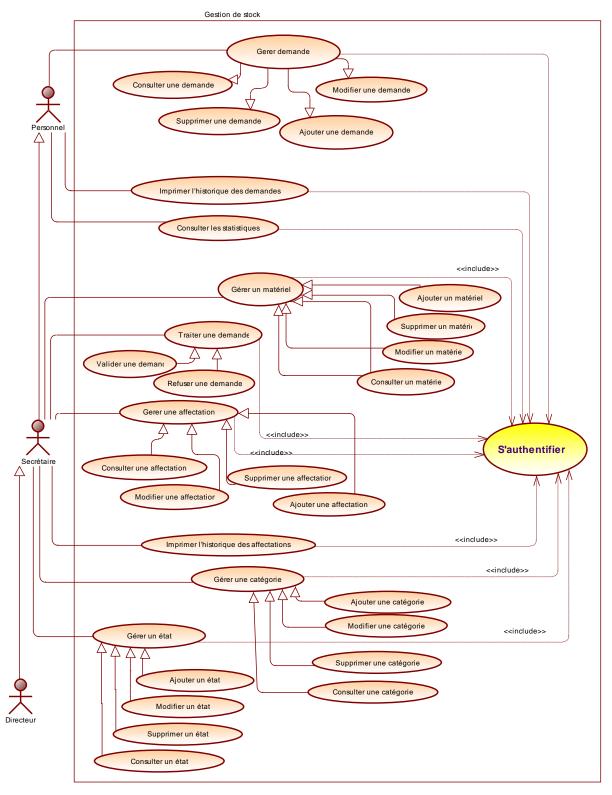


Figure 3:Diagramme de cas d'utilisation de l'administrateur et du personnel.

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 : s'authentifier, ajouter une demande, valider une demande et ajouter une affectation car ils constituent l'essence même de notre projet et il est capital que tous ses contours soient compris.

CU: S'authentifier

SOMMAIRE D'IDENTIFICATION

Titre: S'authentifier

Résumé: Permet de s'authentifier.

Acteur : Directeur, Secrétaire & Personnel

Date de création : 16/06/2022 **Date mis à jour :** 30/07/2022

Version: 1.0

Responsable: AWESSO Lidaou Espoir

DESCRIPTION DES SCENARII

Pré - conditions

- Être connecté au réseau local de l'entreprise.
- L'application doit être fonctionnelle.

Scénario nominal

- 1. L'utilisateur accède à la page d'authentification.
- 2. Le système lui affiche le formulaire d'authentification.
- 3. L'utilisateur saisit les informations requises et valide.
- 4. Le système vérifie les informations saisies (A1).

Le système redirige l'utilisateur sur une page selon son profil.

Scénarii alternatifs

(A1): Les données reçues par le serveur sont erronées.

Ce scénario démarre au point 4 du scénario nominal lorsque les données reçues par le système sur les champs requis ne sont pas valides.

- a. Le système envoie un message d'alerte : " Nom d'utilisateur ou mot de passe incorrect ".
 - Le scenario reprend au point 3 du scenario nominal.

- Le système est toujours fonctionnel ;
- L'utilisateur est connecté.

CU: Ajouter une demande

SOMMAIRE D'IDENTIFICATION

Titre: Ajouter une demande

Résumé : Permet au personnel de faire une demande de sortie de matériel

Acteur : Directeur, Secrétaire & Personnel

Date de création : 16/06/2022 **Date mis à jour :** 30/07/2022

Version: 1.0

Responsable: AWESSO Lidaou Espoir

DESCRIPTION DES SCENARII

Pré - conditions

• L'utilisateur est authentifié.

• L'utilisateur dispose des droits pour faire une demande.

Scénario nominal

- 1. Dans le menu, l'utilisateur clique sur le lien **Demandes & Affectations** puis sur **Liste des demandes**.
- 2. Le système ouvre la page de gestion des demandes.
- 3. L'utilisateur clique sur le bouton Ajouter une demande.
- 4. Le système affiche le formulaire d'ajout d'une demande de sortie du matériel.
- 5. L'utilisateur saisit les informations requises et valide.
- 6. Le système vérifie les informations et enregistre les informations de la demande (A1).
- 7. Le système envoie un message de succès.

Scénarii alternatifs

(A1): Les données reçues par le serveur sont erronées.

Ce scénario démarre au point 6 du scénario nominal lorsque les données reçues par le système sur les champs requis ne sont pas valides.

a. Le système envoie un message d'alerte avec les données erronées. Le scénario reprend au point 5 du scénario nominal.

- Le système est toujours fonctionnel ;
- La demande est ajoutée et attend d'être validée.

CU: Valider une demande

SOMMAIRE D'IDENTIFICATION

Titre: Valider une demande

Résumé : Permet d'accepter une demande de sortie de matériel.

Acteur : Directeur & Secrétaire Date de création : 16/06/2022 Date mis à jour : 30/07/2022

Version: 1.0

Responsable: AWESSO Lidaou Espoir

DESCRIPTION DES SCENARII

Pré - conditions

• L'utilisateur est authentifié.

• L'utilisateur dispose des droits validation des demandes.

Scénario nominal

- 1. Dans le menu, l'utilisateur clique sur le lien **Demandes & Affectations** puis sur **Liste des demandes**.
- 2. Le système ouvre la page de gestion des demandes.
- 3. Dans la liste des demandes, l'utilisateur clique sur le bouton **Détail** de la demande à traiter.
- 4. Le système affiche la liste des détails de la demande.
- 5. L'utilisateur clique sur le bouton Tout valider.
- 6. Le système envoie un message de confirmation de validation.
- 7. L'utilisateur confirme la validation de la demande.

Le système envoie un message de succès.

Scénarii alternatifs

(A1) : Rejet d'un détail de la demande.

Le scénario se déclenche au point 4 lorsque l'utilisateur clique sur le bouton **Rejeter** de l'un des détails de la demande.

- a. Le système ouvre le formulaire de confirmation de rejet, avec saisi du motif de refus.
- b. L'utilisateur confirme le rejet et valide.
- c. Le système envoie un message de succès.

Le scénario reprend au point 5 du scénario nominal.

Scénario d'exception

(E1):

Le scénario se déclenche lorsque l'utilisateur clique sur le boutons **Rejeter** du seul détail contenu dans la demande.

Le scénario se termine par un Rejet de la demande.

- Le système est toujours fonctionnel ;
- La demande est validée et attend d'être servie.

CU: Ajouter une affectation

SOMMAIRE D'IDENTIFICATION

Titre: Ajouter une affectation

Résumé : Permet de faire une affectation de matériel au personnel demandeur.

Acteur : Directeur & Secrétaire Date de création : 16/06/2022 Date mis à jour : 30/07/2022

Version: 1.0

Responsable: AWESSO Lidaou Espoir

DESCRIPTION DES SCENARII

Pré - conditions

• L'utilisateur est authentifié.

• L'utilisateur dispose des droits pour faire une affectation de matériel.

Scénario nominal

- 1. Dans le menu, l'utilisateur clique sur le lien **Demandes & Affectations** puis sur **Liste des affectations**.
- 2. Le système ouvre la page de gestion des affectations.
- 3. L'utilisateur clique sur le bouton Ajouter une affectation.
- 4. Le système affiche le formulaire d'ajout d'une affectation de matériel.
- 5. L'utilisateur saisi les informations requises et valide.
- 6. Le système vérifie les informations et enregistre les informations de l'affectation (A1).
- 7. Le système envoie un message de succès.

Scénarii alternatifs

(A1): Les données reçues par le serveur sont erronées.

Ce scénario démarre au point 6 du scénario nominal lorsque les données reçues par le système sur les champs requis ne sont pas valides.

b. Le système envoie un message d'alerte avec les données erronées. Le scénario reprend au point 5 du scénario nominal.

- Le système est toujours fonctionnel ;
- L'affectation est ajoutée.

2.4.4 Diagramme d'activité

❖ S'authentifier

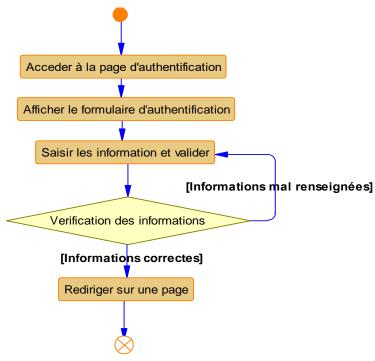


Figure 4:Diagramme d'activité de « s'authentifier »

❖ Ajouter une demande

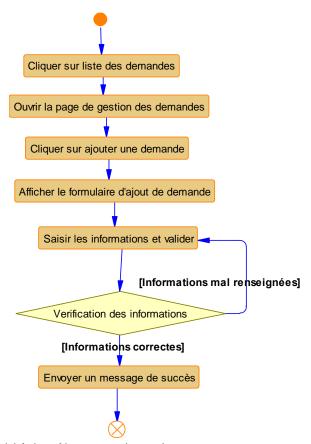


Figure 5:Diagramme d'activité de « Ajouter une demande »

Valider une demande

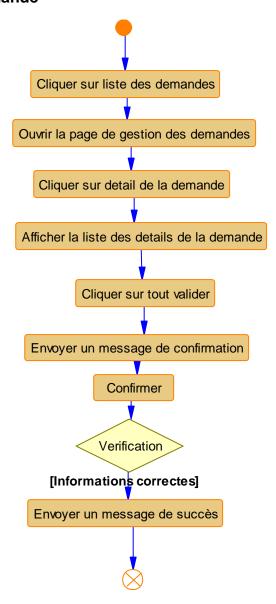


Figure 6:Diagramme d'activité de « Valider un demande »

❖ Ajouter une affectation

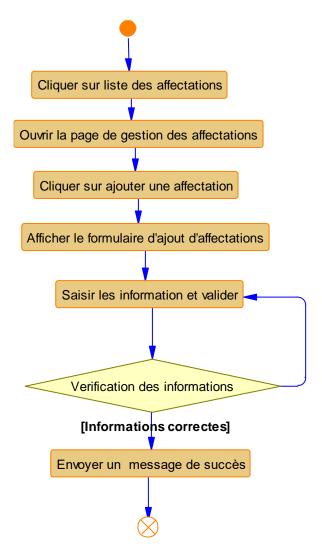


Figure 7:Diagramme d'activité de « Ajouter une affectation »

2.4.5 Diagramme de séquence système

Un diagramme de séquence système est un diagramme qui représente la séquence de message entre l'utilisateur et le système.

❖ S'authentifier

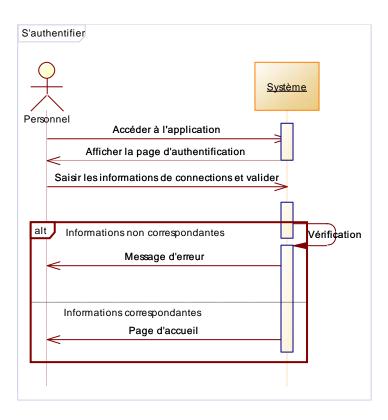


Figure 8:Diagramme de séquence de « S'authentifier »

* Ajouter une demande

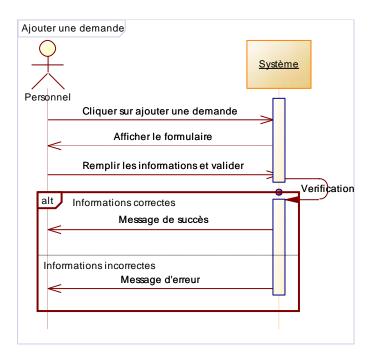


Figure 9:Diagramme de séquence de « Ajouter une demande »

Valider une demande

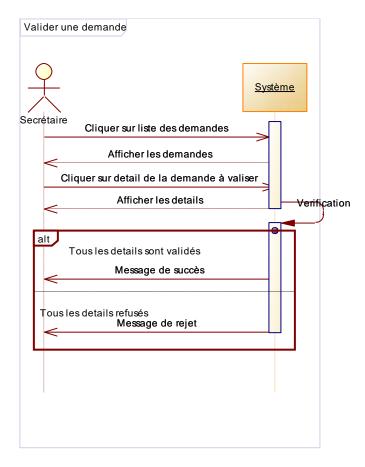


Figure 10:Diagramme de séquence de « Valider une demande »

❖ Ajouter une affectation

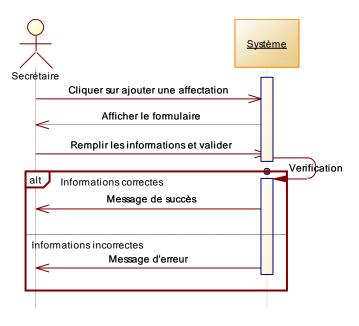


Figure 11:Diagramme de séquence de « Ajouter une affectation »

2.4.6 Diagramme de classe

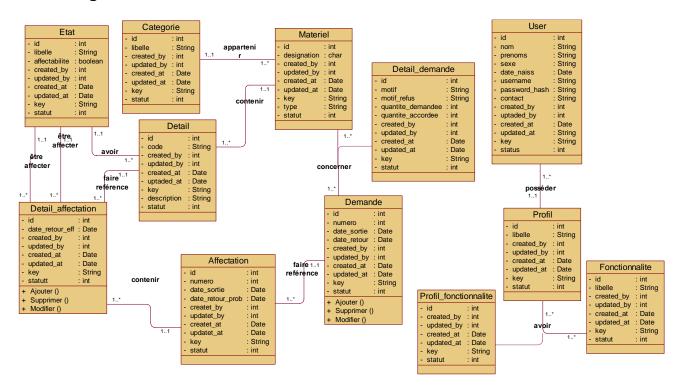


Figure 12:Diagramme des classes

PARTIE 3 : REALISATION ET MISE EN OEUVRE

PARTIE3: REALISATION ET MISE EN ŒUVRE

3.1 Matériels et logiciel utilisé

3.1.1 Matériels

Comme matériel nous avons utilisé un ordinateur portable dont les caractéristiques sont les suivantes :

♣ Marque : DELL LATITUDE E7240

♣ Processeur : Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz 2.50 GHz

♣ Mémoire installée (RAM) : 4.00 Go

♣ Disque dur : 120Go♣ Carte réseau : Oui♣ Lecteur : Non

3.1.2 Logiciels

3.1.2.1 Système de gestion de base de données

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. Il a été acheté le 16 janvier 2008 par Sun Microsystems. En 2009, Sun Microsystems a été acquis par Oracle Corporation, mettant entre les mains d'une même société les deux produits concurrents que sont Oracle Database et MySQL. Ce rachat a été autorisé par la Commission européenne le 21 janvier 2010. Son nom vient du prénom de la fille du cocréateur Michael Widenius, My. 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. 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 générale GNU (GPL) qui s'applique.



Figure 13:MySQL

3.1.2.2 Editeur de code

Visual Studio Code

Visual studio code est un éditeur de code extensible développé par Microsoft, Windows, Linux et MacOs.

Les fonctionnalités incluent la prise en charge du déblocage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis claviers, les préférences et installer des extensions qui ajoutent des fonctionnalités supplémentaires.

Le code source de Visual Studio Code provient du projet logiciel libre et open source VS Code de Microsoft publié sous licence MIT permissive, mais les binaires compilés sont des logiciels gratuits pour toute utilisation.

Dans le Stack Overflow 2019 Developer Survey. Visual Studio Code a été classé comme l'outil d'environnement de développement le plus populaire, avec 50,7% des 87317 répondant favorablement à son utilisation.



Figure 14: Visual Studio Code

3.1.2.3 Framework

¥ Yii2

Yii est un framework PHP haute performance basé sur des composants pour le développement rapide d'applications web modernes. Le nom Yii (prononcé ou) signifie « simple et évolutif » en chinois. Il peut également être considéré comme un acronyme pour Yes It Is! Yee [ji:]

Yii est un framework de programmation web générique, ce qui signifie qu'il peut être utilisé pour développer toutes sortes d'applications web en utilisant PHP. En raison de son architecture basée sur des composants et de sa prise en charge sophistiquée de la mise en cache, il est particulièrement adapté au développement d'applications à grande échelle telles que des portails, des forums, des systèmes de gestion de contenu (CMS), des projets de commerce électronique, des services web RESTful, etc.

Si vous êtes déjà familier avec un autre framework, vous apprécierez peut-être de savoir comment Yii se compare:

Comme la plupart des frameworks PHP, Yii implémente le modèle architectural MVC (Model-View-Controller) et promeut l'organisation du code en fonction de ce modèle.

Yii prend la philosophie selon laquelle le code doit être écrit d'une manière simple mais élégante. Yii n'essaiera jamais de sur-concevoir des choses principalement dans le but de suivre strictement un modèle de conception.

Yii est un framework full-stack offrant de nombreuses fonctionnalités éprouvées et prêtes à l'emploi : générateurs de requêtes et ActiveRecord pour les bases de données relationnelles et NoSQL ; Prise en charge du développement d'API RESTful; prise en charge de la mise en cache multiniveau; et plus encore.

Yii est extrêmement extensible. Vous pouvez personnaliser ou remplacer presque tous les éléments du code du noyau. Vous pouvez également tirer partie de l'architecture d'extension solide de Yii pour utiliser ou développer des extensions redistribuables.

La haute performance est toujours un objectif principal de Yii.



Figure 15:Yii2

3.2 Sécurité de l'application

Pour assurer la sécurité de notre application, un certain nombre de règles de sécurité et de bonnes pratiques ont été mises en place dans notre développement :

- L'authentification : Avant d'effectuer une action quelconque sur la plateforme, l'utilisateur doit obligatoirement s'authentifier ;
- Un utilisateur a accès uniquement aux fonctionnalités correspondant à son profil
 :
- L'application du principe du moindre privilège au niveau de la sécurité de la base de données et de la gestion des utilisateurs dans l'application.

3.3 Evaluation financière de la solution

Tableau 3:Evaluation financière de la solution

Désignation Désignation	Description	Prix Unitaire	Quantité	Total
Matériel	Ordinateur 4gb de ram 2,00 GHz	180000	2	360000
Logiciel	Configuration des outils (Xamp, SSL)	5000	2	10000
Conception	Montant pour la conception de l'application	2000/heure	200 (8h x 25)	400000
Formation	Montant de formation du personnel	10000/catégorie	2	20000
Total				790000

3.4 Présentation de l'application

Notre application est une application web qui permet essentiellement la gestion de stock de matériels. Elle se décompose en plusieurs modules suivant l'utilisateur qui s'authentifie. Ainsi le système permet au agents de l'entreprise de gérer chacun de ses cotés les taches soumises à leur niveau.

3.4.1 Mise en place de la base de données

- Code source de création de la base de données :
- CREATE DATABASE clin_db_stock;
- Code source de création des tables :
- CREATE Table Etat (
- id int AUTO_INCREMENT PRIMARY KEY,
- libelle VARCHAR(50) NOT NULL,
- affetabilite boolean NOT NULL,

```
key_etat CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated by int(11),
    created at DATETIME not NULL,
    updated_at DATETIME
);
CREATE Table Detail affectation (
    id int AUTO_INCREMENT PRIMARY KEY,
    date retour eff DATETIME,
    key detailaffectation CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated at DATETIME,
    idEtatsortie int NOT NULL,
    idEtatretour int,
    idDetail int NOT NULL,
    idAffectation int NOT NULL
);
CREATE Table Categorie (
    id int AUTO INCREMENT PRIMARY KEY,
    libelle VARCHAR(50) NOT NULL,
    key_categorie CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated at DATETIME
);
CREATE Table Detail (
    id int AUTO_INCREMENT PRIMARY KEY,
    code VARCHAR(50) NOT NULL,
    key detail CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated_at DATETIME,
    idEtat int NOT NULL,
    idMateriel int NOT NULL
);
CREATE Table Affectation (
    id int AUTO INCREMENT PRIMARY KEY,
    numero char(10) NOT NULL,
    date sortie DATETIME NOT NULL,
    date_retour_prob DATETIME NOT NULL,
    key affectation CHAR(32) not NULL,
    statut int(1) not NULL,
    created by int(11) NOT NULL,
```

```
updated_by int(11),
    created_at DATETIME not NULL,
    updated_at DATETIME,
    idDemande int NOT NULL
);
CREATE Table Materiel (
    id int AUTO_INCREMENT PRIMARY KEY,
    designation VARCHAR(50) NOT NULL,
    type_materiel VARCHAR(50),
    description TEXT,
    key materiel CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated at DATETIME,
    idCategorie int NOT NULL
);
CREATE Table Demande (
    id int AUTO_INCREMENT PRIMARY KEY,
    numero char(10) NOT NULL,
    date sortie DATETIME NOT NULL,
    date retour DATETIME NOT NULL,
    key_demande CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated_at DATETIME
);
CREATE Table Detail demande (
    id int AUTO_INCREMENT PRIMARY KEY,
    motif TEXT,
    motif refus TEXT,
    quantite demandee int(11) NOT NULL,
    quantite_accordee int(11),
    key_detaildemande CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated at DATETIME,
    idMateriel int NOT NULL,
    idDemande int NOT NULL
);
CREATE Table Profil (
    id int AUTO INCREMENT PRIMARY KEY,
    libelle VARCHAR(50) NOT NULL,
    key profil CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
```

```
updated_by int(11),
    created_at DATETIME not NULL,
    updated_at DATETIME
);
CREATE Table Fonctionnalite (
    id int AUTO INCREMENT PRIMARY KEY,
    libelle VARCHAR(50) NOT NULL,
    code char(50) NOT NULL,
    key_fonctionnalite CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created_at DATETIME not NULL,
    updated_at DATETIME
);
CREATE Table User (
    id int(11) AUTO_INCREMENT PRIMARY KEY,
    nom VARCHAR(50) NOT NULL,
    prenoms VARCHAR(50) NOT NULL,
    sexe CHAR(1) NOT NULL,
    date_naiss DATETIME not NULL,
    username VARCHAR(50) NOT NULL,
    password hash VARCHAR(255) NOT NULL,
    password_reset_token VARCHAR(255),
    telephone VARCHAR(20) NOT NULL,
    email VARCHAR(100) NOT NULL,
    role smallint(6) NOT NULL,
    auth_key CHAR(32) not NULL,
    status smallint(6) not NULL,
    created by int(11) NOT NULL,
    updated_by int(11),
    created at DATETIME not NULL,
    updated_at DATETIME,
    idProfil int NOT NULL
);
CREATE Table profil_fonctionnalite (
    id int AUTO_INCREMENT PRIMARY KEY,
    key profilfonctionnalite CHAR(32) not NULL,
    statut int(1) not NULL,
    created_by int(11) NOT NULL,
    updated_by int(11),
    created at DATETIME not NULL,
    updated at DATETIME,
    idProfil int NOT NULL,
    idFonctionnalite int NOT NULL
);
```

3.4.2 Plan de navigation

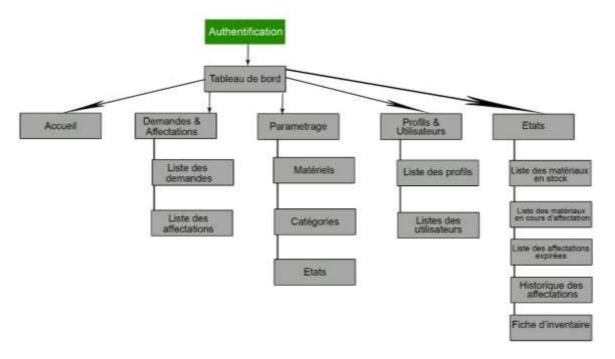


Figure 16:Plan de navigation

3.4.3 Quelques masques de saisie

3.4.3.1 Formulaire d'authentification des agents

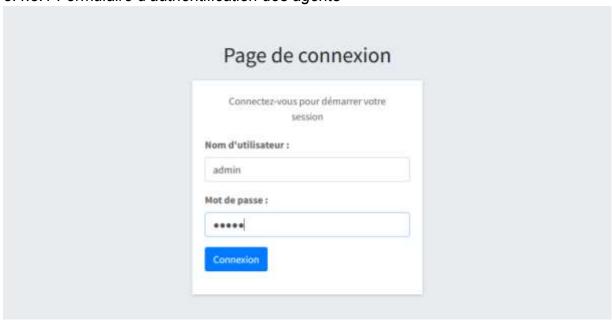


Figure 17:Formulaire d'authentification des agents

3.4.3.2 Formulaire d'ajout d'une demande

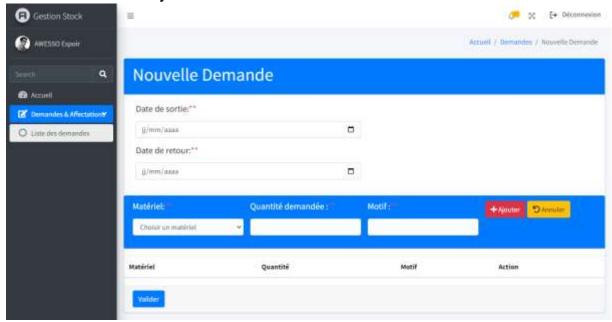


Figure 18:Formulaire d'ajout d'une demande

3.4.3.3 Formulaire d'ajout d'une affectation

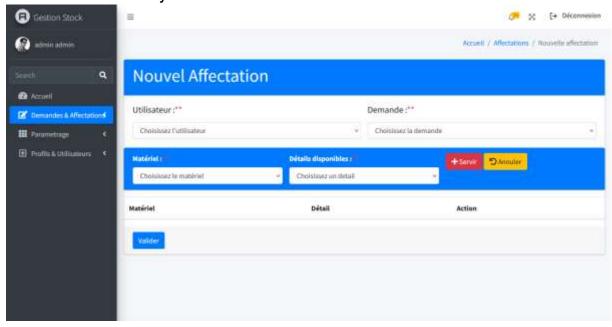


Figure 19:Formulaire d'ajout d'une affectation

3.4.4 Quelques états et statistiques

3.4.4.1 Historique des affectations

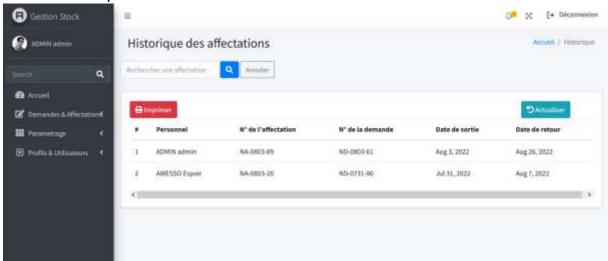


Figure 20:Historique des affectations

3.4.4.2 Répartition des matériels par catégorie



Figure 21:Répartition des matériels par catégorie

3.4.4.3 Répartition des matériels selon leurs états et répartition des affectations des matériels



Figure 22:Répartition des matériels selon leurs états et répartition des affectations des matériels

CONCLUSION

Notre projet avait pour objectif de mettre en place une application de gestion de stock. 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-àvis 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

Année académique : 2021-2022

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

WEBOGRAPHIE INDICATIVE

Année académique : 2021-2022

- https://www.google.com/, Tout le long du projet ;
- https://stackoverflow.com/, Tout le long du projet ;
- https://www.youtube.com/, tout le long du projet ;

DOCUMENTS ANNEXES

- EKOUE TOULAN Dédé Cornela Faisance, Automatisation de la gestion de stock : cas d'une quincaillerie (2019 – 2020).

TABLE DES MATIERES

REMERCIEMENTS	i
SOMMAIRE	ii
GLOSSAIRE	iii
INTRODUCTION	1
PARTIE1 : CAHIER DES CHARGES	3
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 : ANALYSE ET CONCEPTION	6
2.1 ETUDE DE L'EXISTANT	6
2.2 CRTITIQUE DE L'EXISTANT	6
2.3 Planning prévisionnel de réalisation	7
2.4 ETUDE DETAILLEE DE LA SOLUTION	7
2.4.1 Choix de l'outils d'analyse	7
2.4.2 Présentation de l'outils de modélisation	8
2.4.3 Diagramme de cas d'utilisation	9
2.4.4 Diagramme d'activité	16
2.4.5 Diagramme de séquence système	18
2.4.6 Diagramme de classe	21
PARTIE3 : REALISATION ET MISE EN ŒUVRE	23
3.1 Matériels et logiciel utilisé	23
3.1.1 Matériels	23
3.1.2 Logiciels	23
3.2 Sécurité de l'application	25
3.3 Evaluation financière de la solution	26
3.4 Présentation de l'application	26
3.4.1 Mise en place de la base de données	26
3.4.2 Plan de navigation	30
3.4.3 Quelques masques de saisie	30
3.4.4 Quelques états et statistiques	32
CONCLUSION	34
BIBLIOGRAPHIE INDICATIVE	35
WEBOGRAPHIE INDICATIVE	36
DOCUMENTS ANNEXES	37