Table des matières

INTRODUCTION GENERALE 1

CHAPITRE I : CONTEXTE GENERAL DU PROJET MOBIPOS 4

I. INTRODUCTION 5
II. PRESENTATION DE L'ORGANISME D'ACCUEIL 5
II.1 Présentation "Eurequat Algérie" 5
II.2 Domaine d'expertise 6
II.3 Les projets réalisés 6
III. SYSTEME MOBIPOS 7
III.1 Description des systèmes TPV 7
III.2 Description du système MOBIPOS 8
III.3 Comparatif entre les systèmes équivalents à MOBIPOS 11
IV. CYCLE DE VIE ADOPTE 13
V. CONCLUSION 14

CHAPITRE II : ANALYSE ET CONCEPTION DU MODULE BACK OFFICE MOBIPOS 15

I. INTRODUCTION 16
II. CAHIER DE CHARGE DU PROJET 16
II.1. Contraintes techniques 16
II.2. Contraintes fonctionnelles 16
II.3. Contraintes de sécurités 18
III. LA PHASE D'ANALYSE 18
III.1. Le langage UML 18
III.2. Entreprise Architect 19
III.3. PowerMockup 19
IV. ANALYSE GLOBALE DE L'APPLICATION MOBIPOS 19
IV.1. Description des acteurs 19
IV.2. Identifications des messages 20
IV.3. Spécification des besoins fonctionnels 20
IV.4. Spécification des besoins non fonctionnels 21
V. CONCEPTION 22
V.1. Diagramme de cas d'utilisations 22
V.2. Diagramme de séquence 25
V.3. Diagramme de classe 33
VI. CONCEPTION DES MAQUETTES 35
VII. CONCLUSION 36

CHAPITRE III : REALISATION ET TESTS DU MODULE BACK OFFICE MOBIPOS 37

I. INTRODUCTION 38
II. METHODES DE GESTION DE PROJET : SCRUM ET DIAGRAMME DE GANTT 38
II.1. Méthodes Scrum 38
II.2. Digramme Gantt 40
III. OUTILS DE DEVELOPPEMENT 41
III.1. Programmation 41
III.2. Gestion de version « Git » 44
III.3. Google docs 44
IV. L'ARCHITECTURE DE L'APPLICATION 44
V. GESTION DES TESTS ET ANOMALIES 45
V.1. Les tests 45
V.2. Les anomalies 46
VI. LES DIFFICULTES RENCONTREES 48
VII. LES INTERFACES GRAPHIQUES 49

VII.1. Authentification 49
VII.2. Page d'accueil 50
VII.3. Menu Administration 50
VII.4. Menu vente 56
VII.5. Menu stock 58
VII.6. Menu rapport 59
VII.7. Menu configuration 63
VIII. CONCLUSION 65

CONCLUSION GENERAL 66

REFERENCES BIBLIOGRAPHIQUES 68

ANNEXES A : 70

LISTE DE FIGURES 83

LISTE DES TABLEAUX 84

LISTE DES ABREVIATIONS 85

INTRODUCTION GENERALE

Contexte du projet

Dans le cadre de la formation de Génie Logiciels (GL) à l'université Abou Bekr Belkaid, nous avons eu l'occasion d'effectuer notre projet de fin d'études pour l'obtention du diplôme de Master en Informatique au sein de l'entreprise « Eurequat-Algérie [1]».

Ce stage s'est déroulé du 1 février 2017 au 30 mai, à Tlemcen sur le thème : «Réalisation d'un système MOBIPOS : un module back office d'un système terminal point de vente».

Généralement ce choix de stage vise à compléter notre formation universitaire acquise, durant cinq ans, au sein de notre université, et de nous introduire dans la vie professionnelle grâce à une mise en pratique de nos connaissances, à l'utilisation des compétences acquises et à mettre en épreuve notre esprit d'équipe.

La société Eurequat Algérie a démarré son activité en mars 2006, avec le soutien et les moyens techniques de la société Eurequat Technologie Europe.

C'est au travers de son offre globale en matière de traçabilité, d'acquisition de données et de gestion des points de vente, sur le marché de l'identification par code à barres & RFID « Radio Frequency Identification », qu'Eurequat s'est imposé comme un précurseur en Algérie.

Eurequat s'appuie sur une expérience de plus de 10 ans en conduite et implémentation projets stratégiques dans la fourniture des matériels, des logiciels, des conseils et des services associés.

L'un des projets les plus efficaces qui est lancé par Eurequat le MOBIPOS qui consiste à développer un système conçu pour la gestion et le contrôle des magasins.

Aujourd'hui, vu l'intérêt de vouloir gagner en temps, de conserver les données, de limiter le nombre d'employés et pas mal d'autres raisons, ont poussé petites, moyennes et grandes entreprises à chercher des solutions informatiques capables de répondre à leurs besoins. Algérie ferries[2] est l'un des entreprises qui était intéressé par ces solution informatiques spécialement MOBIPOS qui est proposé par Eurequat. Sachant que Algérie ferries est une entreprise nationale de transport maritime de voyageurs.

[1] http://www.eurequat-algerie.com

[2] http://algerieferries.dz

MOBIPOS est une solution informatique conçu pour la gestion et le contrôle des magasins. Elle se compose de deux modules qui sont :

- Le module Front office qui aide les vendeurs de point de vente à automatiser la tache de la vente.
- Le module Back Office qui aide le responsable du magasin à automatiser les tâches de la gestion des vendeurs, la gestion des caisses. Et aussi assurer une meilleure gestion des stocks, un bon suivi les ventes réalisées par le Front office. C'est sur ce module que nous allons nous intéresser dans ce travail.

Après avoir pris connaissance de ce projet, nous avons été fortement motivés de travailler dessus non seulement pour son intérêt mais également pour pouvoir vivre une expérience professionnelle au sein de l'entreprise.

Problématique

A l'issue de notre entretien avec le chef de projet on a conclu que seulement un module du système a été réalisé, c'est la solution front office pour gérer les ventes et l'encaissements. Dans le cas de notre client qui a des multi magasins avec ses caisses. Nous Seront obligée de développer le module Back office qui assure les fonctions essentielles au pilotage de points de vente : ventes, stocks, gestion des produits, statistiques, inventaires…

Objectifs

Le meilleur moyen c'est de complémenter la solution avec un autre module qui sera dédié à un type d'utilisateur :

Un administrateur qui a la possibilité de :

- Gérer les utilisateurs.
- Gérer les caissiers / vendeurs.
- Gérer les catégories avec ses produits.
- Gérer les traversées.
- Consulter les ventes réalisées.
- Générer des rapports.

- Configurer les magasins avec ses caisses.

Plan du mémoire

Ce rapport présente l'ensemble des étapes suivies pour développer la solution. Il contient trois chapitres organisés comme suit :

Le premier chapitre intitulé « *Contexte général du projet MOBIPOS* » est consacré à la présentation du contexte général du projet ainsi que l'organisme d'accueil.

Le chapitre suivant intitulé « *Analyse et conception du module Back office MOBIPOS* » cette partie consiste à collecter, analyser et définir les besoins de haut niveau (les besoins fonctionnels et les besoin non fonctionnels). Par la suite, nous élaborons une conception détaillée des cas d'utilisation, les diagrammes de séquence, ainsi que le diagramme de classe complet.

Le dernier chapitre intitulé « *Réalisation et tests du module Back office MOBIPOS* » présente l'environnement de travail ainsi que les outils logiciels que nous avons utilisés pour la réalisation de notre projet. Il illustre aussi le travail réalisé avec un ensemble d'interfaces graphiques conçues pour l'application.

Nous terminerons ce rapport par une conclusion générale qui résume notre travail avec des perspectives pour les futurs travaux de ce projet.

CHAPITRE I

CONTEXTE GENERAL DU PROJET MOBIPOS

Rapport-Gratuit.com

I. Introduction

Nous présentons dans ce chapitre une étude préliminaire du projet MOBIPOS. Dans un premier temps, nous commençons par présenter l'organisme d'accueil « Eurequat Algérie » dans lequel notre stage s'est déroulé. Par la suite, nous décrivons le système MOBIPOS ainsi que ses différentes parties. Nous terminons par une comparaison entre notre système et d'autres systèmes existants.

II. Présentation de l'organisme d'accueil

Cette partie sera consacrée à la présentation de notre organisme d'accueil : Eurequat Algérie.

II.1 Présentation "Eurequat Algérie"

Eurequat Algérie est une société de services informatiques spécialisée dans l'intégration et la mise en œuvre de solutions professionnelles de traçabilité et d'identification automatique ainsi que les points de vente. Elle a démarré son activité en mars 2006, avec le soutien et les moyens techniques de la société Eurequat Technologie Europe.

Figure I.1 Le logo de la société Eurequat [1]

Les compétences d'Eurequat s'articulent autour des métiers liés à l'identification automatique par code à barres & RFID, aux systèmes embarqués, aux points de vente, et aux solutions mobiles. [1]

II.2 Domaine d'expertise

A- Identification :

L'identification englobe le domaine du badge et du contrôle d'accès (imprimantes, logiciel de conception de cartes, consommables, lecteurs de carte chifa, portes-badges, cordons, clips, …). [1]

B- Traçabilité :

La traçabilité est assurée aujourd'hui avec outils matériels et logiciels. Depuis de nombreuses années l'équipe Eurequat, spécialisée dans la traçabilité, a réalisé des logiciels "standards" afin de répondre aux exigences des entreprises manufacturières et de stockage. Ces logiciels de traçabilité sont également très ouverts afin d'évoluer et de respecter les normes et les spécificités liées à un métier. Une solution clé en main chez Eurequat comprend l'analyse du besoin, le développement, l'installation, la formation par experts de la traçabilité, tout en s'appuyant sur du matériel provenant des plus grands fournisseurs de matériels.[1]

II.3 Les projets réalisés

Eurequat Algérie propose différents projets dans le domaine de l'identification et la traçabilité, parmi ces projets :

- ✓ **« Track & Sale 2016 » :** est une solution de vente mobile qui permet la supervision des agents de distribution, et la traçabilité de l'ensemble des événements utiles du processus de distribution et de vente.
- ✓ **« Track 'in 2015 » :** est une solution de suivi des livraisons qui permet de garantir le suivi d'une livraison et la supervision des chauffeurs. Elle permet d'améliorer la qualité de service des logisticiens.
- ✓ **« IventWay 2013 » :** est une solution de gestion des immobilisations qui permet gérer les inventaires des immobilisations de manière optimale, rapide et fiable. Cette solution est destinée aux grandes organisations ayant un parc d'immobilisations très important.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

- ✓ « **MOBIPOS 2017 » :** est une solution point de vente conçue pour la gestion et le contrôle des magasins. Elle est adaptée au TPV (Terminal point de vente) et autres tablettes Android. Elle contient les modules suivants :
  - Gestion du stock.
  - Gestion des ventes et encaissement.
  - Gestion des utilisateurs.

Dans notre projet de fin d'étude, nous allons nous intéresser et étudier le projet MOBIPOS afin d'y contribuer à son développement.

III. Système MOBIPOS

Actuellement, l'entreprise "Eurequat" travaille sur un projet MOBIPOS de version spécifique à Algérie Ferries[3]. Ce projet appartient à la catégorie des systèmes terminaux point de vente TPV.

Cette version actuelle permet de gérer et contrôler les magasins, les stocks et de générer les rapports. Elle permet de gérer les stocks et visualiser les statistiques.

III.1 Description des systèmes TPV

Un terminal point de vente ou TPV désigne une caisse connectée au système d'information du magasin.

Le passage des simples caisses aux terminaux points de vente a permis d'améliorer la gestion des points de vente et de traiter un certain nombre d'opérations en temps réel (gestion des stocks, gestion des prix, gestion des opérations promotionnelles, etc.).

[3] http://algerieferries.dz/

La communication des informations entre le TPV et le système d'information se fait dans les deux sens. [2]

Figure I.2 Terminal point de vente (TPV) [1]

III.2 Description du système MOBIPOS

MOBIPOS est un système conçu pour la gestion et le contrôle des multi-magasins avec ses caisses. Il se compose de deux modules qui sont :

1- Le Front office qui est déjà réalisés par l'entreprise pour gérer des caisses modernes.

2- Le Back office qui permet de gérer des multi-magasins avec ses caisses.

Notre travail consiste à concevoir et développer ce module.

III.2.1 Architecture fonctionnelle MOBIPOS

La figure ci-dessous détaille l'architecture du système MOBIPOS dans lequel s'insère notre module Back Office.

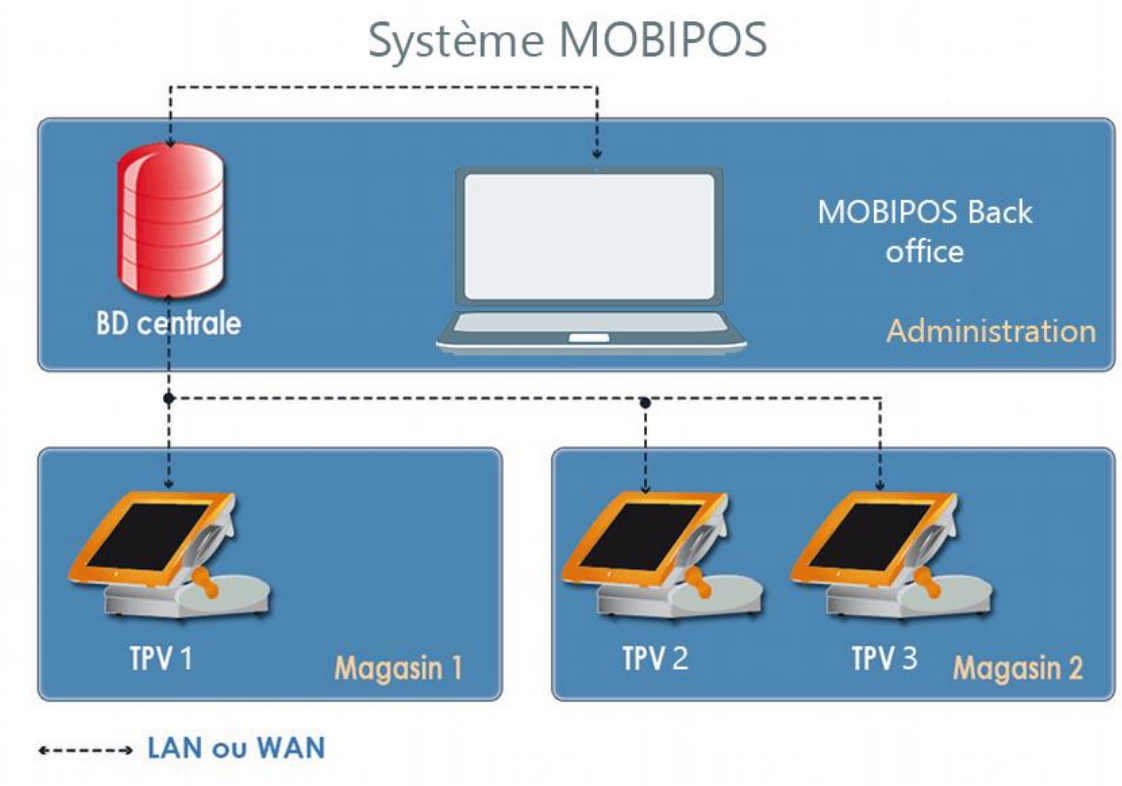


Figure I.3 Architecture globale du système MOBIPOS

III.2.2 Description du module front office MOBIPOS

Une application de front office est la partie d'un système informatique accessible aux utilisateurs finaux ou aux clients. [3]

Dans notre système, le module front office est une application conçue pour la gestion et le contrôle des caisses. Elle est adaptée au TPV et autres tablettes Android. Elle permet d'effectuer des ventes, encaissement et gestion du stock.

La figure ci-dessous détaille les fonctionnalités de Front office MOBIPOS.

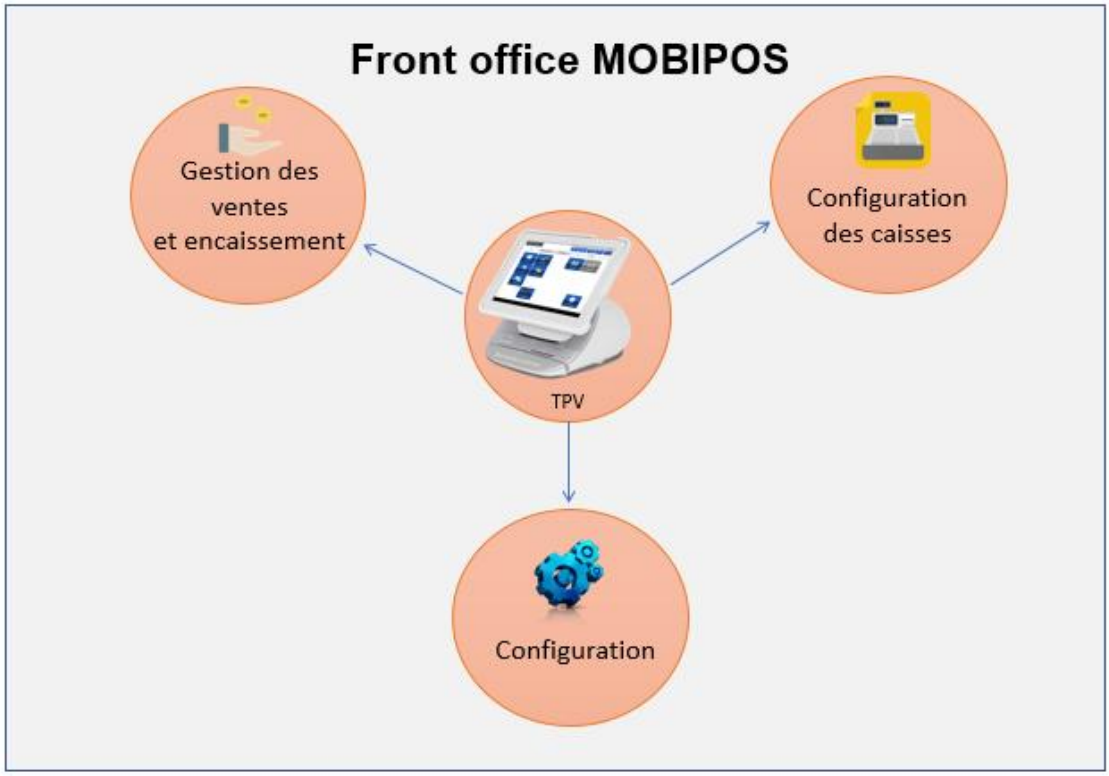


Figure I.4 Fonctionnalités de Front office MOBIPOS

Dans l'architecture montrée dans la figure I.4 nous avons les fonctionnalités globales de Front office MOBIPOS qui sont :

- La gestion des ventes et l'encaissement.
- La configuration des caisses.
- La configuration.

III.2.3 Description du module back office MOBIPOS

Dans le jargon du web, une interface d'administration (ou Back office) est la partie qui permet à l'entreprise d'administrer et de gérer son site ou son application. Accessible depuis un navigateur web, il faut généralement un identifiant et un mot de passe pour pouvoir s'y connecter. [4]

Dans notre système, le module Back office est un module qui permet de contrôler les ventes sur les bateaux en respectant les règles européennes pour limiter les fraudes. En plus, il permet aussi de gérer, si nous disposons d'un réseau de magasins avec ses

multi caisses (plusieurs stocks et ventes à superviser), des statistiques propres à chaque point de vente. Ensuite il gérer des profils de voyageurs, même des navigants.

La figure ci-dessous détaille les fonctionnalités de Back office MOBIPOS.

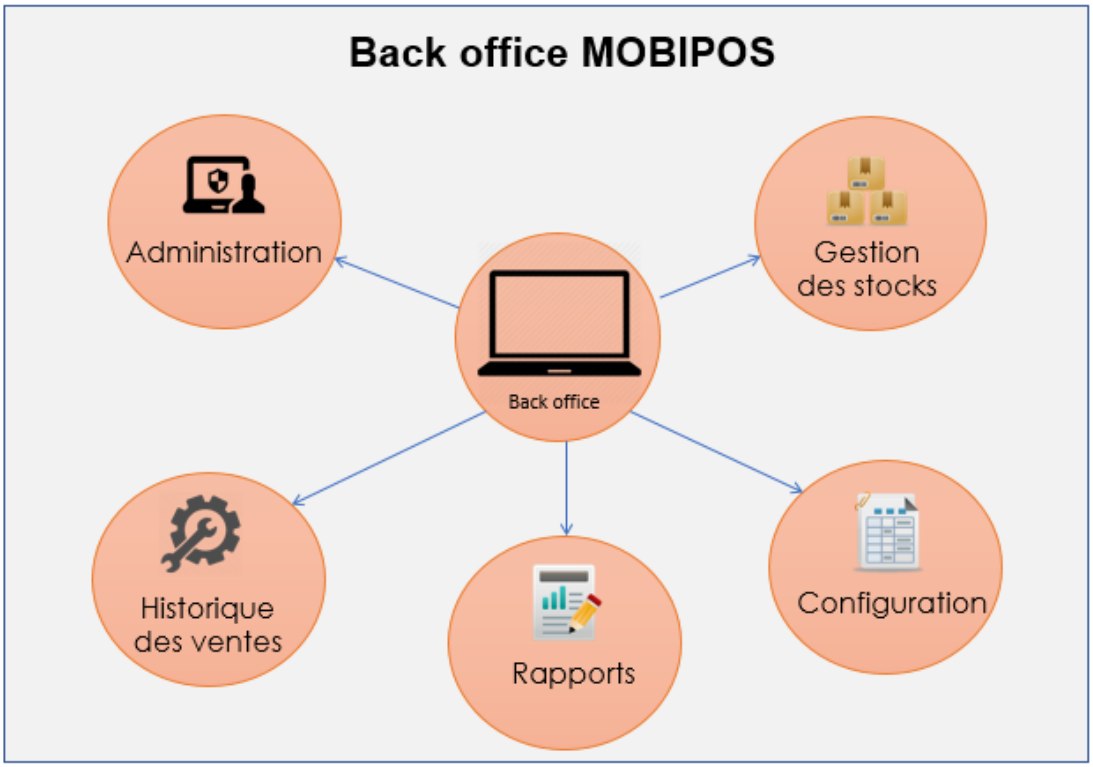


Figure I.5 Fonctionnalités de Back office MOBIPOS

Dans l'architecture montrée dans la figure I.5 nous avons les fonctionnalités globales de Back office MOBIPOS qui sont :

- L'administration.
- L'historique des ventes.
- Gestion des stocks.
- Les rapports.
- La configuration.

III.3 Comparatif entre les systèmes équivalents à MOBIPOS

Sur le marché international, il apparaît plusieurs systèmes de gestion du magasin. Nous avons choisi ceux qui se rapprochent le plus de notre projet. Dans ce qui suit nous

donneront un comparatif entre trois différents systèmes qui sont : RoverCash, Phasis et Store POS, selon des critères bien définis.

- **RoverCash :** une application de point de vente disponible sur IOS et Android permet de gère un magasin, est développer par l'entreprise LUNDI MATIN [5][6]
- **Phasis :** un logiciel de point de vente compatible pour **caisse enregistreuse** et PC tactiles, est développer par l'entreprise Crisalid [7].
- **Store POS :** une solution de point de vente contient deux modules, Store POS front office une application Android et Store POS back office une application web , cette solution est développer par l'entreprise AEM Softs [8] .

Ci-dessous un tableau comparatif entre MOBIPOS et trois différents systèmes.

	MOBIPOS	RoverCash	Phasis	Store POS
Licence	Payant	Gratuit/ Payant	Payant	Payant
Langue	Français	Anglais/ Français	Français	Français
Plateformes	Android/Web	Android/IOS	Windows	Android/Web
Standard/Spécifique	Spécifique	Standard	Standard	Standard
Back office	Oui	Non	Non	Oui
Multi-magasin	Oui	Oui	Oui	Oui
Satatistiques	Oui	Oui	Oui	Oui
Import/Export	Oui	Non	Oui	Oui
Gestion de la relation client	Oui	Oui	Oui	Oui

Tableau I.1 Comparatif entre différents systèmes [5][6][7][8]

Remarque :

Standard : adapté à tous les clients.

Spécifique : l'application est développée pour un client spécifique, adapté seulement pour ce client comme notre cas pour l'entreprise algerie ferries.

IV. Cycle de vie adopté

Pour chaque produit (ou application) conçu et développé on choisit une démarche pour le suivre tout au long du projet que l'on appelle cycle de vie du produit. Le cycle de vie se base sur la notion de développement durable en fournissant un moyen efficace pour évaluer les impacts environnementaux d'un produit.

Nous avons choisi, pour la conception et le développement de notre projet, le cycle de vie incrémental car il prend en compte le fait qu'un logiciel peut être conçu étape par étape, et chaque incrément implémente plusieurs fonctionnalités triées par ordre de priorité et est intégrée par la suite à l'ensemble des précédents. À chaque étape, le produit est testé, exploité et maintenu dans son ensemble jusqu'à ce que la totalité du produit soit finie.

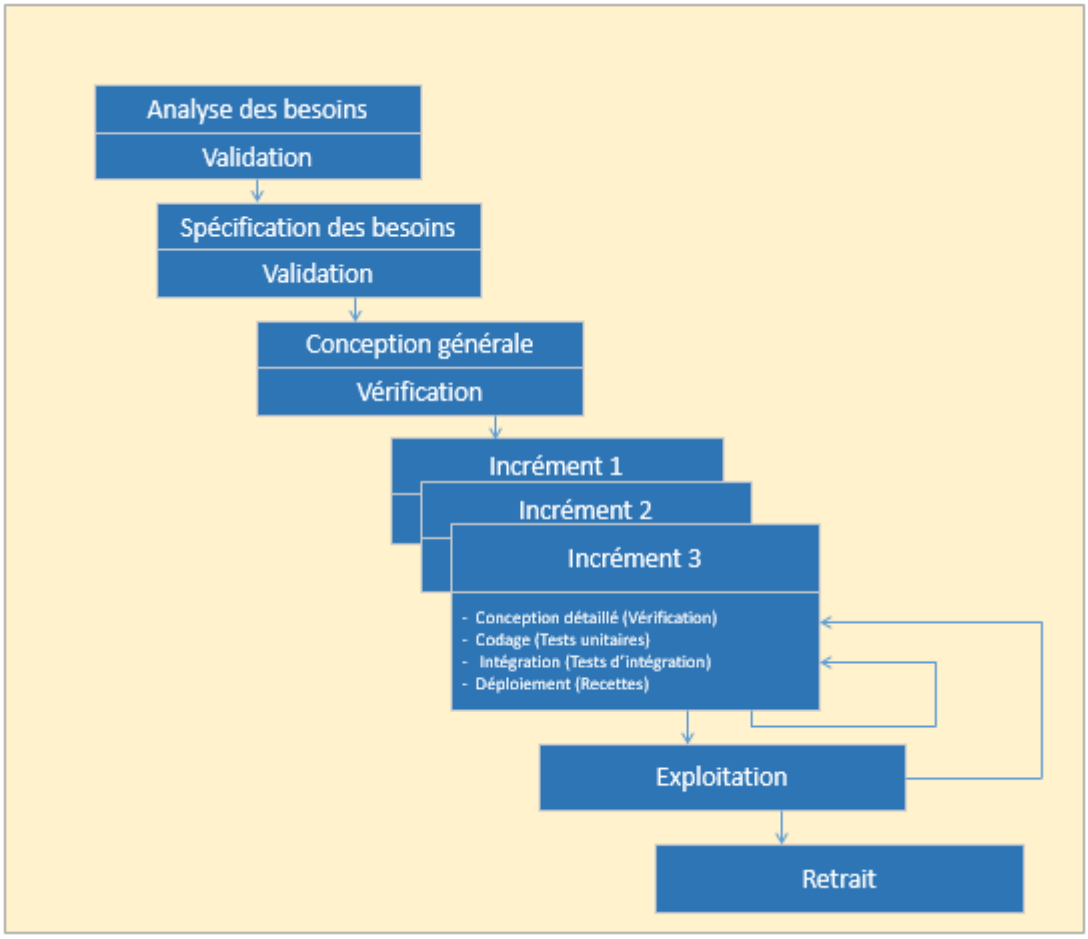


Figure I.6 Cycle de vie de Back office MOBIPOS

V. Conclusion

Dans ce chapitre introductif, nous avons présenté l'organisme d'accueil où nous avons effectué notre stage. Ensuite, nous avons donné une vue générale sur le système MOBIPOS. Nous allons, dans le deuxième chapitre, décortiquer le cahier des charges pour analyser les besoins recensés et présenter la conception et la modélisation du système Back office.

Rapport-Gratuit.com

CHAPITRE II

ANALYSE ET CONCEPTION DU MODULE BACK OFFICE MOBIPOS

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

I. Introduction

Ce chapitre vise à capter les besoins, identifier les rôles des utilisateurs et préparer le plan de réalisation. Dans un premier lieu, nous allons présenter le cahier de charge de notre projet MOBIPOS. Ensuite, nous allons identifier les acteurs du projet, ceux qui toucheront de façon directe notre application back-office, lister les exigences fonctionnelles et non fonctionnelles du produit puis nous allons finir par utiliser une démarche objet basé sur le processus unifié (UP) qui s'appuie sur le langage de modélisation UML (Unified Modeling Language).

Les diagrammes UML utilisés dans la démarche suivie sont :

- Les diagrammes de cas d'utilisation pour spécifier les besoins de notre système.
- Les diagrammes de séquence qui détaillent quelques cas d'utilisation.
- Un diagramme de classes global.

II. Cahier de charge du projet

Au début de ce projet, Nous avons fait une réunion avec le chef de projet pour savoir les besoins et les résultats attendus par le client.
Et s'appuyant sur cette réunion, de rédiger un cahier de charges qui reprend tous les besoins exprimés ainsi que toutes les futures fonctionnalités.

II.1. Contraintes techniques

La solution fonctionne avec tous les navigateurs WEB : Firefox, chrome, internet explorer.

II.2. Contraintes fonctionnelles

- **L'administration globale du système**

Elle contient les éléments suivants :

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

- **La gestion des produits :** La solution doit offrir un mécanisme de gestion des produits et leurs prix. Elle consiste à créer, modifier, supprimer et visualiser la liste des produits avec leur prix.
- **La gestion des catégories :** La solution doit offrir un mécanisme de gestion des catégories qui consiste à définir une quantité limitée pour certaines catégories bien spécifiées par notre client.
- **La gestion des caissiers/vendeurs :** La solution doit offrir un mécanisme de gestion des caissiers/vendeurs qui consiste à créer, modifier, supprimer et visualiser la liste des caissiers / vendeurs.
- **La gestion des cartes d'embarquements navigants / passagers :** La solution doit importer le fichier de type (xslx, xsl, csv) qui contient la liste des navigants et passagers. Ces derniers peuvent faire des ventes par leurs cartes d'embarquements au niveau du front office.
- **La gestion des utilisateurs :** La solution doit offrir un mécanisme de gestion des utilisateurs qui consiste à créer, modifier, supprimer et visualiser la liste des utilisateurs.
- **La gestion des traversées :** La solution doit offrir un mécanisme de gestion des traversées qui consiste à créer, modifier, supprimer et visualiser la liste des traversées.

- **L'historique des ventes**

Le système doit visualiser en temps réel les ventes effectuées dans différents magasins et selon le critère sélectionné (par produit, par ticket et par carte d'embarquent). Par la suite, générer des rapports correspondants à chaque historique des ventes.

- **Génération des rapports**

Le système doit générer des rapports qui sont :

- ✓ **Sorties de stocks :** Correspond à la sortie de stock qui ne sont pas liées à une vente.
- ✓ **Etat des mouvements de stocks :** recensent les sorties de stock suite à une vente et les entrées en stock
- ✓ **Historique du stock :** Consiste à générer des rapports qui contient la photo du stock au moment de la génération de l'inventaire.

- **Gestion des stocks :**

Le système doit visualiser l'historique des stocks. Par la suite générer des rapports correspondants à l'historique des stocks sélectionnés.

- **Configuration du système :**

La solution doit offrir un mécanisme de configuration qui consiste à gérer les magasins et d'affecter chaque caisse à une magasin.

II.3. Contraintes de sécurités

La contrainte de la sécurité a été très importante tout au long de la mise en place du projet.

Lors de la connexion à un système, celui-ci demande tout le temps un identifiant et un mot de passe pour y accéder.

III. La phase d'analyse

La démarche que nous avons adoptée consiste à l'implémentation de versions successives, qui recommence en proposant un produit de plus en plus complet.
Nous avons utilisé UML comme formalisme de conception.

III.1. Le langage UML

UML (Unified Modeling Language), se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML modélise l'ensemble des données et des traitements en élaborant des différents diagrammes. En clair, il ne faut pas designer UML en tant que méthode (Il y manque la démarche) mais plutôt comme une boite d'outils qui sert à améliorer les méthodes de travail [9].

III.2. Entreprise Architect

Enterprise Architect est un logiciel de modélisation et de conception UML complet, Couvre le développement de logiciels de la collecte des besoins jusqu'à la phase d'analyse, des modèles de conception, les tests et la maintenance [10].

III.3. PowerMockup

C'est une plateforme qui nous a aidé à faire la conception des écrans c'est à dire le prototype des maquettes.

IV. Analyse globale de l'application MOBIPOS

L'objectif de notre travail est de concevoir et développer une application back-office qui permettra à un administrateur connecté à partir de l'application de pouvoir contrôler et gérer ses magasins.

IV.1. Description des acteurs

Un acteur représente un élément externe qui interagit avec un système dans le but de le faire fonctionner et d'en tirer profit [11].

Nous commençons notre analyse par identifier les acteurs qui agissent sur notre système à savoir :

❖ **Administrateur :**

C'est l'acteur principale dans notre projet c'est à dire dans la partie back-office, il possède le privilège de plus haut niveau. Donc il est capable de manipuler toutes les fonctionnalités proposées par l'application notamment la gestion des employés, des utilisateurs, des navigants et des voyageurs. Il prend aussi en charge la gestion des produits / prix, des magasins, des stocks et la consultation des ventes.

Back-Office

IV.2. Identifications des messages

Un message représente la spécification d'une communication unidirectionnelle entre objets qui transporte de l'information avec l'intention de déclencher une activité chez le récepteur [11].

Message émis au système :

- Demande d'authentification.
- Visualisation du stock.
- Visualisation de l'historique des ventes
- Visualisation de la liste des vendeurs, des utilisateurs.
- Visualisation de la liste des catégories, des produits et ses prix.

Message émis par le système :

- Visualisation des interfaces, de validation, de modification
- Affichage des messages d'erreurs et de confirmations

IV.3. Spécification des besoins fonctionnels

Il s'agit des fonctionnalités du système. Ce sont les besoins spécifiant un comportement d'entrée / sortie du système.

Dans cette partie nous allons identifier les besoins fonctionnels de notre acteur :

- **Gestion des utilisateurs :** Consiste à gérer les utilisateurs du système.
- **Gestion des employées :** Consiste à gérer la liste des caissiers ou des vendeurs qui travaillent dans les magasins.

- **Gestion des catégories :** Consiste à gérer la liste des catégories des produits qui est une forme de regroupement des produits.
- **Gestion des cartes d'embarquements :** Consiste à importer la liste des cartes d'embarquements pour pouvoir garder l'historique des cartes qui sont actives ou non actives. On distingue deux types de carte :
  a) **Cartes pour passagers.**
  b) **Cartes pour navigants.**
- **Gestion des produits / prix :** Consiste à gérer la liste des produits qui sont définis par des données générales (code à barres, libellé, description et caractéristiques
- **Visualisation l'historique des ventes :** Consiste à visualiser la liste des ventes :
  - ✓ **Par produit :** C'est-à-dire regroupement par produit.
  - ✓ **Par ticket :** C'est-à-dire regroupement par ticket.
  - ✓ **Par carte :** C'est-à-dire regroupement par numéro de carte d'embarquement.

- **Gestion des stocks :** Consiste à visualiser les historiques des stocks et par la suite générer des rapports correspondants au opérations effectuées.
- **Gestion des magasins :** Consiste à gérer la liste des magasins qui sont définis par nom de magasin, l'emplacement et son responsable.
- **Contrôle des caisses :** Consiste à contrôler de l'ouverture automatique de la caisse. Et aussi gérer la liste des caisses et les affecter aux magasins.
- **Génération des rapports :** Consiste à générer des rapports pour pouvoir :
  - Consulter l'historique des stocks.
  - Consulter l'état des mouvements.
  - Consulter les sorties des stocks.

IV.4. Spécification des besoins non fonctionnels

Il s'agit des besoins qui caractérisent le système. Ce sont des besoins nécessaires pour atteindre la perfection et la bonne qualité du système.

Dans le cadre de ce travail, nous allons distinguer les besoins non fonctionnels pour fournir un produit performant qui respecte les exigences de client et qui sont :

- **L'extensibilité :** L'application devra être extensible, c'est-à-dire qu'il pourra y avoir une possibilité d'ajouter ou de modifier de nouvelles fonctionnalités.
- **La convivialité :** L'application doit offrir une interface simple à utiliser pour tout type d'utilisateur. Elle doit aussi assurer une facilité de navigation et une compréhensibilité d'écriture.
- **Fiabilité :** L'application doit offrir aux utilisateurs des résultats fiables et corrects.
- **Confidentialité :** L'application devra être hautement sécurisée car les informations ne devront pas être accessibles à tout le monde (Authentification).
- **La rapidité** : En effet, vu le nombre important des transactions quotidiennes, il est impérativement nécessaire que la durée d'exécution des traitements s'approche le plus possible du temps réel.
- **La performance :** Un système doit être avant tout performant c'est-à-dire à travers ses fonctionnalités, répond à toutes les exigences des usagers d'une manière optimale.
- **Portabilité :** Elle fonctionne sur tout système d'exploitation.
- **L'ergonomie :** le thème (Template) adopté par application doit être inspiré des couleurs de l'entreprise.

V. Conception

V.1. Diagramme de cas d'utilisations

Un cas d'utilisation (use case) représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier [11].

Dans cette partie nous allons concentrer à capturer les besoins de notre projet qui consiste à réaliser la partie back office du projet global.

V.1.1. Diagramme de cas d'utilisation globale

Nous avons regroupé l'ensemble de cas d'utilisation dans un seul diagramme

qui est ci-dessous (voir figure II.1)

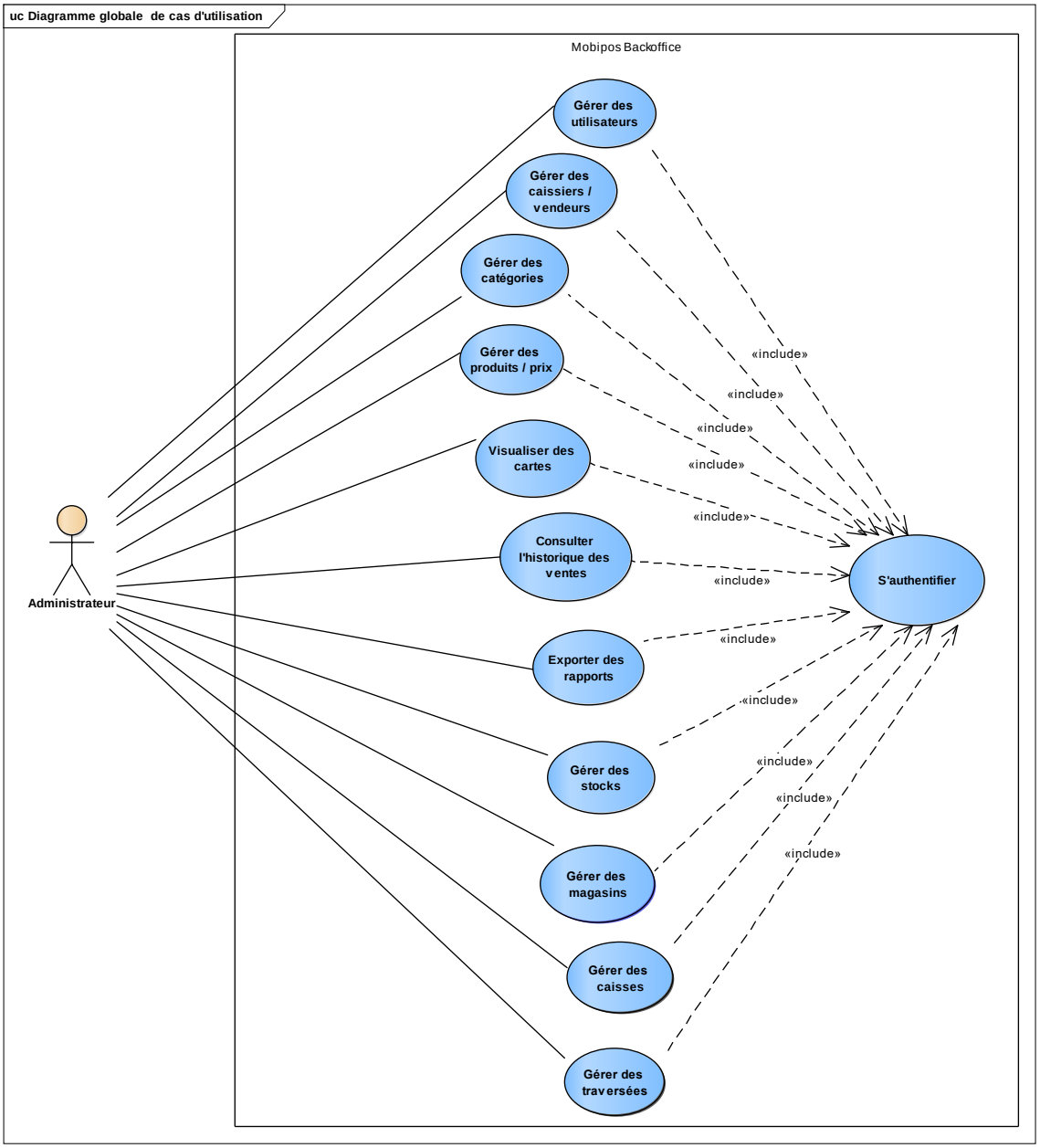


Figure II.1 : Diagramme globale de cas d'utilisation

V.1.2. Diagramme de cas d'utilisation du cas « Consulter l'historiques des ventes »

Nous détaillons le cas « Consulter les historiques des ventes » par un sous diagramme de cas d'utilisation qui se présente dans la figure suivante suivant : (voir figure II.2)

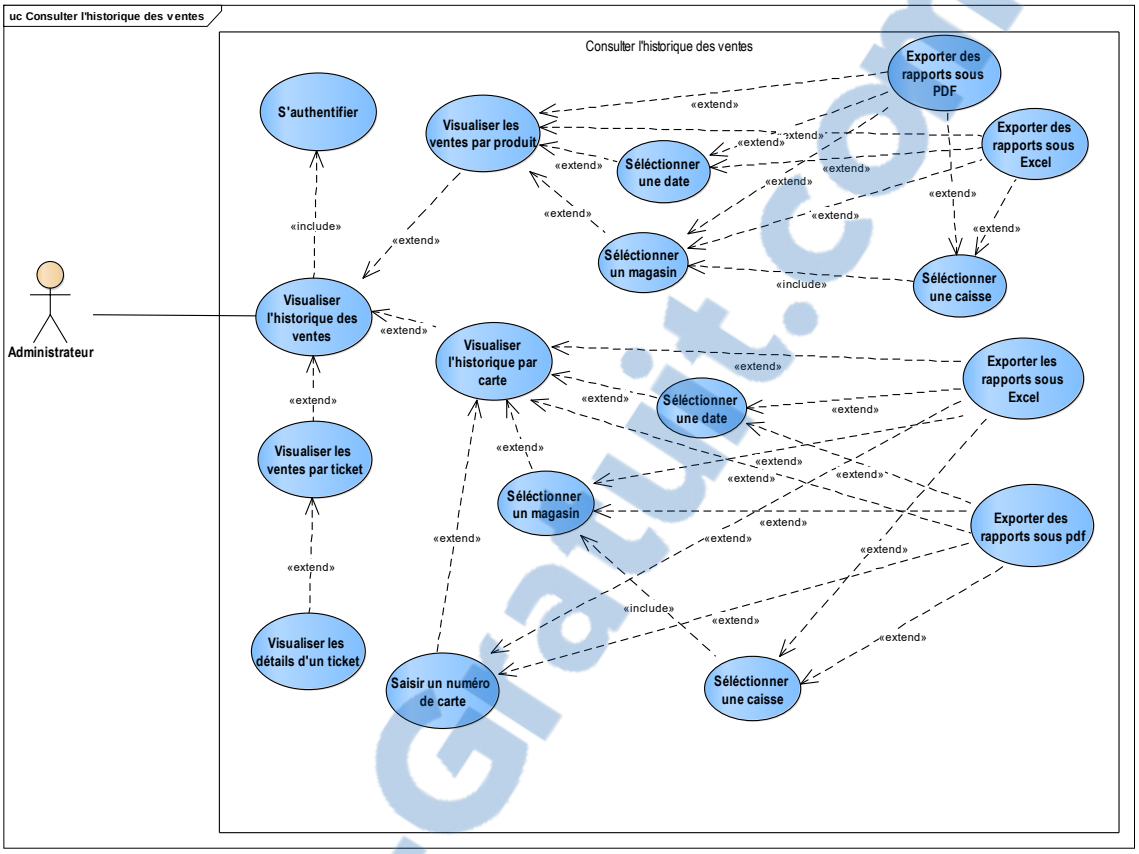


Figure II.2 : Diagramme du cas « Consulter l'historique des ventes »

V.1.3. Diagramme de cas d'utilisation du cas « Exporter des rapports »

Nous détaillons le cas « Générer des rapports » par un sous diagramme de cas d'utilisation qui se présente dans la figure suivante suivant : (voir figure II.3)

Rapport-Gratuit.com

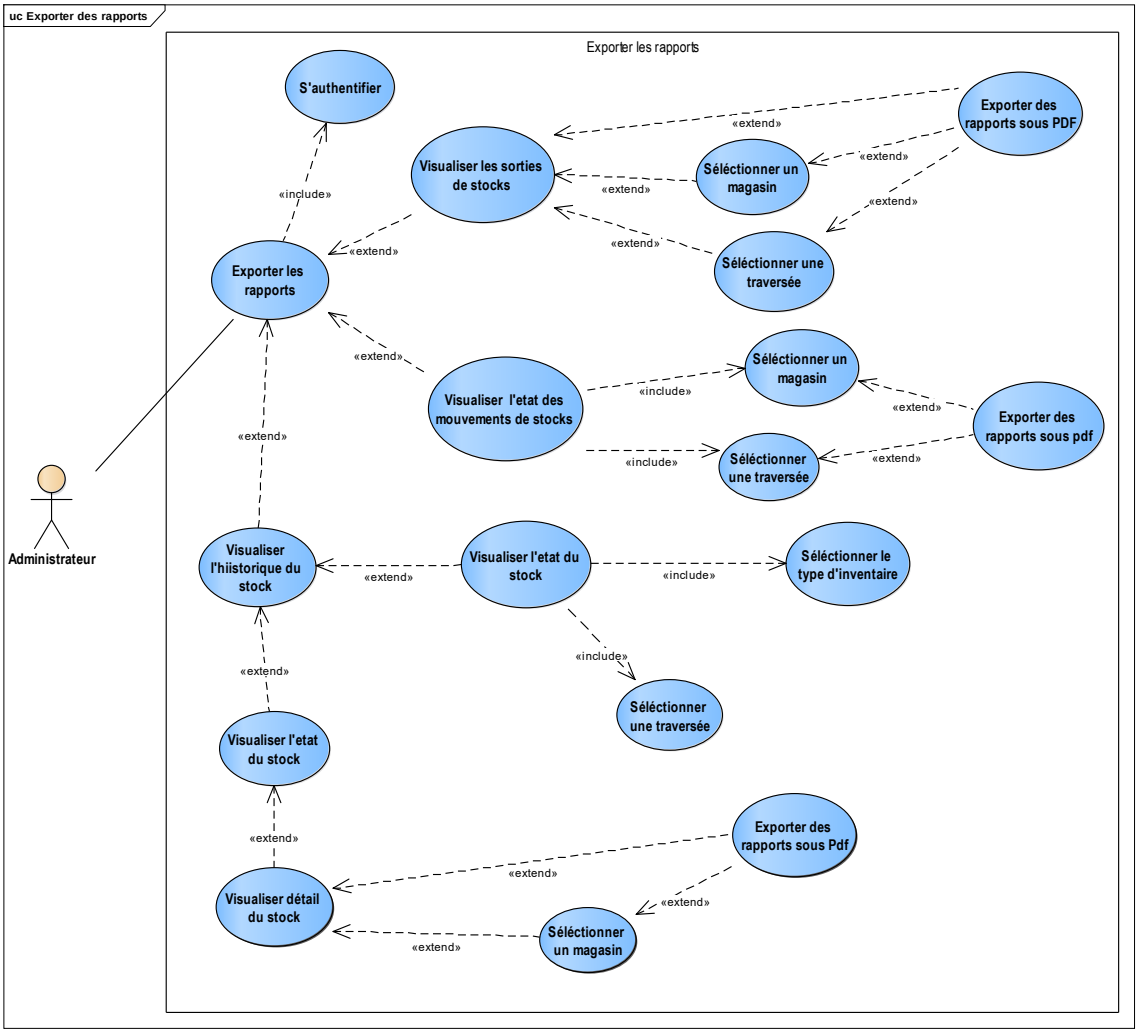


Figure II.3 : Diagramme du cas « Exporter des rapports »

V.2. Diagramme de séquence

Pour schématiser la vue comportementale de notre système, nous faisons recours au diagramme de séquence d'UML [12]. Ce diagramme permet de présenter les interactions entre l'acteur et le système avec des messages présentés dans un ordre chronologique.

On présentera ci-après les diagrammes de séquences, que nous avons jugés importants de notre application.

V.2.1 Diagramme de séquence du cas « S'authentifier » :

La Figure II.4 montre le diagramme de séquence de l'authentification.

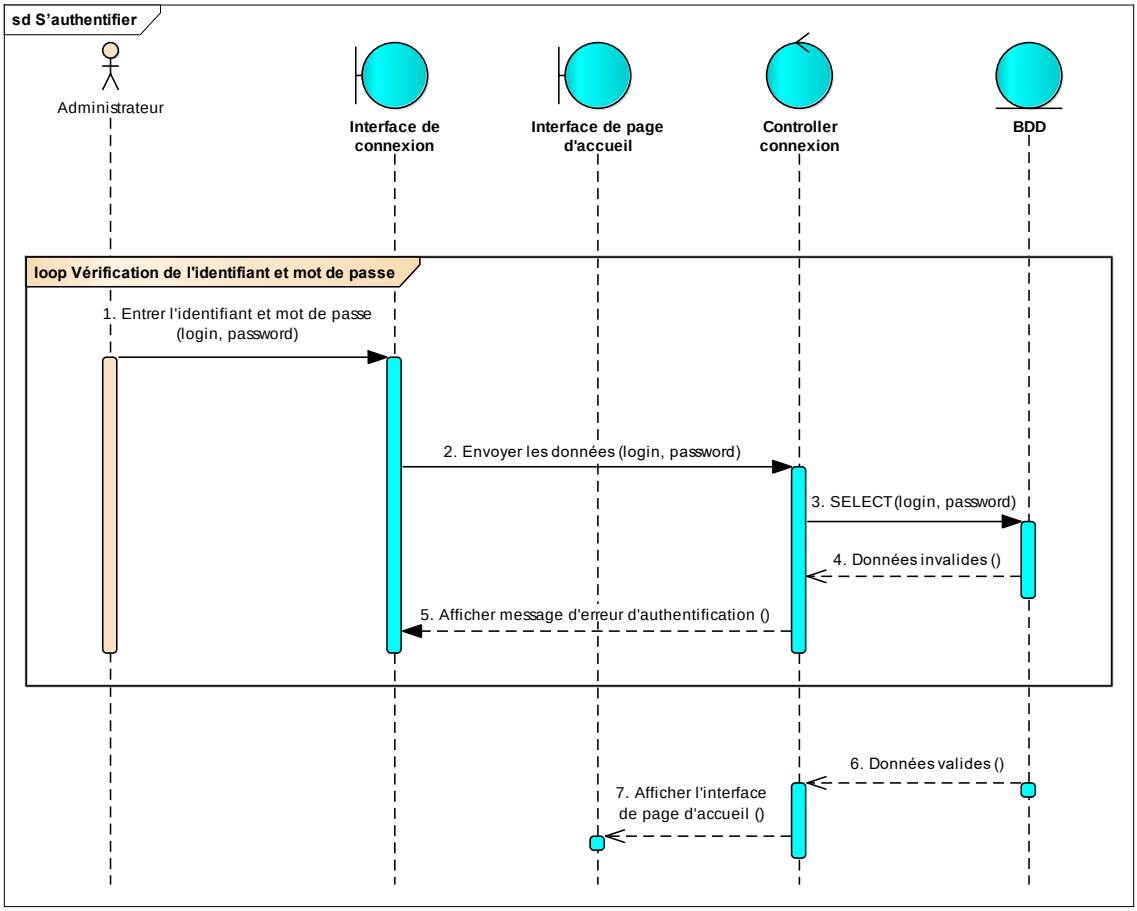


Figure II.4 : Diagramme de séquence « S'authentifier »

Description textuelle du diagramme de séquence « S'authentifier »

Acteur : Utilisateur.

Précondition : Serveur disponible.

Postcondition : Utilisateur authentifié.

Description des scénarios :

a) Scénario normal :

1. L'utilisateur accède à l'interface de connexion et saisit son identifiant et son mot de passe.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

2. Les données saisies lors de la demande de connexion seront envoyées vers le contrôleur de connexion qui va vérifier l'existence de l'utilisateur dans la base de données.
3. La base de données annonce au contrôleur de connexion que les données d'authentification de l'utilisateur sont bien valides qui à son tour le redirige vers l'interface d'accueil.

b) Scénario d'erreur :

A1 : Login et mot de passe erronés. L'enchainement d'A1 démarre du point 3 du scénario normal.

3. La base de données annonce au contrôleur que les données sont invalides.
   Un message d'erreur est envoyé à l'utilisateur en lui indiquant que le login ou le mot de passe est incorrecte et lui demandant de les vérifiées

V.2.2 Diagramme de séquence « Ajouter une catégorie » :

La Figure II.5 met en évidence le diagramme de séquence d'ajout de catégorie.

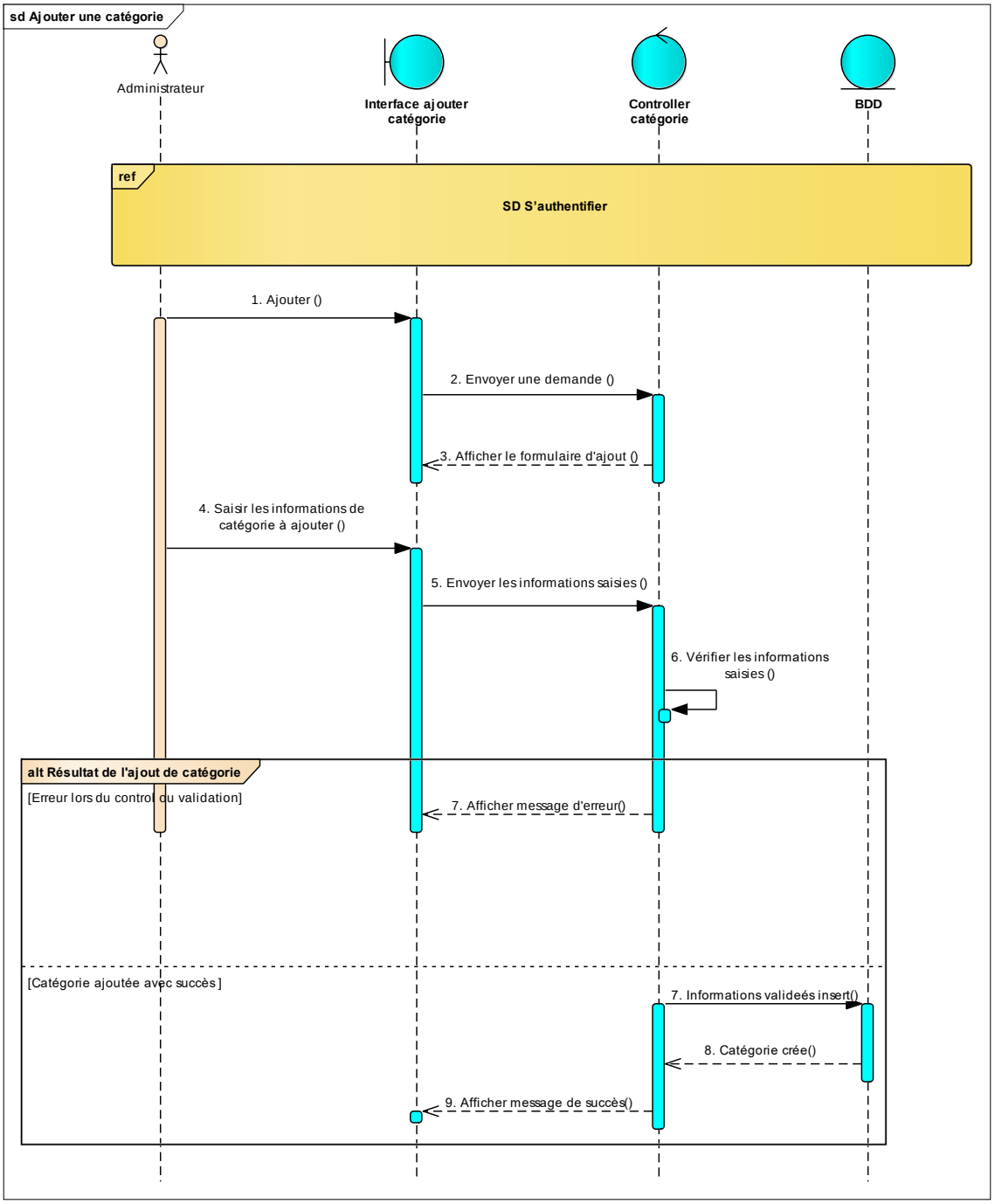


Figure II.5 : Diagramme de séquence « Ajouter une catégorie »

Description textuelle du diagramme de séquence « Ajouter une catégorie »

Acteur : Utilisateur.

Précondition : Utilisateur authentifié.

Postcondition : Catégorie ajoutée.

Description des scénarios :

c) Scénario normal :

1. L'Administrateur accède à l'interface d'ajout de catégorie.

2. Saisir les informations de catégorie à ajouter.

3. Les données sont envoyées au contrôleur d'ajout qui à son tour vérifie les informations à saisir.

4. La catégorie sera ajouté comme une nouvelle entrée dans la table catégorie et une vue de succès d'ajout sera affiché.

d) Scénario d'erreur :

A1 : Login et mot de passe erronés. L'enchainement d'A1 démarre du point 3 du scénario normal.

3. La base de données annonce au contrôleur que les données sont invalides.
4. Un message d'erreur est envoyé à l'utilisateur en lui indiquant que le login ou le mot de passe est incorrecte et lui demandant de les vérifiées

V.2.3 Diagramme de séquence « Visualiser rapport sortie stock » :

La Figure II.6 montre le diagramme de séquence de Visualisation des rapport sortie stock.

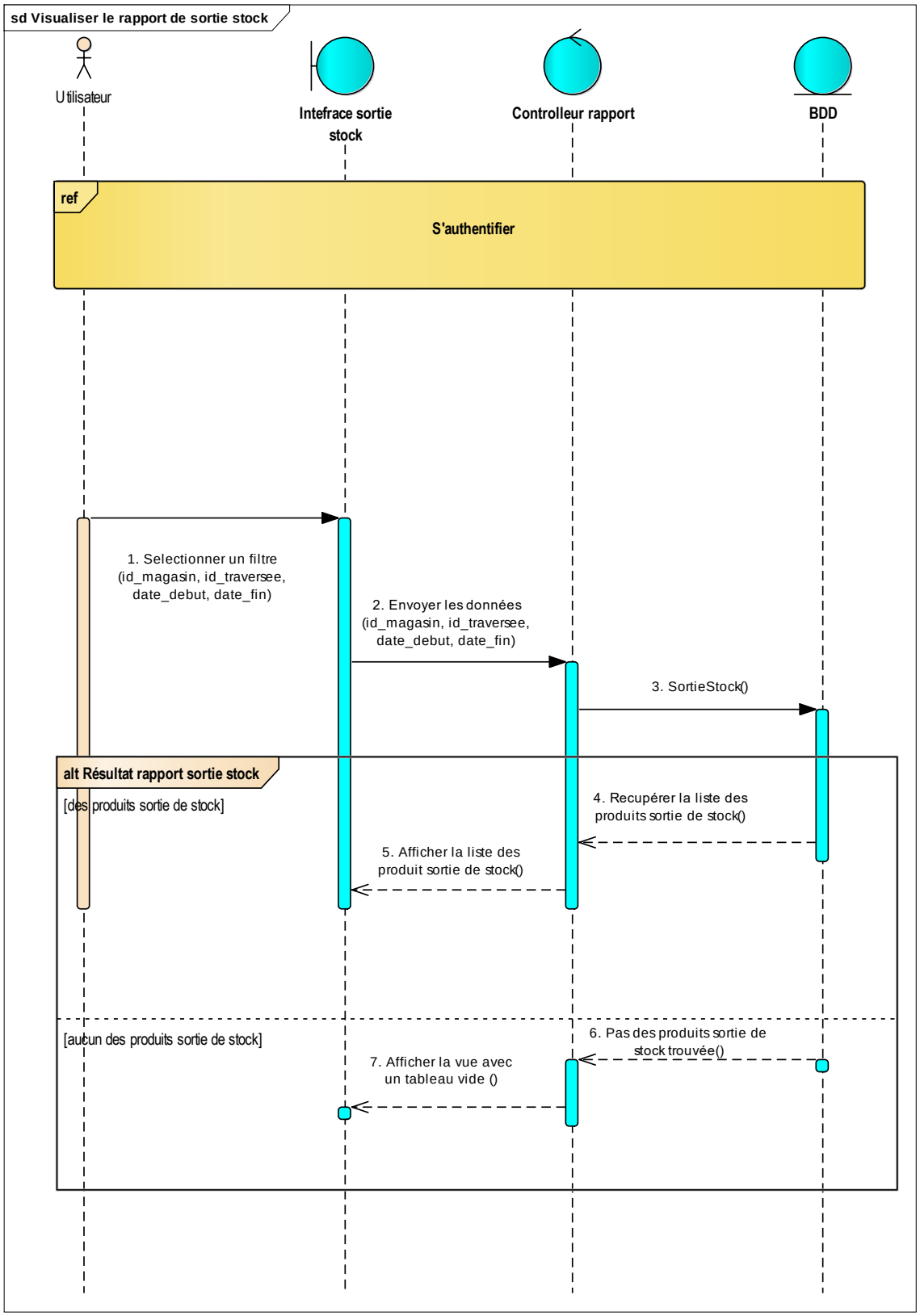


Figure II.6 : Diagramme de séquence « Visualiser rapport sortie stock »

Description textuelle du diagramme de séquence « Visualiser rapport sortie stock »

Acteur : Utilisateur

Précondition : S'authentifier

Description des scénarios :

a) Scénario normal :

1. L'utilisateur sélectionner un filtre (le nom de traversée et ses dates, le nom de magasin).
2. Les données seront envoyées au contrôleur de rapport qui par la suite recherche les produits sortis de stock avec les conditions sélectionner au filtre précèdent.
3. Le système affiche la liste des produits sortie de stock par rapport au filtre sélectionner

b) Scénario d'erreur :

A1 : Aucun des produits sortis de stock. L'enchainement d'A1 démarre du point 2.

3. Le système affiche un tableau vide avec un message dans le table "Aucune donnée disponible "

V.2.4 Diagramme de séquence « Export PDF rapport sortie stock » :

La Figure II.7 montre le diagramme de séquence de l'export rapport sortie stock.

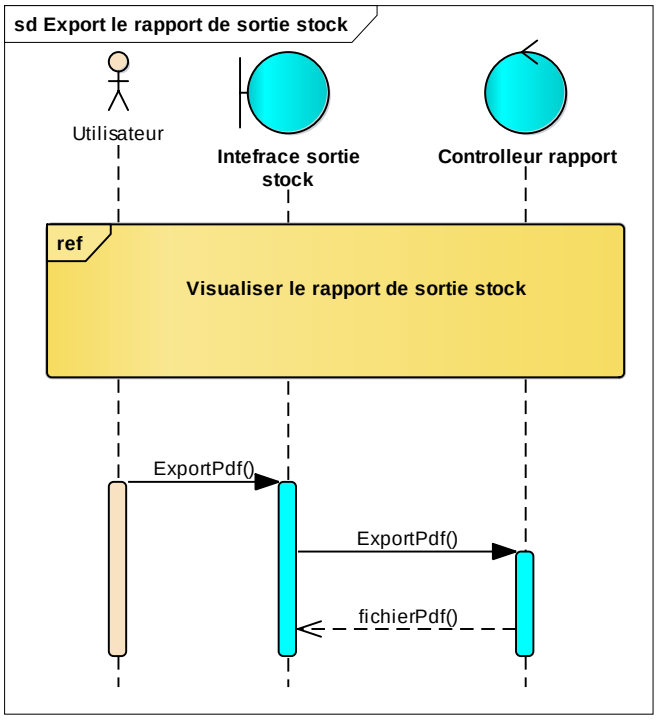


Figure II.7 : Diagramme de séquence « Exporter PDF rapport sortie stock »

Description textuelle du diagramme de séquence « Export PDF rapport sortie stock »

Acteur : Utilisateur

Précondition : Visualiser rapport sortie stock

Description des scénarios :

a) Scénario normal :

1. L'utilisateur demande d'exporter un fichier PDF
2. La demande sera envoyée au contrôleur de rapport qui par la suite généré un fichier PDF.
3. Le système retourne un fichier PDF téléchargé automatiquement par le navigateur ; il contient la liste des produits sortie de stock et les informations sélectionnées par le filtre

V.3. Diagramme de classe

Le diagramme de classe permet de représenter la relation entre les éléments de modélisations statiques comme les associations et leurs contraintes qui, par conséquent montrent la structure du modèle.

Nous avons fait plusieurs allers retours sur ce diagramme par rapport à chaque itération afin d'avoir un diagramme de classe final.

La figure II.8 montre le diagramme de classe du projet MOBIPOS backoffice qui représente les données.

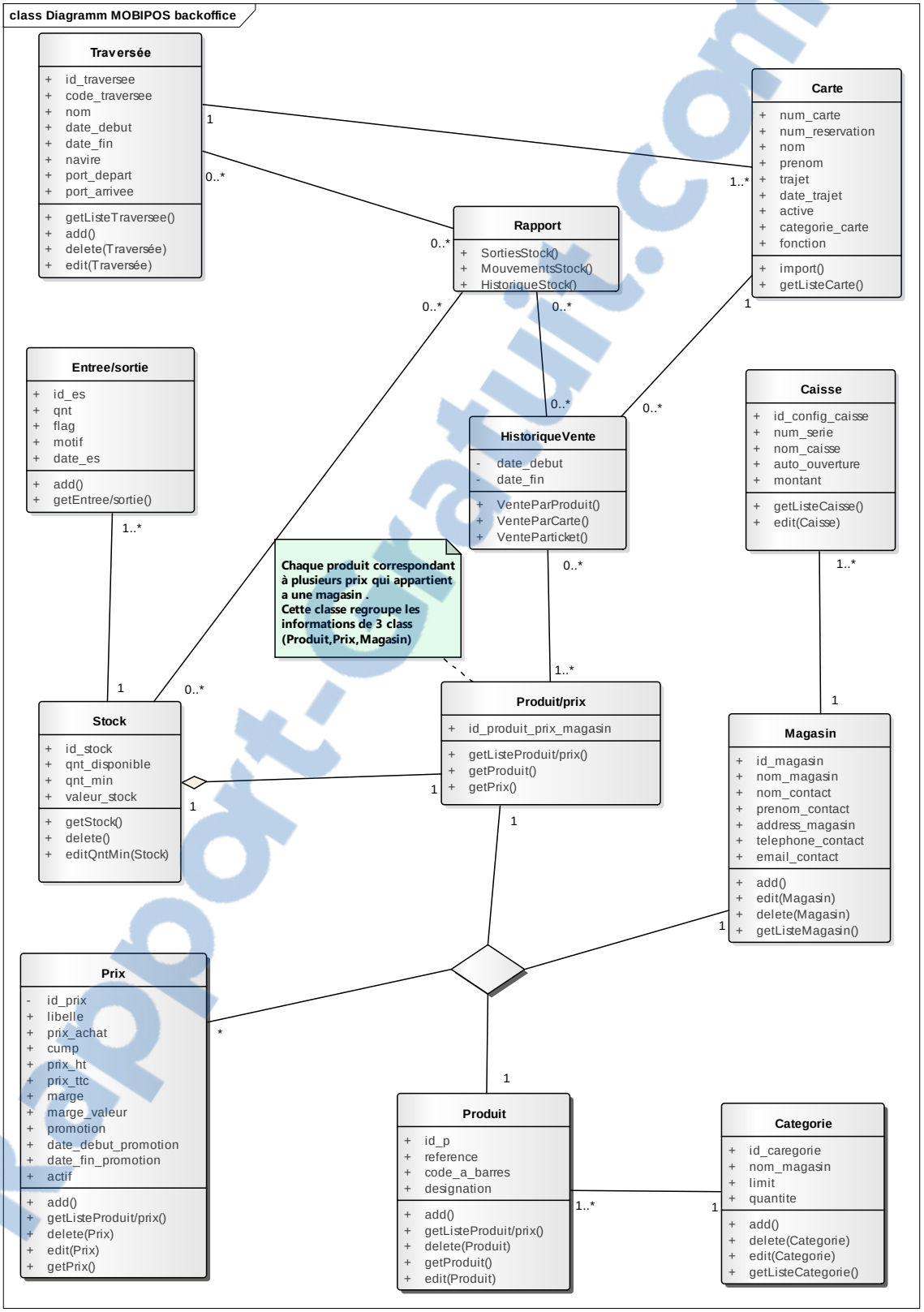


Figure II.8 Diagramme de class « projet MOBIPOS backoffice »

Rapport-Gratuit.com

VI. Conception des maquettes

Une fois le cahier des charges réalisé, commence la phase de maquettage, aussi appelée « wireframing ». Une étape indispensable pour se faire une idée plus précise de l'ergonomie de l'application.

Pour la réalisation des maquettes du Back Office nous sommes passés par deux étapes essentielles :

La définition de la mise en page de la maquette fonctionnelle, l'organisation des différents éléments et voir les principaux textes. Ensuite, nous avons a développé une maquette graphique avec l'outil dédié à la réalisation des maquettes « PowerMockup », qui fournit une grande collection d'éléments de l'interface utilisateur et des icônes entièrement réalisés à partir de formes PowerPoint permettant de créer des prototypes d'écran de web.

Voilà un des exemples de maquettes de notre application (Voir la figure II.9)

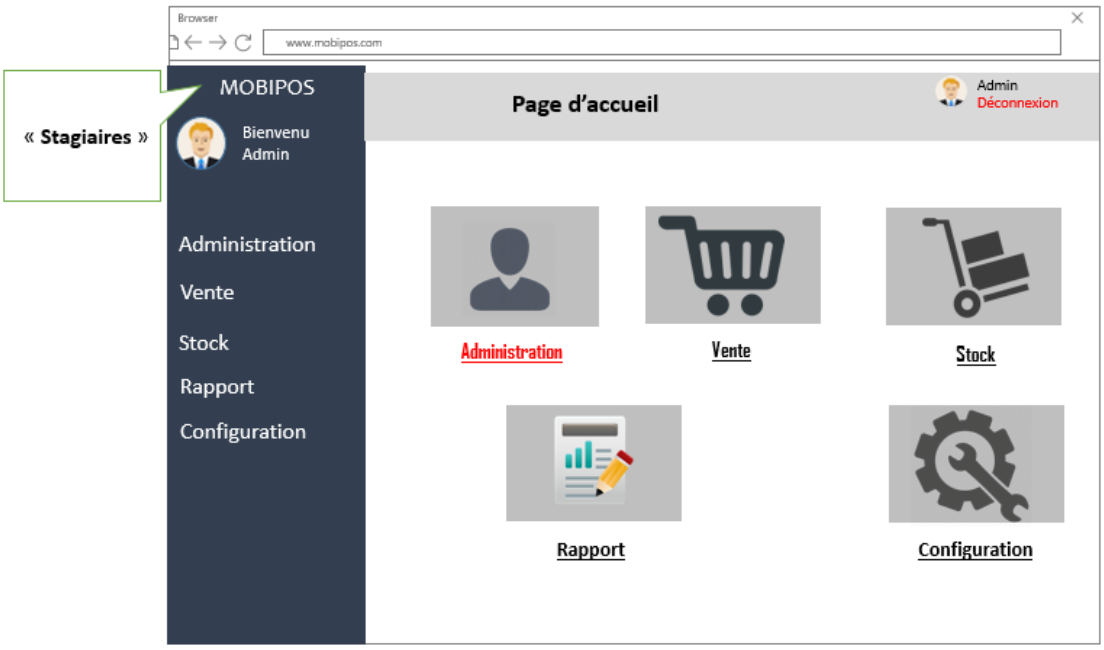


Figure II.9 Exemple de maquette « page d'accueil »

VII. Conclusion

Dans ce chapitre, Nous avons présenté le cahier charge de notre système. Par la suite, nous avons identifié les besoins fonctionnels et non fonctionnels de notre application. Ensuite, nous avons fait une étude conceptuelle du système qui consiste à produire un schéma qui offre une vision claire tout en satisfaisant les besoins de notre client, en appliquant le formalisme du langage Objet UML.

Les diagrammes de cas d'utilisation illustrent la vue fonctionnelle du système, ensuite, la vue statique permet de définir la structure du système et de dégager les différentes entités, cette vue est réalisée par le diagramme des classes. Enfin, la vue dynamique modélisé par les diagrammes de séquence détaillés nous a permis d'avoir une vue générale sur le déroulement des cas d'utilisation et leurs exécutions, afin de modéliser les interactions entre les différents objets du système.

Le chapitre suivant sera consacré à la phase de réalisation du projet, ainsi que l'environnement de développement, les différents éléments de structures utilisés.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

CHAPITRE III

REALISATION ET TESTS DU MODULE BACK OFFICE MOBIPOS

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

I. Introduction

Après avoir achevé la phase d'analyse de notre projet et la phase de la conception qui assure une bonne mise en œuvre d'un système fonctionnel répondant au besoin de notre client. Maintenant, notre application est prête à être codée.

Dans ce chapitre, nous avons décrire la méthode SCRUM adoptée pour ce projet. Ensuite nous avons présenté l'environnement de développement et les différents outils utilisés pour mener à terme ce travail. Par la suite, la description de notre application à l'aide des captures d'écran.

II. Méthodes de gestion de projet : SCRUM et Diagramme de Gantt

II.1. Méthodes Scrum

Durant la réalisation notre projet, Nous allons adopter la méthode agile : SCRUM comme un processus de développement logiciel afin de livrer du logiciel de qualité qui répond au besoin de l'utilisateur, dans les temps prescrits par le client. La méthode SCRUM est une méthode agile qui définit un cadre de travail permettant la réalisation de projets complexes. [13]

En première étape, Notre directrice (**Scrum Master**) effectue une première planification de l'itération (**Sprint Planning**). Ces itérations sont constituées d'un ensemble exigences appelé histoire utilisateur (**User Story**), en détaillant les différentes fonctionnalités qui seront développées. Avant chaque sprint, Notre directrice organise une réunion de planification (le sprint planning meeting). Ce planning sélectionné dans le **Product Backlog** les exigences les plus prioritaires pour le client (**Product Owner**). Elles seront développées, testées et livrées au client à la fin de cette itération (qui dure de 2 à 4 semaines suivant le nombre des user stories présentent dans le **Backlog**).

Le **Backlog** est l'ensemble des User Story à développer durant l'itération en cours. [13]

Au cours du sprint, chaque jour, il est organisé une réunion d'avancement (Daily Scrum) à 9h avec tous les membres de l'équipe afin de faire un point de coordination sur les tâches en cours et sur les difficultés rencontrées. Le tour de parole doit être

scrupuleusement respecté pour éviter que la mêlée ne dérive sur des discussions techniques et déborde la limite des 15 minutes.

La figure III.1 montre le processus de Scrum.

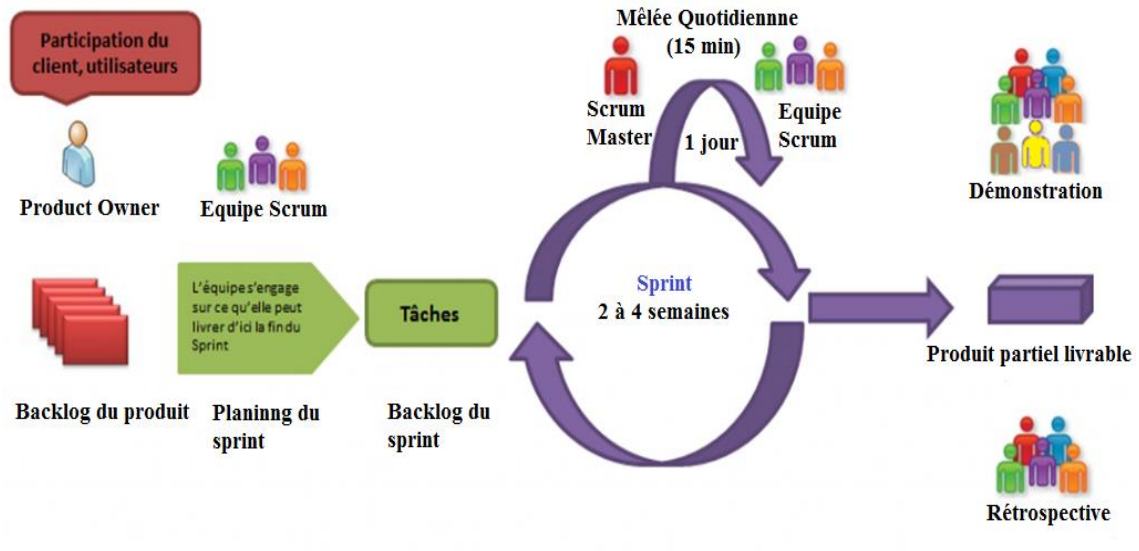


Figure III.1 Le processus de Scrum [14]

À la fin de notre stage, nous avons réussi à réaliser trois itérations, nous avons donné un nom pour chacune. Durant chaque sprint, nous avons rencontrés des bugs. Nous les avons traités par catégories (section V.). À la fin de chaque sprint, après la réalisation de toutes les fonctionnalités du sprint, nous passons à la phase test (section V).

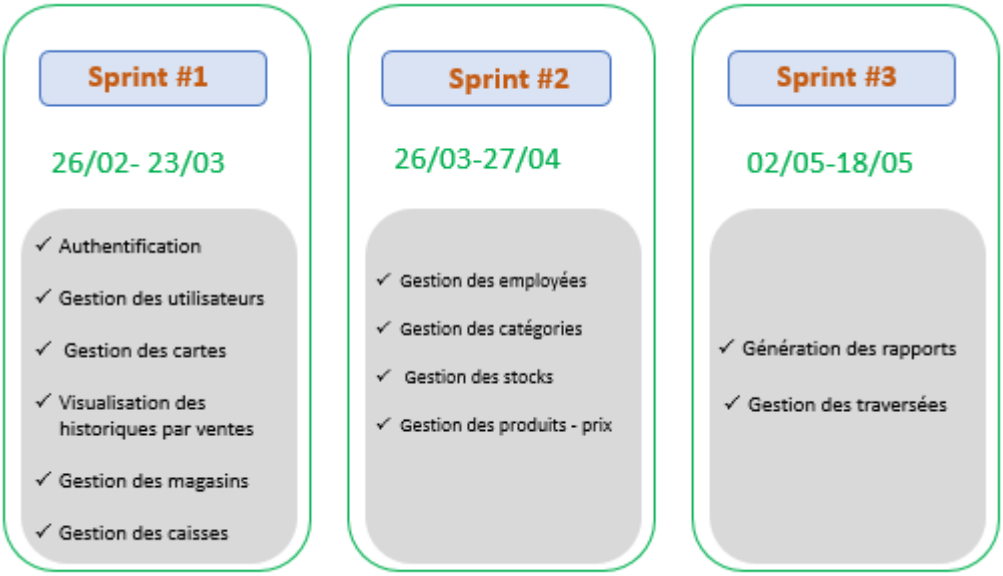


Figure III.2 Les sprints réalisés du MOBIPOS Back office

II.2. Digramme Gantt

Afin d'assurer une gestion optimale du projet et d'avoir une vue graphique sur l'avancement du projet, nous avons utilisé « le diagramme de Gantt » [15]. À l'aide de Tom's planner[4], un logiciel en ligne, permettant de modéliser sous forme d'un diagramme de Gantt la planification des différentes tâches qui sont nécessaires à un projet.

Le tableau III.1. suivant illustre nos tâches sous un diagramme qui est le diagramme de Gantt de notre projet :

[4] https://www.tomsplanner.com/

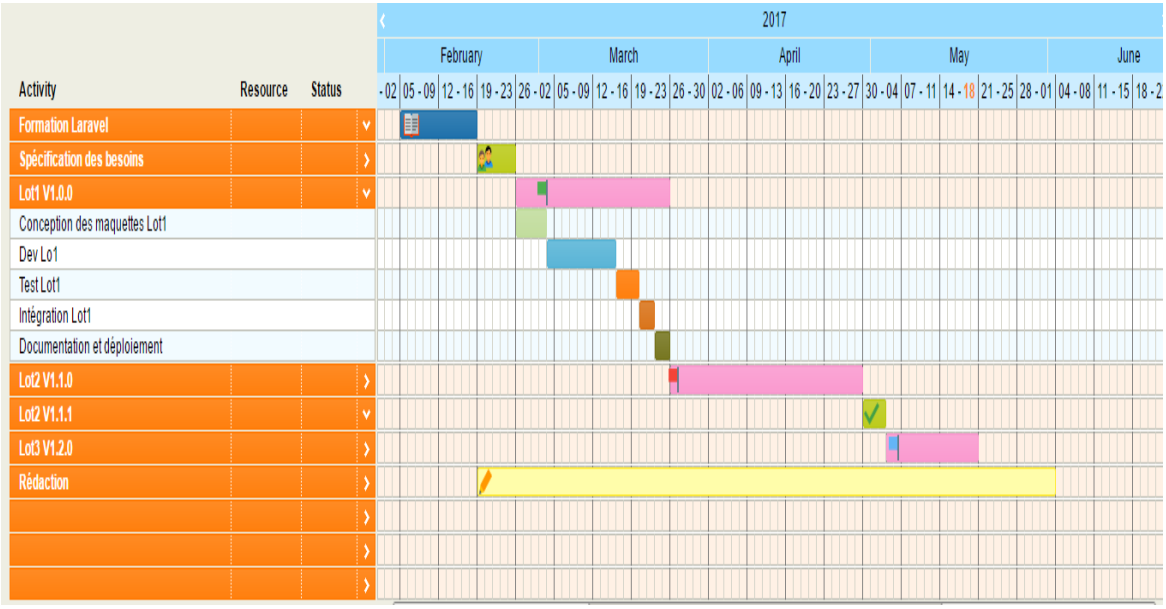


Tableau III.1. Diagramme de Gantt

III. Outils de développement

Nous décrivons les outils de développement utilisé pour réaliser ce projet.

III.1. Programmation

Pour le développement de notre application, nous avons utilisé les langages de programmation suivants : Le Framework PHP Laravel, JavaScript [16], HTML 5 [17] & CSS 3 [18] et comme environnement de travail nous avons opté pour l'utilisation de phpStorm qui est un environnement de développement intégré (IDE) développé par Jetbrains. Cet éditeur, spécialisé dans les langages web essentiellement en PHP, qui permet de développer de manière plus efficace et plus propre.

- **Le Framework Laravel**

Sachant que nous avons commencé notre stage par une formation « Laravel » de deux semaines sur ce Framework [19]. C'est ce qui nous a également permis de perfectionner nos connaissances sur le PHP objet. Par ailleurs ce Framework nous a aidé à avoir un code plus structuré, plus claire et mieux organisé.

Laravel adopte le patron MVC mais ne l'impose pas, il est totalement orienté objet.

Le principe d'une telle structure est de diviser l'application en 3 parties distinctes :

- Le modèle correspond à une table d'une base de données. C'est une classe qui étend la classeModel qui permet une gestion simple et efficace des manipulations de données et l'établissement automatisé de relations entre tables.
- Le contrôleur intercepte toutes les requêtes faites par les utilisateurs.
- La vue est soit un simple fichier avec du code HTML, soit un fichier utilisant le système de template Blade de Laravel.

La figure III.3 montre l'architecture de Laravel.

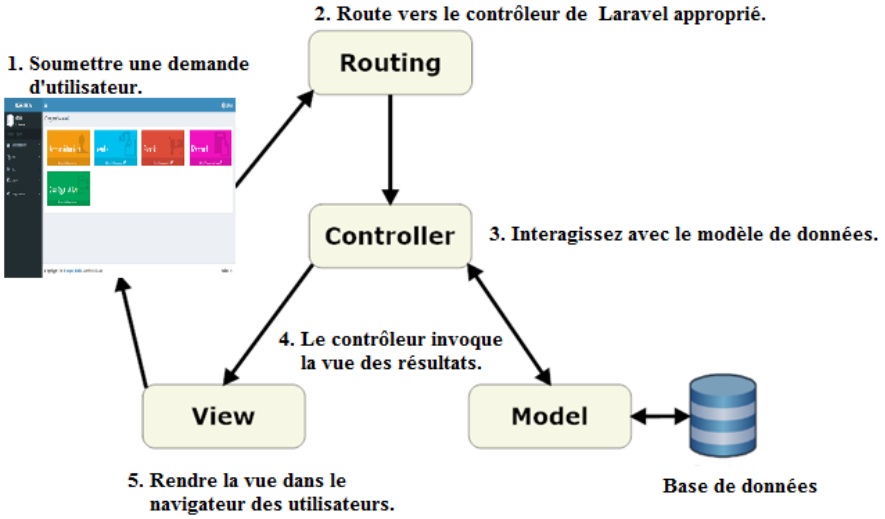


Figure III.3 Architecture de Laravel [20]

- **Le système de gestion de base de données « My SQL »**

MySQL est un système de gestion de base de données relationnelle (SGBDR). C'est un produit gratuit et Open Source constitué d'un serveur SQL qui supporte différents systèmes de stockage, outils d'administration, ainsi que de nombreuses interfaces de programmation. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde [21].

III.2. Gestion de version « Git »

Afin de faciliter le développement, le suivie et la collaboration nous utilisons un outil de gestion des versions. Cet outil s'appelle Git, un outil de gestion de versions de code source, s'est répandu très rapidement dans la communauté Open Source, d'autre part sa rapidité, sa flexibilité et sa fiabilité. L'hébergeur Git le plus connu et le plus utilisé est GitHub, qui offre de l'hébergement gratuit de projets publics, et propose une version payante pour particuliers ou entreprises, pour des projets privés. [22]

III.3. Google docs

Google Docs est un logiciel de traitement de texte développé par Google. Avec cet éditeur on peut créer, modifier et partager des documents, des présentations ainsi que des feuilles de calcul en ligne. Il permet aussi de travailler en équipe en temps réel. Nous l'utilisons pour permettre d'avoir un travail collaboratif en partageant nos documents.

IV. L'architecture de l'application

Le module back-office se présente sous la forme d'une architecture trois tiers ou ce qu'on appelle également architecture à trois niveaux est l'application du modèle plus général qu'est le multi-tiers [23]. L'architecture logique du système est divisée en trois niveaux ou couches :

- ✓ **Client** : L'ordinateur demandeur de ressources, équipé d'une interface utilisateur (navigateur web) chargé de la présentation.

Rapport-Gratuit.com

- ✓ **Serveur web -Apache** : est un serveur HTTP chargé de fournir la ressource mais faisant appel à un autre serveur, dans le serveur web se trouve le code source de notre application back-office.
- ✓ **Serveur de base de données - MySQL** : est un serveur de base de données relationnelles SQL, fournissant au serveur web les données dont il a besoin.
  Ce serveur est aussi utilisé par l'application front office, les deux module (back office et front office) ont une base de données commun.

La figure ci-dessus montre l'architecture du module back-office. (Voir la figure III.4).

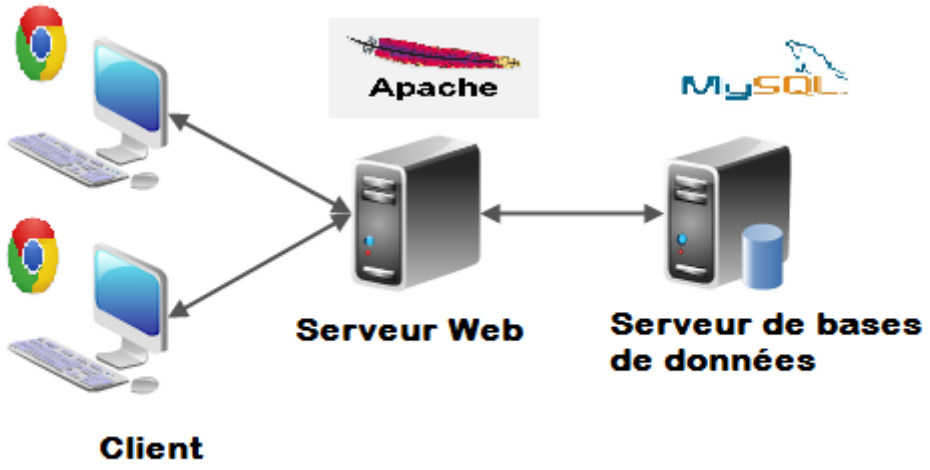


Figure III.4 Architecture du module back-office

V. Gestion des tests et anomalies

V.1. Les tests

La phase test, représente une partie importante de tout projet logiciel dont le but est d'arriver à un produit « zéro défaut ». Elle permet de détecter les problèmes logiciels ou les bugs de manière à pouvoir les corriger avant le déploiement de l'application. Elle est également réalisée pour développer la confiance et la connaissance du produit livré. Généralement 40% du budget global est consacré à l'effort de test.

Nous utilisons trois types de tests comme décrit ci-dessous.

- **Tests unitaires :**

Ces tests sont faits sur une unité de notre application. Ils présentent une procédure permettant de vérifier et tester une partie précise de notre application ou un module, indépendamment du reste du programme, ceci afin de s'assurer qu'il répond aux spécifications fonctionnelles (Backlog). Et qu'il fonctionne correctement en toutes circonstances. Dans le cadre d'utilisation de ce premier type, nous faisons nos tests par rapport à ce qui a été définie dans la conception.

- **Tests d'intégration :**

Les tests d'intégration suivent les tests unitaires. Une fois que nous validons notre partie développement et nos correctifs (autrement dit, après chaque sprint), ces tests d'intégration consistent à regrouper nos modifications. Dans le cadre d'utilisation de ce deuxième type, nous testons pour valider leur cohérence après l'intégration

- **Tests d'acceptation**

Les tests d'acceptation ou (recette) sont passés à chaque sprint. Ils vérifient que le système livré est opérationnel et fonctionne conformément aux besoins initiaux du « Product Owner ».

V.2. Les anomalies

Une application informatique a forcément quelques bugs. Ces anomalies sont mises en évidence par la phase des tests cités ci-dessus. Pour pouvoir mesurer la qualité globale de notre application, il est important d'associer à chaque déclaration d'anomalie un type précise. Pour cela nous avons classé les anomalies en trois types : 1) bloquante, 2) majeure et 3) mineure.

Le tableau ci-dessus montre quelques anomalies durant la réalisation notre application.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Type d'anomalie	Signification	Exemple
Bloquante	Le système ne peut plus être utilisé	- Impossible de modifier un produit. - La configuration de caisse ne marche pas. - Accès à la liste des cartes impossible.
Majeure	Impact réglementaire	- Des calcules fausses. - Duplication de la quantité vendue dans l'historique des ventes par carte.
Mineure	Pas d'impact réglementaire	- Des erreurs d'orthographes sur les messages de validations. - Problème de l'extension du fichier Excel. - Agrandir le tableau dans le rapport PDF.

Tableau III.2. Quelques exemples d'anomalies

Nous avons saisi les bugs d'une manière claire et précise dans Git hub [24] pour que le correcteur (développeur) puisse les comprendre, les reproduire et les corriger.

La figure ci-dessus montre quelques anomalies durant la réalisation notre application dans la plateforme Git hub (Voir la figure III.5).

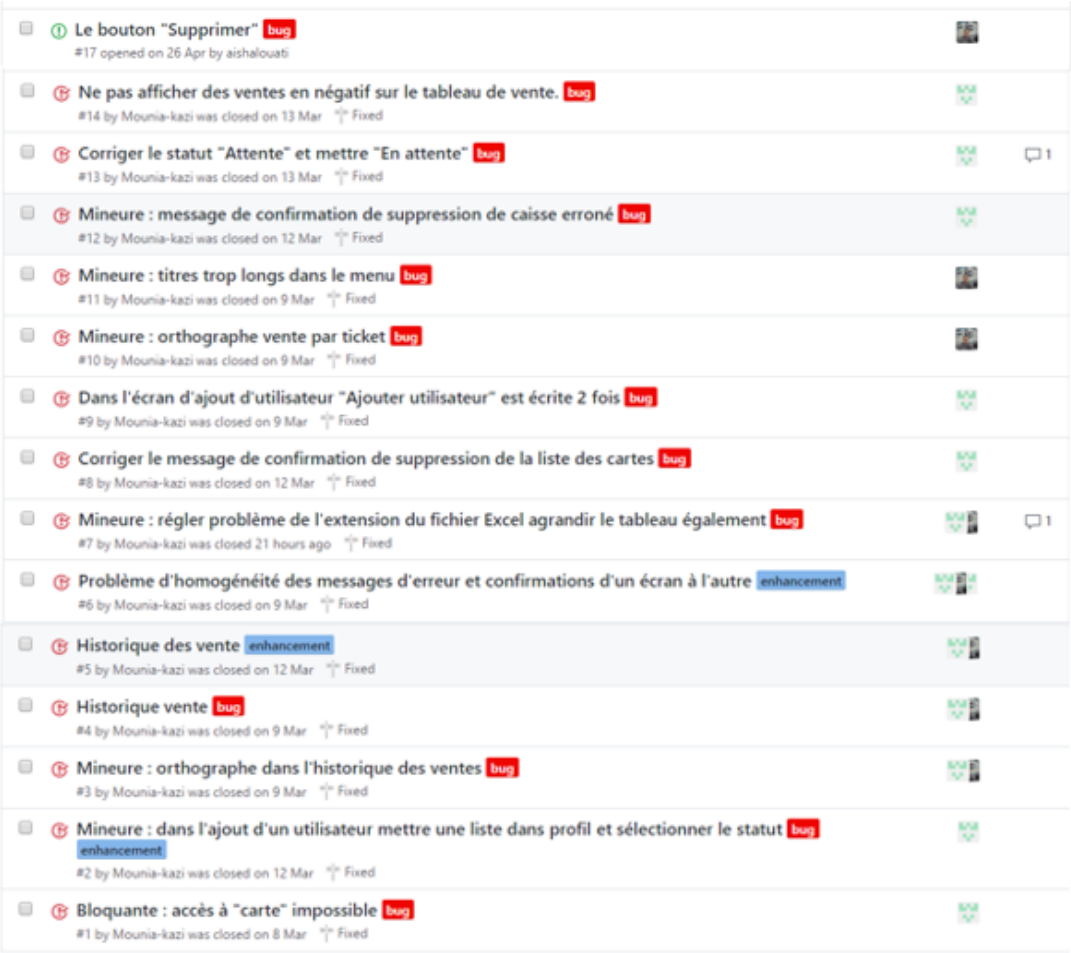


Figure III.5 Quelque anomalies de projet MOBIPOS Back office

VI. Les difficultés rencontrées

Tout au long du projet, nous avons rencontré des difficultés, parmi ces difficultés, nous pouvons citer :

- Temps pour maîtriser les outils utilisés pour réaliser le système.
- Identification des besoins du client.
- Perdre le temps de travailler sur des fonctionnalités inutiles.
- Les besoins du client ne sont pas clairs et il a du mal à formuler avec précision ce qu'il veut.

VII. Les interfaces graphiques

Nous présentons dans cette section les interfaces les plus importantes qui illustrent les différentes fonctionnalités déjà vus dans le **c**hapitre II**.**

VII.1. Authentification

Au lancement de notre application, une interface d'authentification démarre comme le la figure III.6 montre.

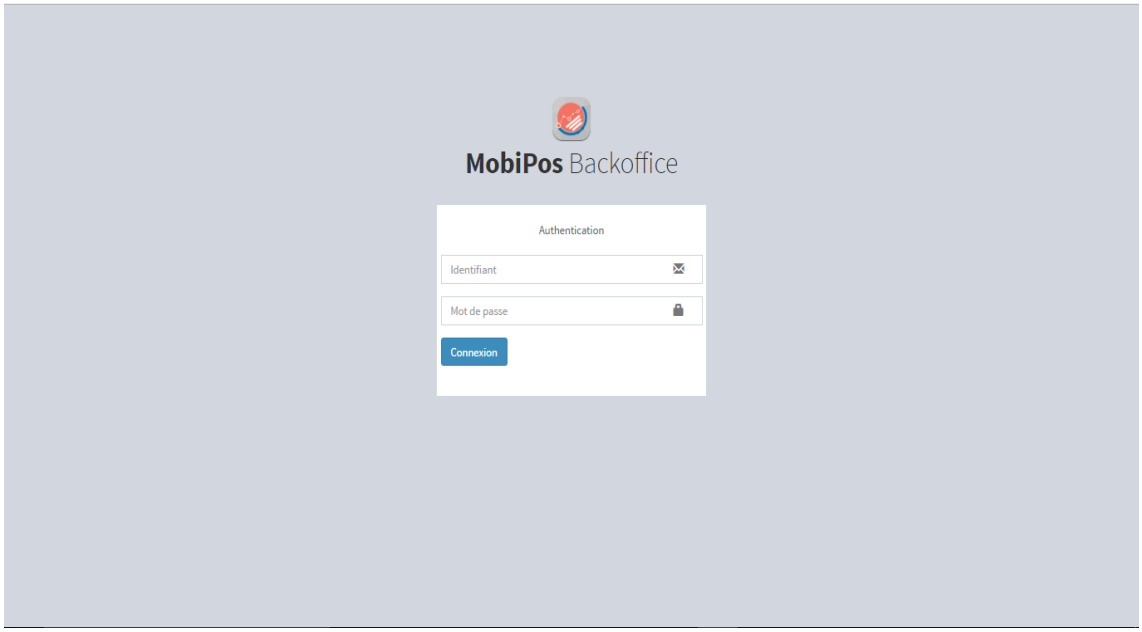


Figure III.6 Interface d'authentification

L'accès au back office est protégé par un identifiant et un mot de passe. Avec un type d'utilisateur c'est l'administrateur, ce dernier possède un accès à toutes les fonctionnalités.

VII.2. Page d'accueil

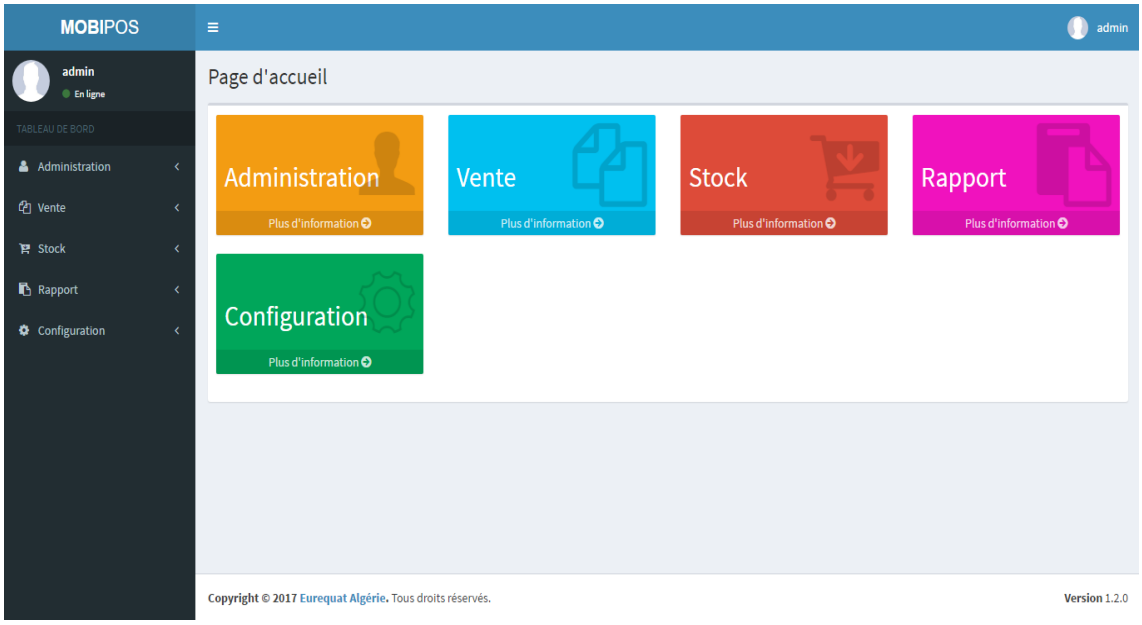


Figure III.7 Interface d'Accueil

Après l'authentification l'administrateur est rediriger directement vers le tableau de bord qui permet de visualiser toutes les menus principaux du back office (Administration, Vente, Rapports et configuration).

VII.3. Menu Administration

Ce menu contient les fonctionnalités pour administres et gérer notre système qui sont :

- Utilisateurs
- Caissiers / vendeurs
- Cartes
- Catégories
- Produits /prix
- Traversée

Nous avons choisi les sous menus les plus importants pour les montrer.

VII.3.1. Caissiers / Vendeurs

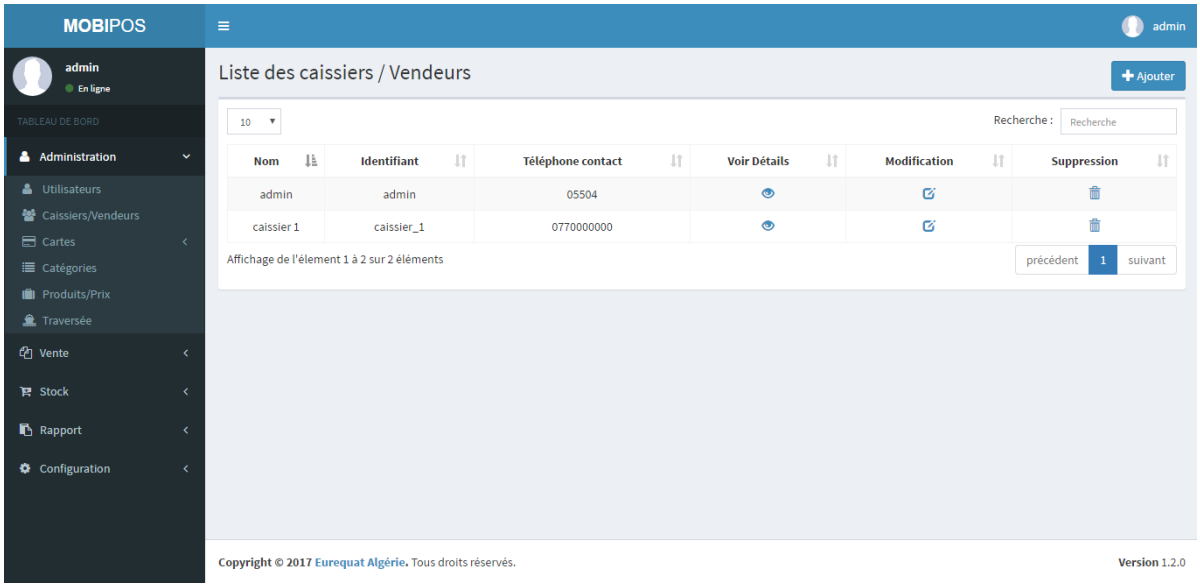


Figure III.8 Liste des caissiers

Cette interface permet à l'administrateur d'ajouter un caissier (l'utilisateur de front office) et lui affecter le privilège, consulter, modifier et supprimer.

VII.3.2. Catégories

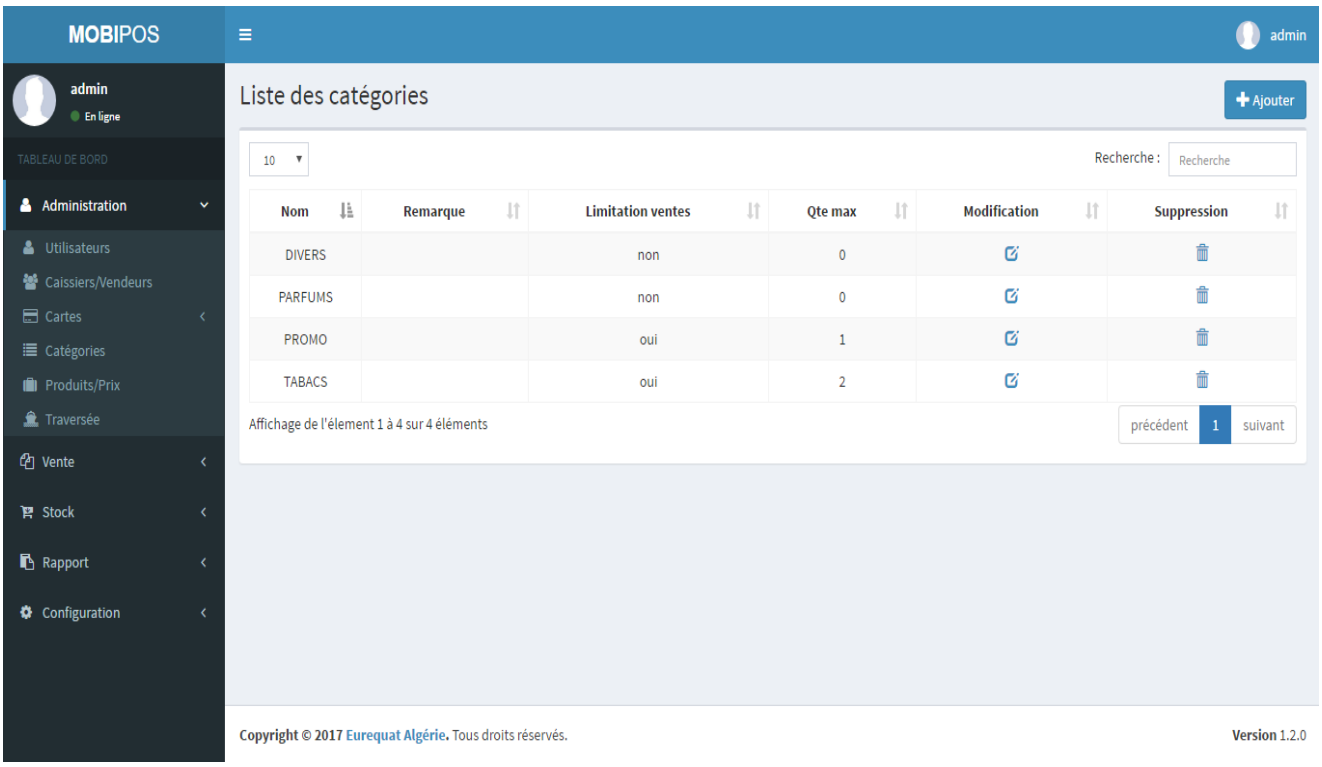


Figure III.9 Liste des catégories

Cette interface permet à l'administrateur de consulter les catégories, ajouter, modifier et supprimer.

S'il veut supprimer une catégorie mais des produits sont affectés à cette catégorie un pop-up qui s'affiche avec un message (Voir la figure III.10)

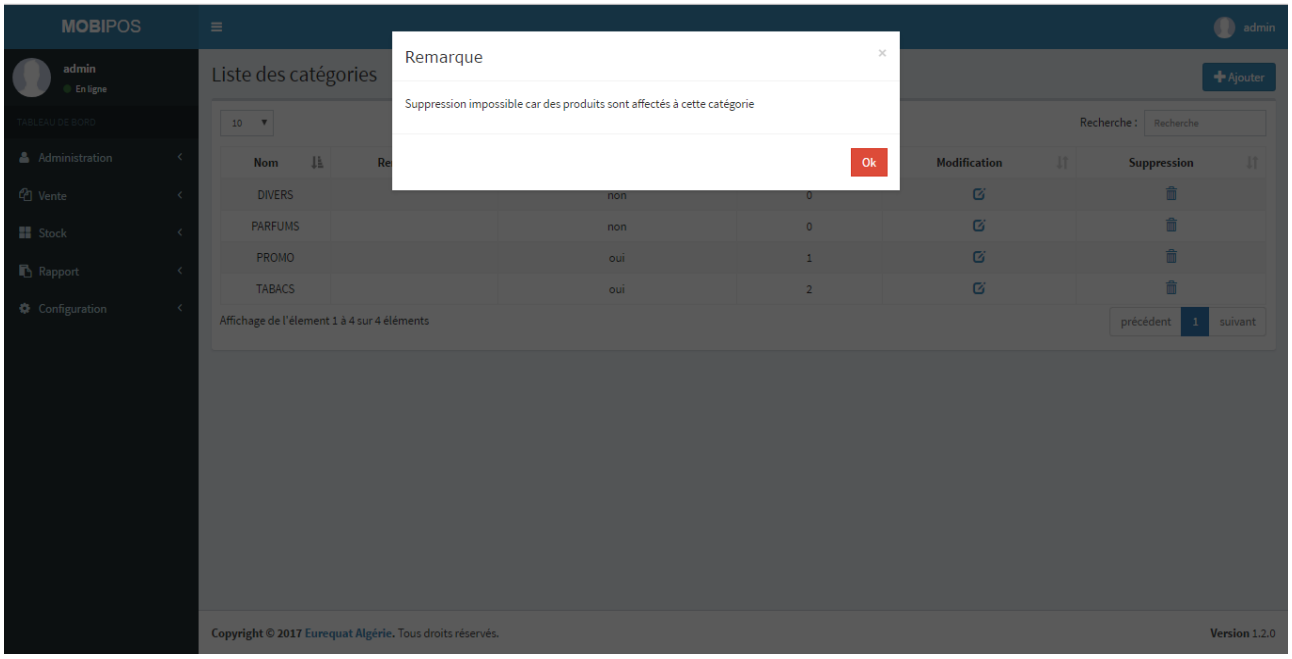


Figure III.10 Message d'avertissement

L'administrateur a la possibilité d'ajouter des catégories avec limitation des ventes ça veut dire que la quantité de vente des produits qui sont affectés à cette catégorie sont limités.

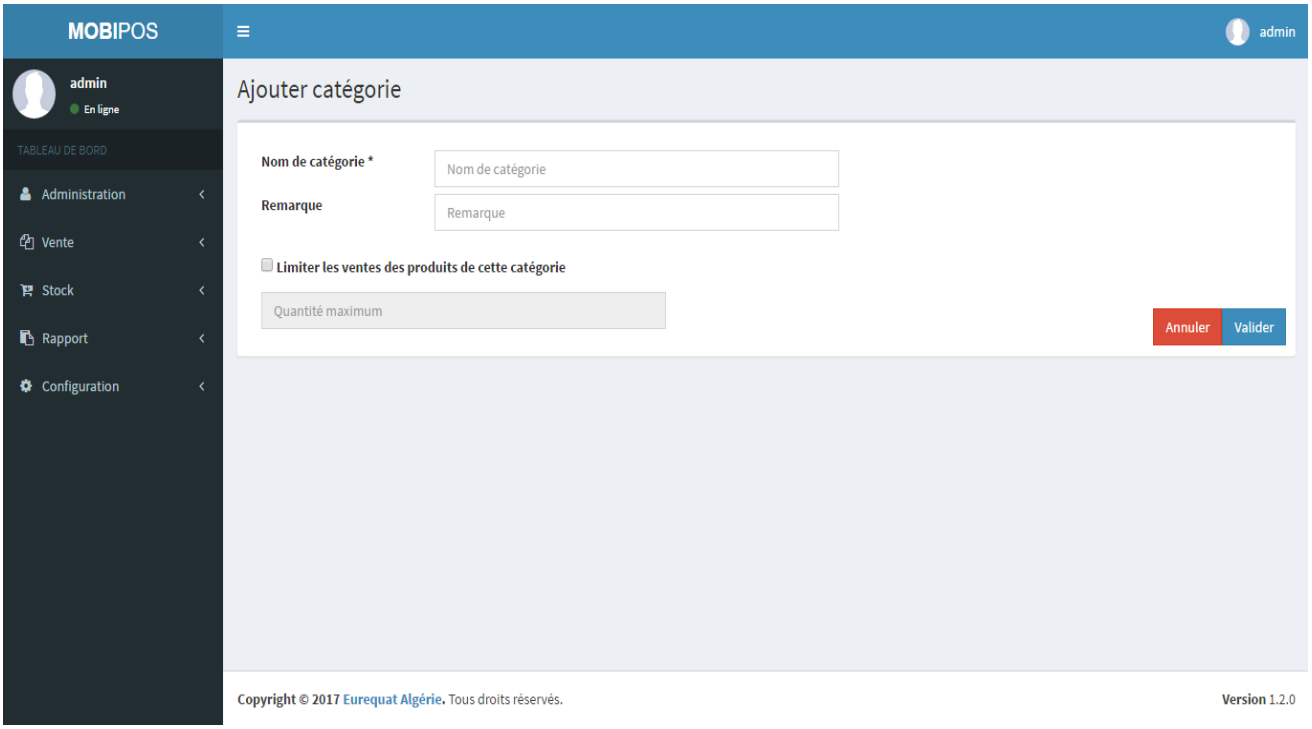


Figure III.11 L'ajout d'une catégorie

VII.3.3. Produits / Prix

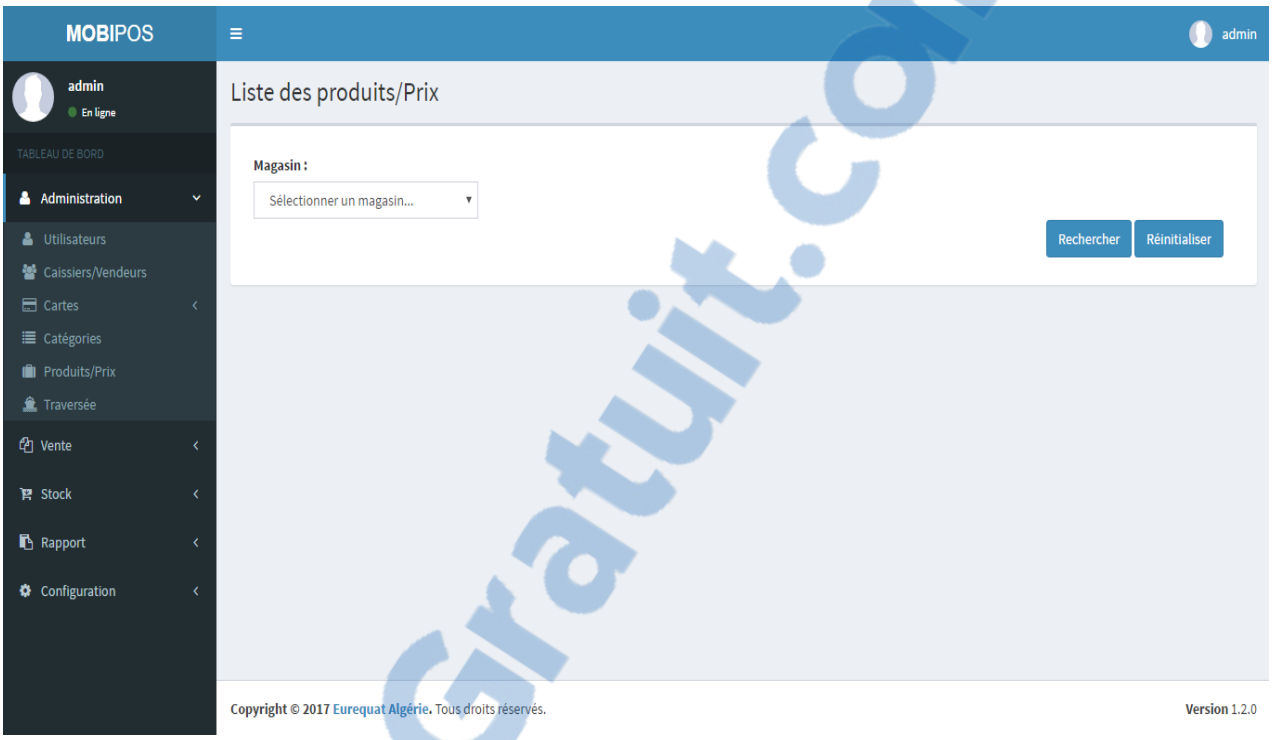


Figure III.12 Liste des produits

Cette interface permet à l'administrateur de sélectionner le magasin pour visualiser la liste des produits et leur prix correspondant au magasin sélectionné.

Rapport-Gratuit.com

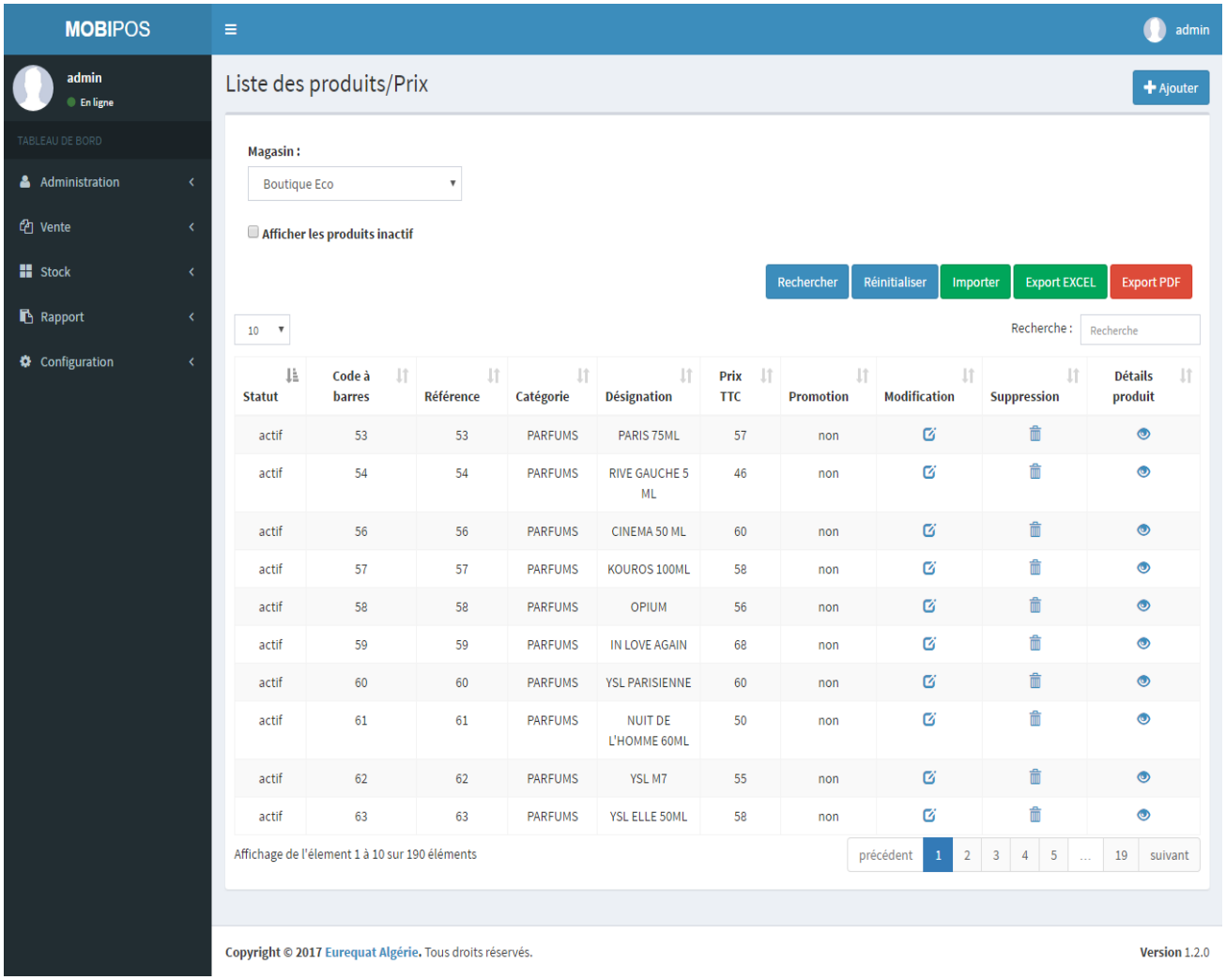


Figure III.13 Listes des produits au magasin sélectionné

Cette interface affiche la liste des produits et leur prix d'un magasin. Elle permet aussi à l'administrateur d'ajouter, modifier, supprimer, visualiser les détails, importer la liste des produits et exporter des rapports.

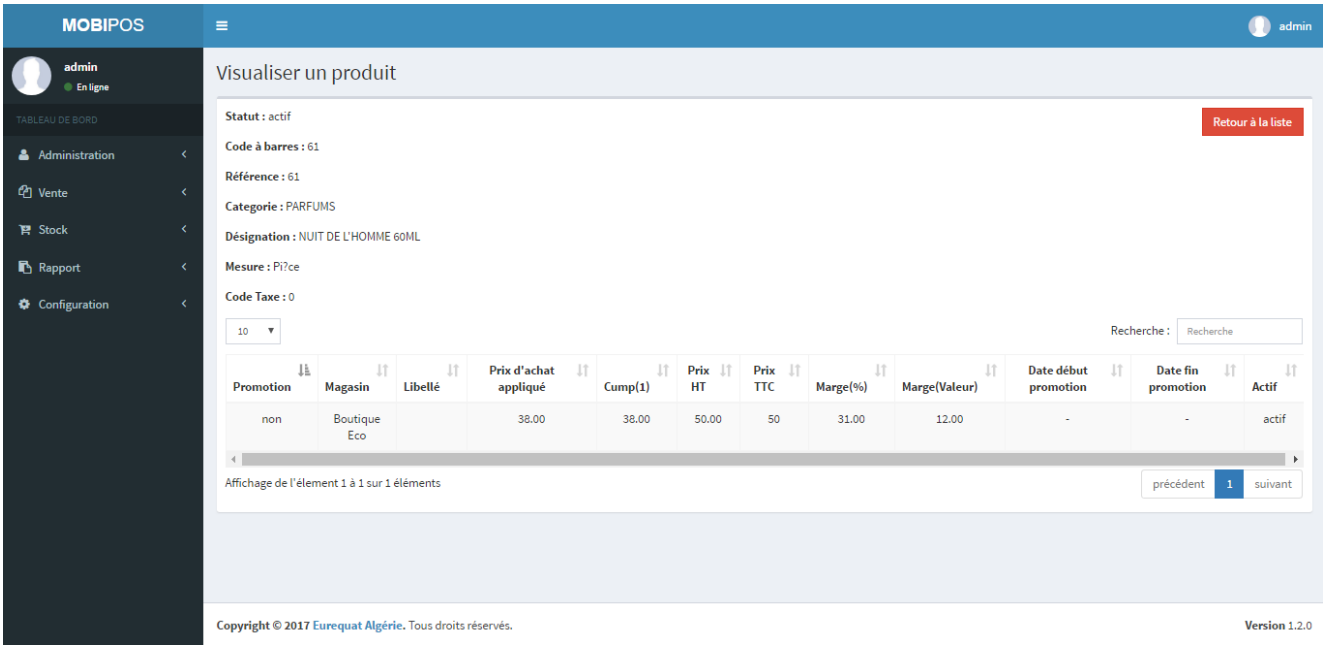


Figure III.14 Visualiser un produit

Cette interface permet de visualiser un produit et ses prix.

VII.4. Menu vente

Ce menu permet de visualiser l'historique des ventes regroupé par produit, par ticket et par carte.

Nous avons choisi le sous menus l'historique des ventes regroupé par carte pour le montrer.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

VII.4.1. Vente par carte

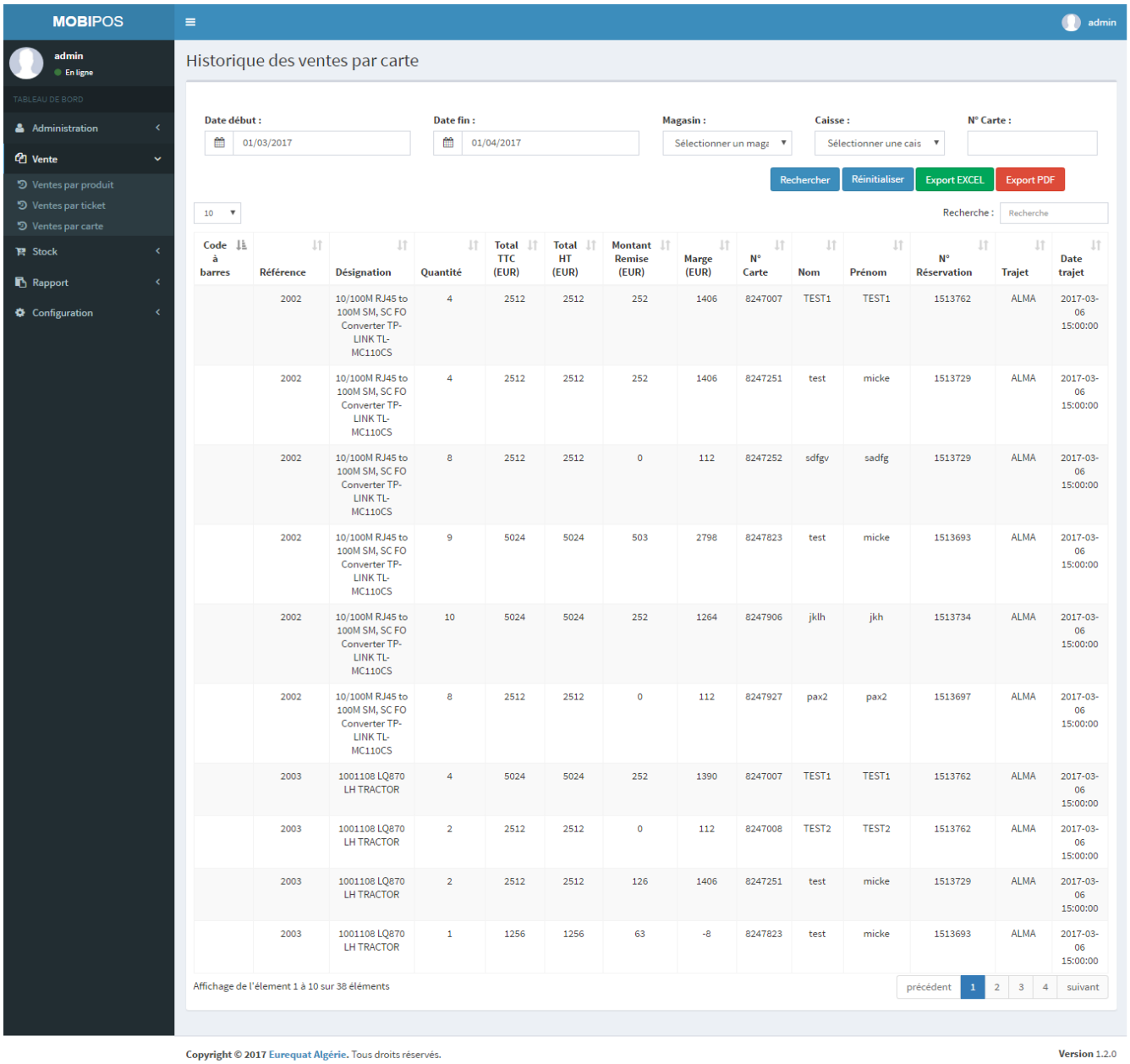


Figure III.16 Historique des ventes par carte

Cette interface permet de visualiser l'historique des ventes regrouper par carte d'embarquement, avec la possibilité de faire un filtre par date, par magasin avec ses caisses et par numéro de carte d'embarquement.

VII.5. Menu stock

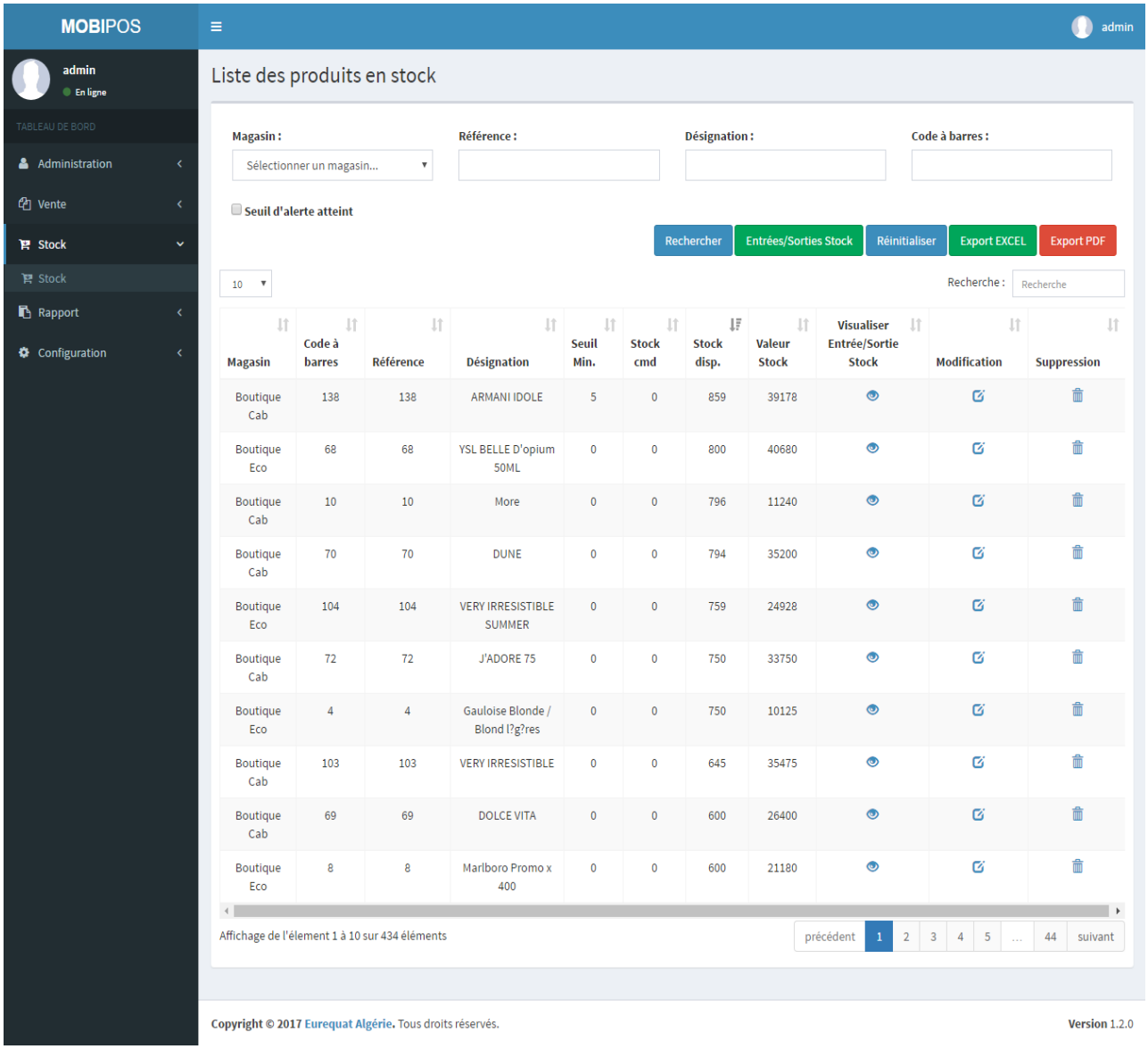


Figure III.16 Listes des produits en stock

Cette interface permet à l'administrateur de modifier, supprimer et visualiser la liste des produits en stock et la liste des entrées /sorties.

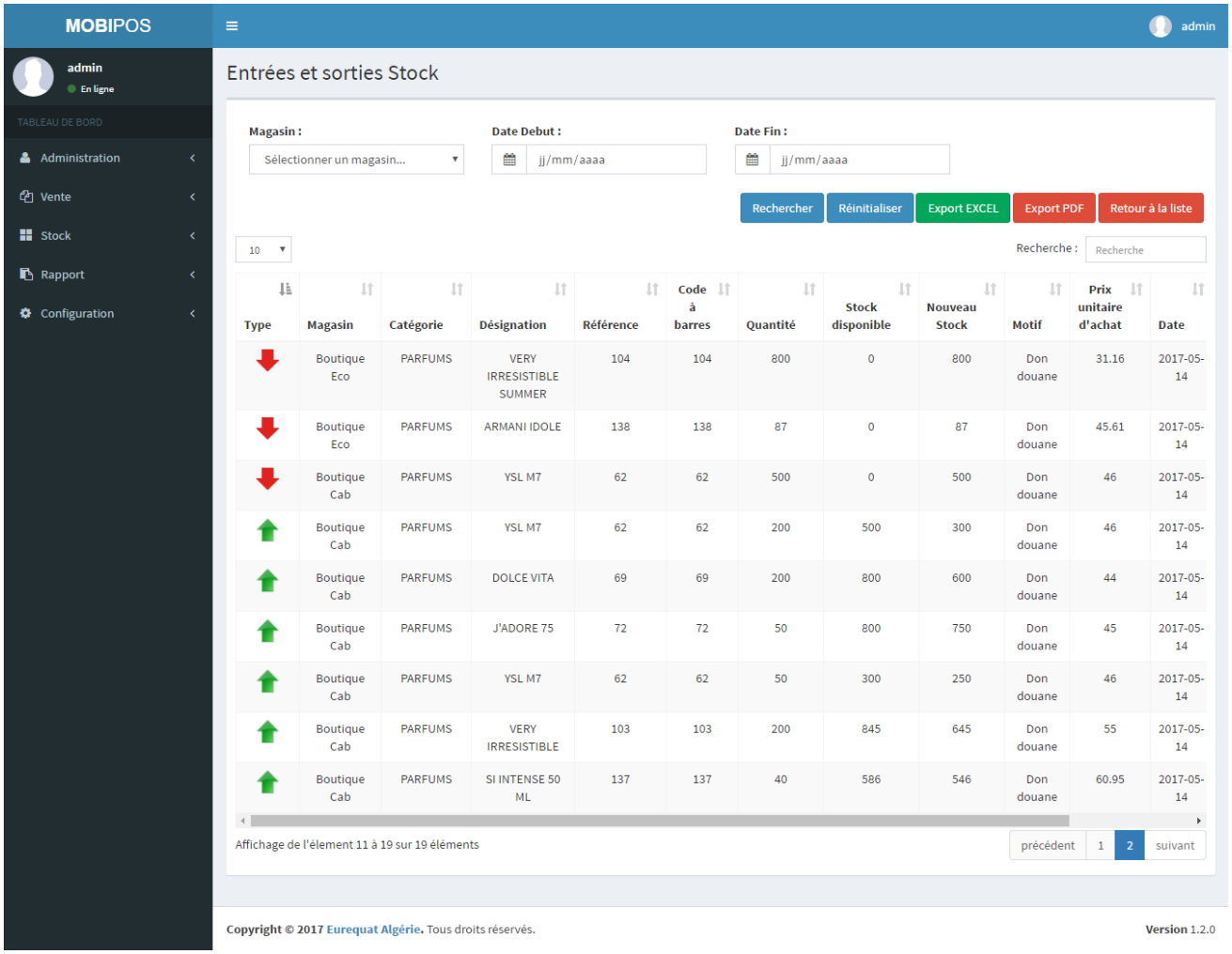


Figure III.17 Entrées/sorties en stock

A partir de cette l'interface (figure III.17) l'administrateur peut visualiser la liste des entrées/sorties en stock. Il aussi a la possibilité de filtrer par magasin, par date et exporter des rapports associés.

VII.6. Menu rapport

Ce menu contient des rapports qui aide l'administrateur à bien contrôler ces magasins qui sont :

- Sorties de stocks
- Etat des mouvements de stocks
- Historique du stock

Nous avons choisi les sous menus les plus importants pour les montrer.

Rapport-gratuit.com

VII.6.1 Etat des mouvements de stocks

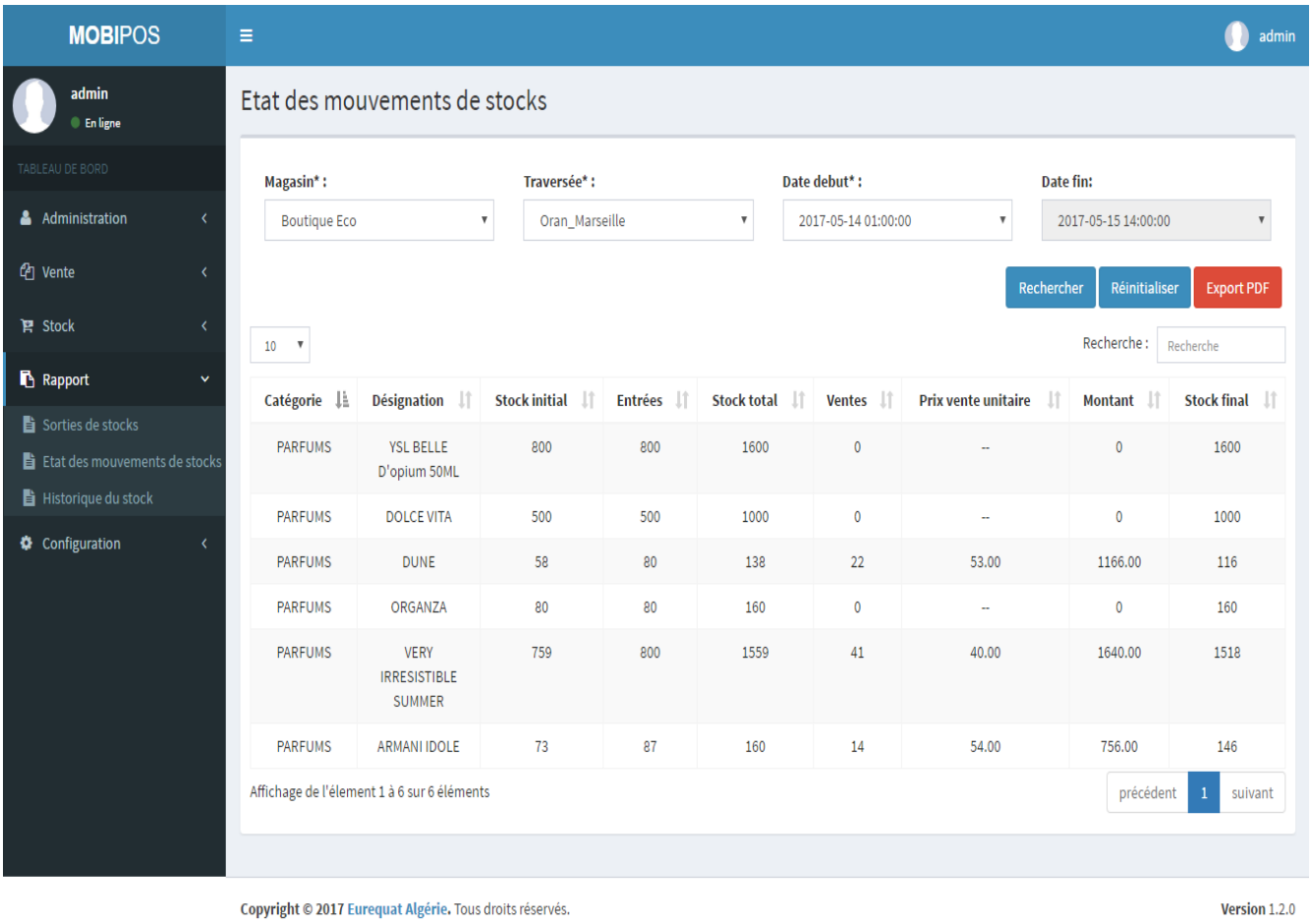
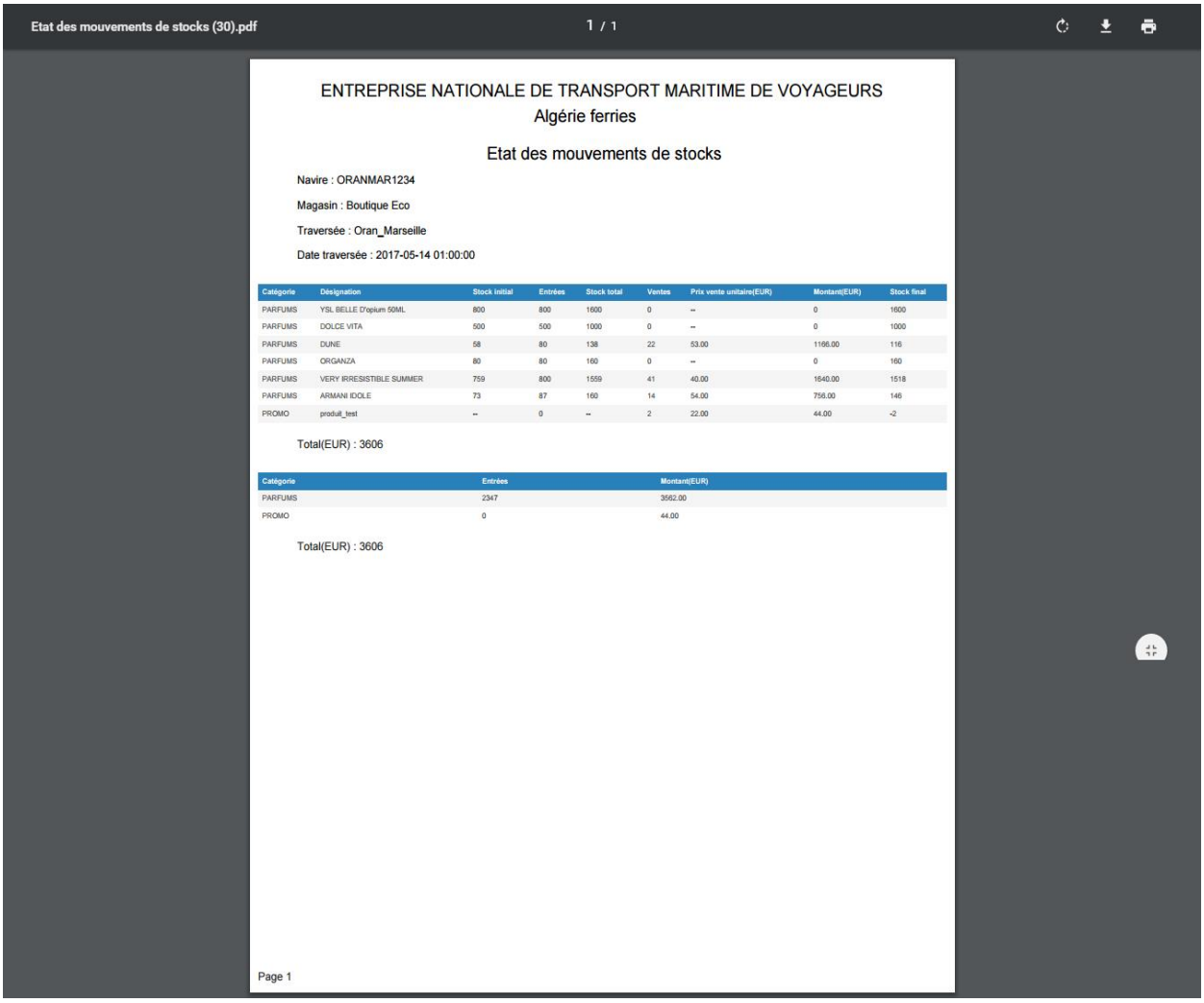


Figure III.18 Rapport de l'état des mouvements de stocks

Cette interface permet de visualiser la liste des produits vendu par rapport à les entrées de stock dans une traversée et un magasin, si l'administrateur ne fait pas le filtre, il ne peut pas visualiser ce rapport.

L'administrateur a la possibilité d'exporter ce rapport en PDF, la figure III.19 montre un exemple d'un fichier exporter en PDF.



Etat des mouvements de stocks (30).pdf 1 / 1

ENTREPRISE NATIONALE DE TRANSPORT MARITIME DE VOYAGEURS
Algérie ferries

Etat des mouvements de stocks

Navire : ORANMAR1234

Magasin : Boutique Eco

Traversée : Oran_Marseille

Date traversée : 2017-05-14 01:00:00

Catégorie	Désignation	Stock initial	Entrées	Stock total	Ventes	Prix vente unitaire(EUR)	Montant(EUR)	Stock final
PARFUMS	YSL BELLE D'opium 50ML	800	800	1600	0	--	0	1600
PARFUMS	DOLCE VITA	500	500	1000	0	--	0	1000
PARFUMS	DUNE	58	80	138	22	53.00	1166.00	116
PARFUMS	ORGANZA	80	80	160	0	--	0	160
PARFUMS	VERY IRRESISTIBLE SUMMER	759	800	1559	41	40.00	1640.00	1518
PARFUMS	ARMANI IDOLE	73	87	160	14	54.00	756.00	146
PROMO	produit_test	--	0	--	2	22.00	44.00	-2

Total(EUR) : 3606

Catégorie	Entrées	Montant(EUR)
PARFUMS	2347	3562.00
PROMO	0	44.00

Total(EUR) : 3606

Page 1

Figure III.19 Rapport sous format PDF

VII.6.2 Historique du stock

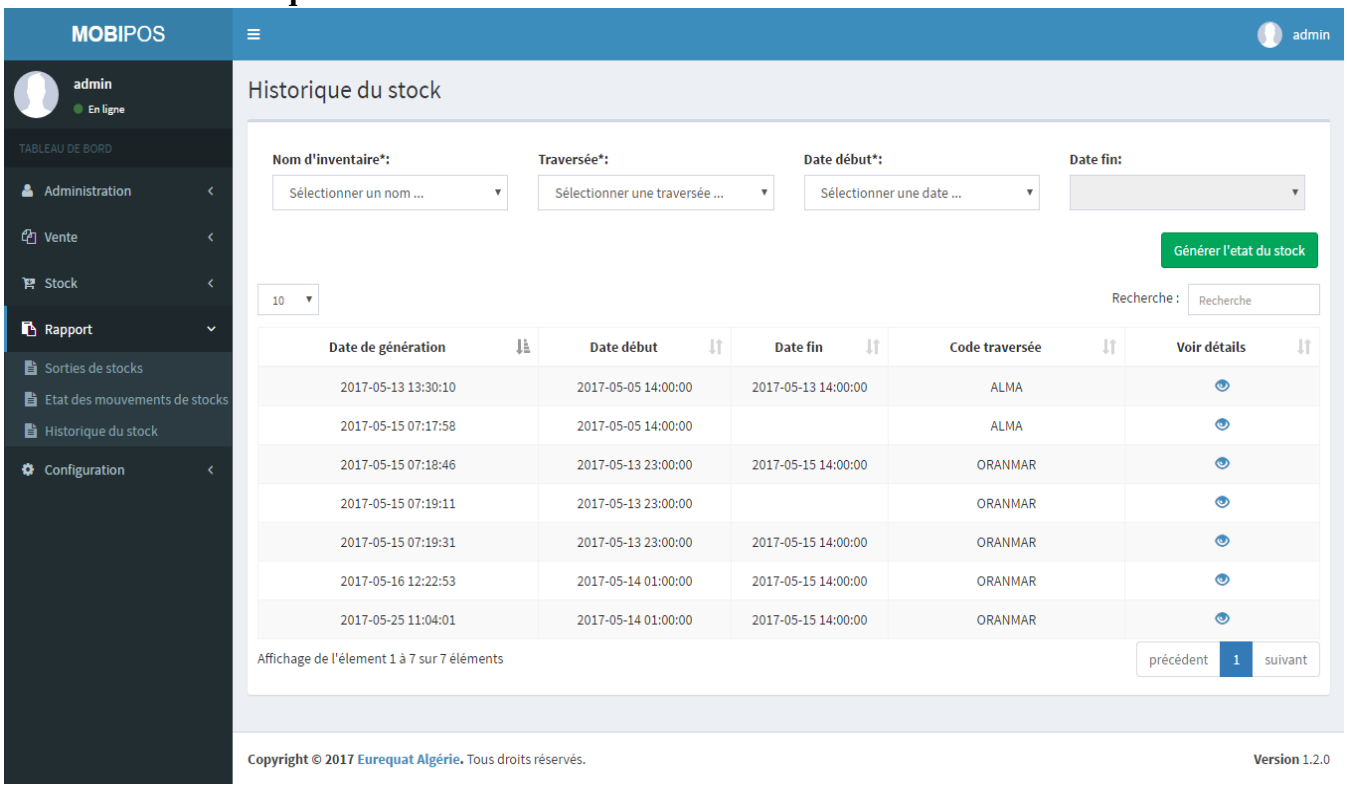


Figure III.20 La liste des inventaires générés

Cette interface permet de visualiser la liste des inventaires générés à partir de sélectionner le type d'inventaire et le nom de traversée et ses dates.

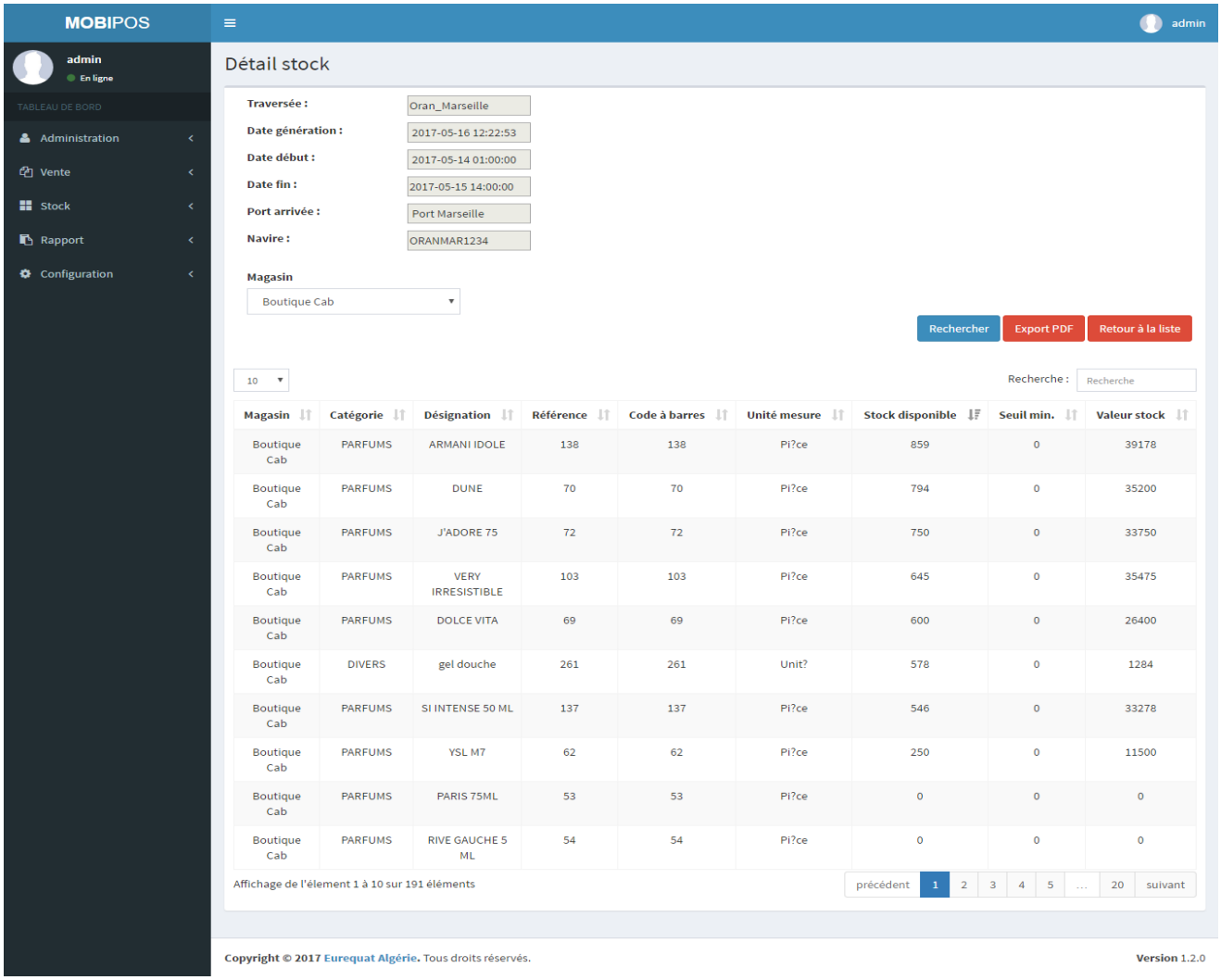


Figure III.21 Rapport de l'historique du stock

Cette interface permet à l'administrateur de visualiser le détail du stock par rapport l'inventaire sélectionnée. Il a aussi la possibilité d'exporter ce rapport sous format PDF.

VII.7. Menu configuration

Ce menu contient les fonctionnalités pour configurer les traversées, magasin et ses caisses.

VII.7.1 Caisse

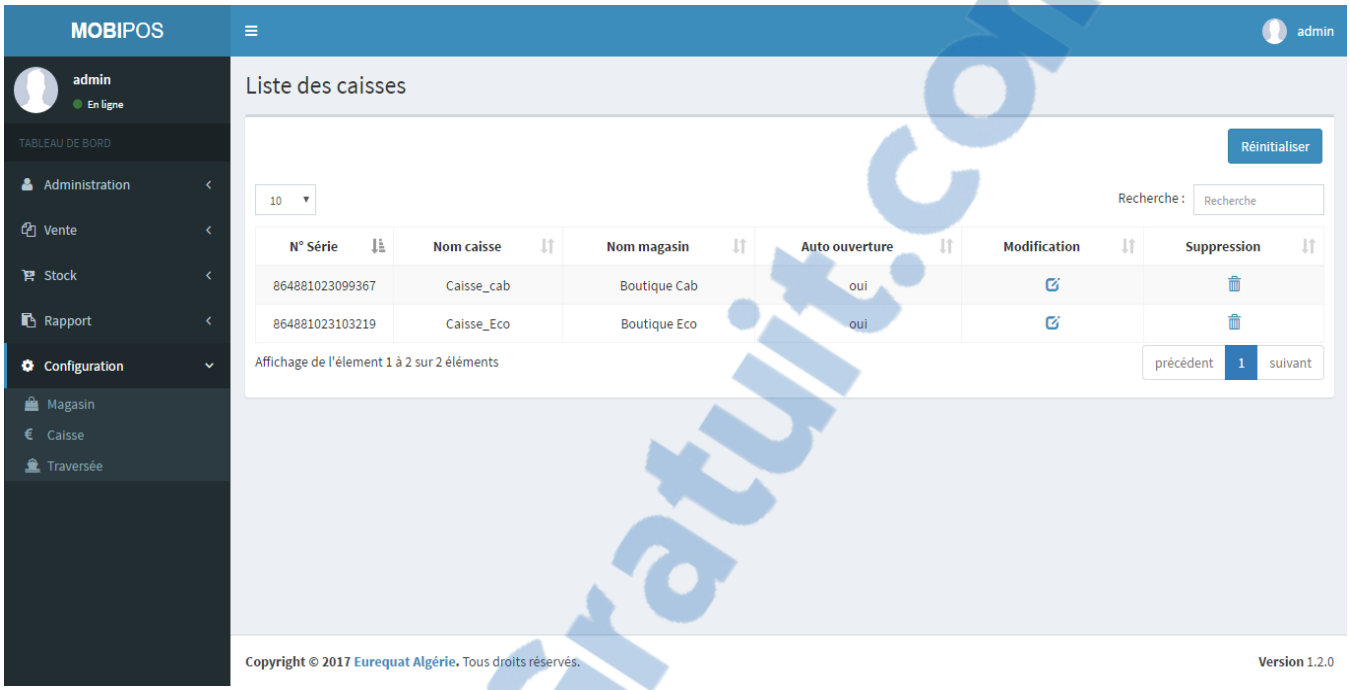


Figure III.22 La liste des caisses

Cette interface permet à l'administrateur de visualiser la liste des caisses. Il a la possibilité de supprimer, modifier et affecte chaque caisse à un magasin (voir figure III.23).

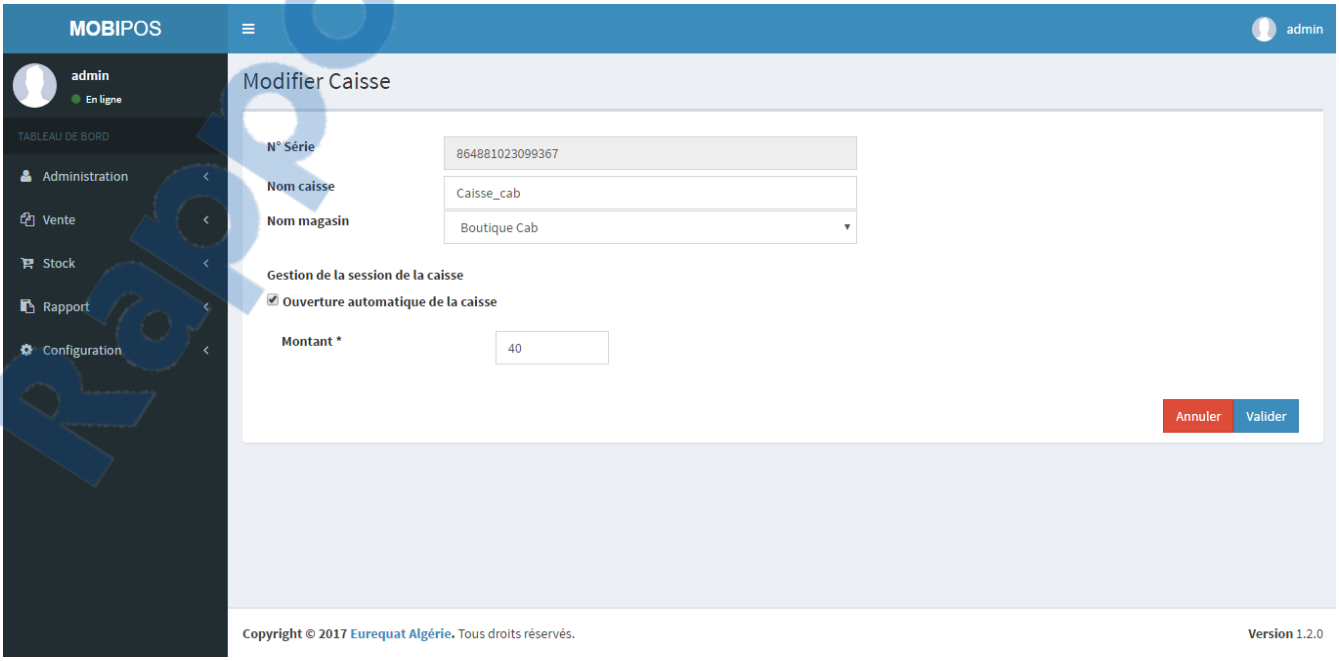


Figure III.23 Modifier une caisse

Rapport-Gratuit.com

VIII. Conclusion

A travers ce dernier chapitre, nous avons décrit l'architecture de l'application et les outils de développement utilisé pour la réalisation de notre. Ensuite, nous avons illustré quelques fonctionnalités de base de l'application à travers un ensemble de captures d'écran.

CONCLUSION GENERAL

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Nous avons réalisé dans ce projet de fin d'étude le système Back office MOBIPOS d'un système terminal point de vente. Ce dernier a été effectué lors de notre stage qui s'est déroulé au sein de l'entreprise « Eurequat Algérie ». L'objectif principal était de concevoir et développer une application back-office qui permettra à un administrateur de contrôler et gérer ses magasins, visualiser des rapports des ventes et gestion des stocks.

L'application a été implémenté par diverses technologies en se basant sur l'étude conceptuelle. Le système de base de données choisi c'est MYSQL. L'application a été implémenter avec Laravel 5 qui est un framework PHP, javascript pour la programmation des scriptes, HTML5 et CSS3 pour la mise en page et design de notre projet.

Ce stage a été sous plusieurs aspects riches d'enseignements, c'est une expérience intéressante, qui nous a permis de découvrir le monde professionnel et de mettre en pratique ce que nous avons acquis durant notre cursus et d'améliorer nos connaissances et nos compétences sur le plan technique.

Malgré toutes les difficultés rencontrées au niveau d'analyse des besoins du client et les contraintes de temps, nous avons réussi à réaliser la totalité de notre application tout en respectant les objectifs fixés au début du stage et répondre aux besoins du client.

Finalement, notre travail ne s'arrête pas à ce niveau, en effet plusieurs fonctionnalités peuvent être ajoutées notamment la gestion des commandes et fournisseur, et améliorer le back-office pour une version standard qui est adaptée à tous les clients et pas pour un client spécifique comme notre cas.

Références Bibliographiques

[1] « Eurequat Algerie ». [En ligne]. Disponible sur : http://www.eurequat-algerie.com/aProposDeNous.php.Consulté le 26 février 2017.

[2] « Terminal point de vente », Définitions marketing. [En ligne]. Disponible sur: http://www.definitions-marketing.com/definition/terminal-point-de-vente/.Consulté le 05 mars 2017.

[3] « Application de front office », Wikipédia. 20-juill-2016.Consulté le 05 mars 2017.

[4] « Interface d'administration ou Back Office ». [En ligne]. Disponible sur: http://www.morphemzero.be/lexique/interface-administration/.Consulté le 05 mars 2017.

[5] L. M. SAS, « RoverCash - Caisse tactile enregistreuse sur tablette Apple et Androïd ». [En ligne]. Disponible sur: http://www.rovercash.fr.Consulté le 15 mars 2017.

[6] L. M. SAS, « Logiciel de gestion commerciale - ERP - CRM by LUNDI MATIN ». [En ligne]. Disponible sur: https://www.lundimatin.fr/tarifs/modules-optionnels/commerce-connecte/caisse-tactile.php. Consulté le 15 mars 2017.

[7] « Phasis, le logiciel de caisse tactile tous commerces ». [En ligne]. Disponible sur: http://www.crisalid.com/materiel-caisse-enregistreuse/logiciel-encaissement/logiciel_caisse_enregistreuse_tous_commerces.Consulté le 15 mars 2017.

[8] « Logiciel d'encaissement et de gestion Marque NF 525, caisses enregistreuses, terminaux points de vente, périphériques ». [En ligne]. Disponible sur: http://www.aemsofts.com/.Consulté le 15 mars 2017.

[9] F. Juliard UML Unified Method Language, Journal Université de Bretagne Sud UFR SSI-IUP Vannes, 2001-2002.

[10] Sparx System Ltd and SparxSystems Software Gmbh, Entreprise Architect [en ligne].publier en 2016, Disponible sur <http://www.sparxsystems.eu/start/home/> (traduit). Consulté le 05 avril 2017.

[11] P. ROCK, F. VALLE. UML 2 en action de l'analyse des besoins à la conception. EYROLLES, 4ème édition, 2007

[12] Cian.Developpez.com, Diagramme de séquence [Cours en ligne], publier le 03/12/03, Disponible sur <http://cian.developpez.com/uml2/tutoriel/sequence/> (traduit). Consulté le 16 avril 2017.

[13] Benoit Feron,Découvrez la méthode SCRUM , 14 novembre 2014. [En ligne]. Disponible sur : http://www.dynamique-mag.com/article/methode-scrum-definition.4749. Consulté le 17 avril 2017.

[14] DCUBE : Scrum vs cycle en v, avril 2014. http://blog.dcube.fr/blog/2014/04/28/scrum-vs-cycle-en-v/

[15] Gregory Mcbride. The Gantt Chart Handbook – Everything you need to knew about Gantt chart . Emereo Publishing.29 avril 2016.46p.ISBN 1489120416

[16]« JavaScript »,Wikipedia.[Enligne].Disponible sur:https://fr.wikipedia.org/wiki/JavaScript.Consulté le 14 mai 2017.

[17]« HTML5 », Wikipédia. [En ligne].Disponible sur: https://fr.wikipedia.org/wiki/HTML5.Consulté le 14 mai 2017.

[18]« Feuilles de style en cascade », Wikipédia. [En ligne]. Disponible sur: https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade.Consulté le 14 mai 2017.

[19]« Laravel - The PHP Framework For Web Artisans ». [En ligne]. Disponible sur: https://laravel.com/. Consulté le 14 mai 2017.

[20] Architecture of Laravel Applications.[En ligne]. Disponible sur: http://laravelbook.com/laravel-architecture/ .Consulté le 25 mai 2017.

[21]« MySQL », Wikipédia. [En ligne]. Disponible sur: https://fr.wikipedia.org/wiki/MySQL.Consulté le 25 mai 2017.

[22]« Git ». [En ligne]. Disponible sur: https://git-scm.com/.Consulté le 25 mai 2017.

[23]« Architecture trois tiers : définition et explications ». [En ligne]. Disponible sur : http://www.techno-science.net/?onglet=glossaire&definition=5266.Consulté le 25 mai 2017.

[24]« What is GitHub? - Definition from WhatIs.com », SearchITOperations. [En ligne]. Disponible sur: http://searchitoperations.techtarget.com/definition/GitHub. Consulté le 25 mai 2017.

Annexes A :

Manual d'utilisation du back office MOBIPOS V1.0.0(Sprint #1)

Version : V1.0.0

Statut : Validé

Usage : Livrable

SOMMAIRE

1.INTRODUCTION **72**

2.UTILISATION DU BACK-OFFICE MOBIPOS **72**

2.1 Accès au back-office du Mobipos 72
2.2 Administration 74
2.2.1 Utilisateurs 75
2.2.2 Cartes 75
2.3 Vente 76
2.3.1 Historique des ventes par produit 77
2.3.2 Historique des ventes par ticket 77
2.3.3 Historique des ventes par carte 78
2.3 Configuration **Erreur ! Signet non défini.**
2.3.1 Magasin 79
2.3.2 Caisse 81

1.Introduction

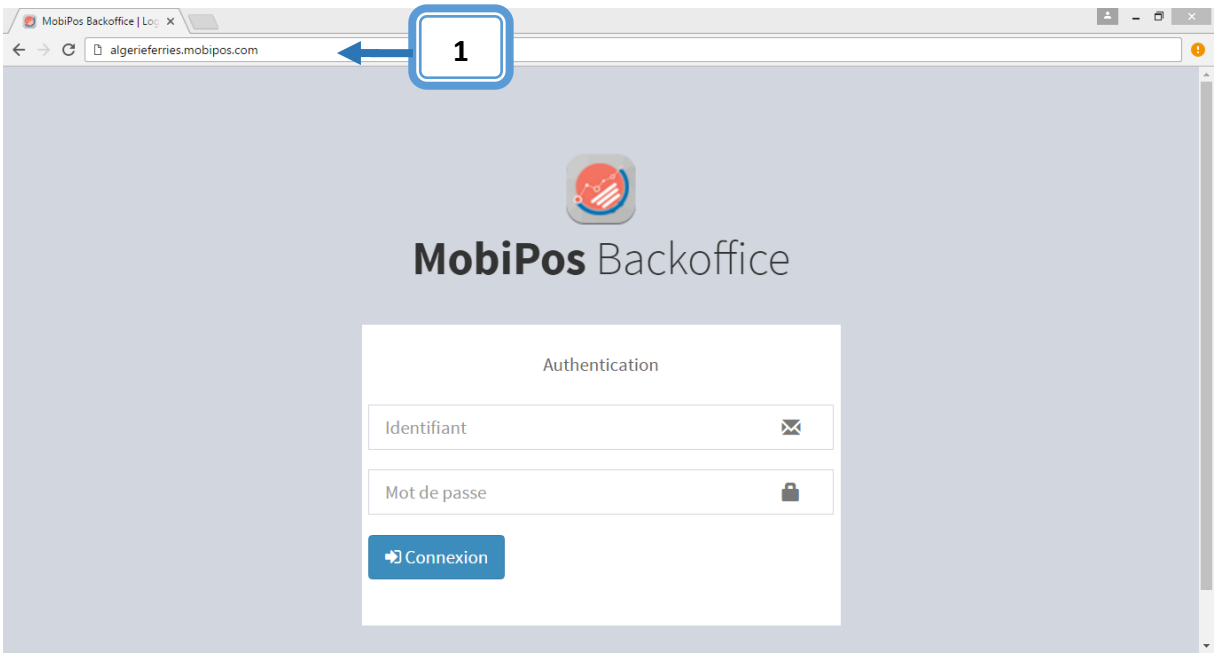
Le back-office de Mobipos est une application web qui assure le pilotage de l'application Mobipos. Elle contient les modules suivants :

- Administration
- Vente
- Configuration

Il est recommandé d'utiliser le navigateur Google Chrome, ainsi qu'activer les pop-ups, pour cela il faut cliquer sur Paramètres => Paramètres avancés => Confidentialité => Paramètre de contenu => Fenêtres pop-up => Choisir "Autoriser tous les sites à afficher des fenêtres pop-up "=> Ok.

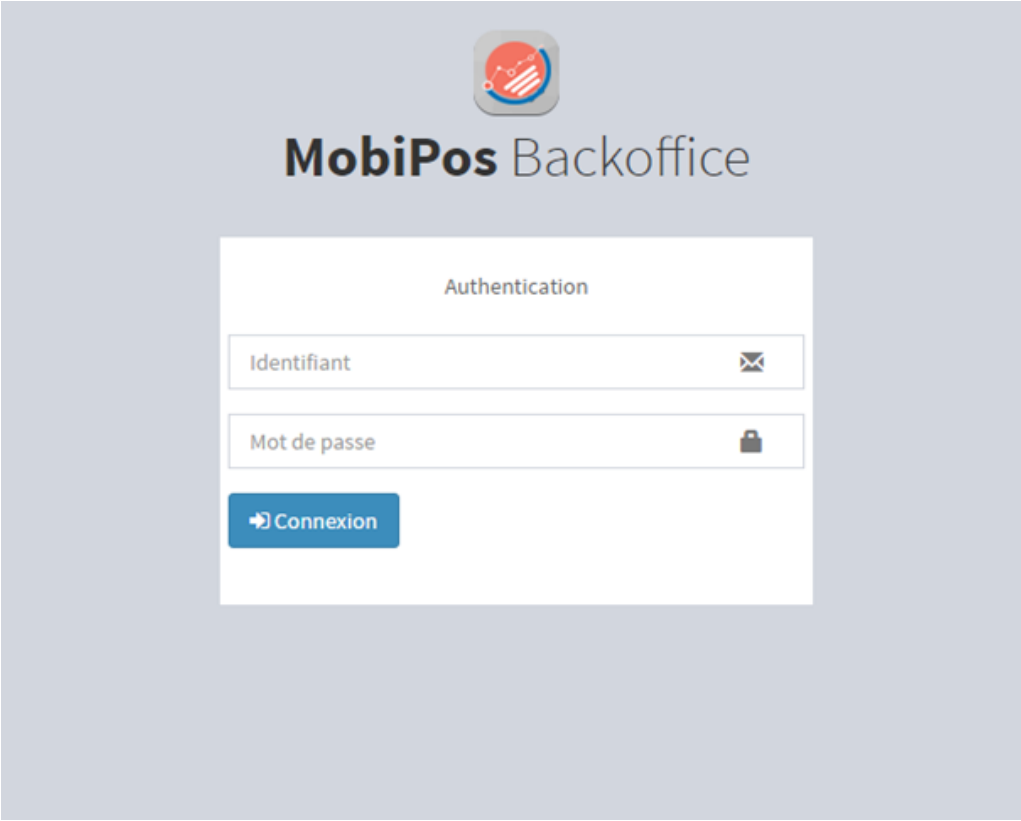
2.Utilisation du Back-office MOBIPOS

2.1 Accès au back-office du MOBIPOS



Url du Back-office Mobipos

En utilisant le navigateur "*Google chrome*", taper l'URL" *algerieferries.mobipos.com*" **1**, l'interface d'authentification apparaîtra. L'utilisateur doit saisir son 'Identifiant' et son 'Mot de passe' puis cliquer sur le bouton 'Connexion' pour se connecter à l'application.

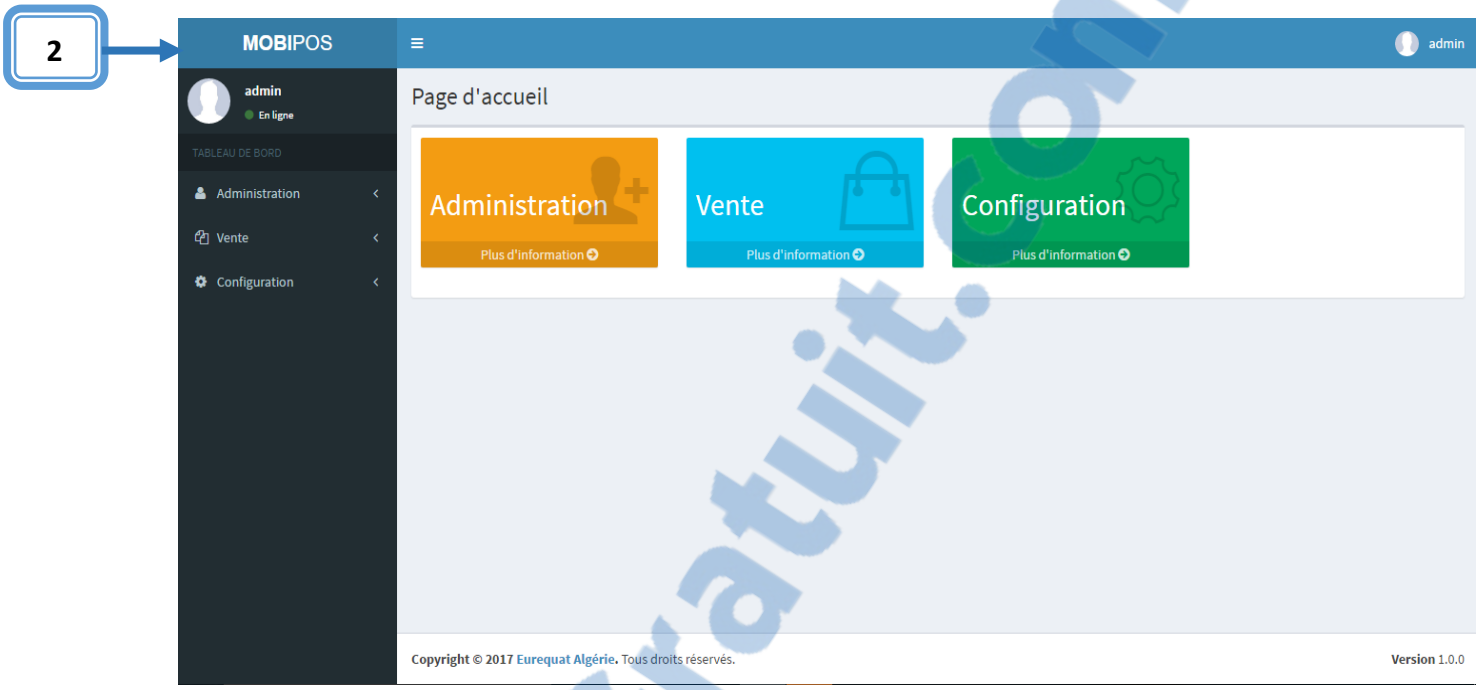


Fenêtre d'authentification

Remarque : ***pour la première utilisation de MOBIPOS le nom d'utilisateur et mot de passe sont : admin /admin***

Le Back-office de MOBIPOS a un accès administrateur, ce dernier possède un accès à toutes les fonctionnalités.

Après l'authentification de l'administrateur, Une « Page d'accueil » se présente.

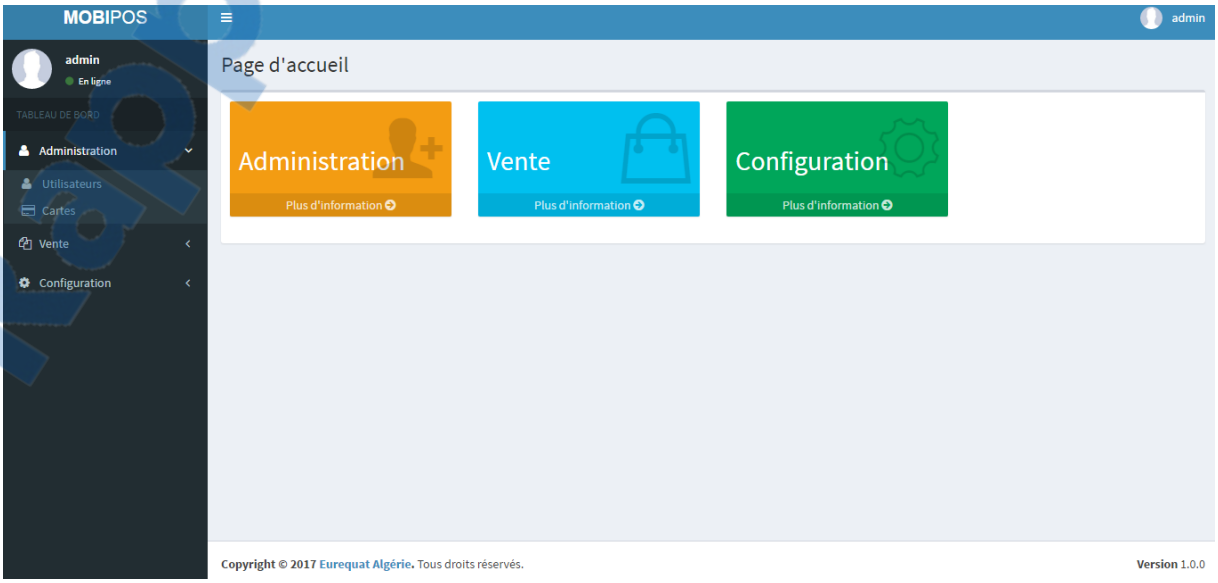


Interface de la page d'accueil

Remarque :

Il suffit de cliquer sur **2** pour accéder à la page d'accueil.

2.2 Administration

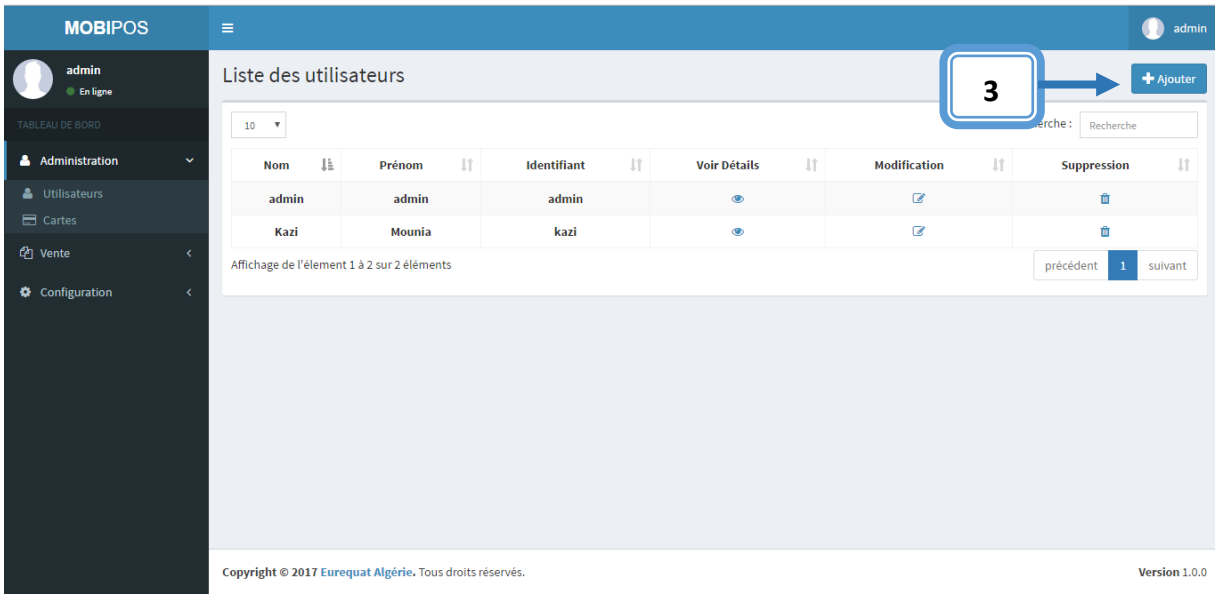


Dans la partie "Administration", on peut gérer les "Utilisateurs" et "Cartes".

Rapport-Gratuit.com

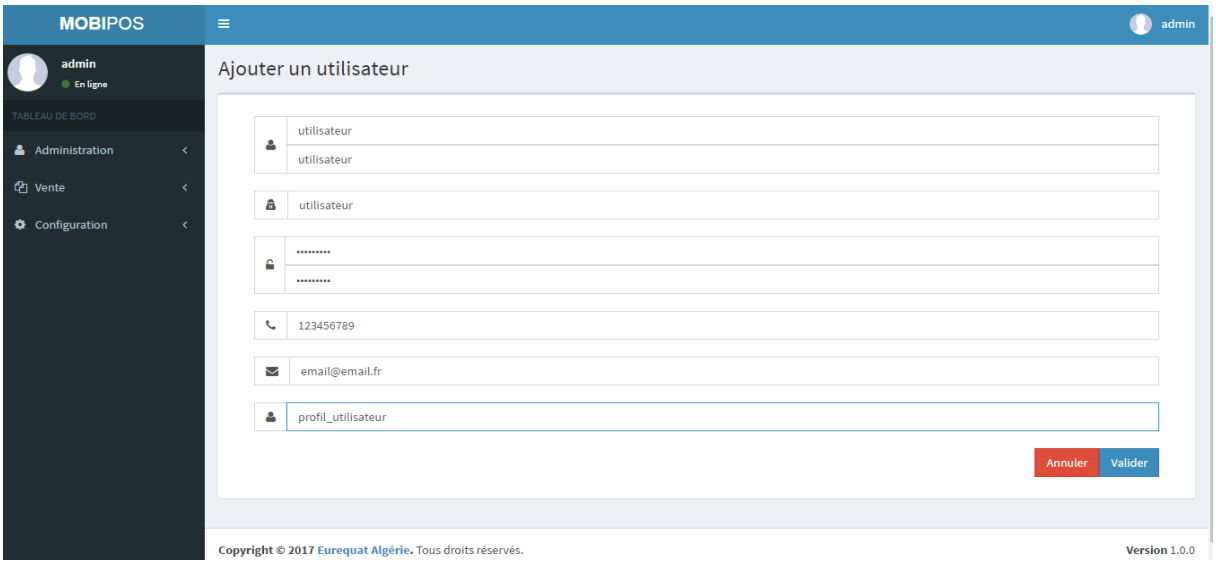
2.2.1 Utilisateurs

A partir du menu de navigation, cliquez sur "Administration", un sous menu s'affiche, cliquer sur "Utilisateurs" :



Interface Gestion des Utilisateurs

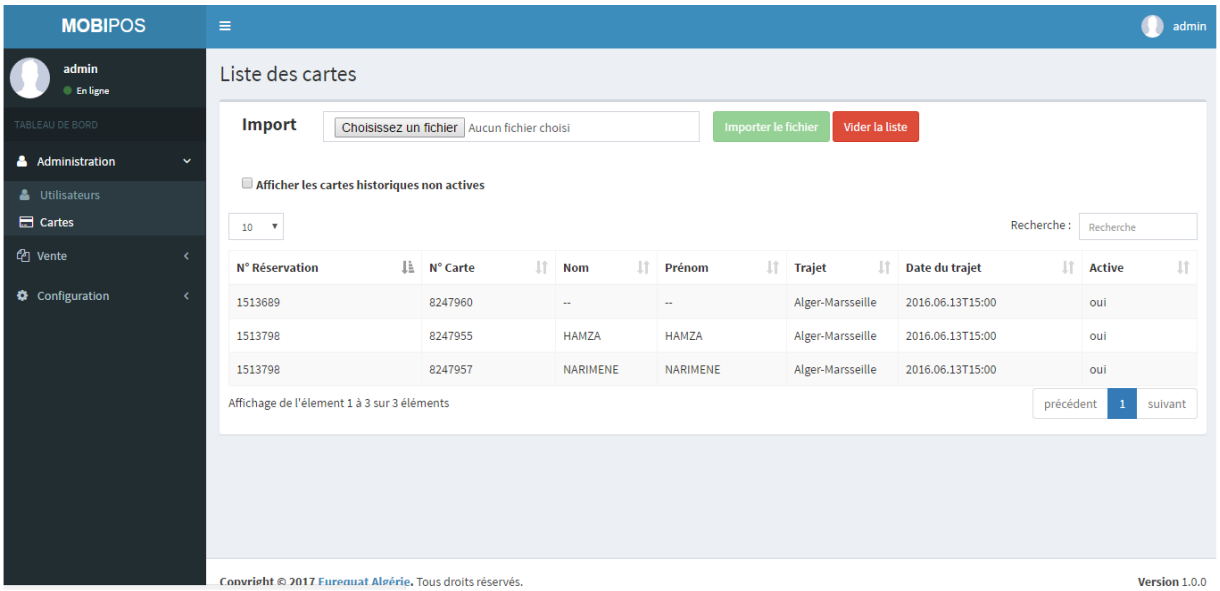
Pour ajouter un utilisateur, cliquer sur **3** comme indiquer sur l'image ci-dessus. Puis remplissez les informations, ensuite cliquer sur le bouton « Ajouter ». Il est également possible de supprimer ou modifier un utilisateur.



Interface d'ajout d'un Utilisateur

2.2.2 Cartes

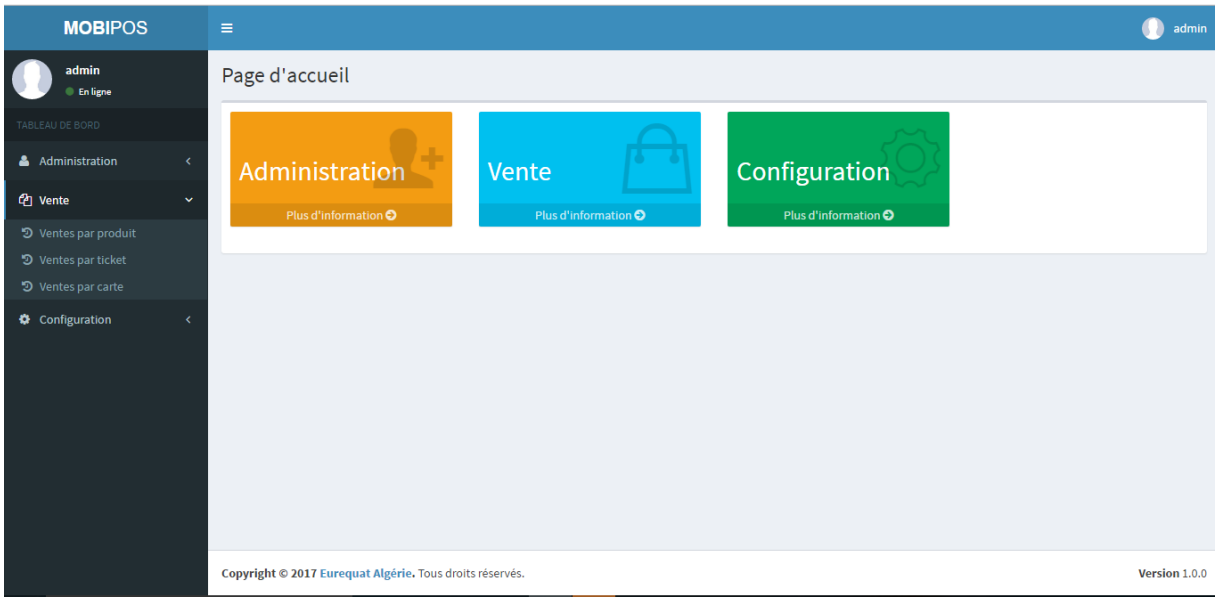
A partir du menu de navigation, cliquez sur "Administration", un sous menu s'affiche, cliquer sur "Cartes" :



Interface Carte

A partir de l'interface "Cartes", il est possible d'importer un fichier Excel contentant les numéros d'accès ,voir les informations des cartes importées, qui porteront le statut "Active => oui ", ainsi que l'historique des cartes, il est indispensable de vider la liste avant d'en importer une nouvelle ,par la suite les anciennes cartes deviennent inactives et porteront le statut "Active => non ".

2.3 Vente



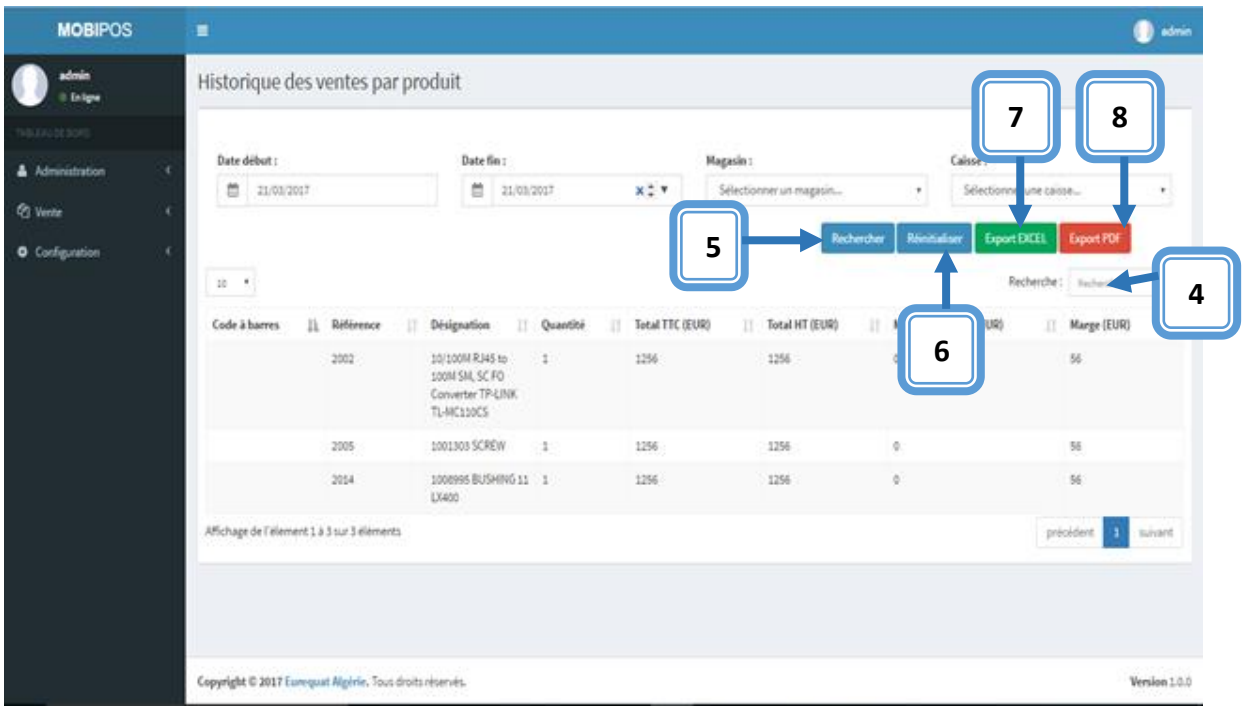
Interface Vente

A partir du menu, cliquer sur « Vente », trois sous menu s'affichent, "Ventes par produit", "Ventes par ticket" et "Ventes par carte", ces sous menu permettent de visualiser l'historique des ventes regroupé par produit, ticket, carte.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

2.3.1 Historique des ventes par produit

A partir du menu de navigation, cliquer sur "Vente" puis "Ventes par produit", on accède à la fenêtre suivante :



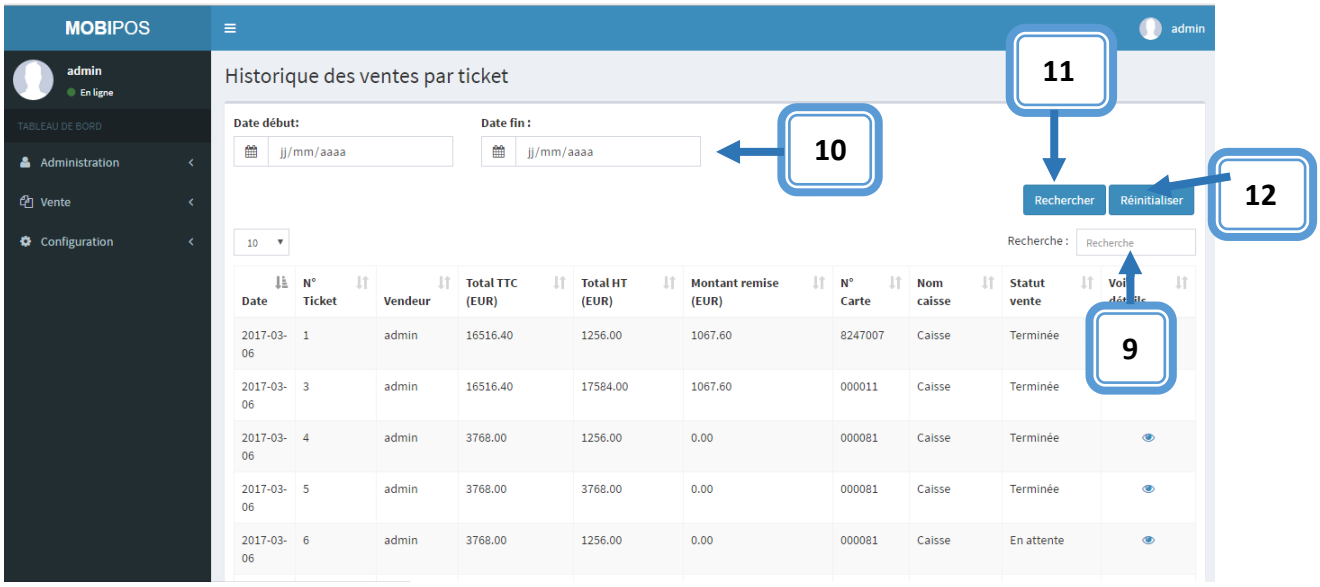
Interface Historique des Ventes par produit

Cette fenêtre affiche la liste des ventes réalisées regroupée par produit. Pour faire une recherche multi champs ,il suffit de saisir dans la case **4**, et pour faire un filtre, il suffit de saisir les champs date, le nom du magasin, le nom de la caisse (tous les champs ne sont pas obligatoires) ensuite lancer la recherche **5**, et par défaut le filtre afficher par rapport à la date du jour. Pour exporter l'historique en Excel, appuyer sur **7**. Pour exporter l'historique en PDF, appuyer sur **8**.

Pour réinitialiser le filtre par défaut appuyer sur **6.**

2.3.2 Historique des ventes par ticket

A partir du menu de navigation, cliquer sur "Vente" puis "ventes par ticket", on accède à la fenêtre suivante :



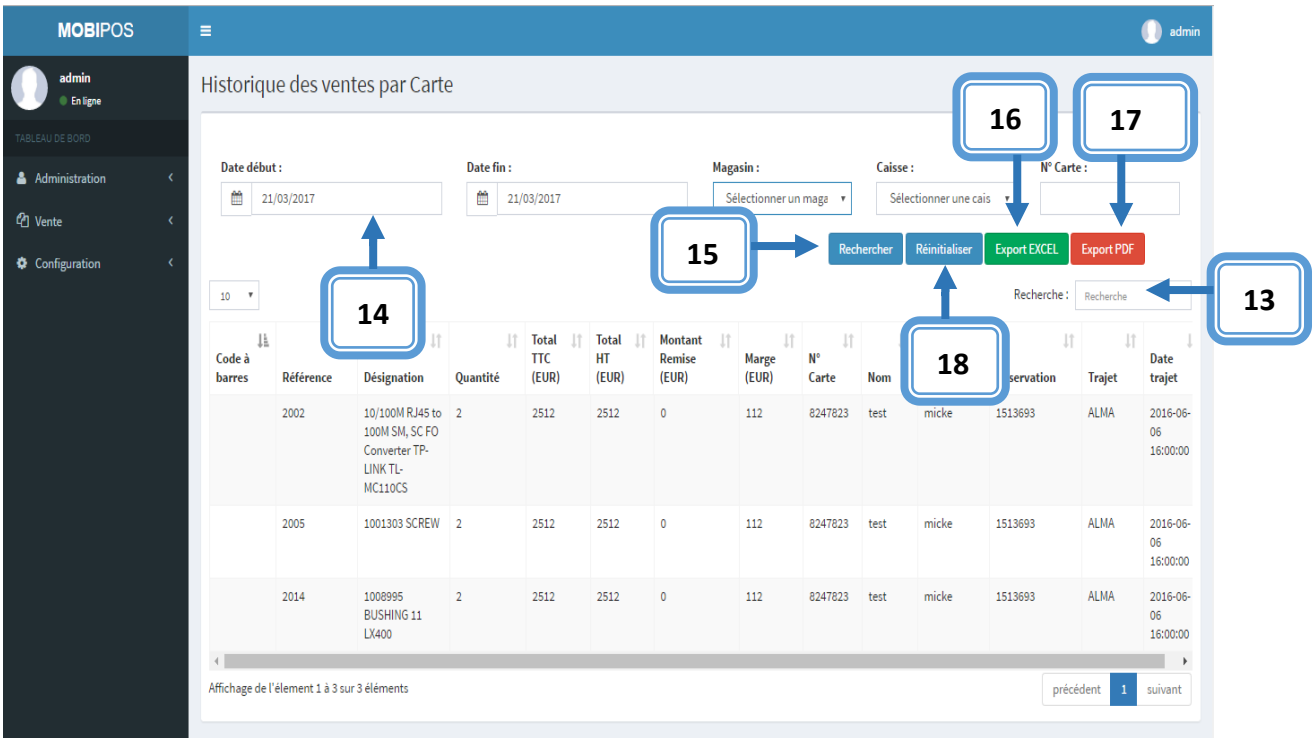
Interface Historique des Ventes par ticket

Cette fenêtre affiche la liste des ventes réalisées. Pour faire une recherche multi champs saisir dans la case **9**. Pour faire une recherche, saisir les champs **10** date début et date fin (tous les champs ne sont pas obligatoires) ensuite lancer la recherche **11**. Pour réinitialiser le filtre par défaut appuyer sur **12.**

.

2.3.3 Historique des ventes par carte

A partir du menu de navigation, cliquer sur "Vente" puis "Ventes par carte", on accède à la fenêtre suivante :



Interface Historique des Ventes par carte

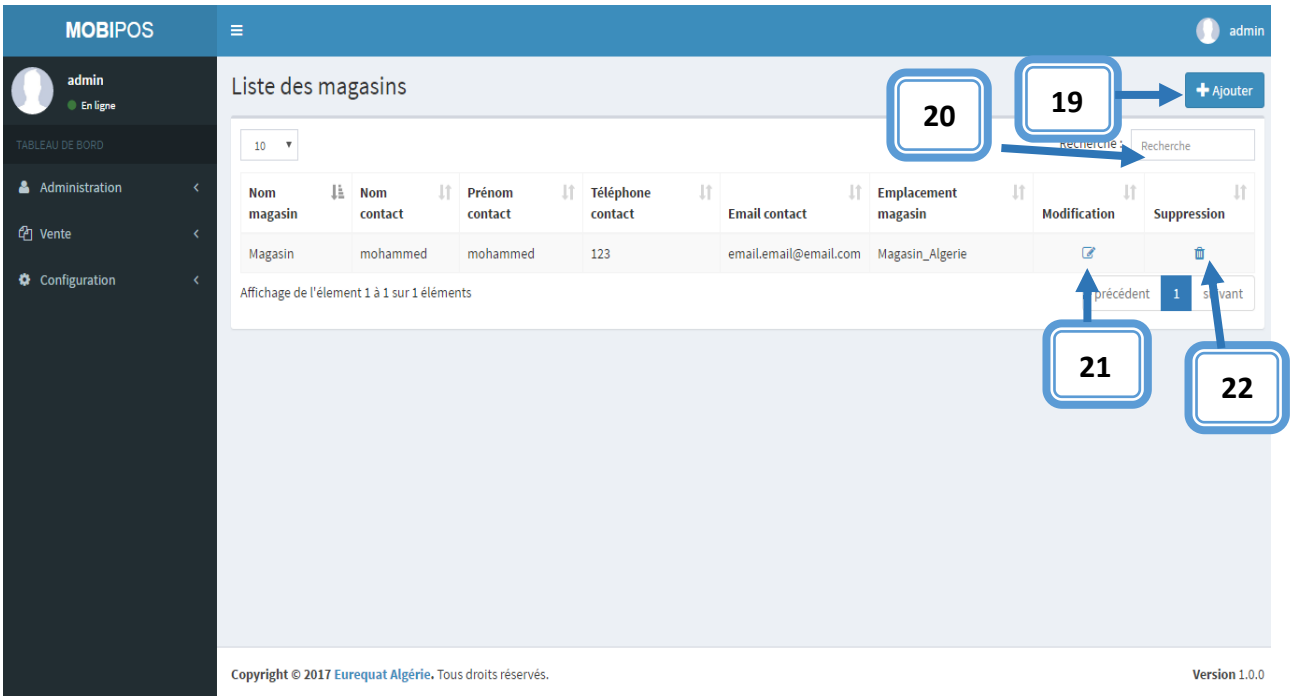
Cette fenêtre affiche la liste des ventes réalisées regroupé par carte du client. Pour faire une recherche multi champs saisir dans la case **13**, et pour faire un filtre saisir les champs **14** date, le nom du magasin, le nom de la caisse, le numéro de carte (tous les champs ne sont pas obligatoires) ensuite lancer la recherche **15**, et par défaut le filtre afficher par rapport à la date du jour. Pour exporter l'historique en Excel, appuyer sur **16**. Pour exporter l'historique en PDF, appuyer sur **17**.

Pour réinitialiser le filtre par défaut appuyer sur **18.**

2.3 Configuration

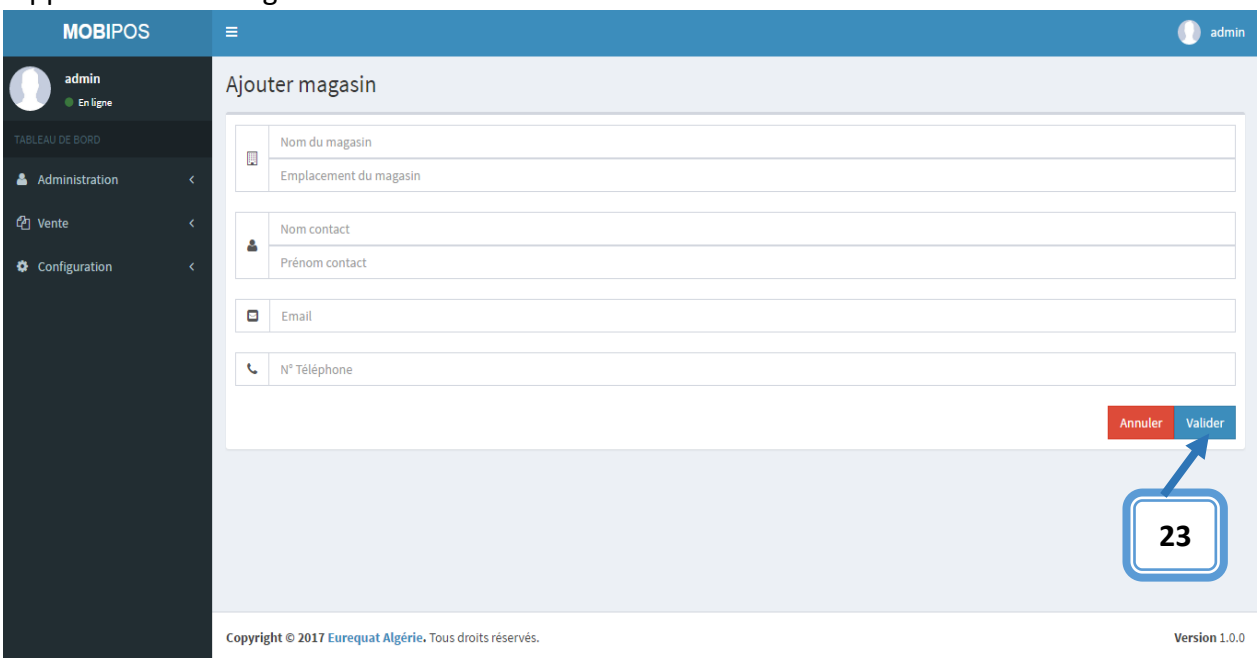
2.3.1 Magasin

A partir du menu de navigation, cliquez sur "Configuration", un sous menu s'affiche, cliquer sur "Magasin" :



Interface magasin

A partir de cette interface il est possible d'ajouter des magasins **19** , voir les magasins existants, faire une recherche **20**, il est aussi possible de modifier **21**, supprimer **22** un magasin.



Interface d'ajout d'un magasin

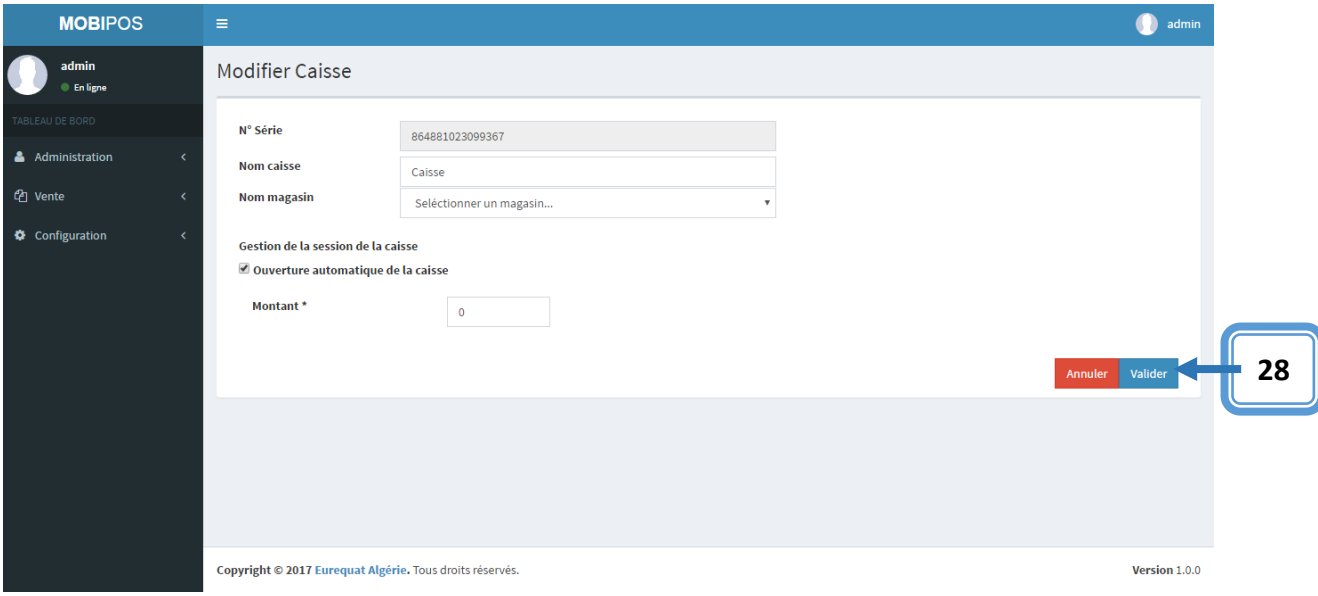
Afin d'ajouter un magasin, il suffit de remplir les champs et cliquer sur le bouton "Valider" **23** pour terminer l'opération avec succès.

2.3.2 Caisse

A partir du menu de navigation, cliquez sur "Configuration", un sous menu s'affiche, cliquer sur "Caisse" :

Interface caisse

L'interface "Caisse", permet de faire la configuration des caisses **24** ,voir les caisses existantes ,vérifier si les caisses sont ouvertes automatiquement ou pas, et cela à partir du statut "Auto ouverture" **25** , ainsi que faire la suppression des caisses **26**, il est également possible de réinitialiser le contenu afin de voir les nouvelles caisses ajoutées **27**.



Interface de la configuration / modification d'une caisse

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Après avoir cliquer sur **24**, l' interface de la configuration des caisses se présente, dans laquelle, il est possible de modifier le nom, attribuer un magasin, ainsi que faire l'ouverture automatique en saisissant un montant à la caisse, par la suite cliquer sur le bouton "Valider" pour terminer la configuration **28**.

Liste de figures

Figure I.1 Le logo de la société Eurequat 5
Figure I.2 Terminal point de vente (TPV) 8
Figure I.3 Architecture globale du système MOBIPOS 9
Figure I.4 Fonctionnalités de Front office MOBIPOS 10
Figure I.5 Fonctionnalités de Back office MOBIPOS 11
Figure I.6 Cycle de vie de Back office MOBIPOS 13
Figure II.1 : Diagramme globale de cas d'utilisation 23
Figure II.2 : Diagramme du cas « Consulter l'historique des ventes » 24
Figure II.3 : Diagramme du cas « Exporter des rapports » 25
Figure II.4 : Diagramme de séquence « S'authentifier » 26
Figure II.5 : Diagramme de séquence « Ajouter une catégorie » 28
Figure II.6 : Diagramme de séquence « Visualiser rapport sortie stock » 30
Figure II.7 : Diagramme de séquence « Exporter PDF rapport sortie stock » 32
Figure II.8 Diagramme de class « projet MOBIPOS backoffice » 34
Figure II.9 Exemple de maquette « page d'accueil » 35
Figure III.1 Le processus de Scrum 39
Figure III.2 Les sprints réalisés du MOBIPOS Back office 40
Figure III.3 Architecture de Laravel 43
Figure III.4 Architecture du module back-office 45
Figure III.5 Quelque anomalies de projet MOBIPOS Back office 48
Figure III.6 Interface d'authentification 49
Figure III.7 Interface d'Accueil 50
Figure III.8 Liste des caissiers 51
Figure III.9 Liste des catégories 52
Figure III.10 Message d'avertissement 53
Figure III.11 L'ajout d'une catégorie 53
Figure III.12 Liste des produits 54
Figure III.13 Listes des produits au magasin sélectionné 55
Figure III.14 Visualiser un produit 56
Figure III.16 Historique des ventes par carte 57
Figure III.16 Listes des produits en stock 58
Figure III.17 Entrées/sorties en stock 59
Figure III.18 Rapport de l'état des mouvements de stocks 60
Figure III.19 Rapport sous format PDF 61
Figure III.20 La liste des inventaires générés 62
Figure III.21 Rapport de l'historique du stock 63
Figure III.22 La liste des caisses 64
Figure III.23 Modifier une caisse 64

Liste des tableaux

Tableau I.1 Comparatif entre différents systèmes 12
TableauIII.1.Diagramme de Gantt……. 43
TableauIII.2.Quelques exemples d'anomalies…….....49

Rapport-Gratuit.com

Liste des abréviations

TPV : Terminal Point de Vente.

RFID : Radio Frequency Identification.

IOS : iPhone Operating System.

UML : Unified Modeling Language.

UP : Unified process.

POS : Point Of Sale.

Résumé

Les entreprises commerciales ont besoins de solutions à faible coût pour la gestion des ventes. Pour répondre à ce besoin, l'entreprise Eurequat Algérie a lancé le projet « MOBIPOS »,Nous avons participé à la réalisation de ce dernier en développant le module back office MOBIPOS qui aide le responsable du magasin à automatiser les tâches de la gestion des vendeurs et la gestion des caisses. Il permet aussi d'assurer une meilleure gestion des stocks et un bon suivi des ventes. Pour cela, nous avons utilisé MYSQL comme un système de gestion de base de données et ce module a été implémenté avec Laravel 5 qui est un framework PHP, javascript pour la programmation des scriptes, HTML5 et CSS3 pour la mise en page et le design de notre projet. Le système développé a été testé et approuvé par le client.

Mots clés : Back office, Terminal de point de vente, Gestion des ventes.

Abstract

Commercial companies need low-cost solutions for the sales management. In order to meet this need, Eurequat Algeria initiate the project entitled "MOBIPOS". We have been a part of the development team of application in charge of the back office module development . The back office is designed to help the store manager automating some of the tasks relative to the sale process (Management of vendors & funds), also the application is designed to ensures better inventory management and good sales tracking. In order to fulfill the design requirements , we have used MYSQL as a management system database . The PHP framework Laravel 5 for programming , Javascript is adopted for scripting , HTML5 and CSS3 for layout design of our project. The developed system has been tested and approved by the customer.

Keywords: Back office, Terminal POS, Sales management.

الملخص

لإدارة المبيعات تحتاج المؤسسات التجارية لحلول منخفضة التكلفة. لتلبية هذه الحاجة، أطلقت شركة Eurequat الجزائر مشروع "MOBIPOS"، حيث شاركنا في انجاز هذا الأخير من خلال تطويرينا لوحدة وهي back office MOBIPOS التي تساعد مدير المحل لأتمتة مهام إدارة البائعين وإدارة الأموال. كما يسمح بضمان إدارة ممتازة للمخزون ومتابعة جيدة للمبيعات. لتطوير هذه الوحدة قمنا باستخدام MySQL كنظام إدارة قواعد البيانات و Laravel 5 الذي يعتبر إطارا PHP، كما استخدمنا Java script لمخطوطات البرمجة وHTML5، CSS3 للتخطيط لتصميم مشروعنا.

لقد تم اختبار مشروعنا من قبل العملاء والموافقة عليه.

كلمات الرئيسية: المكتب الخلفي، نقاط البيع، إدارة المبيعات.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES