RÉPUBLIQUE DU BÉNIN
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ D'ABOMEY-CALAVI

INSTITUT DE FORMATION ET DE RECHERCHE EN INFORMATIQUE

BP 526 Cotonou Tel : +229 21 14 19 88
http://www.ifri-uac.net Courriel : contact@ifri.uac.bj

MÉMOIRE

pour l'obtention du

Diplôme de Licence en Informatique

Option : Génie Logiciel

Présenté par :
Kpèdétin Elfriede **BOTON**

Conception d'une application web pour la gestion des ressources matérielles : cas de AFRITECH

Sous la supervision :
Ing. Miranda **GNONLONFOUN**

Membres du jury :

Patrick SOTINDJO	Dr	IFRI, UAC	Président
Matine OUSMANE	Dr	IFRI, UAC	Examinateur
Miranda GNONLONFOUN	Ing	IFRI, UAC	Rapporteur

Année Académique : **2023-2024**

Sommaire

Dédicace ii

Remerciements iii

Résumé iv

Abstract v

Liste des figures vi

Liste des tableaux vii

Liste des acronymes viii

Introduction 1

1 Revue de littérature 3

2 Analyse, Modélisation, Choix techniques 11

3 Résultats et discussion 22

Conclusion 31

Bibliographie 32

Webographie 33

Table des matières 34

Dédicace

Je dédie ce travail à :

- l'Eternel, mon Dieu, pour son accompagnement et sa protection tout au long de ce parcours académique qui s'achève ;
- Mon feu père Germain BOTON ;
- Ma mère Augustine HOUNWANOU ;

Remerciements

Nous tenons à remercier :

• L'Institut de Formation et de Recherche en Informatique, en particulier son Directeur le Professeur Eugène C. EZIN pour tout ce qu'il met en oeuvre pour la réussite de l'Institut et de ses étudiants ;

• Le Directeur Adjoint de l'Institut de Formation et de Recherche en Informatique, le Professeur Gaston EDAH pour tout le travail et l'accompagnement le long de la formation des étudiants ;

• Madame Miranda GNONLONFOUN, notre maître de mémoire pour sa disponibilité, sa patience, son accompagnement et la confiance qu'elle a placée en nous ;

• Monsieur Josaphat AMOUSSOU, notre maitre de stage pour le suivi, la disponibilité, les conseils et la patience qu'il nous a accordé durant toute la période de stage, ainsi que son collaborateur Mr Chidiac pour son soutien et sa collaboration précieuse ;

• Tout le personnel administratif ainsi que tous les enseignants de l'Institut de Formation et de Recherche en Informatique pour leur contribution à notre formation ;

• Toute notre famille pour leur soutien et leur encouragement ;

• Tous nos camarades et amis ;

Que tous ceux qui de près ou de loin ont contribué à l'élaboration et à la concrétisation de ce travail trouvent en celui-ci l'expression de notre profonde reconnaissance.

Résumé

AfriTech est une entreprise technologique confrontée à des difficultés dans la gestion de ses équipements, ce qui engendre des retards et des dysfonctionnements logistiques. Ces contraintes nuisent à l'efficacité opérationnelle et à la traçabilité des ressources. Pour y remédier, nous proposons la mise en place d'une plateforme web automatisée, conçue à l'aide des technologies PHP, HTML, CSS et JavaScript, afin de centraliser, suivre et optimiser la gestion des équipements en temps réel. Cette solution permettra de rationaliser l'affectation du matériel entre les différents services, d'assurer une meilleure visibilité sur l'état et la localisation des équipements, et de faciliter le processus de décision grâce à des rapports détaillés. La structure de l'application reposera sur un modèle UML définissant les interactions entre les différents types d'utilisateurs (administrateurs, employés, gestionnaire de matériel) et les fonctionnalités du système, à travers une interface intuitive et ergonomique.

Mots clés : AfriTech, la gestion des équipements, plateforme web, services, système, application.

Abstract

AfriTech is a technology company facing challenges managing its equipment, which leads to delays and logistical malfunctions. These constraints hamper operational efficiency and resource traceability. To address this, we propose the implementation of an automated web platform, designed using PHP, HTML, CSS, and JavaScript technologies, to centralize, track, and optimize equipment management in real time. This solution will streamline equipment allocation between different departments, provide better visibility into equipment status and location, and facilitate the decision-making process through detailed reports. The application structure will be based on a UML model defining the interactions between different types of users (administrators, employees, equipment manager) and system functionalities, through an intuitive and ergonomic interface.

Key words: AfriTech, equipment management, web platform, services, system, application.

Liste des figures

1.1 Organigramme de AFRITECH 4

2.1 Diagramme des cas d'utilisation 13
2.2 Diagramme des classes 14
2.3 Diagramme de séquence d'authentification 15
2.4 Diagramme de séquence de demande d'actif 15
2.5 Méthode de cycle en v [9] 16
2.6 Architecture MVC 17
2.7 VS Code 19
2.8 StarUML 20

3.1 Page de connexion 22
3.2 Page d'Accueil 23
3.3 Page d'ajout d'actif 23
3.4 Liste des affectations 24
3.5 Enregistrement de demande 24
3.6 Liste des demandes 25
3.7 Ajouter un fournisseur 25
3.8 Les actifs en stock 26
3.9 Ajout de catégorie d'actif 26
3.10 Liste des actifs 27
3.11 Amortissement 27
3.12 Ajout d'un service 28
3.13 Liste des services 28
3.14 Commande 29
3.15 Demande d'actif 29

Liste des tableaux

1.1 Inventaire des équipements informatiques 5
1.2 Avantages et limites des solutions 9

2.1 Rôles des acteurs 12
2.2 Résumé des outils, langages et frameworks employés. 20

Liste des acronymes

AFRITECH :

African Technologies

CSS :

Cascading Style Sheets

HTML :

Hypertext Markup Language

HTTP :

Hypertext Transfer Protocol

ITAM :

IT Asset Management

MVC :

Modèle-Vue-Contrôleur

PHP :

Hypertext Preprocessor

PWA :

Progressive Web Apps

SaaS :

Software as a Service

SGBD :

Système de Gestion de Base de données

SQL :

Structured Query Language

UAC :

Université d'Abomey-Calavi

UML :

Unified Modeling Language

VSC :

Visual Studio Code

XML :

eXtensible Markup Language

Introduction Générale

1. Contexte et justification

Le développement des entreprises technologiques nécessite une gestion efficace des ressources matérielles, y compris les équipements informatiques et autres actifs. Afritech utilise des feuilles Excel pour gérer ses ressources, cette méthode devient inefficaces avec l'augmentation du volume de matériel, entraînant des erreurs, des retards, une mauvaise allocation des ressources et un manque de traçabilité en temps réel. Pour remédier à ces problèmes et améliorer l'efficacité opérationnelle, Afritech a besoin d'une solution logicielle sur mesure, permettant une gestion centralisée et proactive des actifs matériels.

2. Problématique

L'utilisation de feuilles Excel par Afritech entraîne des erreurs, un manque de traçabilité et une gestion inefficace des ressources matérielles. Comment informatiser la gestion des stocks pour optimiser le suivi, l'allocation et l'utilisation des équipements ?

3. Objectifs

Notre objectif principal est de concevoir une plateforme de gestion des ressources matérielles pour Afritech. De façon spécifique, notre solution permettra de :

- Enregistrer chaque actif avec ses caractéristiques (nom, quantité, catégorie, etc.).
- Effectuer des affectations et suivre les actifs affectés aux différents services, ainsi que les réaffectations.
- Gérer les contrats d'approvisionnement et les informations des fournisseurs.
- Fournir des statistiques sur l'utilisation, l'état des équipements et la disponibilité des actifs pour une meilleure prise de décision.
- Suivre en temps réel l'état et la disponibilité des ressources.

4. Organisation du mémoire

Le présent travail est structuré en trois (03) chapitres. Le premier chapitre est consacré à la présentation de la structure d'accueil ainsi qu'à la revue de littérature. Le deuxième chapitre traite de l'analyse, de la modélisation et des choix techniques effectués pour la conception de la solution proposée. Enfin, le troisième chapitre porte sur la réalisation de l'application.

Chapitre 1

Revue de littérature

Introduction

Le présent travail s'inscrit dans un contexte où la gestion des ressources matérielles est un enjeu clé pour les entreprises. Ce chapitre propose une revue de littérature sur le sujet. Il commencera par une présentation générale de la gestion des équipements et des stocks, suivie de l'examen des solutions existantes comme Odoo, Zoho, Asset Panda, et Snipe-IT, en soulignant leurs avantages et leurs limites par rapport aux besoins spécifiques de Afritech.

1.1 Présentation de la structure d'accueil : AFRICAN TECHNOLOGIES

African Technologies (AFRITECH) est une entreprise actuellement présente au Bénin et au Togo. Elle est spécialisée dans la réalisation des logiciels, site web, et offre des services informatique à sa clientèle. Elle est située au Bénin au quartier Saint Michel à Cotonou. Créée en 2019, elle est composée d'une jeune équipe dynamique ayant une grande expertise dans le domaine du développement des applications, des télécoms, des réseaux et des systèmes d'information. AFRITECH est une entreprise révolutionnaire et innovatrice, satisfaisant toute sa clientèle et l'aidant à augmenter sa valeur ajoutée. Pour ce faire, AFRITECH entend étendre ses agences partout en Afrique, un projet d'installation au Mali, au Sénégal, au Burkina-Faso, au Niger est en cours. AFRITECH s'est donnée la mission de conseiller et d'orienter sa clientèle dans la prise de décisions sur les plans stratégiques, organisationnels et sécuritaires, liée au déploiement et à la mise en œuvre des systèmes d'information et de gestion. Son rôle est aussi d'informer, former et assister ses clients et leurs partenaires dans la mise en œuvre des solutions relevant de la gestion, de l'informatique, des réseaux, et de la télécommunication. Elle assiste de même ses partenaires et clients dans le renforcement des capacités et le recyclage du personnel.

1.1.1 Domaines d'interventions

Quelques domaines d'interventions de AFRITECH

- Développement mobile et web avec FlutterFlow ;
- Infographie/design (PHOTOSHOP, INDESIGN, ILLUSTRATOR.)
- réseau (interconnexion de routeur, Réseau local et Etendu, câblage…)
- administration (LINUX, WINDOWS SERVER) ;
- maintenance informatique ;
- audit informatique ;
- politiques et référentiel de sécurité informatique ;
- assistance RSSI (Responsable Sécurité Système Informatique) ;
- développement informatique général web (PHP, HTML, JAVASCRIPT, CSS, JAVA, C, C++), maitrise des logiciels standards ;
- etc.

1.1.2 Organigramme

L'administration de AFRITECH peut être représenter par l'organigramme ci-dessous :

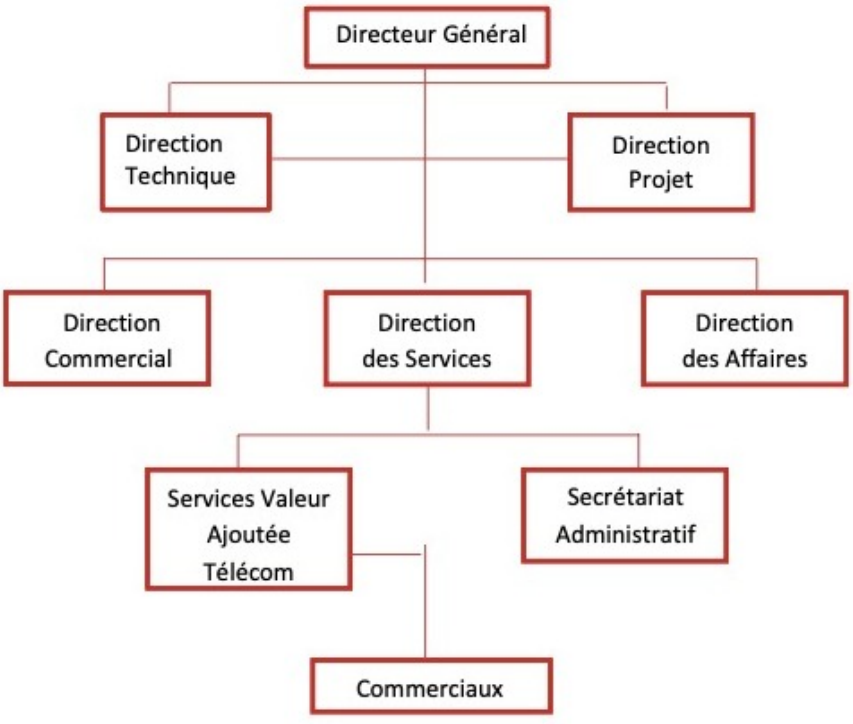


FIGURE 1.1 : Organigramme de AFRITECH

Matériel	**Quantité**
Ordinateur portable	15
Ordinateur bureautique All-in-One	12
Imprimante	11
Copieur	2
Onduleur	31
Régulateur	8
Scanner	9
Classeurs de dossiers	43
Armoire mobile	73
Bureaux	63
Fauteuil bureaux	72
Fauteuil ordinaire	115
Chaise d'attente (3 en 1)	25
Boîte Ancre imprimante	150
Cartouche encre imprimante	76
Caméra de surveillance	32
Serveur PROLIANT dl380 GEN10	2
switch	16
Routeur	5
Paquet de papier ram	40
Clavier a fil	50
Clavier sans fil	12
Souris sans fil	20
Sousris à fil	50
Clé USB	24
Câble	30
Scanner	8
Caméra de surveillance	32
Téléphones fixes	10

TABLE 1.1 : Inventaire des équipements informatiques

L'inventaire de AFRITECH Bénin regroupe du matériel informatique, bureautique, de sécurité et du mobilier. Le siège de l'entreprise est situé à Lomé, au Togo, et elle dispose de quatre agences réparties dans différentes villes du pays.

1.2 Clarification conceptuelle

1.2.1 Application web

a. Définition [1]

Une application web est un programme accessible via un navigateur (comme Chrome, Safari ou Firefox), hébergé sur un serveur, qui ne nécessite aucune installation sur les appareils des utilisateurs et permet d'accomplir des tâches spécifiques.

b. Fonctionnement des applications web

Toutes les applications qui existent ont leurs modes de fonctionnement. Pour savoir comment les applications web fonctionnent dans le cadre de mon projet, il nous faut connaître leurs besoins.

Les applications web ont besoin d'abord d'un navigateur compatible avec les langages de programmation utilisés pour le codage, afin de rendre le programme exécutable. Elles nécessitent ensuite un serveur d'applications pour exécuter les tâches demandées, un serveur web pour gérer les requêtes des utilisateurs, et une base de données pour stocker les informations.

Les besoins déjà connu, abordons maintenant le fonctionnement des applications web. Voici comment elles fonctionnent :

- Un utilisateur lance une requête au serveur web via un navigateur ou l'interface utilisateur de l'application web ;
- Le navigateur envoie ces requêtes au serveur d'applications web approprié ;
- Conception de tableaux de bord interactifs ;
- La base de données génère ensuite les résultats de la demande lorsqu'elle est sollicitée.

c. Les différents types d'applications web [7]

- Applications web statiques : Ces applications affichent des contenus fixes préconfigurés. Elles sont légères et rapides, mais manquent d'interactivité.

  Exemple : Portfolios en ligne ou sites vitrines.

- Applications web dynamiques : Elles permettent une interaction en temps réel entre l'utilisateur et le serveur. Les données affichées changent en fonction des actions ou des requêtes de l'utilisateur.

  Exemple : Réseaux sociaux, blogs avec zones de commentaires.

- Applications e-commerce : Ces applications permettent de vendre des produits ou services en ligne, avec des fonctionnalités telles que des paniers d'achat et des passerelles de paiement.

  Exemple : Shopify, Amazon.

- Progressive Web Apps (PWA) : Les PWA combinent les fonctionnalités des applications web et des applications mobiles natives, offrant une expérience hors ligne et une installation sur le bureau ou mobile.

  Exemple : Twitter Lite, Starbucks.

- Applications SaaS (Software as a Service) : Ces applications fournissent des services en ligne sous forme d'abonnement.

  Exemple : Dropbox, Slack.

1.3 Solutions existantes

1.3.1 Odoo

[12] Odoo est un logiciel de gestion d'entreprise open source qui propose une suite d'applications pour gérer diverses fonctions d'une entreprise, telles que la gestion de la comptabilité, des ventes, des achats, de l'inventaire, des ressources humaines, du CRM, du marketing, et bien plus encore. Odoo est modulaire, ce qui permet aux entreprises de personnaliser leur système en fonction de leurs besoins spécifiques.

1.3.2 Zoho Inventory

[18] Zoho Inventory est un logiciel de gestion des stocks et des actifs qui permet aux entreprises de suivre leurs produits, gérer les commandes, optimiser les niveaux de stock et automatiser les flux d'inventaire. Il offre des fonctionnalités comme la gestion multi-entrepôts, l'intégration avec les plateformes de e-commerce, le suivi des expéditions et la génération de rapports analytiques.

1.3.3 Asset Panda

[2] Asset Panda est un logiciel de gestion des actifs qui permet aux entreprises de suivre, gérer et optimiser l'utilisation de leurs équipements en temps réel. Il offre des fonctionnalités comme le suivi par code-barres ou RFID, la gestion des audits, la maintenance préventive et des rapports détaillés. Accessible sur le web et mobile, il permet une personnalisation avancée selon les besoins de l'entreprise.

1.3.4 NetSuite

[11] NetSuite est une solution ERP utilisée pour gérer les actifs de l'entreprise, les achats, les ventes, l'inventaire, les finances et les ressources humaines. Bien qu'elle soit principalement une version cloud payante, certaines fonctionnalités sont disponibles dans des versions limitées ou open-source. Elle permet une intégration complète des actifs dans la gestion globale de l'entreprise.

1.3.5 CMDBuild

[3] CMDBuild est un outil open source conçu pour la gestion des biens d'entreprise et des processus métiers. Il permet de créer une base de données centralisée des actifs et de suivre leur cycle de vie, tout en offrant des fonctionnalités avancées comme la gestion des tickets, la planification des interventions et l'intégration avec d'autres systèmes IT.

1.3.6 OpenMAINT

[13] OpenMAINT est un logiciel open source spécialisé dans la gestion des biens d'entreprise, incluant les équipements, bâtiments, infrastructures et services de maintenance. Il permet de planifier

et de suivre les interventions de maintenance, de gérer les stocks et d'optimiser l'utilisation des ressources.

1.3.7 Snipe-IT

[15] Snipe-IT est un logiciel open-source de gestion des actifs informatiques (ITAM). Il permet aux entreprises de suivre l'inventaire de leurs équipements (ordinateurs, serveurs, périphériques, etc.), de gérer les affectations aux utilisateurs et services, et d'organiser la maintenance. Il propose des fonctionnalités comme le suivi par QR code, les notifications de maintenance, la gestion des licences et des historiques de mouvement. Étant open-source, il peut être auto-hébergé et personnalisé selon les besoins spécifiques d'une organisation.

Synthèse

Le tableau ci-dessous présente un récapitulatif mettant en évidence les avantages et limites des différentes solutions analysées dans notre étude de l'existant.

Solutions	Avantages	Limites
Odoo	- Solution ERP complète avec modules intégrés. - Gestion multi-entrepôts et suivi des mouvements. - Interface intuitive et évolutive.	- Nécessite une configuration avancée pour une utilisation optimale. - Certaines fonctionnalités avancées sont payantes.
Zoho Inventory	- Gestion avancée des stocks et des actifs. - Intégration avec les plateformes e-commerce et ERP. - Interface intuitive et rapports détaillés.	- Conçu principalement pour la gestion des stocks plutôt que des équipements d'entreprise. - Payant avec des coûts élevés pour les fonctionnalités avancées.
Asset Panda	- Suivi des actifs en temps réel avec QR code/RFID. - Haut niveau de personnalisation. - Application mobile intuitive.	- Coût élevé selon le nombre d'actifs suivis. - Peut être complexe à configurer pour des besoins spécifiques.
OpenMAINT	-Gestion avancée des équipements, bâtiments et infrastructures. - Planification des interventions de maintenance. - Solution robuste pour les grandes entreprises.	- Interface moins intuitive. - Nécessite une configuration avancée. - Moins adapté à la gestion des affectations dynamiques d'actifs.
Snipe-IT	- Open-source et gratuit. - Suivi détaillé des équipements IT. - Gestion des licences et notifications de maintenance.	- Principalement axé sur les actifs informatiques, pas sur les autres équipements. - Nécessite des compétences techniques pour l'installation et la maintenance.
CMDBuild	- Solution très flexible et personnalisable. - Intégration avec d'autres outils IT et gestion des tickets. - Peut gérer tous types de ressources matérielles.	- Configuration technique complexe. -Interface moins intuitive. - Moins adapté à un usage non IT (services généraux, gestion administrative).
NetSuite	- Solution ERP robuste qui intègre la gestion des ressources matérielles avec la comptabilité, les ventes et les achats. - Très bonne gestion de l'inventaire et des achats.	-Très complexe à implémenter et à configurer. - Moins personnalisable pour les besoins spécifiques des entreprises de taille moyenne.

TABLE 1.2 : Avantages et limites des solutions

Conclusion

Les solutions comme Odoo, Zoho, Asset Panda, Netsuite, CMDBuild et Snipe-IT sont performantes mais souvent génériques ou rigides. Notre solution, conçue sur mesure pour Afritech, offre une gestion centralisée, un suivi en temps réel et des automatisations adaptées aux besoins spécifiques de l'entreprise. Les chapitres suivants détailleront notre approche et nos choix techniques.

Chapitre 2

Analyse, Modélisation, Choix techniques

Introduction

Dans le cadre de ce projet, divers outils, technologies et services ont été sélectionnés en fonction des besoins spécifiques d'Afritech. [8]Ce chapitre présente les choix effectués, ainsi que l'architecture sécurisée du système, conçue pour répondre aux exigences du projet tout en garantissant une gestion efficace des ressources matérielles.

2.1 Analyse des besoins

Notre système a pour objectif principal la gestion efficace des ressources matérielles au sein d'AFRITECH. L'analyse des besoins permettra de spécifier les exigences fonctionnelles et techniques nécessaires à l'optimisation de la gestion des stocks, des affectations et de la maintenance des actifs.

2.2 Modélisation

Pour modéliser les fonctionnalités de notre application web, nous avons choisi le langage UML (Unified Modeling Language). UML est un langage de modélisation visuelle standard, utilisé pour spécifier, visualiser et documenter les modèles de systèmes logiciels, y compris leur structure et leur conception, afin de répondre à toutes les exigences.

2.2.1 Identification des acteurs

Un acteur est une entité extérieure au système modélisé, qui interagit directement avec lui. Pour notre système de gestion des ressources matérielles, nous avons identifié cinq (05) principaux acteurs :

- Admin
- Gestionnaire de matériel
- Employé

- Fournisseur

2.2.2 Rôle des acteurs

Le tableau 2.1 ci-dessous résume les différents rôles associés à chaque acteur qui interagit avec le système.

Numero	Acteurs	Rôles
1	Admin	Gestion des utilisateurs, rôles et accès
2	Gestionnaire de matériel	Gestion des stocks, transferts, et supervision des actifs, Gestion des utilisateurs, configurations, et rapports.
3	Employé	Faire des demandes, les suivre et signaler des problèmes.
4	Fournisseur	Gérer les livraisons d'actifs et les commandes.

TABLE 2.1 : Rôles des acteurs

2.2.3 Diagramme des cas d'utilisation

Le diagramme des cas d'utilisation constitue une étape essentielle de l'analyse UML, car il résume la manière dont les différents acteurs interagissent avec le système. Ce diagramme facilite la collecte, l'analyse, et l'organisation des besoins tout en identifiant les principales fonctionnalités du système. Dans le contexte de notre projet de gestion des ressources matérielles, le diagramme de cas d'utilisation offre une vue globale des interactions entre les acteurs et le système, permettant de mieux comprendre le comportement fonctionnel de ce dernier.

Le diagramme des cas d'utilisation relatif à notre système se présente à la page 10

2.2.4 Diagramme des classes

Le diagramme des classes est un élément clé d'UML, utilisé pour représenter la structure statique du système en termes de classes et des relations qui les lient. Il permet de modéliser les entités principales du système d'information et leurs interactions.

La figure 2.2 illustre le diagramme des classes de notre système de gestion des ressources matérielles qui se trouve à la page 11.

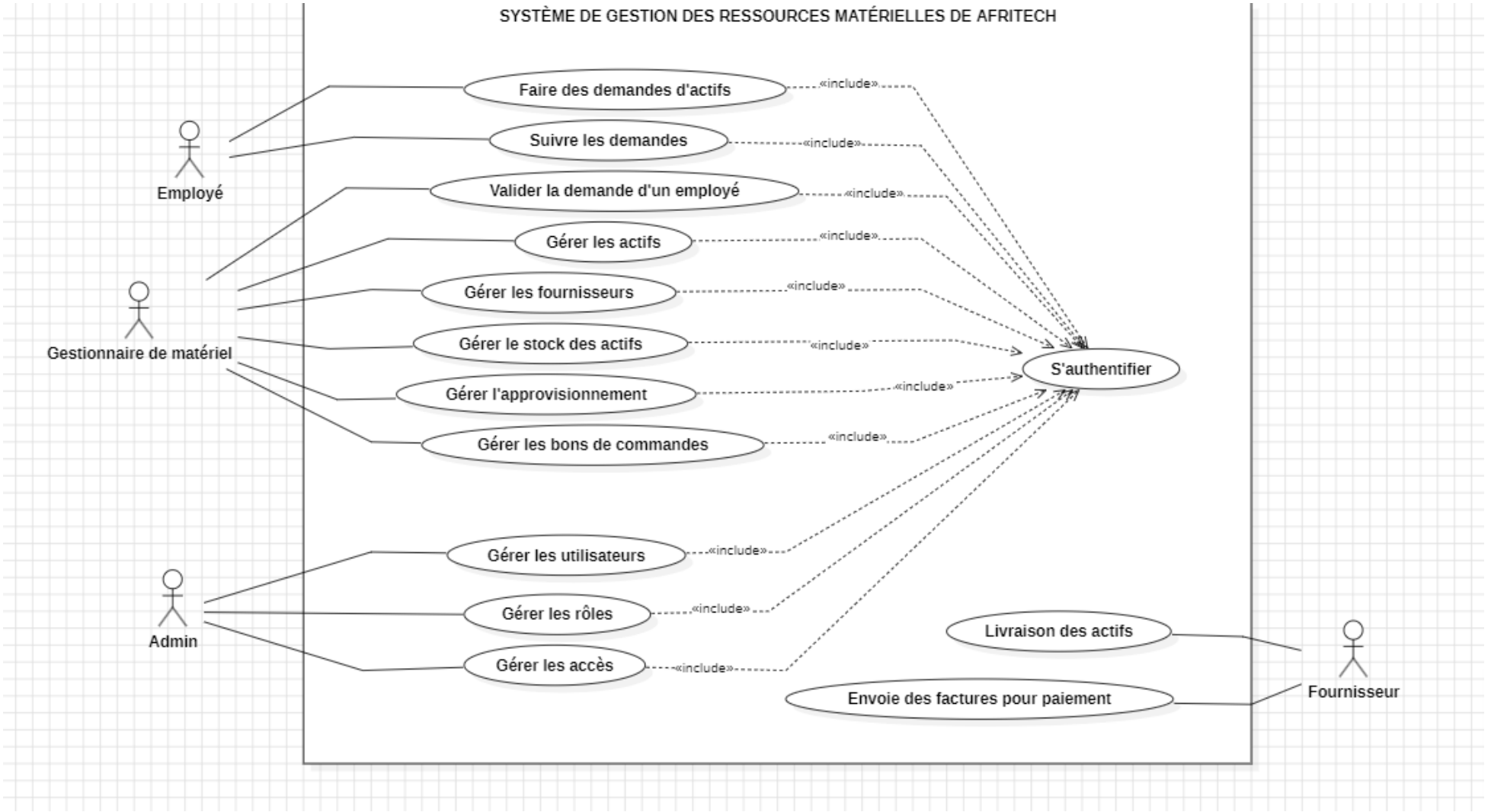


FIGURE 2.1 : Diagramme des cas d'utilisation

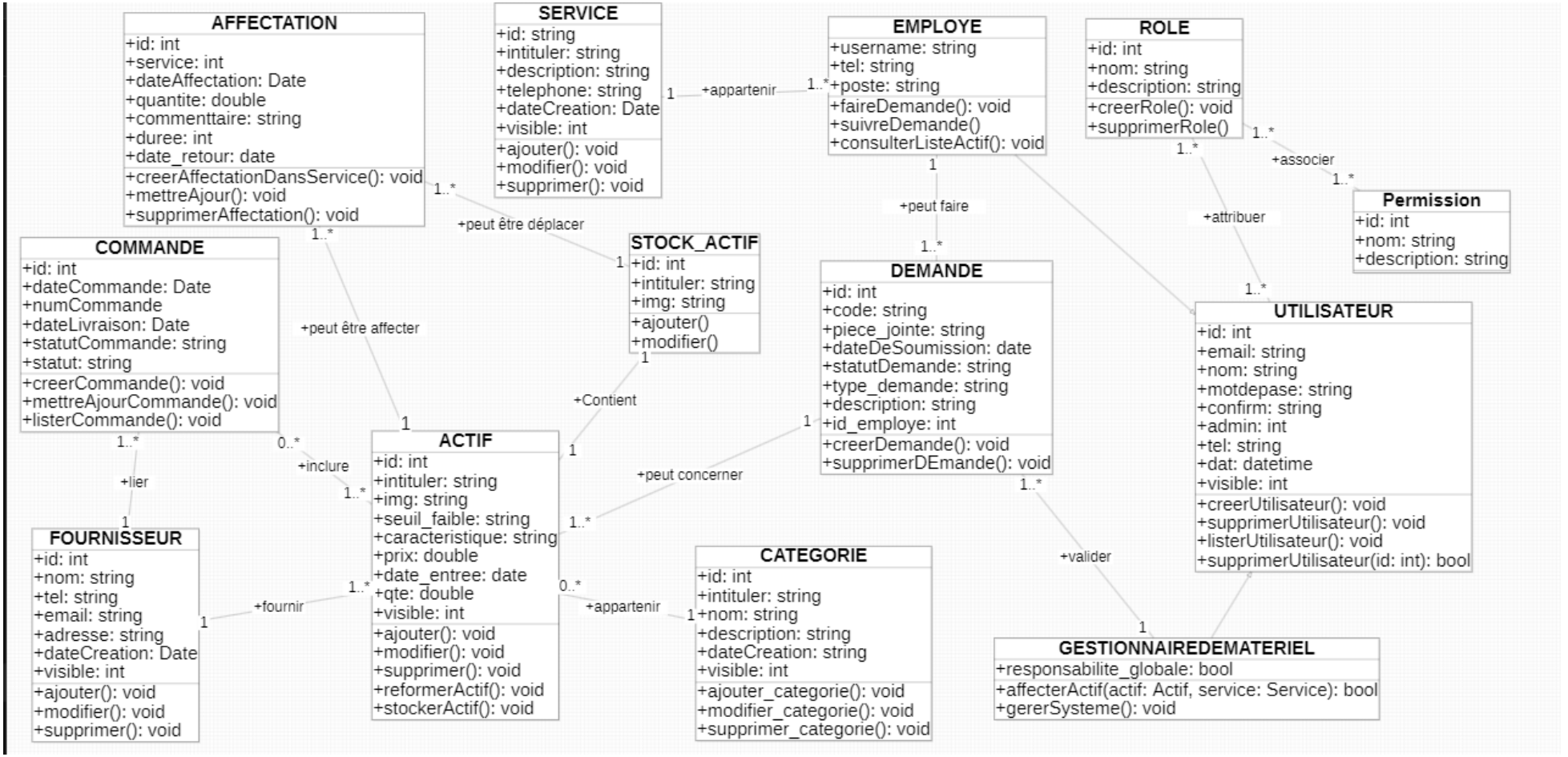


FIGURE 2.2 : Diagramme des classes

2.2.5 Diagramme de séquence

Le diagramme de séquence illustre les interactions entre les différents éléments du système et les acteurs externes. Il offre une représentation graphique détaillée des échanges qui ont lieu entre les acteurs et le système, en mettant en évidence l'ordre des messages échangés et les processus correspondants.

2.2.5.1 Diagramme de séquence pour l'authentification

La figure 2.3 illustre le diagramme de séquence décrivant le processus d'authentification sur la plateforme.

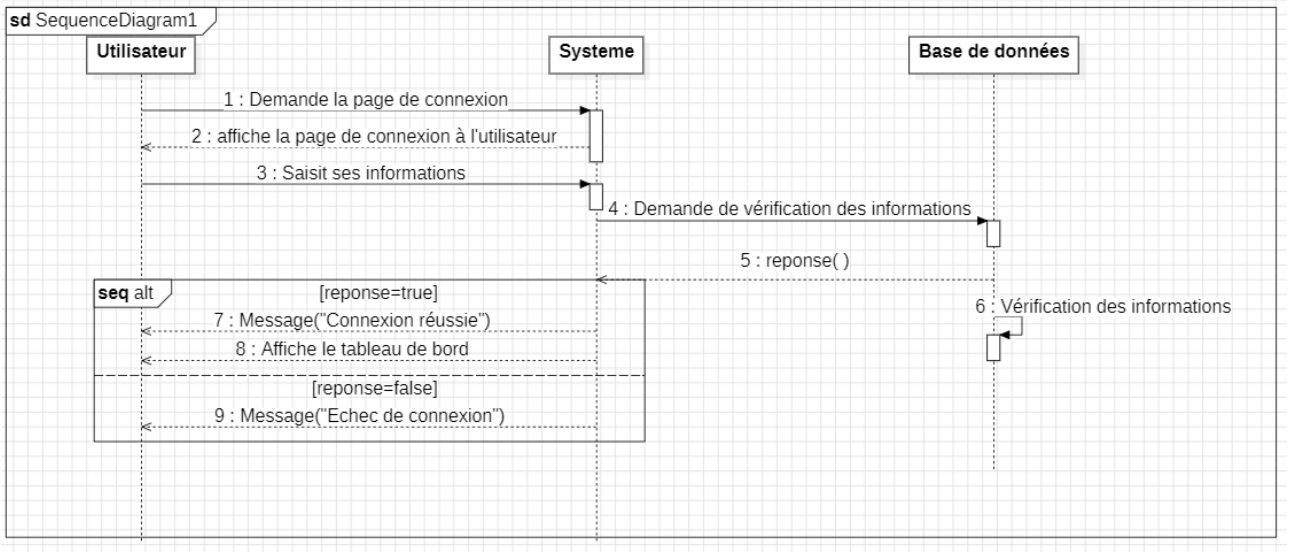


FIGURE 2.3 : Diagramme de séquence d'authentification

2.2.5.2 Diagramme de séquence pour la demande d'actif

A travers ce diagramme de séquence nous allons montrer comment un utilisateur faire sa demande d'actif. La figure 2.4

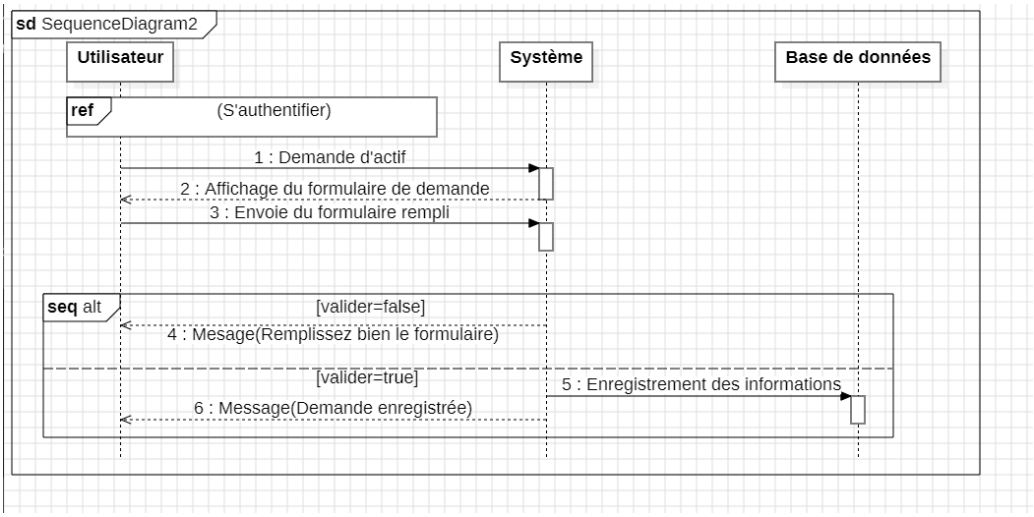


FIGURE 2.4 : Diagramme de séquence de demande d'actif

2.3 Choix techniques

2.3.1 La méthodologie du projet

Pour mener à bien le développement de notre projet, nous avons choisi d'adopter le modèle de cycle de vie en V, qui est l'un des modèles couramment utilisés pour la gestion de projet.

2.3.2 Le modèle de cycle en V

En gestion de projet, le cycle en V se divise en une phase descendante et une phase ascendante. Il est dérivé du modèle en cascade, également appelé « waterfall », qui représente les processus de développement de manière linéaire et en étapes successives. L'un des principaux avantages du cycle en V est qu'il limite les retours en arrière incessants pour redéfinir les spécifications initiales. Cette méthode repose sur des étapes séquentielles, allant de l'analyse des besoins jusqu'aux tests d'acceptation. Dans le cadre de notre projet, nous avons structuré notre travail en deux phases :

1. Phase d'analyse

- **Étude documentaire :** collecte, traitement et stockage des informations ;
- **Spécifications :** définition des caractéristiques, fonctionnalités et services requis ;
- **Conception générale :** élaboration de l'architecture globale et de l'ergonomie du système ;
- **Conception détaillée :** modélisation des différentes composantes du système en UML.

2. Phase de réalisation

- **Codage :** implémentation des fonctionnalités spécifiées ;
- **Tests unitaires :** vérification individuelle de chaque composant du système ;
- **Tests d'intégration :** validation de l'interaction entre les différents modules.

Cette approche est illustrée par la figure 2.5 ci-dessus :

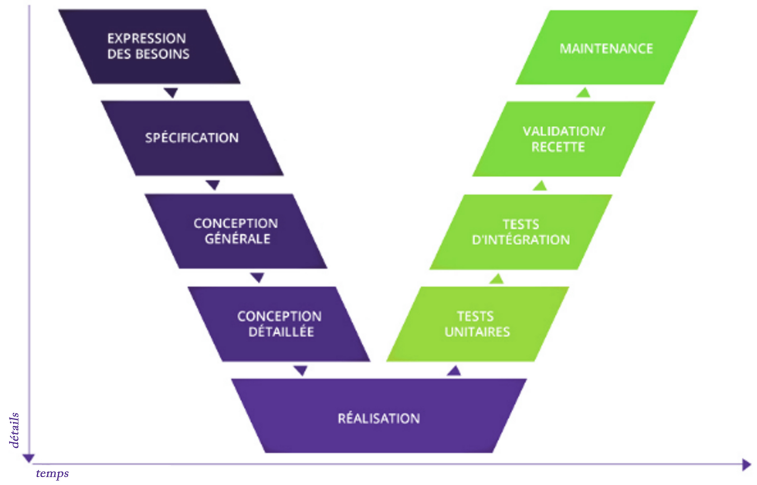


FIGURE 2.5 : Méthode de cycle en v [9]

2.4 Architecture du Système pour la Gestion des Ressources Matérielles

Notre système est une application web accessible à un large nombre d'utilisateurs répartis sur l'ensemble du pays. Les utilisateurs peuvent accéder à la plateforme via un simple navigateur web, depuis n'importe quel poste de travail. L'architecture de l'application repose sur trois niveaux distincts, inspirés du modèle [10]MVC (Modèle-Vue-Contrôleur) pour garantir une séparation claire des responsabilités et une meilleure évolutivité :

1. Le Modèle (Model) : Gère les données et la logique métier (par exemple, les équipements, affectations, incidents). Il est indépendant de la vue et du contrôleur, mais peut envoyer des mises à jour aux autres composants.

2. La Vue (View) : Présente les données à l'utilisateur sous forme d'interface graphique. Elle lit les données du modèle et les affiche, tout en réagissant aux changements envoyés par celui-ci.

3. Le Contrôleur (Controller) : Gère les actions de l'utilisateur (clics, formulaires). Il interagit avec le modèle pour effectuer des modifications et met à jour la vue en conséquence.

Le modèle est indépendant de la vue et du contrôleur, bien qu'il puisse leur envoyer des messages. La vue lit les données du modèle et reçoit des mises à jour de celui-ci. Un modèle peut être utilisé par plusieurs vues. La gestion des entrées est assurée par le contrôleur, tandis que la gestion des sorties relève de la vue.

La vue dépend du modèle, car elle l'interroge pour afficher les données.

Le contrôleur dépend de la vue et du modèle. Il agit comme intermédiaire en demandant les données au modèle, les analysant, puis envoyant le contenu à afficher à la vue.

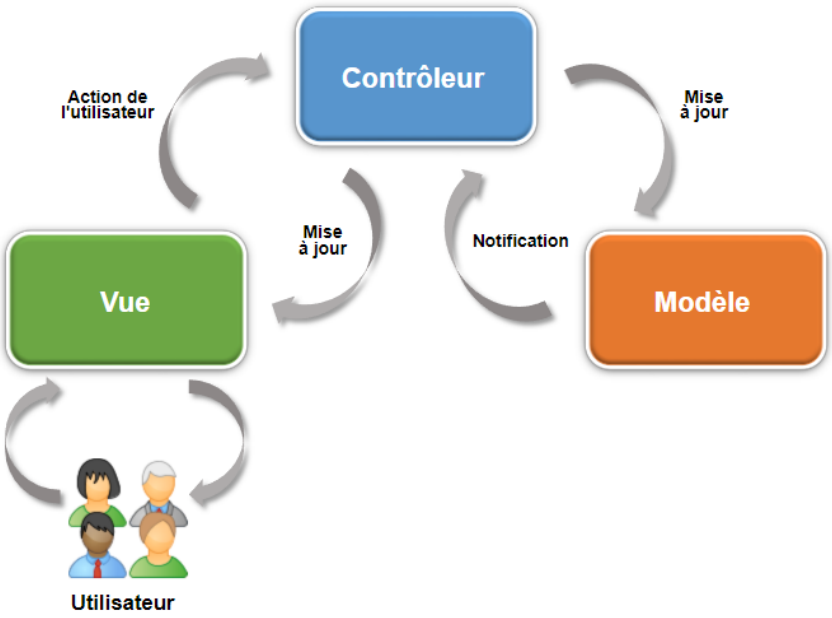


FIGURE 2.6 : Architecture MVC

Cette approche de développement repose sur le modèle MVC (Modèle-Vue-Contrôleur), qui permet de mieux structurer le code. L'un de ses principaux avantages est la clarté de l'architecture, facilitant ainsi la maintenance et les améliorations. Par exemple, la modification des traitements, comme

le passage d'une base de données SQL à XML, n'affecte pas la vue, car celle-ci reste indépendante de la logique métier.

2.5 Technologies utilisées

2.5.1 Frontend

Le front-end représente l'interface utilisateur, c'est-à-dire la partie avec laquelle l'utilisateur interagit directement. Pour le développement de l'application, nous avons intégré diverses technologies.

2.5.1.1 HTML5

[5] HTML est un langage de balisage utilisé pour concevoir des pages web en écrivant de l'hypertexte, d'où son nom. Il permet de structurer de manière sémantique et de mettre en forme le contenu des pages. De plus, il facilite l'inclusion d'éléments variés, tels que des ressources multimédias (images, vidéos), des formulaires de saisie, des titres, des paragraphes, ainsi que des éléments interactifs comme des applets, grâce à l'utilisation de balises

2.5.1.2 CSS3

[4] Le CSS est un langage informatique utilisé pour styliser et mettre en forme des documents web, tels que ceux conçus en HTML ou XML. Grâce à ses nombreuses propriétés liées à l'apparence et au positionnement, il permet d'améliorer l'esthétique des pages web sans avoir à modifier le code HTML sous-jacent.

2.5.1.3 JavaScript

[6] JavaScript est un langage de programmation de scripts largement utilisé pour rendre les pages web interactives. Il repose sur un modèle basé sur des prototypes, où les bases du langage et ses principales interfaces sont fournies sous forme d'objets. La grande majorité des sites web l'intègrent, et la plupart des navigateurs disposent d'un moteur JavaScript dédié pour l'exécuter.

2.5.1.4 Bootstrap

Bootstrap est un framework front-end qui repose sur les langages HTML, CSS et JavaScript, offrant aux développeurs des outils pour concevoir rapidement des sites web. Conçu pour faciliter la création de designs responsives adaptés à tous types d'écrans, notamment les smartphones, il propose des styles prédéfinis pour les typographies, les boutons, et bien d'autres éléments accessibles via des classes. De plus, Bootstrap intègre un système de grille performant permettant de positionner facilement les éléments HTML sur une page web.

2.5.2 Backend

Le backend constitue la partie invisible mais indispensable d'une application, qui permet de gérer et de traiter les actions réalisées par l'utilisateur depuis l'interface.

2.5.2.1 PHP

PHP est un langage de programmation open-source, principalement utilisé pour générer des pages web dynamiques via un serveur HTTP. Il peut également être exécuté localement, comme tout langage interprété. PHP est un langage impératif qui prend également en charge la programmation orientée objet. [14]

2.5.2.2 Système de gestion de base de données (SGBD)

Notre choix de SGBD s'est porté sur MySQL, et pour cela, nous avons opté pour l'utilisation locale du serveur de développement WAMP. WAMP est un ensemble de logiciels permettant de configurer un serveur web local pour le développement d'applications. Accessible à un large public, il ne nécessite pas de connaissances techniques approfondies et est compatible avec les systèmes d'exploitation Windows. [17]WAMP regroupe le serveur web Apache, la base de données relationnelle MySQL, ainsi que les langages scripts PHP et Perl. Il est couramment utilisé pour configurer un environnement de test local pour des sites ou des applications web.

2.6 Outils de développement

2.6.1 Visual Studio Code

[16]Visual Studio Code (VSC) est un éditeur de code open-source, gratuit et multi-plateforme développé par Microsoft. Construit avec Electron et basé sur Monaco Editor, il est conçu principalement pour JavaScript, TypeScript et Node.js, mais s'adapte à d'autres langages via un large écosystème d'extensions.

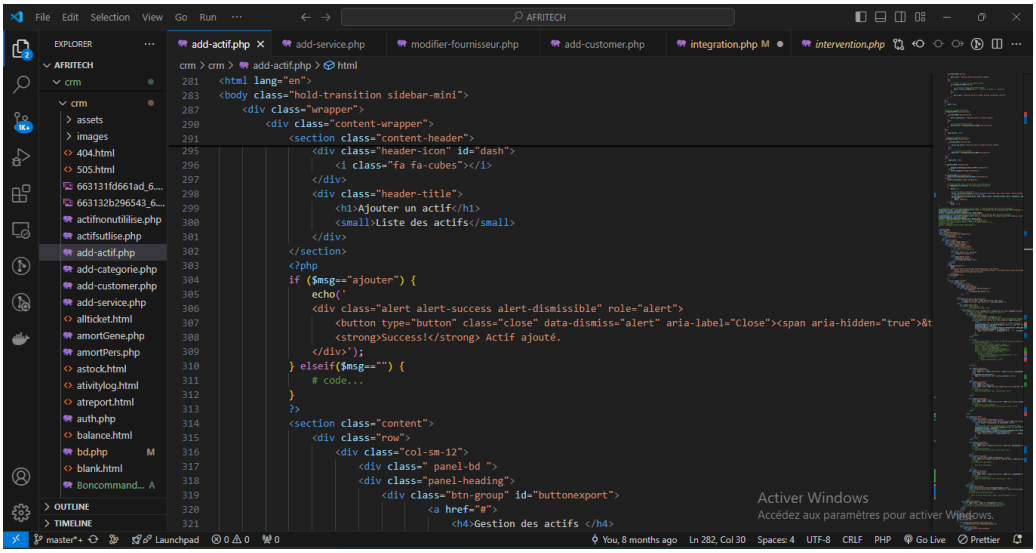


FIGURE 2.7 : VS Code

2.6.2 StarUML

Pour la réalisation de nos diagrammes UML, nous avons opté pour l'utilisation de l'outil StarUML. StarUML est un outil puissant permettant de créer des diagrammes UML et de générer des codes dans des langages tels que Java, PHP, C, entre autres, à partir des diagrammes de classe UML. Il permet également de produire des requêtes SQL pour la création de bases de données en fonction du SGBD choisi. Ce type d'outil est particulièrement utile pour la conception et le développement d'applications et de plateformes robustes. Dans le cadre de notre étude, nous avons utilisé StarUML pour ses fonctionnalités complètes et son interface intuitive.



FIGURE 2.8 : StarUML

Voici un tableau récapitulatif des outils et langages utilisés dans notre projet.

Outils, langages et framework
Windows 11 Professionel
UML HTML CSS PHP JavaScript Bootstrap
MySQL (SGBD)
Visual Studio Code
StarUML

TABLE 2.2 : Résumé des outils, langages et frameworks employés.

Conclusion

Ce chapitre a permis de présenter en détail l'analyse de la problématique, la modélisation de la solution proposée, ainsi que les choix techniques retenus pour le développement de l'application. Dans le dernier chapitre, nous présenterons les résultats obtenus suite à la mise en œuvre de cette solution.

Chapitre 3

Résultats et discussion

Introduction

Dans ce chapitre, nous présentons les résultats obtenus suite à l'implémentation de la solution proposée. Nous analysons ensuite ces résultats, en les comparant aux attentes initiales et aux objectifs du projet, afin d'évaluer l'efficacité de l'application et d'identifier les points d'amélioration éventuels.

3.1 Présentation du protoype de l'application

3.1.1 Page de connexion

La page de connexion permet aux utilisateurs d'accéder à la plateforme en toute sécurité via leur adresse e-mail et leur mot de passe. Elle offre également la possibilité de s'inscrire s'ils n'ont pas encore de compte.

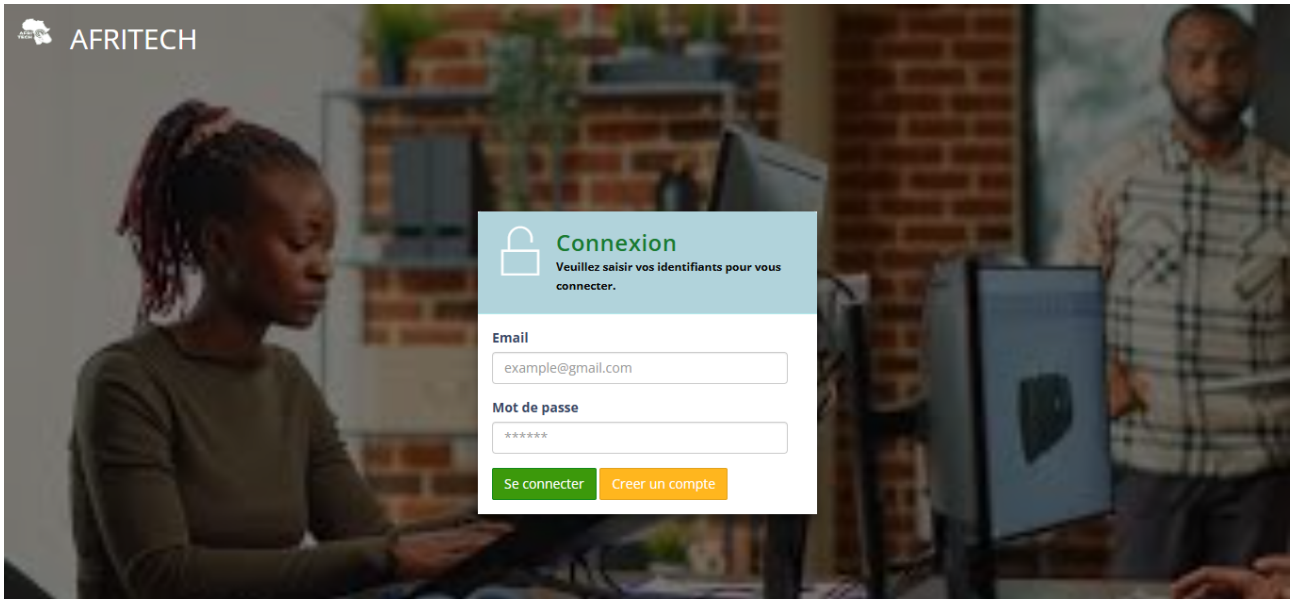


FIGURE 3.1 : Page de connexion

3.1.2 Page d'accueil

La page d'accueil joue un rôle central dans l'expérience utilisateur en offrant un accès rapide aux principales fonctionnalités de la plateforme. Elle présente un aperçu clair et structuré des ressources disponibles, facilitant ainsi la prise de décision et l'organisation des tâches. Grâce à une navigation intuitive, les utilisateurs peuvent accéder directement aux sections essentielles, améliorant ainsi l'efficacité et la fluidité de l'utilisation.

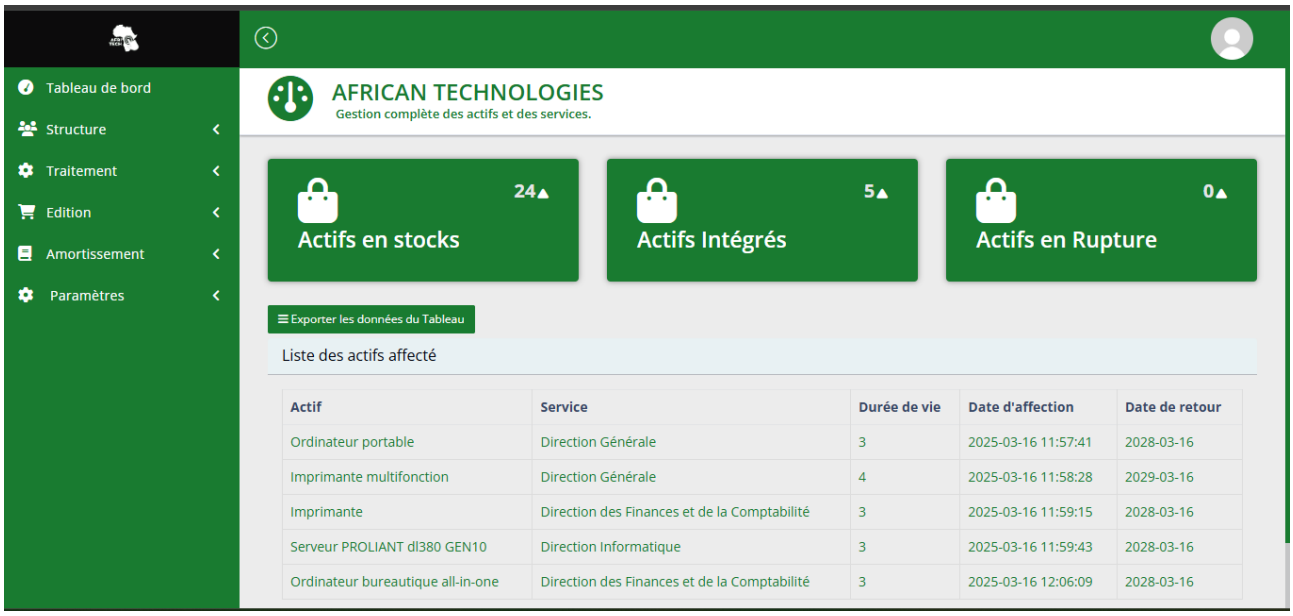


FIGURE 3.2 : Page d'Accueil

3.1.3 Ajout d'actif

Pour ajouter un nouvel actif, il faut renseigner des informations essentielles afin d'assurer son identification, son suivi et une gestion optimisée.

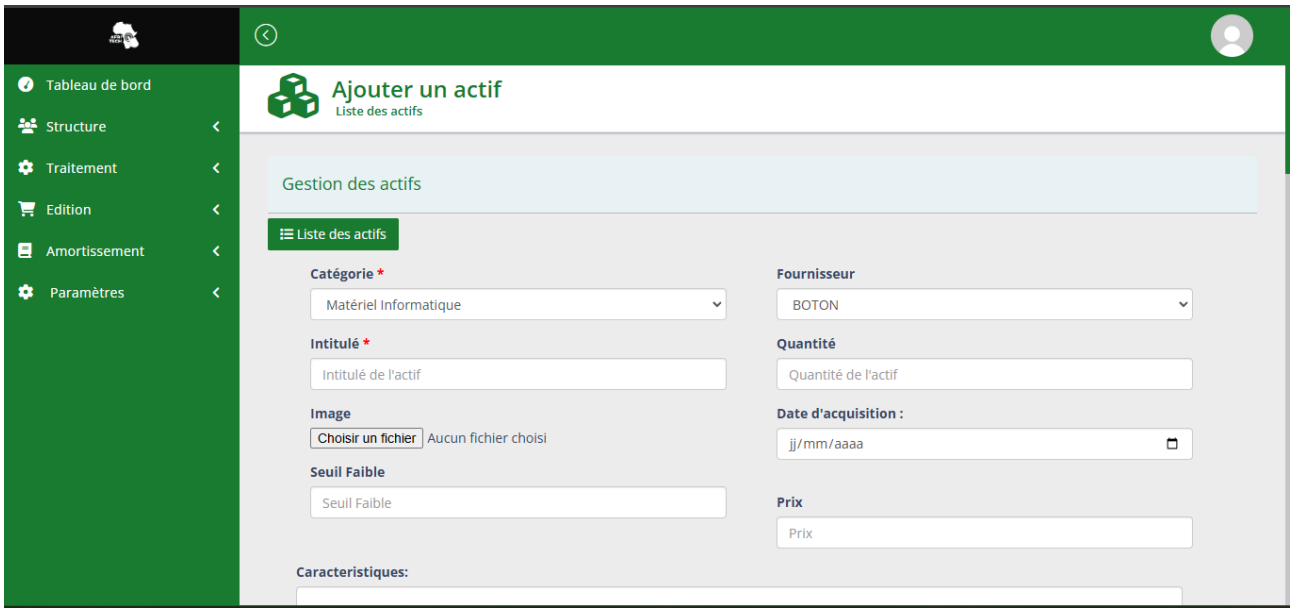


FIGURE 3.3 : Page d'ajout d'actif

3.1.4 Consultation des Affectations

Cette page permet de visualiser l'ensemble des affectations des actifs, assurant ainsi une traçabilité optimale des équipements. Vous pouvez consulter les détails de chaque affectation, notamment le service concerné, l'actif attribué, la date d'affectation et la durée prévue.

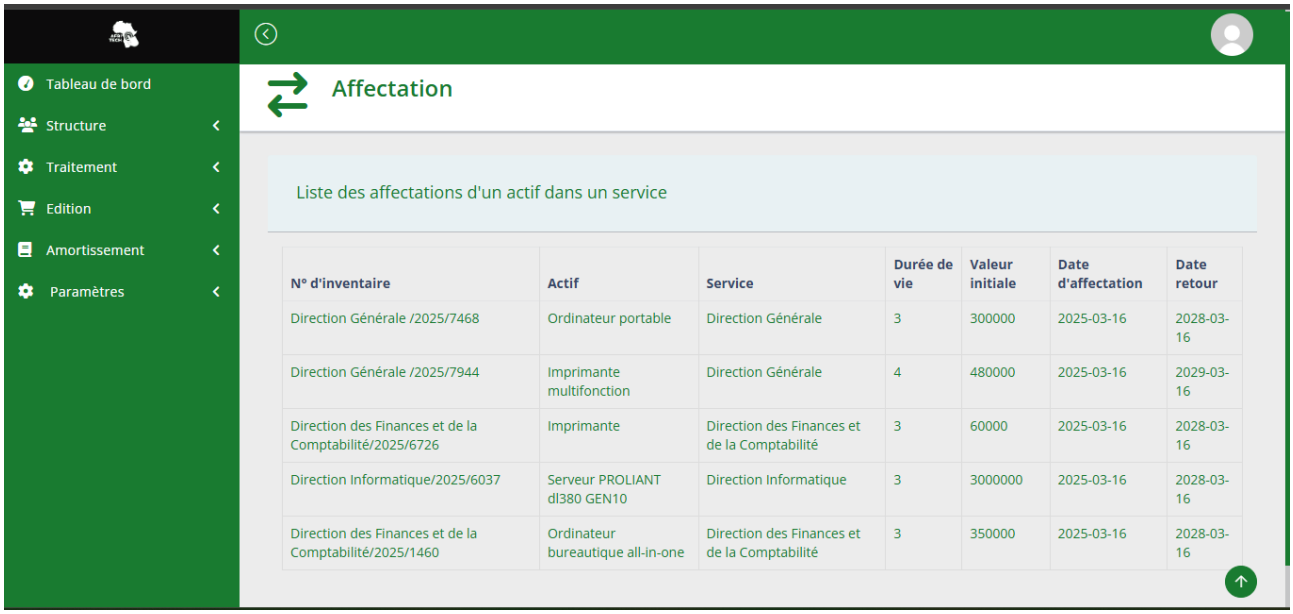


FIGURE 3.4 : Liste des affectations

3.1.5 Page d'enregistrement d'une demande

Cette page permet aux employés de soumettre une demande d'affectation d'actif. Pour garantir un traitement efficace, il est essentiel de renseigner avec précision les informations requises, telles que le type d'actif demandé, l'employé concerné et la justification de la demande.

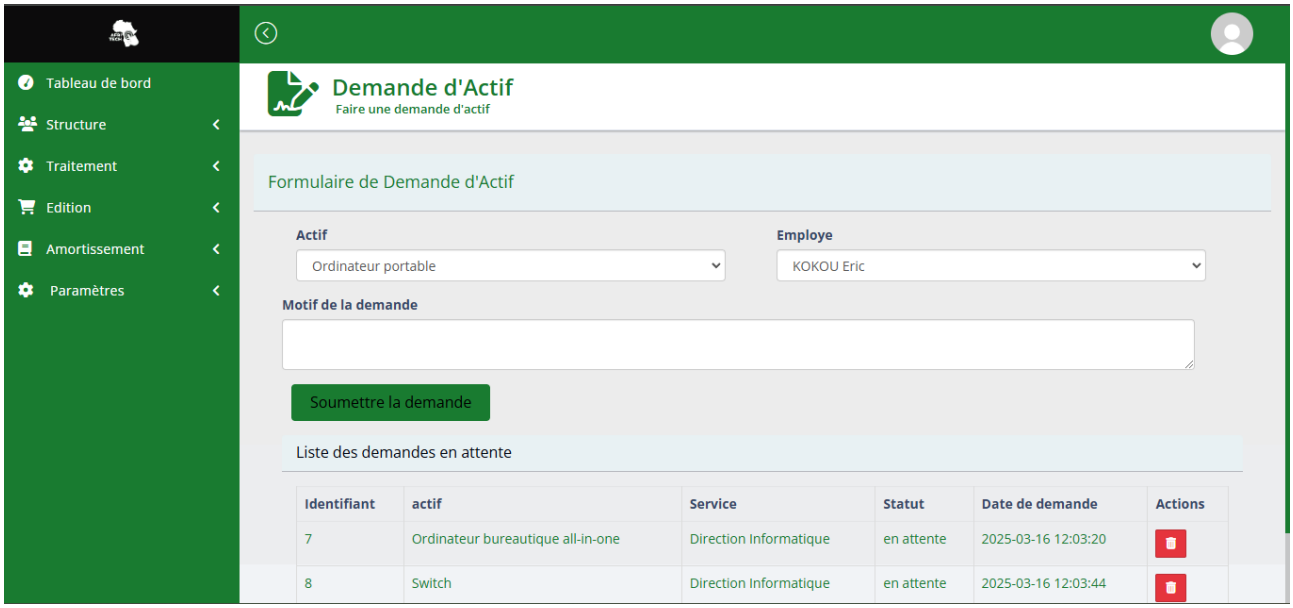


FIGURE 3.5 : Enregistrement de demande

3.1.6 Liste des demandes

Cette page permet de valider ou d'annuler les demandes d'affectation d'actifs. Vous pouvez consulter les détails de chaque demande et effectuer les actions nécessaires pour valider ou annuler la demande, en fonction de son statut et des besoins du service.



FIGURE 3.6 : Liste des demandes

3.1.7 Ajout de fournisseur

Pour enregistrer un nouveau fournisseur dans le système, il est nécessaire de renseigner certaines informations essentielles. Cela permet d'identifier le fournisseur, de centraliser ses coordonnées et d'assurer une gestion efficace des commandes.

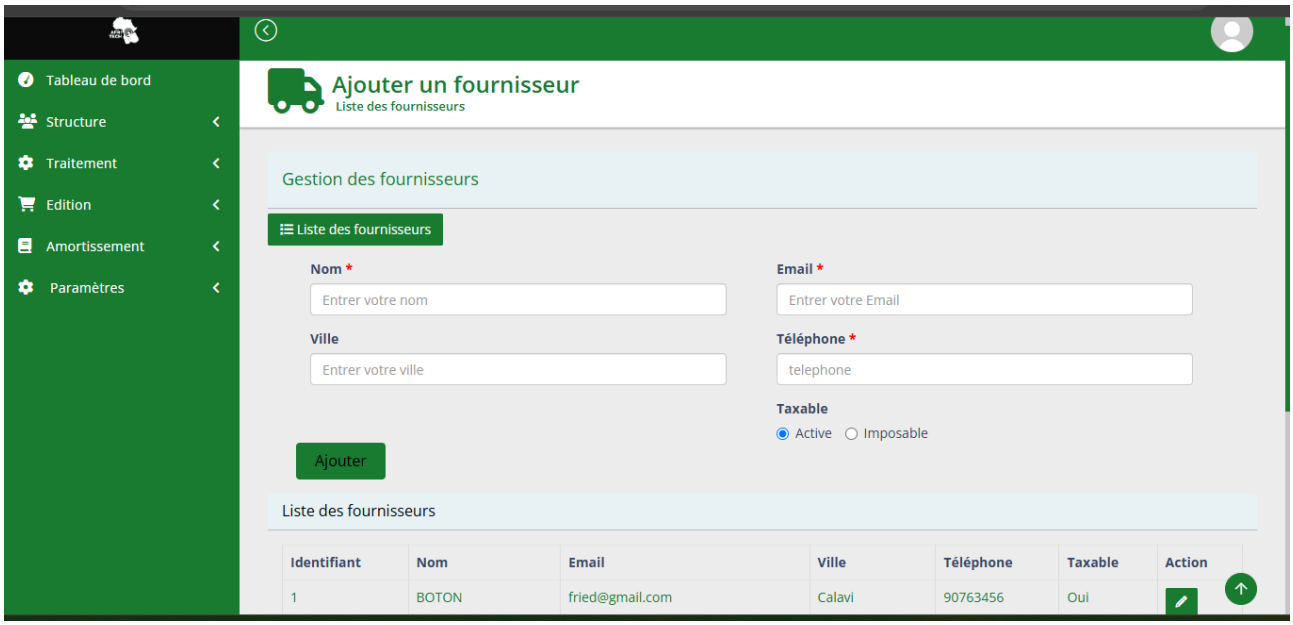


FIGURE 3.7 : Ajouter un fournisseur

3.1.8 Gestion du Stock

Cette page permet de consulter la liste des actifs et d'ajouter des quantités aux actifs existants afin de maintenir un suivi précis du stock.

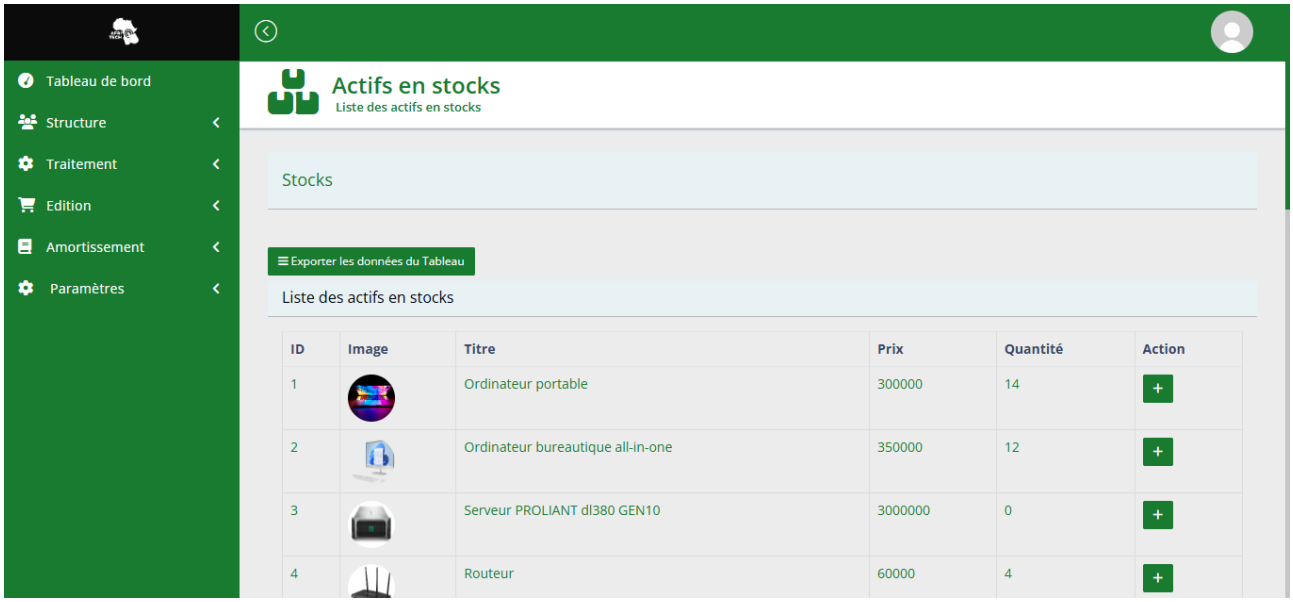


FIGURE 3.8 : Les actifs en stock

3.1.9 Ajout de catégorie d'actif

Pour ajouter une nouvelle catégorie d'actifs, il est nécessaire de renseigner les informations suivantes :

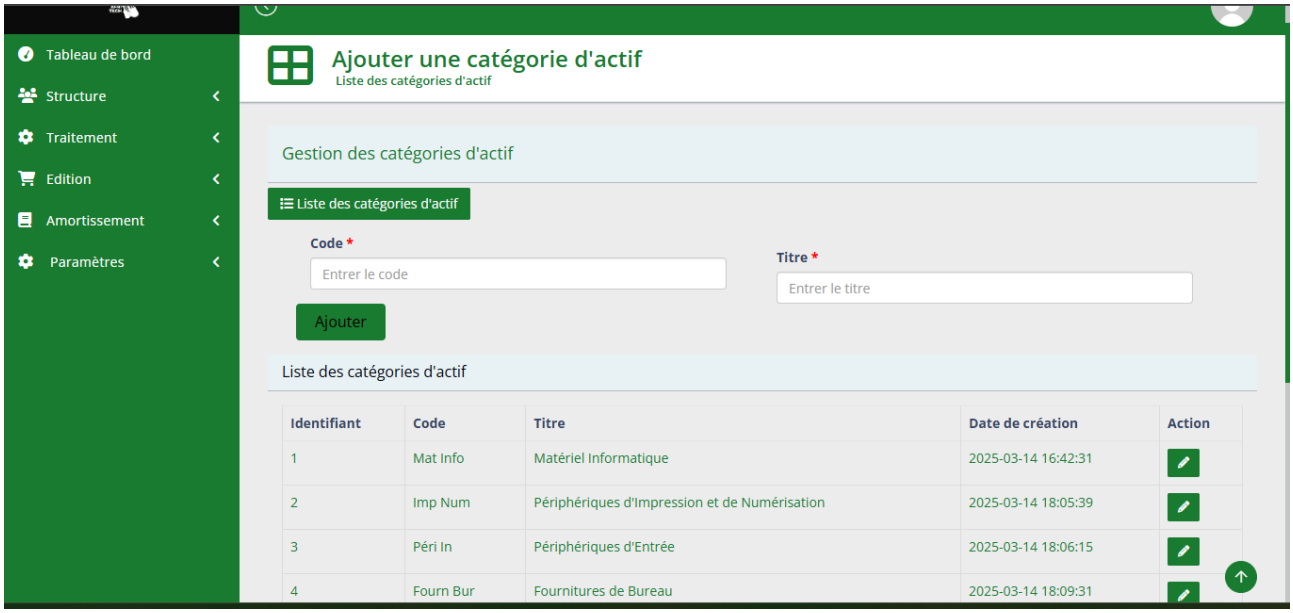


FIGURE 3.9 : Ajout de catégorie d'actif

3.1.10 Liste des Actifs

Cette page permet de consulter la liste des actifs et de modifier leurs informations si nécessaire.

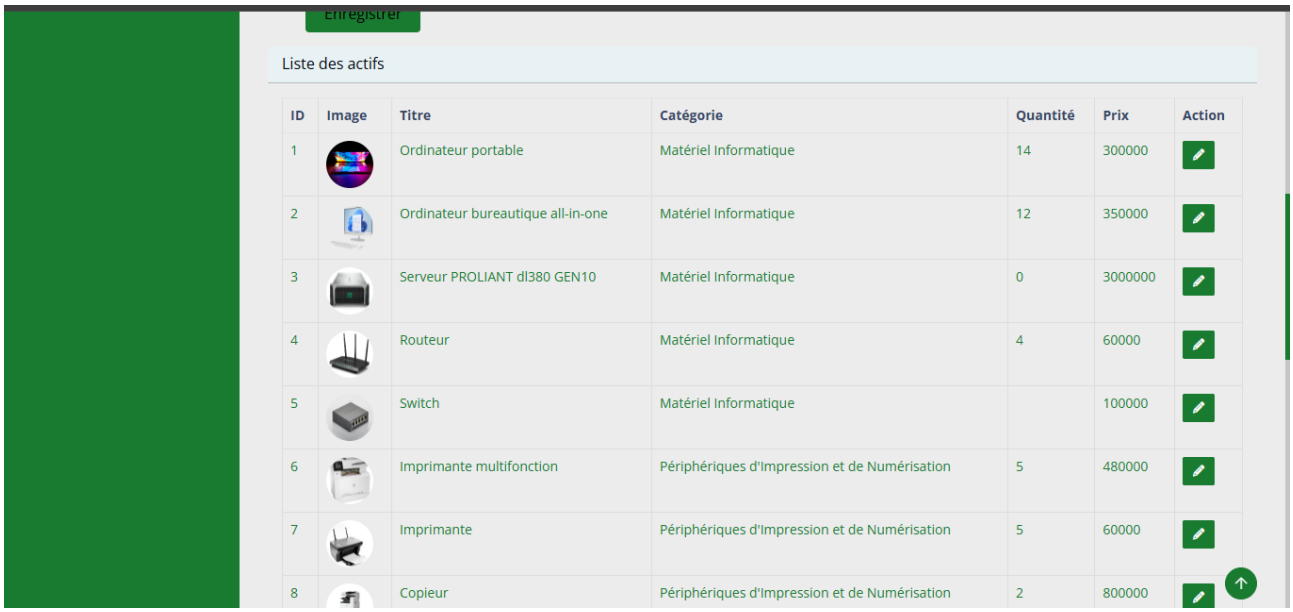


FIGURE 3.10 : Liste des actifs

3.1.11 Gestion de l'Amortissement

Cette page permet de suivre l'amortissement des actifs en fonction de leur durée de vie et de leur valeur d'usage. Vous pouvez consulter les détails de l'amortissement pour chaque actif afin d'assurer une gestion financière optimale.



Num inventaire	Actif	Service	Durée de vie	Date d'affectation	Date de retour	Jour restant	Valeur initiale	Amortissement pour l'année	Valeur résiduelle
Direction Générale /2025/7468	Ordinateur portable	Direction Générale	3 ans	2025-03-16	2028-03-16	1096 jours	300000	100000	300000
Direction Générale /2025/7944	Imprimante multifonction	Direction Générale	4 ans	2025-03-16	2029-03-16	1461 jours	480000	120000	480000
Direction des Finances et de la Comptabilité/2025/6726	Imprimante	Direction des Finances et de la Comptabilité	3 ans	2025-03-16	2028-03-16	1096 jours	60000	20000	60000
Direction Informatique/2025/6037	Serveur PROLIANT dl380 GEN10	Direction Informatique	3 ans	2025-03-16	2028-03-16	1096 jours	3000000	1000000	3000000
Direction des Finances et de la Comptabilité/2025/1460	Ordinateur bureautique all-in-one	Direction des Finances et de la Comptabilité	3 ans	2025-03-16	2028-03-16	1096 jours	350000	116667	350000

FIGURE 3.11 : Amortissement

3.1.12 Ajout d'un Service

L'ajout d'un nouveau service dans l'organisation permet d'assurer une gestion précise des ressources et des affectations d'actifs. Chaque service est associé à des responsabilités spécifiques, et l'enregistrement dans le système garantit une meilleure organisation et une traçabilité des ressources affectées à chaque service. Pour ajouter un service, voici les informations nécessaires :

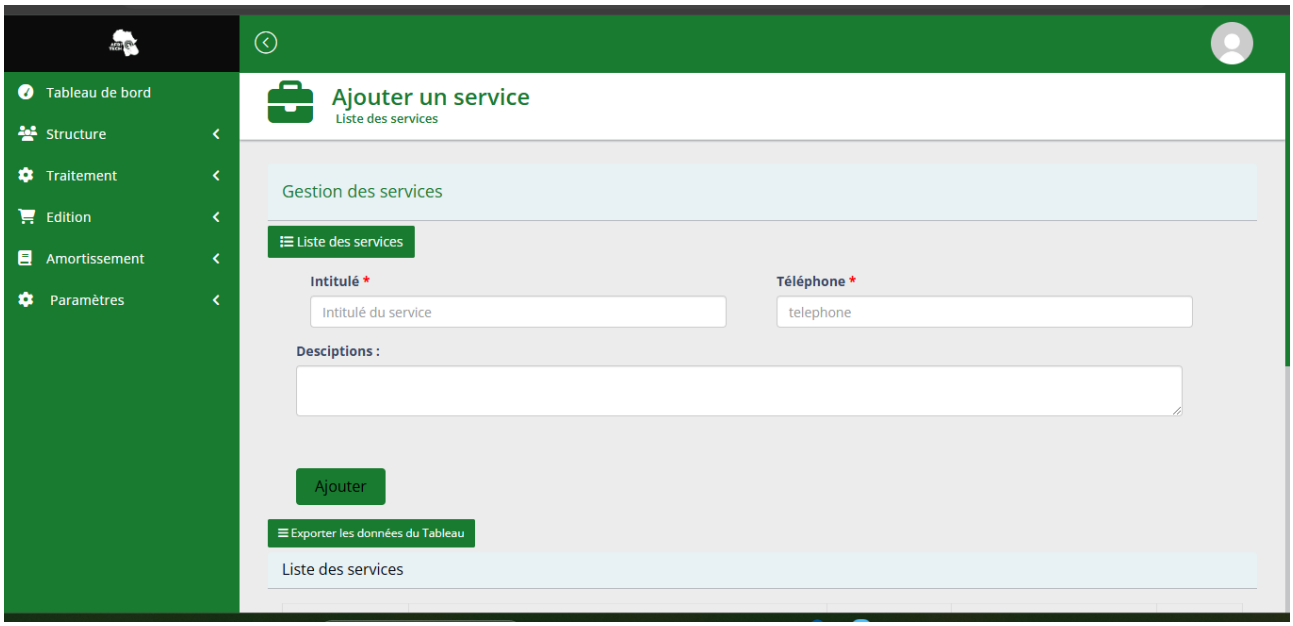


FIGURE 3.12 : Ajout d'un service

3.1.13 Liste des services

Cette section affiche l'ensemble des services enregistrés dans le système, permettant ainsi de suivre l'organisation et l'affectation des ressources au sein de l'entreprise. Chaque service est associé à des informations clés qui facilitent la gestion et la coordination entre les différentes entités de l'entreprise.



FIGURE 3.13 : Liste des services

3.1.14 Gestion des Commandes

Cette page permet d'enregistrer les commandes d'actifs, de suivre leur traitement, et de finaliser la commande en générant et imprimant la facture correspondante.

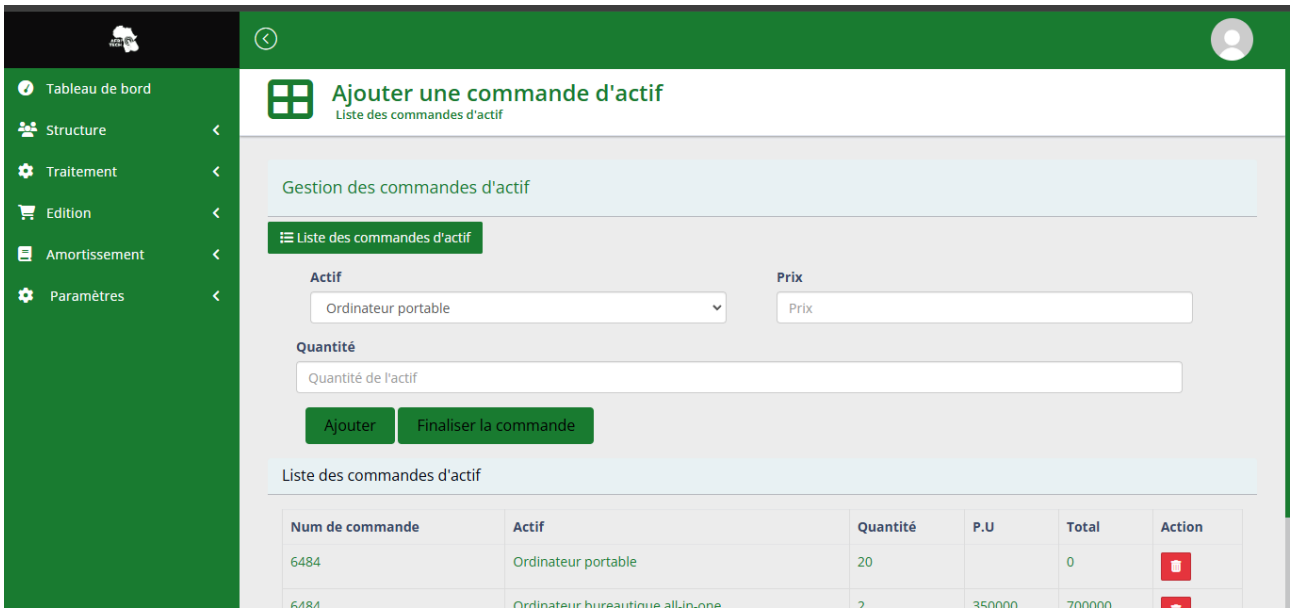


FIGURE 3.14 : Commande

3.1.15 Demande d'actif

Cette section permet aux employés de soumettre des demandes d'actifs en fonction de leurs besoins professionnels. Elle offre un suivi détaillé des demandes, depuis leur soumission jusqu'à leur validation par les responsables concernés. Chaque demande est associée aux informations essentielles pour assurer une gestion efficace et une allocation optimale des ressources au sein de l'entreprise.



FIGURE 3.15 : Demande d'actif

3.2 Sécurité

Notre système repose sur plusieurs dispositifs de sécurité, qui incluent :

3.2.1 Authentification

L'authentification est une étape qui permet à l'utilisateur de prouver son identité en fournissant son adresse e-mail et son mot de passe. Elle garantit un accès sécurisé à la plateforme.

3.2.2 Utilisation d'une fonction de hachage

Le hachage des mots de passe est une pratique de sécurité essentielle qui doit être systématiquement mise en œuvre. Sans cette mesure, chaque mot de passe stocké peut être compromis si le support de stockage, généralement une base de données, est exposé. Cela permettrait à un attaquant d'accéder non seulement à votre application, mais aussi à d'autres services si l'utilisateur réutilise le même mot de passe.

En hachant un mot de passe avant de le stocker, vous compliquez considérablement la tâche d'un attaquant cherchant à retrouver le mot de passe original. De plus, cette méthode permet de comparer un mot de passe haché avec une chaîne reçue lors de l'authentification.

Dans notre travail, nous avons utilisé la fonction password() qui génère un hachage sécurisé et irréversible en utilisant un algorithme de hachage robuste.

3.3 Perspectives

Nous envisageons d'améliorer notre système en y intégrant de nouvelles fonctionnalités telles que :

- **Optimisation du suivi des actifs :** Améliorer la précision du suivi des équipements en intégrant des technologies comme les QR codes pour une gestion en temps réel.
- **Automatisation des notifications :**Mettre en place un système de rappels automatiques pour les retours de matériel, les maintenances et les incidents (vol, panne, etc.).
- **Intégration avec d'autres systèmes :** Permettre l'interopérabilité avec des logiciels de comptabilité ou de gestion des ressources humaines pour une meilleure centralisation des informations.
- **Déploiement à grande échelle :**Étendre la solution à d'autres entreprises ou institutions ayant des besoins similaires en gestion des ressources matérielles.

Conclusion

Ce chapitre a permis d'analyser les résultats obtenus, mettant en lumière la performance de la solution développée. Les conclusions tirées servent à affiner notre approche et à proposer des recommandations pour optimiser la gestion des ressources matérielles chez Afritech.

Conclusion Générale

En conclusion, la mise en place de cette application Web pour la gestion des ressources matérielles d'Afritech marque une avancée majeure vers une gestion plus efficace et centralisée des actifs. Ce projet a permis de répondre aux objectifs initiaux en proposant un système structuré qui optimise le suivi des équipements, améliore la gestion des fournisseurs et automatise les processus d'allocation et de maintenance. Tout au long du développement, une attention particulière a été portée à la traçabilité des actifs et à la fiabilité des données, garantissant ainsi une meilleure transparence et une prise de décision plus éclairée. Malgré les progrès réalisés, certains défis subsistent, notamment en matière d'intégration avec d'autres systèmes et d'optimisation des performances. Toutefois, ces limites constituent aussi des pistes d'amélioration pour les évolutions futures du projet. Dans cette perspective, des améliorations sont envisagées, telles que l'ajout de nouvelles fonctionnalités pour affiner la gestion des mouvements d'actifs, renforcer la sécurité et automatiser davantage les opérations. Ces évolutions permettront d'accroître l'efficacité du système et de répondre aux besoins croissants d'Afritech.

En somme, ce projet illustre l'importance de la transformation numérique dans l'optimisation des processus de gestion en entreprise. Il constitue un premier pas vers une modernisation continue qui renforcera la compétitivité et la capacité d'innovation d'Afritech dans un environnement en constante évolution.

Bibliographie

[8] V. LOZANO. In : *Tout ce que vous avez toujours voulu savoir sur latex sans jamais oser le demander.* (). Consulté le 20 octobre 2024.

Webographie

[1] *Application web*. Consulté le 12 novembre 2024. URL : https://fr.m.wikipedia.org/wiki/Application_web.

[2] *asset*. Consulté, le 5 Décembre 2024. URL : https://www.assetpanda.com/get-a-demo/.

[3] *CMDBuild*. Consulté, le 10 Mars 2024. URL : https://www.cmdbuild.org/en.

[4] *css*. Consulté le 27 octobre 2024. URL : https://developer.mozilla.org/fr/docs/Web/CSS.

[5] *html*. Consulté, le 6 Décembre 2024. URL : https://fr.wikipedia.org/wiki/Hypertext_Markup_Language.

[6] *javaScript*. Consulté le 10 septembre 2024. URL : https://developer.mozilla.org/fr/docs/Web/JavaScript.

[7] *Les différents types d'applications web*. Consulté le 12 novembre 2024. URL : https://blog.osmova.com/types-dapplications-web/.

[9] *Modèle en V*. Consulté le 5 Janvier 2024. 2024. URL : https://www.planilog.com/support/cycle-en-v/.

[10] *MVC*. Consulté le 20 novembre 2024. URL : https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur.

[11] *Netsuite*. Consulté, le 10 Mars 2024. URL : https://bistasolutions.net/netsuite-finance-2/?gad_source=1&gclid=Cj0KCQjwm7q-BhDRARIsACD6-fX5c68Ekb4mXxIA_k8HZ7Pyo4pLPOSHjFKUjDbQs-uzr4isSlurOscaAtxKEALw_wcB.

[12] *Odoo*. Consulté, le 10 Décembre 2024. URL : https://www.odoo.com/.

[13] *OpenMaint*. Consulté, le 10 Mars 2024. URL : https://www.openmaint.org/en.

[14] *php*. Consulté le 06 Novbembre 2024. URL : https://fr.wikipedia.org/wiki/PHP.

[15] *SnipeIT*. Consulté, le 11 Décembre 2024. URL : https://snipeitapp.com/.

[16] *Visual Studio Code*. Consulté le 20 septembre 2024. URL : https://fr.wikipedia.org/wiki/Visual_Studio_Code.

[17] *WampServer*. Consulté le 28 septembre 2024. URL : https://fr.wikipedia.org/wiki/Visual_Studio_Code.

[18] *Zoho*. Consulté le 11 Décembre 2024. URL : https://www.zoho.com/.

Table des matières

Dédicace ii

Remerciements iii

Résumé iv

. iv

Abstract v

. v

Liste des figures vi

Liste des tableaux vii

Liste des acronymes viii

Introduction 1

1 Revue de littérature 3

Introduction 3

1.1 Présentation de la structure d'accueil : AFRICAN TECHNOLOGIES 3

1.1.1 Domaines d'interventions 3

1.1.2 Organigramme 4

1.2 Clarification conceptuelle 6

1.2.1 Application web 6

1.3 Solutions existantes 7

1.3.1 Odoo 7

1.3.2 Zoho Inventory 7

1.3.3 Asset Panda 7

1.3.4 NetSuite 7

1.3.5 CMDBuild 7

1.3.6 OpenMAINT 7

1.3.7 Snipe-IT 8

Conclusion 10

2 Analyse, Modélisation, Choix techniques 11

Introduction 11

2.1 Analyse des besoins 11

2.2 Modélisation 11

2.2.1 Identification des acteurs . . . 11
2.2.2 Rôle des acteurs . . . 12
2.2.3 Diagramme des cas d'utilisation . . . 12
2.2.4 Diagramme des classes . . . 12
2.2.5 Diagramme de séquence . . . 15
2.2.5.1 Diagramme de séquence pour l'authentification . . . 15
2.2.5.2 Diagramme de séquence pour la demande d'actif . . . 15
2.3 Choix techniques . . . 16
2.3.1 La méthodologie du projet . . . 16
2.3.2 Le modèle de cycle en V . . . 16
2.4 Architecture du Système pour la Gestion des Ressources Matérielles . . . 17
2.5 Technologies utilisées . . . 18
2.5.1 Frontend . . . 18
2.5.1.1 HTML5 . . . 18
2.5.1.2 CSS3 . . . 18
2.5.1.3 JavaScript . . . 18
2.5.1.4 Bootstrap . . . 18
2.5.2 Backend . . . 19
2.5.2.1 PHP . . . 19
2.5.2.2 Système de gestion de base de données (SGBD) . . . 19
2.6 Outils de développement . . . 19
2.6.1 Visual Studio Code . . . 19
2.6.2 StarUML . . . 20
Conclusion . . . 20

3 Résultats et discussion **22**
Introduction . . . 22
3.1 Présentation du protoype de l'application . . . 22
3.1.1 Page de connexion . . . 22
3.1.2 Page d'accueil . . . 23
3.1.3 Ajout d'actif . . . 23
3.1.4 Consultation des Affectations . . . 24
3.1.5 Page d'enregistrement d'une demande . . . 24
3.1.6 Liste des demandes . . . 25
3.1.7 Ajout de fournisseur . . . 25
3.1.8 Gestion du Stock . . . 26
3.1.9 Ajout de catégorie d'actif . . . 26
3.1.10 Liste des Actifs . . . 26
3.1.11 Gestion de l'Amortissement . . . 27
3.1.12 Ajout d'un Service . . . 27
3.1.13 Liste des services . . . 28
3.1.14 Gestion des Commandes . . . 28
3.1.15 Demande d'actif . . . 29
3.2 Sécurité . . . 29
3.2.1 Authentification . . . 30

3.2.2 Utilisation d'une fonction de hachage 30
3.3 Perspectives 30
Conclusion 30

Conclusion **31**

Bibliographie **32**

Webographie **33**

Table des matières **34**