REPUBLIQUE DU SENEGAL



UN PEUPLE - UN BUT - UNE FOI

Ministère de l'emploi, de la formation professionnelle, de l'apprentissage et de l'insertion (MEFPAI)

SONATEL ACADEMY



Ecole Police, Immeuble Scalène Mermoz lot B, Rue KA 05,

<u>Contact@sonatelacademy.sn</u> / <u>https://www.academysonatel.com</u> +221 33 839 21 00

<u>Spécialité</u>: Développement web et mobile Mémoire de fin de cycle pour l'obtention du certificat de spécialisation

Sujet : WEBPARTNER, Application de gestion des reportings destinées aux partenaires de OFMS (Orange Finances Mobiles Sénégal)

Présenté et soutenu par :

Fama Zeyna Faye DIOP

Cheikhna Ahmadou SAKHO

Alioune Aldemba SECK

Encadreur:

M. Birane Baila WANE

ANNEE DE SOUTENANCE : 2021-2022

A La Mémoire de

Tous ceux qui ont contribué à mon éducation, à ma formation et à ma réussite et qui ne sont plus là malheureusement :

- Tambedou Aziz, mon grand-père
- FAYE Fama, ma grand-mère

Paix à leur âme et que Dieu les accueille au paradis. Amine.

Dédicace

En préambule de ce mémoire, je souhaiterais dédier ce travail à ma mère Fatou DIOP et à ma tante SOKHNA DIOP en guise de reconnaissance et de gratitude pour leur amour, leur patience, leur soutien, leur encouragement et surtout pour les énormes sacrifices consentis à mon égard.

Aucun mot, aucune dédicace ne pourrait exprimer mon respect, ma considération ainsi que l'expression

de mon profond amour

Remerciements

Je ne saurais remercier une personne sans au préalable rendre grâce à DIEU, magnifier sa grandeur, prier au nom de son prophète MOUHAMED paix et salut sur lui (PSL)...

Je tiens sincèrement à remercier tous ceux qui de près ou de loin ont participé à l'élaboration de ce mémoire. Mes pensées vont particulièrement envers :

- Mes parents pour le soutien moral et leur patience infinie à mon égard. Vous avez été, vous êtes et vous demeureraient les meilleurs parents au monde. MERCI !!!!
- ➤ Mes tantes adorées Sokhna DIOP et Katy DIOP. MERCI !!!!
- L'ensemble du corps professoral de la SONATEL ACADEMY, en particulier et M. Aly Tall NIANG et M. Birane Baila WANE qui n'ont ménagé aucun effort pour la réussite de ma formation et ont inculqué en moi des valeurs qui, certainement, faciliteront mon intégration dans la vie active. Merci !!!
- ➤ M. Birane Baila WANE, en tant que directeur de mémoire. Merci pour l'aide et le temps qu'il a bien voulu me consacrer et sans lui ce travail n'aurait jamais vu le jour.
- Mes amis particulièrement mon rayon de soleil Sokhna Awa DIOP pour m'avoir motivé à croire en mes rêves, à ne jamais baisser les bras et pour le soutien moral.
- ➤ Mes collègues Cheikhna Ahmadou SAKHO et Alioune Aldemba SECK et toute l'équipe de OFMS.
- ➤ Tous ceux dont les remarques et suggestions conduiront à l'amélioration de mon argumentation.

 Je pense en particulier aux membres du jury.

Avant-Propos

Sonatel Academy est la première école de codage gratuite du Sénégal créée par la Sonatel en partenariat avec SIMPLON.CO pour contribuer à l'amélioration de l'offre de formation numérique au Sénégal. Inaugurée le 22 Novembre 2017, elle propose trois (3) référentiels : développement web/mobile, développement data et référent digital.

L'objectif de la Sonatel Academy est de :

- ✓ Répondre aux besoins en compétences numériques
- ✓ Renforcer l'employabilité des jeunes à travers les métiers du numérique
- ✓ Donner aux jeunes la capacité d'apprendre seul et l'envie de travailler en équipe

Pour l'obtention du certificat de spécialisation en développement WEB Mobile, la Sonatel Academy exige aux étudiants la rédaction d'un mémoire. C'est dans ce cadre que nous avons élaboré ce document qui a pour sujet : WEBPARTNER, application de gestion des reportings destinées aux partenaires de OFMS (Orange Finances Mobiles Sénégal)

Ce document constitue notre premier travail de recherche académique, c'est pourquoi nous sollicitons de la part du jury, beaucoup d'indulgence pour ce qui concerne son évaluation.

Sommaire

INTRODUCTION GENERALE	01
I CADRE GENERAL DU PROJET	03
II METHODOLOGIE, OUTILS ET TECHNOLOGIES DE DEVELOPPEME	NT08
III MISE EN ŒUVRE DE L'APPLICATION	27
CONCLUSION GENERALE	44
BIBLIOGRAPHIE	45
WEBOGRAPHIE	46
TABLE DE MATIÈRES	47

Glossaire

OFMS : Orange Finances Mobiles Sénégal

UML : Unified Modeling Language (Langage de Modélisation Unifié)

SGBD : Système de Gestion de Base de Données

Liste des figures

- Figure 1 : Organigramme de OFMS
- Figure 2 : Processus de la méthode Scrum
- Figure 3 : Processus de la méthode EXtreme Programming (XP)
- Figure 4: Backlog, sprint
- Figure 5 : Frormalisme diagramme de contexte
- Figure 6 : Diagramme de contexte
- Figure 7 : Diagramme de classe
- Figure 8 : Diagramme de déploiement
- Figure 9 : Diagramme de cas d'utilisation du super admin
- Figure 10: Maquette page paiement
- Figure 11: Maquette page trackings
- Figure 12: Maquette page transactions
- Figure 13: Maquette commissions hebdo
- Figure 14 : Maquette commissions journalières
- Figure 15: Page de connexion
- Figure 16 : Liste des utilisateurs
- Figure 17 : Création d'utilisateurs
- Figure 18: Page des trackings
- Figure 19 : Page des paiements
- Figure 20: Page des commissions hebdomadaires
- Figure 21: Diagramme de cas d'utilisation du grossiste
- Figure 22 : Diagramme de cas d'utilisation du distributeur
- Figure 23: Maquette page de connexion
- Figure 24: Maquette page d'accueil
- Figure 25: Maquette page des paiements
- Figure 26: Maquette page des transactions
- Figure 27 : Maquette page des commissions
- Figure 28 : Maquette page des demandes
- Figure 29: Page d'accueil
- Figure 30 : Page des paiements
- Figure 31 : Page des commissions jours
- Figure 32 : Page des commissions hebdos
- Figure 33 : Page d'accueil du grossiste
- Figure 34 : Page d'accueil du grossiste
- Figure 35 : Page des paiements du grossiste jours
- Figure 36: Page des transactions du grossiste
- Figure 37: Détail d'une transaction d'un grossiste
- Figure 38 : Commissions journalières du grossiste
- Figure 39: Commissions hebdomadaires du grossiste
- Figure 40 : Points de vente du grossiste
- Figure 41 : Diagramme de cas d'utilisation du marchand
- Figure 42: Maquette page d'accueil du marchand
- Figure 43: Maquette page des virements bancaires
- Figure 44 : Maquette page détails caisses du marchand
- Figure 45: Maquette page des transactions du marchand
- Figure 46 : Maquette liste des caisses du marchand
- Figure 47: Maquette page des demandes du marchand
- Figure 48: Page d'accueil du marchand
- Figure 49: Virements bancaires du marchand
- Figure 50 : Détail d'un virement bancaire du marchand
- Figure 51: Les transactions du marchand
- Figure 52 : Détail d'une transaction du marchand
- Figure 53: Liste des caisses du marchand

Liste des tableaux

Tableau 1 : Les cardinalités dans un diagramme de classe

Résumé

Ce mémoire a pour objectif de résoudre les problèmes de manque de transparence et d'informations pour les partenaires de Orange Finances Mobiles Sénégal (OFMS) en leur fournissant une meilleure visibilité de leurs activités à travers l'application WEBPARTNER.

La méthode de recherche adoptée consiste en des enquêtes menées auprès des partenaires et des responsables acquisition distributeur ORANGE MONEY pour mieux comprendre leurs problèmes.

Ce présent travail sera scindé en trois parties. Dans la première nous parlerons du cadre général du projet. La deuxième partie proposera la méthodologie, les technologies et les outils de développement. La dernière nommée mise en œuvre sera subdivisée en trois modules : le premier module concernera la partie Super Admin, le second parlera du Grossiste et ses Distributeurs. Et enfin le troisième module visera le Marchand.

Les résultats de cette étude montrent que l'application WEBPARTNER permet d'améliorer les performances financières des partenaires d'OFMS et offre une meilleure visibilité de leurs activités. Les recommandations de ce mémoire sont de continuer à développer l'application WEBPARTNER et d'y inclure les autres partenaires d'OFMS tels que les facturiers et les agrégateurs.

Abstract

This thesis aims to solve the problems of lack of transparency and information for the partners of Orange Finances Mobiles Senegal (OFMS) by providing them with better visibility of their activities through the WEBPARTNER application.

The research method adopted consists of surveys conducted among partners and ORANGE MONEY distributor acquisition managers to better understand their problems.

This work will be divided into three parts. In the first we will talk about the general framework of the project. The second part will propose the methodology, technologies and development tools. The last will be divided into three modules: the first module will concern the Super Admin part, the second will talk about the Wholesaler and its Distributors. And finally the third module will target the Merchant.

The results of this study show that the WEBPARTNER application improves the financial performance of OFMS partners and provides better visibility of their activities. The recommendations of this thesis are to continue to develop the WEBPARTNER application and to include the other OFMS partners such as billers and aggregators.

Introduction générale

De nos jours, le monde connaît une avancée technologique considérable dans tous les secteurs et cela à l'aide de l'informatique, qui joue un rôle important dans le développement de nombreuses entreprises et organisations. Aussi, les services financiers mobiles sont devenus une nécessité pour des millions de personnes dans le monde entier, en particulier dans les pays en développement où l'accès aux services bancaires traditionnels est limité. C'est dans cette optique qu'Orange Finances Mobiles Sénégal a mis à la disposition de ses partenaires des plateformes offrant une gamme de services financiers.

Cependant, malgré ses différentes offres, les partenaires se plaignent souvent d'un manque de transparence et d'informations pour comprendre pleinement leur activité, ce qui crée des tensions voire une rupture de contrat.

Pour pallier ces maux, nous avons mis en place une application WEB, WEBPARTNER qui s'est fixé comme objectif d'assurer l'amélioration des performances financières et une meilleure visibilité des activités que mènent les partenaires d'OFMS, en leur fournissant des informations précises.

Pour arriver à nos fins, plusieurs méthodes de recherches ont été adoptées. Notamment des enquêtes menées auprès des partenaires tels que SEDIMA, AUCHAN pour mieux comprendre leurs problèmes. Mais aussi des enquêtes menées auprès des responsables acquisition distributeur ORANGE MONEY.

Ce présent travail sera scindé en trois parties. Dans la première nous parlerons du cadre général du projet. La deuxième partie proposera la méthodologie, les technologies et les outils de développement. La dernière nommée mise en œuvre sera subdivisée en trois modules : le premier module concernera la partie Super Admin, le second parlera du Grossiste et ses Distributeurs. Et enfin le troisième module visera le Marchand.

PREMIÈRE PARTIE CADRE GÉNÉRAL DU PROJET

I Cadre général du projet

1.1 Présentation de OFMS

Orange Finances Mobiles Sénégal (OFMS) est une filiale de l'opérateur de télécommunications Orange basée au Sénégal. Elle offre des services financiers mobiles (mobile money) aux clients sénégalais. Elle leur permet d'effectuer des transactions financières à partir de leur téléphone portable, tels que des paiements, des transferts d'argent, de l'épargne et des retraits d'argent en espèces auprès de partenaires agréés.

Les principales missions de OFMS sont les suivantes :

- Offrir des services financiers mobiles accessibles : Orange Money permet à tout utilisateur disposant d'un téléphone mobile d'accéder à des services financiers de base, sans avoir besoin de posséder un compte bancaire.
- Faciliter les paiements électroniques : OFMS permet aux utilisateurs de payer leurs factures de services publics, d'acheter des biens et des services en ligne, ainsi que de payer les commerçants qui acceptent les paiements Orange Money.
- Favoriser l'inclusion financière : OFMS contribue à l'inclusion financière en permettant à des personnes qui n'ont pas accès à des services bancaires traditionnels de participer à l'économie formelle.
- Développer des partenariats avec d'autres institutions financières : OFMS travaille en étroite collaboration avec les banques et les autres institutions financières pour faciliter les transferts de fonds et l'échange de données.
- Favoriser le développement économique : En facilitant les paiements électroniques et les transferts d'argent, OFMS contribue à la croissance économique en favorisant les transactions commerciales et les échanges entre individus.

En résumé, Orange Finances Mobiles Sénégal vise à offrir des services financiers mobiles abordables et accessibles aux populations du Sénégal, en contribuant à l'inclusion financière et au développement économique du pays.

Organigramme de OFMS

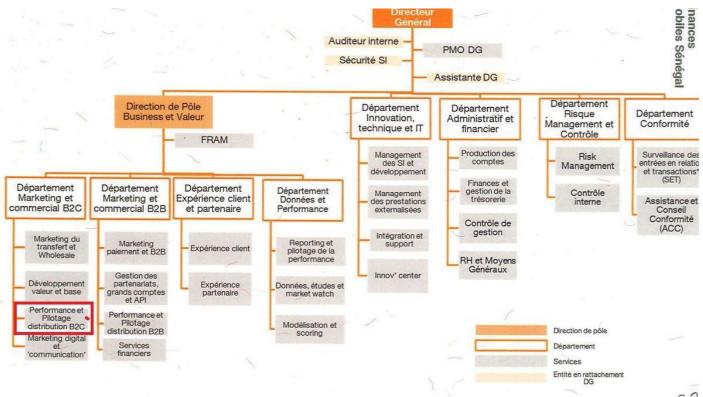


Figure 1 : Organigramme de OFMS

1.2 Contexte du projet

Les entreprises de services financiers mobiles font face à des défis importants en matière de réglementation, de concurrence et de sécurité des données. Orange Finances Mobiles Sénégal (OFMS) en est une qui propose des solutions de paiement mobile pour les clients au Sénégal. Depuis sa création, elle a connu une croissance rapide et a établi des partenariats avec plusieurs entreprises et institutions pour étendre ses services à un public plus large.

Cependant, malgré sa croissance rapide, OFMS a rencontré des problèmes en matière de transparence et de communication avec ses partenaires. Ces derniers ont souvent eu du mal à obtenir des informations précises sur leurs activités, telles que les transactions effectuées et les commissions qu'ils ont gagnées. Cette situation a conduit à une frustration et une insatisfaction croissantes chez les partenaires, qui ont parfois envisagé de chercher des alternatives à OFMS.

Présenter une plateforme qui va améliorer la transparence et la communication avec ses partenaires, est une solution qui contribuera à renforcer la position de l'OFMS sur le marché des services financiers mobiles au Sénégal et à améliorer les relations avec ses partenaires.

1.3 Problématique du projet

Malgré le succès de la plateforme, les partenaires d'ORANGE MONEY manquent de visibilité sur leurs différentes opérations, ils se plaignent de différents maux tels que:

- Retard dans leur comptabilité à cause de données manquantes ce qui peut rendre difficile la gestion de leurs finances
- Impossibilité de voir les détails d'un virement bancaire ce qui peut mener à la confusion
- Une recherche fastidieuse sur leurs différentes opérations via une application archaïque : Global Reporting
- Perte de temps en passant des appels téléphoniques, ou des envois de mails pour demander des extractions de leurs transactions concernant une période
- Le manque d'une vue 360 où ils pourront voir l'ensemble de leurs transactions, gains, virements bancaires et caisses
- Le manque de transparence de OFMS par rapport au concurrent WAVE.
- Doute de la fiabilité des informations fournies par OFMS, ce qui peut affecter leur confiance
- Frustration par le manque de support ou par des réponses inadéquates ou une longue période d'attente de la part du service client d'OFMS

Certains partenaires ont rompu leurs contrats avec OFMS et d'autres menacent de le faire.

D'où la nécessité de formuler notre problématique : Comment OFMS peut-il aider ses partenaires à améliorer leur performance financière et à mieux comprendre leur activité, en leur fournissant des informations précises ?

1.4 Objectifs du projet

L'objectif principal est d'aider les partenaires de OFMS à améliorer leur performance financière et à mieux comprendre leur activité, en leur fournissant des informations précises et ainsi renforcer la position de l'OFMS sur le marché des services financiers mobiles au Sénégal. Pour atteindre ces objectifs nous allons mettre une plateforme où :

Le super Admin pourra accéder à ses :

- Gestion des accès :
 - Utilisateurs
 - Trackings
- Rapport des paiements
- Rapport des transactions
- Rapports des virements bancaires
- Rapport des commissions
 - Journalières
 - Hebdomadaires
- Gestion des demandes

- API
- Marchand
- Distributeur

Le grossiste pourra accéder à ses :

- Paiement de commissions
- Rapport de commissions journalières
- Rapport de commissions hebdomadaires
- Rapport de transactions
- Comptes dormants
- Demandes de changement de rattachement

Le distributeur pourra accéder à ses :

- Rapport de transactions
- Paiement de commissions
- Rapport de commissions journalières
- Rapport de commissions hebdomadaires
- Demandes de rattachement

Le marchand classique pourra accéder à ses :

- Caisses
- Rapport de transactions
- Rapport de factures
- Virements bancaires

En ce qui concerne ces différentes phases ci-dessus, le module Super Admin a été réalisé par M. Alioune Aldemba SECK, le module Grossiste et Distributeur par Mme Fama Zeyna Faye DIOP et enfin le module Marchand par M. Cheikhna Ahmadou SAKHO.

DEUXIÈME PARTIE

METHODOLOGIE, OUTILS ET
TECHNOLOGIES DE DEVELOPPEMENT

II Méthodologie, outils et technologies de développement

Une méthodologie est un système de pratiques, de techniques, de procédures et de règles utilisées par ceux qui travaillent dans une discipline.

2.1 Méthodologie

2.1.1 Présentation des méthodes agiles

Les méthodes agiles caractérisent un mode de gestion des projets informatiques privilégiant le dialogue entre toutes les parties prenantes, clients, utilisateurs, développeurs et autres professionnels du projet, la souplesse en cours de réalisation, la capacité à modifier les plans et la rapidité de livraison. Les méthodes Agiles les plus populaires en usage aujourd'hui sont : Scrum, eXtrême Programming (XP), Feature Driven Development (FDD), Agile Unified Process(AUP)...

2.1.1.1 Scrum

Scrum (qui signifie mêlée au rugby) est aujourd'hui la méthode agile la plus populaire. Elle se caractérise par itérations (appelées sprints) assez courts (maximum 1 mois) et un formalisme réduit : rôles (Product Owner, ScrumMaster, équipe), timeboxes (planification de release, planification de sprint, scrum quotidien, revue de sprint, introspection) et artéfacts (backlog de produit, plan de produit, plan de sprint, burdown/burnup de release, burdown/burnup de sprint).

Le **Scrum** s'appuie sur :

- Le **découpage d'un projet** en unités de temps courtes, nommées **sprints**,ceci afin d'éviter l'**effet tunnel**, dû à une planification tellement étendue dans le temps qu'on n'en voit pas le bout ;
- Des **réunions régulières** pour faire le point ;
- Des rôles :
 - o Une équipe pluridisciplinaire (développeurs, architectes, designers, testeurs);
 - Le product owner, qui a la vision du produit et s'assure de la bonnetraduction des attentes du client à l'équipe projet, en définissant les spécifications fonctionnelles (ou specs) et les priorités;
 - Le scrum master, ou maître de mêlée, qui est le chef d'orchestre, le coordinateur de l'équipe agile, dont il fait partie intégrante.

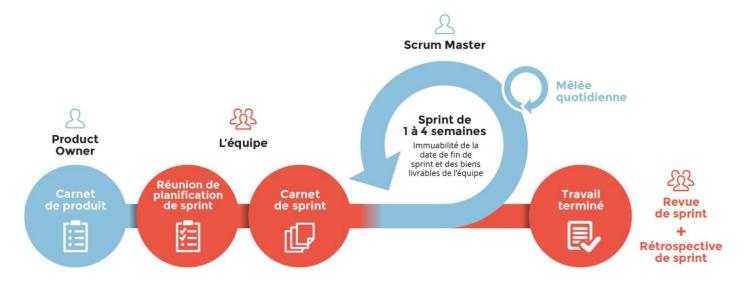


Figure 2 : Processus de la méthode scrum

2.1.1.2 Extreme Programming (XP) Scrum

L'objectif principal de cette méthode est de réduire les coûts du changement. Elle met l'accent sur la revue de code (faite en permanence par un binôme), sur les tests(ils sont faits systématiquement avant chaque développement), la conception continue (refactoring), la simplicité, la traduction des besoins en métaphores.

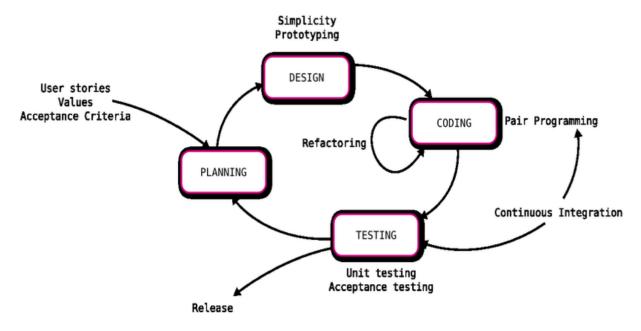


Figure 3 : Processus de la méthode Extreme Programming (XP)

2.1.1.3 Choix de la méthodologie de développement

Nous avons choisi d'utiliser la méthode Scrum car elle implique l'auto-organisation des équipes et permet beaucoup plus de réactivité pour s'adapter aux besoins (parfois changeants) du client. Elle sous-entend aussi l'application de principes agiles, notamment la transparence, la simplicité et la collaboration.

La méthode Scrum soutient la livraison rapide et régulière de fonctionnalités à haute valeur ajoutée.

Trello est une plateforme de gestion de projet en ligne qui permet de créer des tableaux avec des listes et des cartes pour organiser les tâches à réaliser. Les utilisateurs peuvent ajouter des membres, des dates d'échéance et des pièces jointes à chaque carte pour collaborer efficacement. Trello est disponible sur ordinateur et sur mobile, avec une version gratuite et des options premium payantes. Trello est utilisé par de nombreuses entreprises pour gérer leurs projets et améliorer leur productivité.

Trello vous permet d'implémenter la méthode Scrum en toute simplicité :

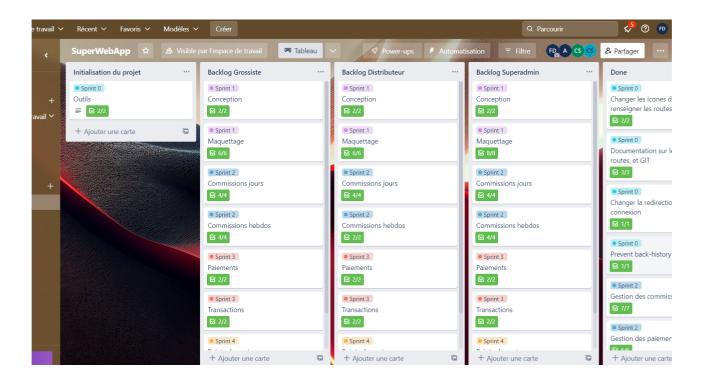


Figure 4: Backlog, Sprint

2.2 Outils

Les logiciels de gestion de projets sont des logiciels ayant pour objectif de faciliter le travail de gestion de projet. Le travail des logiciels de gestion de projet est généralement d'automatiser des tâches de sauvegarde et/ou de la gestion du temps.

Il existe plus outils de gestion parmi lesquels nous allons citer quelque : Trello, Collabtive, Projector...

2.2.1 Outils de gestion de projet

2.2.1.1 Trello

Trello est un outil collaboratif qui organise tous vos projets en une série de listes partagées. D'un seul coup d'œil Trello vous renseignera sur tous vos projets, sur leur état d'avancement et vous dira qui travaille sur quoi dans votre équipe.

Trello se divise en trois (3) éléments principaux :

- Tableaux: Les tableaux regroupent toutes les informations concernant un projet. Ils peuvent être partagés entre plusieurs collaborateurs pour évoluer sur différents projets, et chaque utilisateur peut en créer plusieurs en fonction des équipes ou projets dont il est membre. Les tableaux peuvent prendre différentes formes: classique (cartes et listes), chronogramme (timeline), tableur, calendrier, localisation (basée sur une carte interactive) ou tableau de bord (plusieurs graphiques visuels).
- **Listes** : Le nombre d'ajout possible de listes est illimité et elles peuvent être réorganisées à volonté. Les listes peuvent se déplacer sur le tableau par un système de glisser déposer tout en ayant la possibilité de dupliquer, ou de les archiver.
- Cartes: Les cartes sont situées dans les listes. On peut ajouter dans chaque carte des descriptions et des commentaires, mais aussi des pièces jointes, checklists, liens, étiquettes, échéances, notifications et autant de personnes que nécessaires. On peut aussi créer autant de cartes que nous le souhaitons et les déplacer dans les listes grâce à un système de glisser déposer. Il est possible d'ajouter un deadline à chaque carte, et d'y assigner un ou plusieurs membre(s) du tableau. Les membres assignés à une carte recevront les notifications associées aux actions qui seront faites sur la carte. Les cartes sont personnalisables avec l'ajout de couleur pour les différenciées.

2.2.1.2 Collabtive

Collabtive est doté d'une messagerie interne, ce logiciel open source vous permet de gérer les projets, les tâches ainsi que les utilisateurs. Vous pourrez également suivre les activités et les heures pour ensuite éditer des rapports.

2.2.1.3 Projector

Projector est un organisateur de projet Open-Source. Son nom est issu de « Quality based Project Organizer ». Adapté pour tous les types de projets, cet outil de gestion saura particulièrement

satisfaire les personnes ayant un projet informatique.

2.2.1.4 Choix de l'outil de gestion projet

Nous allons utiliser **Trello** car l'application est très pratique, elle fait partie de notre quotidien et on ne peut plus s'en passer. On peut coordonner des projets à plusieurs et cet outil se prend en main facilement. On a aussi accès à Trello sur mobile si jamais on doit effectuer des déplacements. Même à distance, on est notifié sur son smartphone dès qu'un collaborateur ajoute ou commente une tâche.

2.2.2 Outils de modélisation

Dans la conception d'un système d'information, la modélisation des données est l'analyse et la conception de l'information contenue dans le système afin de représenter la structure de ces informations et de structurer le stockage et les traitements informatiques. Il y a une grande variété d'outils de modélisation qui existe mais nous citerons quelques outils : Star UML, Astah UML, Umbrello, Power AMC...

2.2.2.1 Star UML

Star UML est un outil de modélisation UML gratuit et open source. L'interface utilisateur de Star UML est conviviale et facile à utiliser, ce qui facilite la création de diagrammes UML, et est disponible pour Windows, Mac et Linux, ce qui le rend accessible à un large éventail d'utilisateurs.

2.2.2.2 Astah UML

Astah UML est un logiciel payant qui permet de créer facilement et rapidement n'importe quel diagramme, graphique ou illustration dont votre entreprise a besoin et fonctionne également sur plusieurs plates-formes avec les fonctionnalités de rétro-ingénierie, de génération de code, d'exportation de documents HTML et RTF.

2.2.2.3 Umbrello

Umbrello est un logiciel libre qui vous permettra d'effectuer des modélisations UML. Il s'agit d'un langage utilisé dans le cadre du développement logiciel afin de représenter un modèle informatique sous la forme de pictogrammes.

2.2.2.4 Choix de l'outil de modélisation

Pour faire la modélisation de notre application nous utiliserons **Star UML**. IL offre une large gamme de fonctionnalités pour la modélisation UML, y compris la création de diagrammes de classes, de cas d'utilisation, de séquence, d'activité, de composants et de déploiement. Il permet également de générer du code à partir de ces diagrammes.

2.2.3 Système de gestion de base de données

Un système de gestion de base de données est un logiciel système servant à stocker, à manipuler ou gérer, et à partager des données dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations. Cependant il existe plusieurs SGBD : Mysql, Oracle, MongoDB...

2.2.3.1 **MySQL**

MySQL est un système de gestion de bases de données relationnelles (**SGBDR**). Il est distribué sous une double licence GPL (La licence publique générale GNU) et propriétaire.

2.2.3.2 **Oracle**

Oracle Database est un système de gestion de base de données relationnelle qui depuis l'introduction du support du modèle objet dans sa version 8 peut être aussi qualifié de système de gestion de base de données relationnel-objet.

2.2.3.3 MongoDB

MongoDB est un système de gestion de base de données orienté documents, répartissable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données.

2.2.3.4 Choix du SGBD

Pour gérer la gestion de données de notre application, nous utiliserons **MySQL**. Le serveur MySQL est très rapide. Des tests de performances sont disponibles sur le site de MySQL. MySQL est beaucoup plus simple à utiliser que la plupart des serveurs de bases de données commerciaux. On peut effectuer diverses opérations sur une base MySQL en utilisant des interfaces écrites en C, Perl, C++, Java, Python, PHP.

2.1.1.1 Choix de l'outil de gestion de base de données

Nous avons choisi **DBeaver** qui est un outil de gestion de base de données multiplateforme et gratuit. Il permet aux utilisateurs de se connecter à différents types de bases de données, tels que MySQL, PostgreSQL, Oracle, SQL Server, MongoDB, Cassandra, et bien d'autres.

Les utilisateurs peuvent exécuter des requêtes SQL, gérer des tables et des vues, ainsi que créer, modifier et supprimer des enregistrements de base de données. L'outil prend également en charge la migration de données entre les différentes bases de données et permet l'exportation de données dans différents formats de fichiers.

DBeaver est populaire parmi les développeurs et les administrateurs de bases de données pour sa facilité d'utilisation et sa polyvalence, en particulier pour la gestion de multiples bases de données de différents types.

2.2.4 Les serveurs WEB

Un serveur web est un logiciel informatique qui répond aux demandes des clients web (tels que des navigateurs web) en envoyant des fichiers web (tels que des pages HTML, des images, des vidéos, etc.) via le protocole HTTP (Hypertext Transfer Protocol).

Lorsque vous accédez à un site web, votre navigateur envoie une demande de fichier au serveur web qui héberge le site. Le serveur web répond ensuite en envoyant les fichiers demandés, qui sont affichés dans votre navigateur.

Il existe de nombreux logiciels de serveur web populaires, notamment Apache, Nginx. Les serveurs web sont utilisés pour héberger des sites web de toutes tailles, allant des sites personnels aux sites d'entreprises et aux sites web à haute trafic.

2.2.4.1 Apache

Apache est l'un des serveurs web les plus populaires au monde. Il s'agit d'un logiciel opensource, c'est-à-dire qu'il est disponible gratuitement et que son code source peut être consulté, modifié et distribué par quiconque. Il est un serveur web très fiable et robuste, qui a été largement adopté par les professionnels de l'informatique depuis de nombreuses années.

2.2.4.2 Nginx

Nginx (prononcé "engine-x") est un serveur web open-source populaire qui a été conçu pour être rapide, léger et performant. Il est souvent utilisé comme proxy inverse (reverse proxy) ou comme équilibreur de charge (load balancer) pour des sites web à forte demande.

Il est connu pour sa capacité à gérer des connexions simultanées avec peu de ressources système, ce qui en fait un choix populaire pour les sites web à haute trafic. Il est également capable de gérer de nombreux types de contenu statique et dynamique, y compris les fichiers HTML, les images, les vidéos, les scripts PHP et les applications web en temps réel.

De nombreux grands sites web, tels que Netflix, Airbnb et WordPress.com, utilisent Nginx pour gérer leur trafic web en raison de sa vitesse et de sa capacité à gérer des connexions simultanées avec peu de ressources système.

2.2.4.3 Choix du serveur WEB

Le choix d'un serveur web dépend de nombreux facteurs, tels que la taille du site, les fonctionnalités requises, les compétences techniques disponibles, les ressources système disponibles, etc. Il n'y a pas de "meilleur" serveur web, mais celui qui convient le mieux à vos besoins. Notre choix porte sur Apache qui offre une grande flexibilité et de nombreuses fonctionnalités, notamment la prise en charge de nombreux protocoles, modules et extensions, ainsi qu'une grande capacité de personnalisation. Il peut également être utilisé en conjonction avec d'autres logiciels open-source, tels que PHP, MySQL et Perl, pour créer des applications web dynamiques. Il est connu pour être un serveur web sécurisé, avec de nombreuses fonctionnalités de sécurité intégrées et une grande attention portée à la sécurité.

2.2.5 Les éditeurs de code

Un éditeur de code source est un programme d'édition de texte spécialement conçu pour l'édition du code source d'un programme informatique. Il existe plusieurs types d'éditeur de code : Visual Studio Code, Sublim Text, Brackets ...

2.2.5.1 Visual Studio Code

Visual Studio Code un éditeur de code multiplateforme édité par Microsoft. Cet outil destiné aux développeurs supporte plusieurs dizaines de langages de programmation comme le HTML, C++, PHP, Javascript, Markdown, CSS, etc.

2.2.5.2 Sublime Text

Sublime text est un éditeur de texte générique codé en C++ et Python, disponible sur Windows, Mac et Linux. Le logiciel a été conçu tout d'abord comme une extension pour Vim, riche en fonctionnalités.

2.2.5.3 Brackets

Brackets est un éditeur open source pour le web design et le développement sur des technologies Web telles que HTML, CSS et JavaScript.

2.2.5.4 Choix de l'éditeur de code

Pour écrire le code de notre application nous allons choisir Visual Studio code, car ce dernier est facile à installer, à comprendre, à utiliser et rapide, Il dispose d'une interface graphique responsive et customisable via des thèmes déjà installés.

VS Code assiste les développeurs tout au long de leurs développements afin de simplifier l'édition du code, L'appel des propriétés, ou encore sur l'importation des dépendances. Ce système est disponible par défaut pour plusieurs langages mais aussi peut être configuré pour d'autres.

2.2.6 Outils de travail collaboratif

Les dépôts distants, communément appelées « remotes », permettent de collaborer à plusieurs sur un même projet. De plus, comme une copie du dépôt est synchronisée sur une autre machine, cela permet de s'assurer d'avoir une sauvegarde du projet.

En cas d'incident, il sera toujours possible de récupérer le dépôt depuis la copie distante. Il existe

plusieurs outils intéressants : GitHub, GitLab, BitBucket...

2.2.6.1 Github

GitHub est une plateforme open source de gestion de versions et de collaboration destinée aux développeurs de logiciels.

2.2.6.2 **GitLab**

GitLab est un logiciel libre de forge basé sur git proposant les fonctionnalités de wiki, un système de suivi des bugs, l'intégration continue et la livraison continue.

2.2.6.3 BitBucket

BitBucket est un service web d'hébergement et de gestion de développement logiciel utilisant le logiciel de gestion de versions Git.

2.2.6.4 Choix de l'outil de travail collaboratif

Pour travailler avec un dépôt distant nous utilisons **GitHub**. Ce dernier est un outil de communication et de collaboration entre plusieurs développeurs (ou tout autre personne qui écrit du texte). C'est une interface web créée pour faciliter l'interaction avec Git.

2.2.7 Outils de maquettage

Le maquettage est le processus de création de représentations visuelles d'un produit ou d'un système, généralement dans le but de tester et de valider son fonctionnement et son apparence avant sa production ou son développement complet. Il est souvent utilisé dans la conception de produits tels que les logiciels, les sites web, les applications mobiles, les produits physiques, etc. Le maquettage peut être divisé en trois parties principales : le wireframe, la maquette et le prototype.

- Le wireframe : il s'agit d'une représentation visuelle basique et simplifiée de l'interface utilisateur ou de la structure d'un produit. Le wireframe se concentre sur la disposition des éléments et la hiérarchie de l'information plutôt que sur l'apparence visuelle. Il peut être créé à la main ou avec des outils de conception assistée par ordinateur (CAO) tels que Sketch ou Adobe XD.
- La maquette : elle représente une version plus détaillée et esthétique du produit, incluant l'apparence visuelle et le design graphique. Elle peut inclure des couleurs, des images, des polices de caractères et d'autres éléments visuels pour donner une idée plus précise du produit final. Les maquettes sont souvent créées à l'aide d'outils de conception graphique tels que Photoshop ou Figma.
- Le prototype : il s'agit d'une version interactive et fonctionnelle du produit, qui permet aux utilisateurs de tester et d'interagir avec le produit de manière similaire à celle qu'ils auraient avec le produit final. Le prototype peut être créé à l'aide d'outils de prototypage tels que InVision ou Axure. Les prototypes peuvent être utilisés pour tester la fonctionnalité du produit, recueillir des

commentaires des utilisateurs et valider la conception avant la production ou le développement complet du produit. En somme, ces trois parties du maquettage sont étroitement liées et sont souvent utilisées ensemble pour créer un produit final plus abouti et efficace.

Choix de l'outil pour les maquettes 2.2.7.1

Nous avons choisi Figma pour faire nos maquettes. Figma est un outil de conception d'interface utilisateur (UI) et de conception d'expérience utilisateur (UX) basé sur le cloud. Il permet aux concepteurs de créer des maquettes, des prototypes et des designs graphiques pour des produits tels que des sites web, des applications mobiles, des logiciels et des produits physiques.

Figma offre une interface utilisateur intuitive et facile à utiliser, ainsi que des fonctionnalités collaboratives qui permettent aux équipes de travailler ensemble en temps réel, de partager des maquettes et des prototypes et de recueillir des commentaires des parties prenantes.

2.3 Technologies

L'un des domaines les plus dynamiques et évolutifs dans l'industrie du développement logiciel est le développement Web. Les technologies de développement Web offrent une grande variété d'outils et de frameworks pour la création de sites et d'applications Web modernes et performants.

2.3.1 Les langages

Le langage PHP 2.3.1.1



PHP (Hypertext PreProcessor) désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques. Il s'agit d'un langage de programmation sous licence libre qui peut donc être utilisé par n'importe qui de façon totalement gratuite.

2.3.1.2 Le langage Ruby

Ruby est un langage de programmation open-source qui est idéal pour créer Ruby est un langue and des tâches simples comme la création d'un blog. Il est indépendant de la plateforme et fonctionne sur Unix, Mac et Windows. Créé dans les années 1990 par Yukihiro Matsumoto, Ruby est l'un des meilleurs choix pour le prototypage.

Le langage Python 2.3.1.3



Python est un langage de programmation puissant et facile à apprendre. Il dispose de structures de données de haut niveau et permet une approche simple mais efficace de la programmation orientée objet. Parce que sa

syntaxe est élégante, que son typage est dynamique et qu'il est interprété.

2.3.1.4 Choix du langage

Notre choix sur la technologie porte sur **PHP**, car c'est un langage de programmation à code source ouvert, stable et complet qui produit du contenu dynamique. Ses qualités en font l'un des langages préférés et les plus populaires dans la communauté des développeurs de sites web. Il est compatible avec les systèmes Mac, Windows et Unix. PHP s'intègre nativement avec MySQL, Oracle, Microsoft SQL Server et d'autres bases de données. Depuis quelques années Les programmeurs PHP se tournent souvent vers un framework PHP pour composer leur code.

2.3.2 Les framework

Un framework est, comme son nom l'indique en anglais, un "cadre de travail". L'objectif d'un framework est généralement de simplifier le travail des développeurs informatiques (les codeurs si vous préférez), en leur offrant une architecture "prête à l'emploi" et qui leur permette de ne pas repartir de zéro à chaque nouveau projet.

Les frameworks PHP suivent généralement le modèle de conception du contrôleur de vue de modèle (Model View Controller ou MVC). Ce concept sépare la manipulation des données de leur présentation. En termes d'application PHP, le MVC pourrait correspondre à ce qui suit :

- ✓ Modèle : une base de données
- ✓ Vue : une ou plusieurs pages HTML
- ✓ Contrôleur : fonctions d'accès et de mise à jour de la base de données.

Cependant il existe 2 grands frameworks PHP à savoir Laravel et Symfony :

2.3.2.1 Laravel

Laravel est un framework PHP open-source qui utilise une architecture modulaire et offre une grande variété de fonctionnalités intégrées pour le développement d'applications Web, notamment la gestion des routes, des sessions, des migrations de bases de données, l'authentification, la sécurité, la gestion des erreurs, etc.

Le framework est également livré avec une bibliothèque d'outils et de classes pour aider les développeurs à écrire du code efficace et de qualité, tels que l'ORM (Object-Relational Mapping) Eloquent pour la gestion des bases de données, Blade pour le moteur de templates, et de nombreuses autres fonctionnalités.

2.3.2.2 Symfony

Symfony est construit autour de composants réutilisables, ce qui signifie que les développeurs peuvent facilement composer des applications en utilisant uniquement les composants dont ils ont besoin. Il fournit également une grande variété de fonctionnalités intégrées pour le

développement Web, notamment la gestion des routes, la gestion des formulaires, l'ORM (Object-Relational Mapping), la sécurité, la gestion des sessions, etc.

Le framework est également livré avec une grande bibliothèque d'outils et de classes pour aider les développeurs à écrire du code efficace et de qualité, tels que Twig pour le moteur de templates, Doctrine pour la gestion des bases de données, et de nombreuses autres fonctionnalités.

2.3.2.3 Choix du framework

Notre choix porte sur le **framework Laravel** qui est connu pour sa facilité d'utilisation, sa documentation complète et sa communauté active de développeurs. Il est également compatible avec de nombreux outils et bibliothèques PHP populaires, tels que Composer pour la gestion des dépendances et PHPUnit pour les tests unitaires.

2.4 Analyse et conception

L'analyse et la conception du système sont des étapes importantes dans le développement d'un système informatique. L'analyse du système permet de déterminer les besoins et les exigences du système, tandis que la conception du système implique la création de l'architecture et de la structure du système informatique.

2.4.1 Analyse du système

L'analyse du système implique l'examen approfondi des besoins et des exigences du système informatique. Cette étape permet de déterminer les fonctionnalités et les caractéristiques que doit posséder le système pour répondre aux besoins des utilisateurs.

Les étapes clés pour l'analyse du système sont les suivantes :

- Identification des besoins : Cette étape consiste à identifier les exigences et les besoins des utilisateurs. Cela peut être fait en organisant des réunions avec les parties prenantes, en faisant des enquêtes ou en utilisant d'autres techniques de collecte de données.
- Analyse des processus : L'analyse des processus implique la compréhension des processus métier actuels et des problèmes qui y sont associés.
- Évaluation de la faisabilité : Cette étape implique l'examen des options disponibles et la sélection de la solution la plus appropriée.

2.4.2 Conception du système

La conception du système implique la création de l'architecture et de la structure du système informatique. Cette étape utilise les résultats de l'analyse pour définir les spécifications et les exigences du système.

Les étapes clés pour la conception du système sont les suivantes :

- **Définition des spécifications** : Cette étape consiste à définir les spécifications techniques du système, telles que les fonctionnalités, les performances, la sécurité et la convivialité.
- Architecture du système : L'architecture du système définit la structure globale du système informatique, y compris la disposition des modules et la manière dont ils interagissent.
- Conception de l'interface utilisateur : La conception de l'interface utilisateur implique la création d'une interface utilisateur intuitive et facile à utiliser pour le système.
- **Prototypage et tests** : Cette étape implique la création d'un prototype du système pour tester sa fonctionnalité et sa convivialité.

2.4.2.1 Présentation d'UML

UML (**Unified Modeling Language**) est un langage de modélisation visuelle utilisé pour représenter des systèmes logiciels.

UML est utile pour communiquer efficacement les conceptions logicielles à différents membres de l'équipe de développement. Il peut aider à identifier les lacunes dans les conceptions et à faciliter la collaboration entre les membres de l'équipe.

Il comprend plusieurs types de diagrammes pour modéliser différents aspects du système :

2.4.2.1.1 Le diagramme de contexte

Le diagramme de contexte délimite le domaine d'étude en précisant ce qui est à la charge du système et en identifiant l'environnement extérieur au système étudié avec lequel ce dernier communique (agis ou interagis).

Ses composants sont :

- Les acteurs externes : Un acteur externe est une entité externe au système étudié qui interagit avec le système.
- Un processus unique symbolisant le Système Information étudié :
- Echange entre le système étudié et son environnement

Officiellement le diagramme de contexte n'est pas considéré comme un diagramme UML. Mais il est très utile pour visualiser les acteurs.

Il est important aussi de noter que dans le diagramme de contexte, on représente les interactions des acteurs avec le système étudié. Mais pas les interactions entre acteurs.

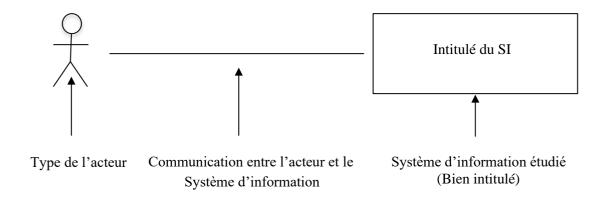


Figure 5: Formalisme diagramme de contexte

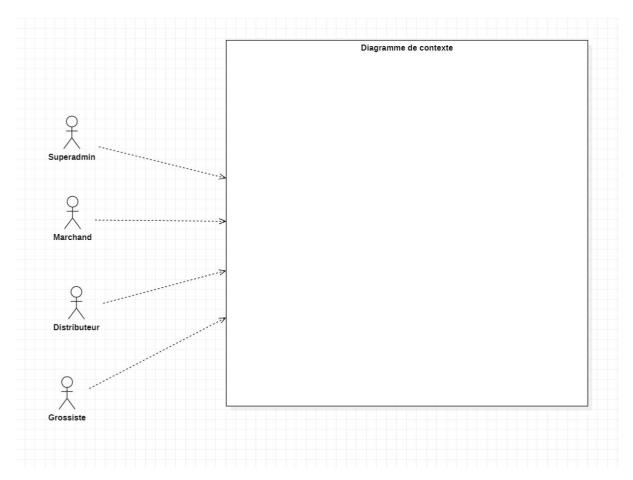


Figure 6 : Diagramme de contexte

2.4.2.1.2 Le diagramme de cas d'utilisation

Le diagramme de cas d'utilisation (use case) représente les différentes interactions entre les utilisateurs et le système. Il montre les acteurs, les cas d'utilisation et les relations entre eux.

• Acteurs : que ce soient des personnes ou des systèmes, sont représentés par des bonhommes en fil de fer.

- **Système** : le système auquel se rapporte le cas d'utilisation est représenté par un rectangle.
- Use Case : le cas d'utilisation lui-même est représenté par une ellipse, dans laquelle il y a généralement une courte phrase décrivant le processus.

Une pointe de flèche montre par ailleurs une relation de dépendance entre cas d'utilisation. Une distinction est faite entre ces deux stéréotypes :

- L'association d'inclusion (include) : le cas d'utilisation d'où part la ligne de connexion en pointillés inclut un deuxième cas d'utilisation, pointé par la pointe de flèche.
- L'association d'extension : le cas d'utilisation d'où part la ligne de connexion en pointillés commence peut, sous certaines conditions, étendre le cas d'utilisation pointé par la tête de flèche. Mais ce n'est pas toujours le cas.

2.4.2.1.3 Le diagramme de classe

Considéré comme le plus important de la modélisation orientée objet, le diagramme de classe est le seul obligatoire lors d'une telle modélisation.

Le diagramme de classes est un diagramme statique. Il représente la vue statique d'une application. Il n'est pas seulement utilisé pour visualiser, décrire et documenter différents aspects d'un système, mais également pour construire le code exécutable de l'application logicielle. Le diagramme de classes décrit entre autres les attributs et les opérations d'une classe ainsi que les contraintes imposées au système. Il est important de noter que chaque langage de Programmation orienté objet donne un moyen spécifique d'implémenter le paradigme objet (pointeurs ou pas, héritage multiple ou pas, etc.), mais le diagramme de classes permet de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier. Les diagrammes de classes sont largement utilisés dans la modélisation des systèmes orientés objet car ce sont les seuls diagrammes UML qui peuvent être mappés directement avec les langages orientés objet.

Les principaux éléments du diagramme de classe sont les classes et leurs relations : association, généralisation et plusieurs types de dépendances, telles que la réalisation et l'utilisation.

- Classe: représente un objet ou un ensemble d'objets qui partagent une structure et un comportement communs. Une classe est une représentation d'un objet ou groupe d'objets partageant un ensemble commun de propriétés (les attributs), de comportements (les opérations) et de relations avec d'autres objets (les associations et les agrégations).
- L'héritage : indique qu'une classe B est une spécialisation d'une classe A. La classe B (appelé classe fille, classe dérivée ou sous classe) hérite des attributs et des méthodes

de la classe A (appelée classe mère, classe de base ou super classe). Il se représente par un triangle vide afin d'indiquer le sens de la généralisation (inverse de la spécialisation).

- Un objet : est une instance d'une classe. C'est une entité concrète dotée d'une identité, d'un état et d'un comportement que l'on peut invoquer. Les objets sont des éléments individuels d'un système en cours d'exécution.
- Une association : est une relation entre deux classes (association binaire) ou plus (association n-aire), qui décrit les connexions structurelles entre leurs instances. Une association indique donc qu'il peut y avoir des liens entre des instances des classes associées.
- Une Composition: indique qu'un objet A (appelé conteneur) est constitué d'un autre objet B. Cet objet A n'appartient qu'à l'objet B et ne peut pas être partagé avec un autre objet. C'est une relation très forte, si l'objet A disparaît, alors l'objet B disparaît aussi.
- L'agrégation: indique qu'un objet A possède un autre objet B, mais contrairement à la composition, l'objet B peut exister indépendamment de l'objet A. La suppression de l'objet A n'entraîne pas la suppression de l'objet B. L'objet A est plutôt à la fois possesseur et utilisateur de l'objet B.
- Les cardinalités : sont des couples de valeur que l'on trouve entre chaque entité et ses associations liées. Donc, pour une association de 2 entités, il y a 4 cardinalités à indiquer (2 de chaque côté). Il y a trois valeurs typiques : 0, 1 et N (plusieurs). Pour les associations à 2 entités, ce sont des valeurs qui permettent d'indiquer combien de fois au minimum et au maximum une occurrence d'entité peut être liée à une autre occurrence d'entité.

Cardinalité	Signification
01	Zéro ou une fois
11 (ou 1)	Une et une seule fois
0* (ou *)	De zéro à plusieurs fois
1*	De une à plusieurs fois
mn	Entre m et n fois
nn (ou n)	n fois

Tableau 1: Les cardinalités dans un diagramme de classe

De manière plus générale, les cardinalités d'une entité dans une association expriment le nombre de fois qu'une occurrence de cette entité peut être impliquée dans une occurrence de l'association, au minimum et au maximum.

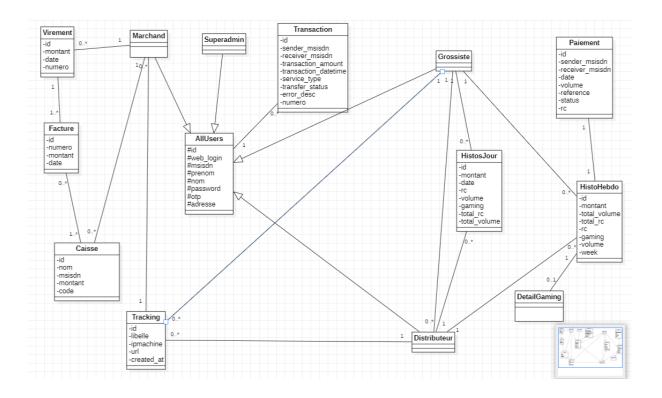


Figure 7 : Diagramme de classe

2.4.2.1.4 Le diagramme de déploiement

Le diagramme de déploiement : représente la manière dont les composants logiciels sont déployés sur l'infrastructure matérielle. Il montre les nœuds, les composants et les relations entre eux.

- Un nœud est une ressource matérielle du système. En général, cette ressource possède au minimum de la mémoire et parfois aussi des capacités de calcul (des ressources humaines ou des périphériques sont des ressources modélisées par des nœuds).
- Le chemin de Communication (Association de communication) est donc un lien représenté par une ligne qui permet de modéliser de façon simpliste la communication entre 2 nœuds (liaison Ethernet, USB, série...).
- L'artefact est le terme générique qui désigne n'importe quel élément produit du travail, c'est un élément concret et existant dans le monde réel (document, exécutable, fichier, base de données...). Un artefact se représente par un rectangle contenant son nom et le stéréotype «artifact» (nous pouvons aussi y faire figurer le symbole). Un artefact qui est la manifestation d'un composant est relié à celui-ci par une relation de dépendance (flèche en pointillées) stéréotypé «manifest» orienté de l'artefact vers le composant.

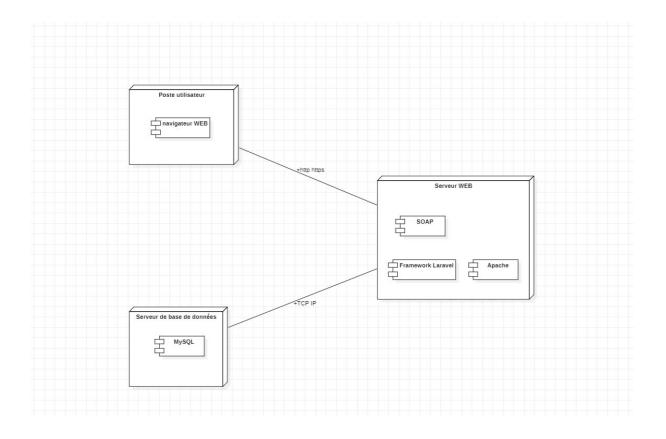


Figure 8 : Diagramme de déploiement

2.4.2.1.5 Déploiement de WEBPARTNER

Le déploiement de notre application web, **WEBPARTNER** codé avec du **Laravel 8** et du **PHP 8** s'est fait en suivant les étapes suivantes :

- Vérifier les prérequis : Avant de commencer, il est important de s'assurer que le serveur d'hébergement dispose des prérequis nécessaires pour exécuter Laravel et PHP 8. Ces prérequis incluent une version récente de PHP, l'extension PHP Mbstring, le gestionnaire de dépendances Composer et la base de données MySQL.
- Cloner le dépôt : Clonez le dépôt de l'application Laravel sur le serveur d'hébergement à l'emplacement souhaité.
- Installer les dépendances : Exécutez la commande "composer install" à partir de la ligne de commande pour installer les dépendances de l'application.
- Configurer l'environnement : Configurez l'environnement en copiant le fichier ".env" et en y ajoutant les informations de configuration de la base de données et d'autres variables d'environnement nécessaires.
- **Générer la clé d'application** : Exécutez la commande "**php artisan key:generate**" pour générer la clé d'application.

- Exécuter les migrations : Si votre application utilise une base de données, exécutez les migrations pour créer les tables nécessaires en exécutant la commande "php artisan migrate".
- Configurer le serveur web : Configurez le serveur web Apache pour servir l'application Laravel en définissant les paramètres de serveur virtuel et en pointant le document **root** sur le répertoire public de l'application.
- Configurer les permissions : Assurez-vous que les fichiers et répertoires nécessaires pour l'application ont les bonnes permissions en exécutant la commande "chmod -R 777 webpartner".
- **Tester l'application** : Enfin, testez l'application pour vous assurer qu'elle fonctionne correctement.

Une fois que ces étapes suivies, notre application **WEBPARTNER** est déployée et prête à être utilisée.

TROISIÈME PARTIE

MISE EN ŒUVRE DE L'APPLICATION

III Mise en œuvre de l'application

3.1 Module 1 : Super Admin

3.1.1 Diagramme de cas d'utilisation

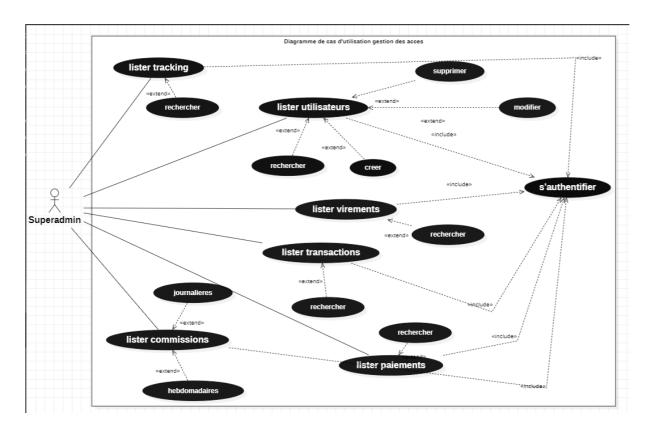


Figure 9 : Diagramme de cas d'utilisation du Super Admin

Ce diagramme de cas d'utilisation permet au Super Admin :

- La possibilité de traquer les actions des autres acteurs du système pour des raisons de sécurité
- La création de comptes miroirs (copie conforme d'un partenaire qui existe déjà dans le système) au besoin
- Une vue globale sur les transactions, les paiements, les commissions et les virements des partenaires. Ainsi que la possibilité de lister et de rechercher.

3.1.2 Maquettage avec Figma



Figure 10: Maquette page paiements

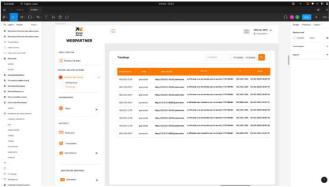


Figure 11: Maquette page Tracking

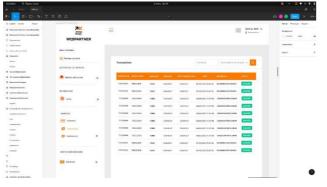


Figure 12: Maquette page transactions

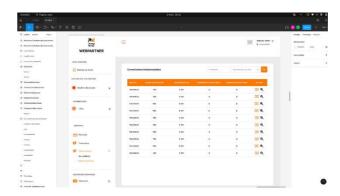


Figure 13: Maquette commissions hebdo

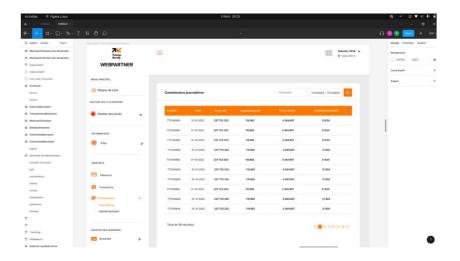


Figure 14: Maquette page commissions journalières

3.1.3 Réalisation

La réalisation de la partie Super Admin a nécessité la mise en place des fonctionnalités de connexion, de création d'utilisateurs, de tracking et des vérifications de transactions, de paiements, de commissions et de virements pour les différents partenaires.

Voici la page de connexion :



Figure 15: Page de connexion

Cette page nous permet de voir les utilisateurs avec les options d'ajout, de suppression et de modification.

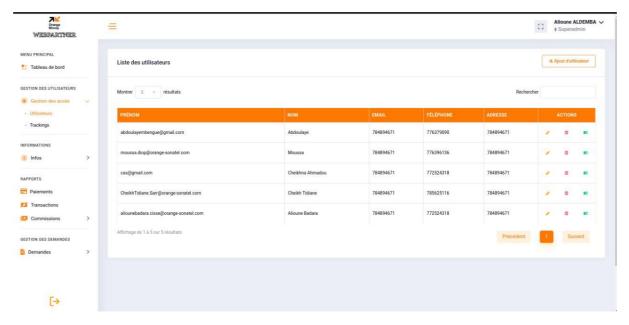


Figure 16: Page de la liste des utilisateurs

Cette page nous permet de créer des utilisateurs.

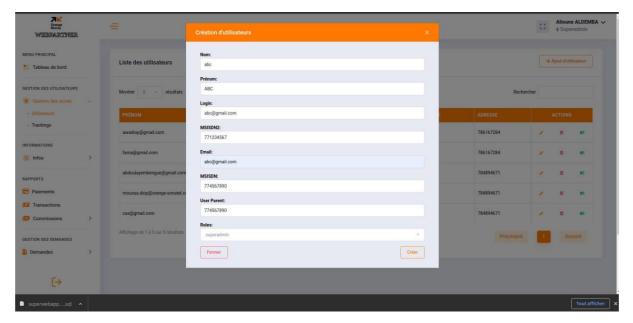


Figure 17 : Création d'utilisateurs

Cette page nous permet de voir les actions que les utilisateurs ont effectuées dans l'application.

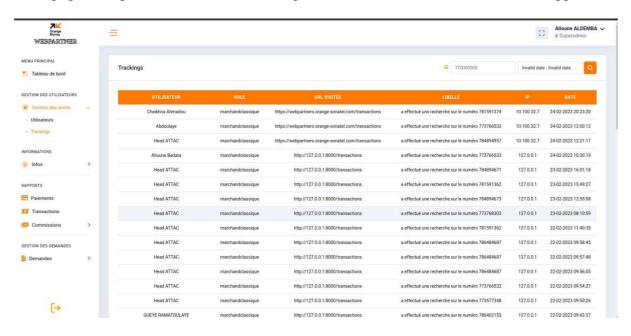


Figure 18: Page des trackings

Cette page nous permet de voir les paiements.

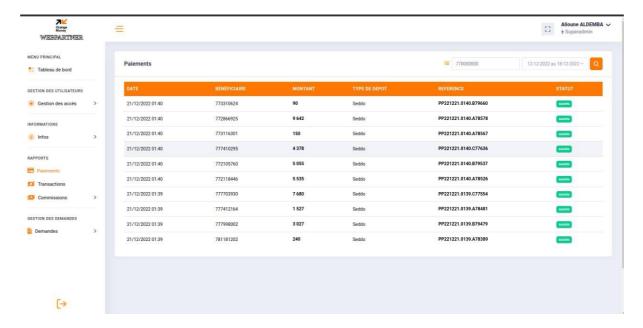


Figure 19: Page des paiements

Cette page nous permet de voir les commissions hebdomadaires des distributeurs ou des grossistes.

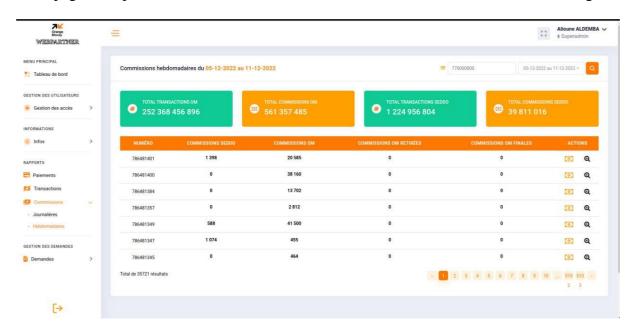


Figure 20: Page des commissions hebdomadaires

3.2 Module 2 : Grossiste et Distributeur

3.2.1 Diagramme de cas d'utilisation

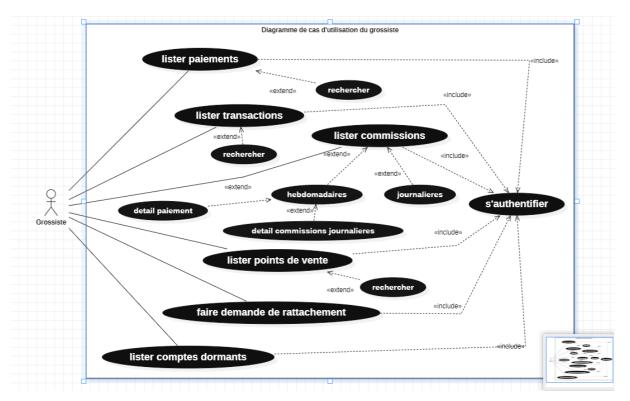


Figure 21 : Diagramme de cas d'utilisation du grossiste

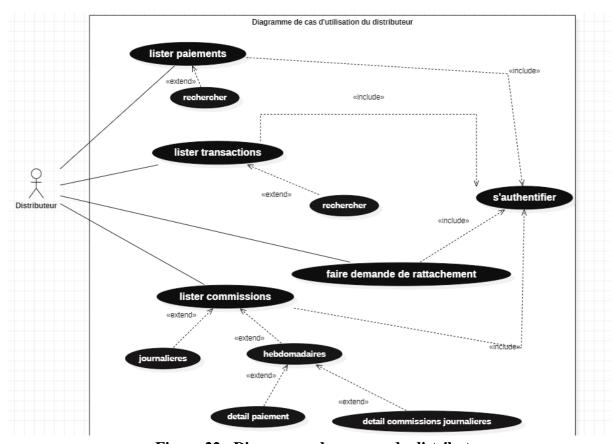


Figure 22 : Diagramme de use case du distributeur

3.2.2 Maquettage avec Figma

3.2.2.1 Maquettes du grossiste



Figure 23: Maquette page de connexion

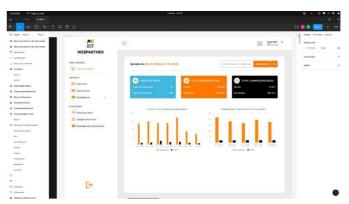


Figure 24: Maquette page d'accueil

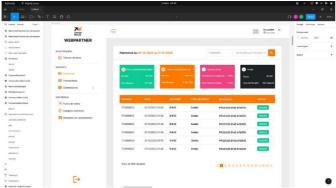


Figure 25: Page de paiements

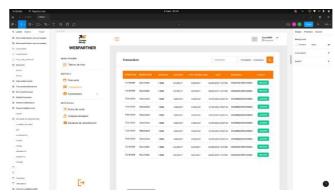


Figure 26: Page des transactions

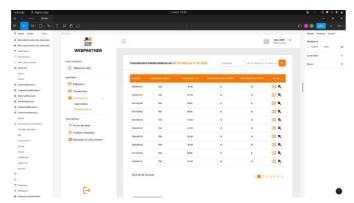


Figure 27: Page des commissions

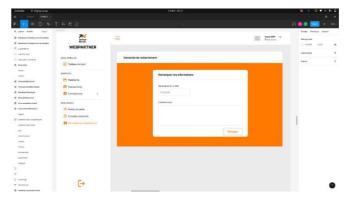
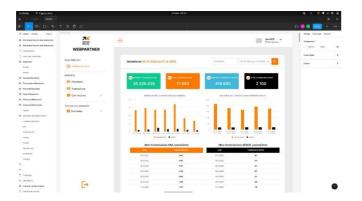


Figure 28 : Page des demandes

3.2.2.2 Maquettes du distributeur



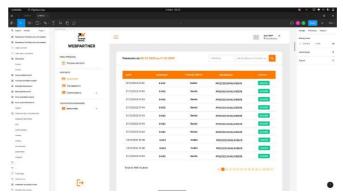
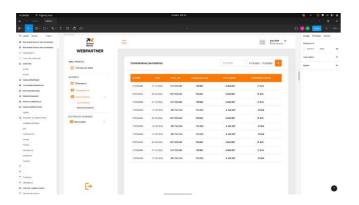


Figure 29: Page d'accueil

Figure 30: Page des paiements



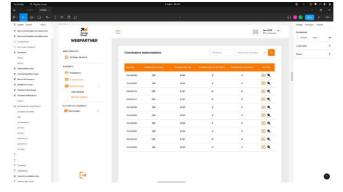


Figure 31: Commissions jours

Figure 32 : Commissions hebdos

3.2.3 Réalisation

Cette page montre la page d'accueil du grossiste avec une vue sur les commissions de sa SIM et de son réseau :

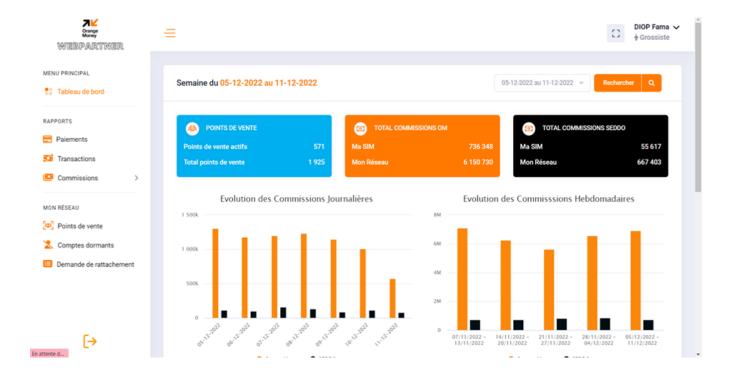


Figure 33: Page d'accueil du grossiste

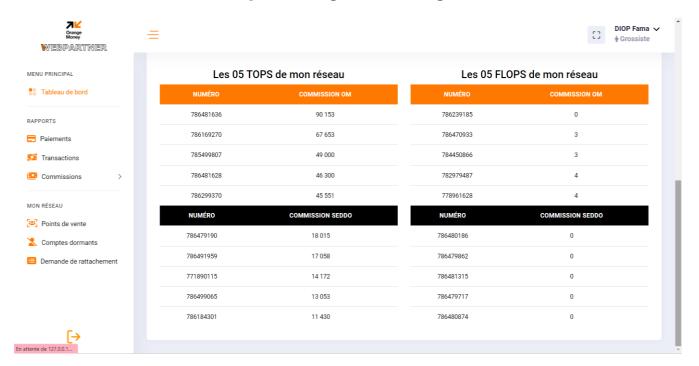


Figure 34: Page d'accueil du grossiste

Cette page montre les paiements du grossiste avec une vue sur les paiements de sa SIM et de son réseau :

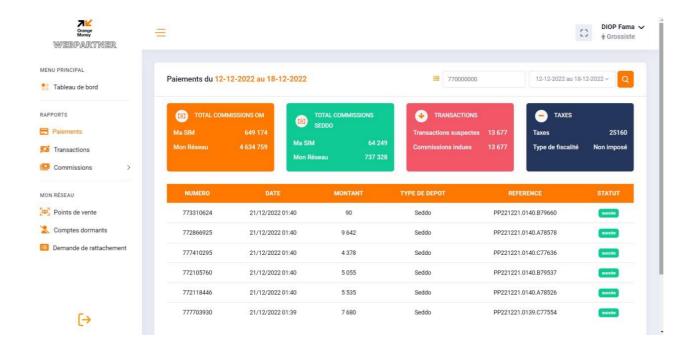


Figure 35 : Page des paiements du grossiste

Cette page montre les transactions du grossiste et de son réseau :

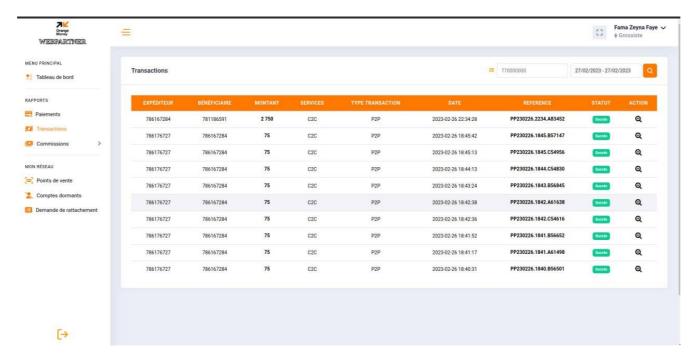


Figure 36: Page des transactions du grossiste

Cette page montre le détail d'une transaction.

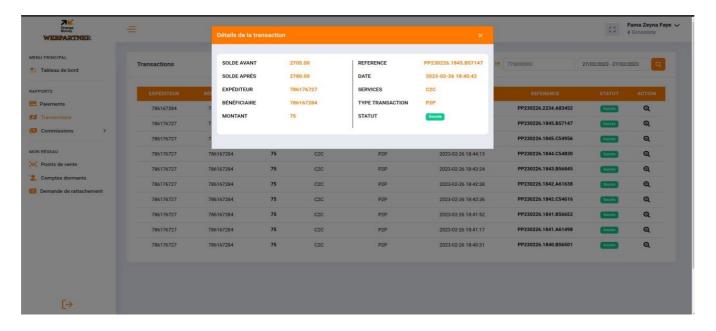


Figure 37 : Détail d'une transaction du grossiste

Cette page montre les commissions journalières du grossiste et de son réseau :



Figure 38 : Commissions journalières du grossiste

Cette page montre les commissions hebdomadaires du grossiste avec une vue sur sa SIM et son réseau :



Figure 39: Commissions hebdomadaires du grossiste

Cette page montre les points de vente du grossistes (ses distributeurs) :

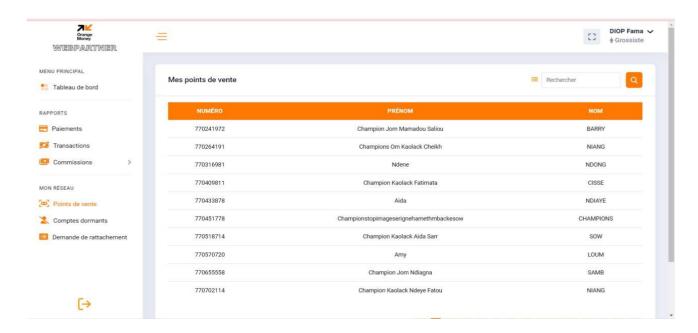


Figure 40 : Points de vente du grossiste

3.3 Module 3: Marchand

3.3.1 Diagramme de cas d'utilisation

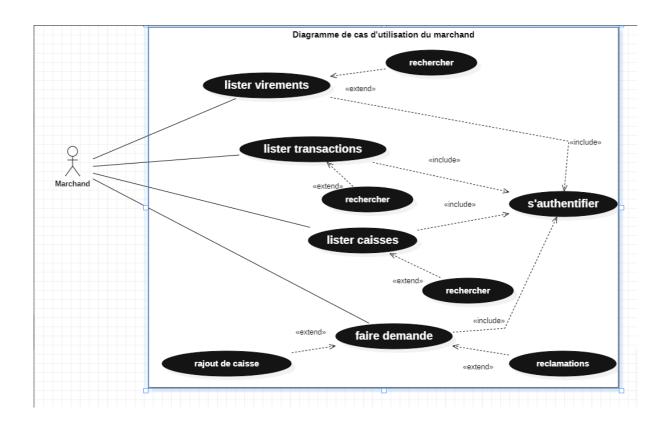


Figure 41: Diagramme de cas d'utilisation du marchand

3.3.2 Maquettes du marchand

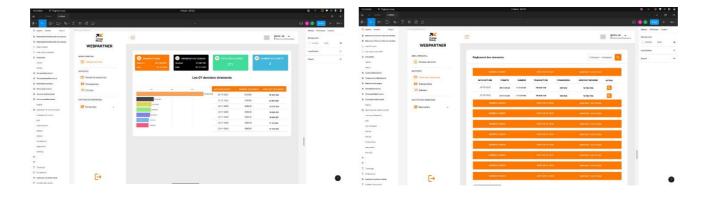


Figure 42: Maquette page d'accueil

Figure 43 : Maquette des virements

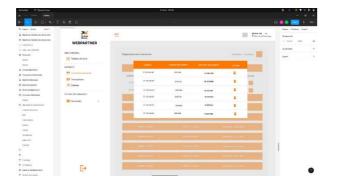


Figure 44 : Page détails caisses

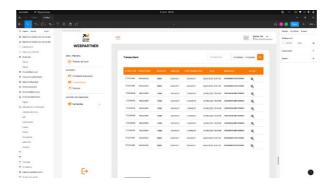


Figure 45: Page des transactions

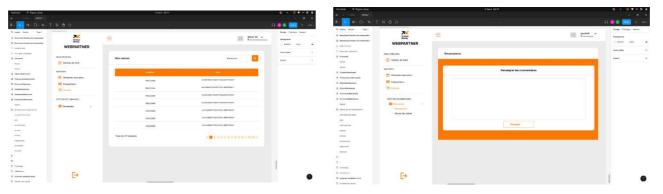


Figure 46: Page liste des caisses

Figure 47 : Page des demandes

3.3.3 Réalisation

Cette page montre la page d'accueil du marchand avec une vue sur ses derniers virements, ses transactions ses caisses et ses nombres de compte.

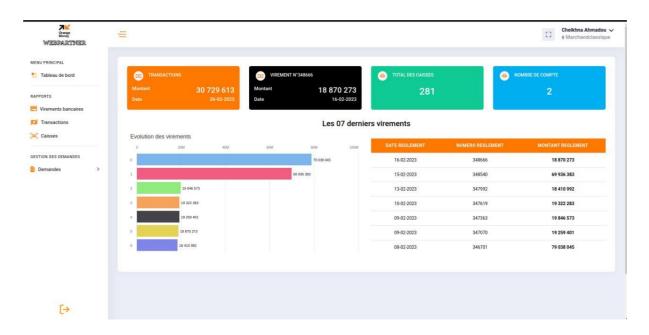


Figure 48: Page d'accueil du marchand

Cette page montre les virements du marchand.

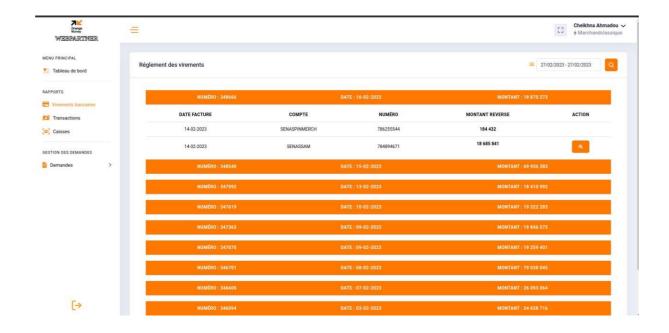


Figure 49: Virements bancaires du marchand

Cette page montre le détail d'un virement bancaire du marchand qui concerne un ensemble de caisses.

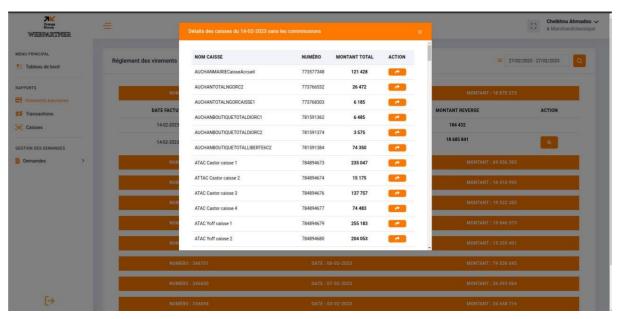


Figure 50 : Détail d'un virement bancaire du marchand

Cette page montre les transactions du marchand.

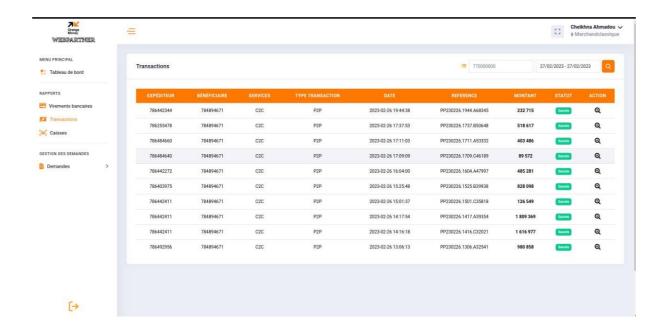


Figure 51: Les transactions du marchand

Cette page montre le détail d'une transaction du marchand.

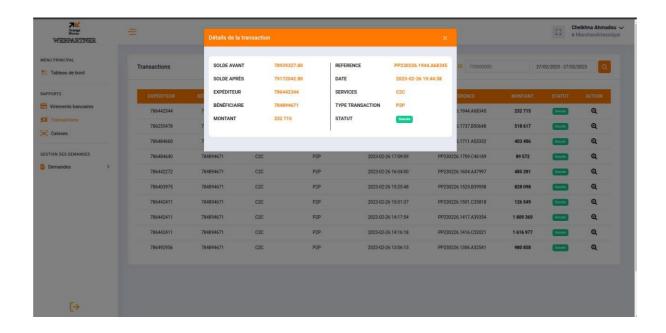


Figure 52: Le détail d'une transaction du marchand

Cette page montre la liste des caisses du marchand.

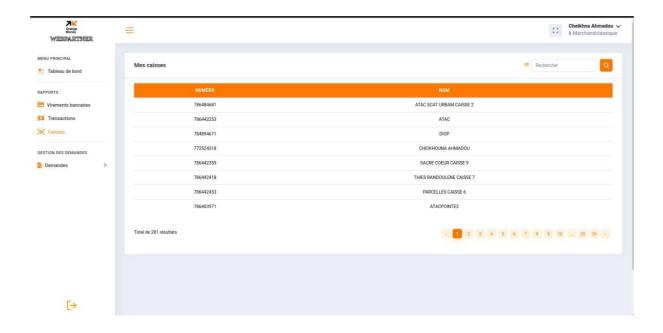


Figure 53: Liste des caisses du marchand

Cette page montre les demandes de réclamations

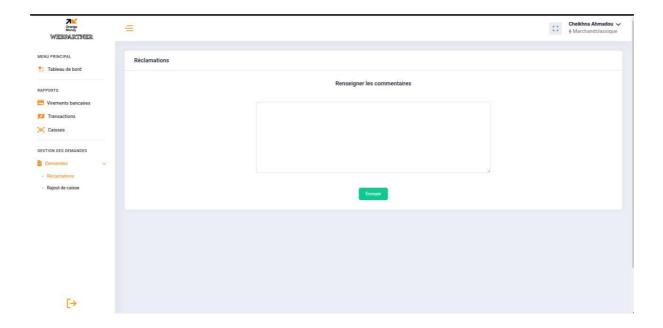


Figure 54 : Demande de réclamation du marchand

Conclusion générale

Au terme de notre analyse, nous avons constaté que les partenaires d'OFMS manquent de visibilité sur leurs opérations et rencontrent des difficultés pour gérer leurs finances. Notre problématique a donc été de savoir comment OFMS peut aider ses partenaires à améliorer leur performance financière et à mieux comprendre leur activité, en leur fournissant des informations précises.

Pour répondre à cette question, nous avons identifié plusieurs objectifs, notamment fournir une vue 360 des transactions et des gains, améliorer la transparence des informations fournies, réduire les temps d'attente pour les demandes de support et simplifier les recherches d'opérations.

Nous avons également conclu que la qualité de l'impact de notre travail sur l'organisation dépendra de la mise en œuvre de ces objectifs et de la satisfaction des partenaires. Si ces derniers obtiennent une visibilité accrue sur leurs opérations, ils pourront mieux gérer leur entreprise et prendre des décisions plus éclairées.

Cependant, notre analyse a rencontré certaines limites, notamment en ce qui concerne l'accès à des données précises et exhaustives sur les partenaires et leurs opérations. Nous avons également constaté que certains problèmes peuvent être dus à des problèmes internes de l'organisation, tels que des problèmes de communication avec le service client.

Enfin, il est important de souligner que cette problématique n'est pas propre à OFMS et peut également s'appliquer à d'autres entreprises dans le secteur des services financiers. Des perspectives d'ouverture peuvent être envisagées, telles que la mise en place d'un réseau de partenaires pour échanger des bonnes pratiques et des solutions communes à l'ensemble du secteur.

Bibliographie

BA Amadou Tidjane, DIEND Cheikh, DIA Mohamadou Lamine, *Plateforme de réservation et de commande pour les restaurants*, SONATEL ACADEMY, 2019-2020, 45 pages.

SECK Abdoulaye, *Application de commande et de livraison pour les restaurants*, SONATEL ACADEMY, 2020-2021, 45 pages.

DIALLO Mamadou Ciré, *Conception et réalisation d'un système de sondage électronique*, ISI, 2021-2022, 74 pages.

Webographie

https://www.scribbr.fr/plan-memoire/cadre-theorique-dun-memoire: 19/02/2023, 20h30

https://www.scribbr.fr/memoire/avant-propos-memoire: 23/02/2023, 21h39

https://www.scribbr.fr/plan-memoire/sommaire: 26/02/2023, 12h50

https://www.scribbr.fr/mise-en-page/pagination-word: 26/02/2023, 18h49

Table des matières

I Cadre g	énéral du projet	
1.1	Présentation de OFMS	
1.2	Contexte du projet	4
1.3	Problématique du projet	4
1.4	Objectifs du projet	5
II Méthod	dologie, outils et technologies de développement	
2.1	Méthodologie	
2.1.1	Présentation des méthodes agiles	
2.1.1.1		
2.1.1.2	0, 7	
2.1.1.3		
2.2	Outils	
2.2.1	Outils de gestion de projet	
2.2.1.1		
2.2.1.2		
2.2.1.3 2.2.1.4	•	
2.2.1.4	Outils de modélisation	
2.2.2		
2.2.2.1		
2.2.2.2		
2.2.2.3		
2.2.2.4	Système de gestion de base de données	
2.2.3	MySQL	
2.2.3.1	·	
2.2.3.2		
2.2.3.4		
2.1.1.1		
2.2.4	Les serveurs WEB	
2.2.4.1		
2.2.4.2	·	
2.2.4.3	Choix du serveur WEB	
2.2.5	Les éditeurs de code	
2.2.5.1		
2.2.5.2		
2.2.5.3		
2.2.5.4		
2.2.6	Outils de travail collaboratif	
2.2.6.1	Github	16
2.2.6.2	GitLab	16
2.2.6.3	BitBucket	16
2.2.6.4	Choix de l'outil de travail collaboratif	16
2.2.7	Outils de maquettage	16
2.2.7.1	Choix de l'outil pour les maquettes	17
2.3	Technologies	17
2.3.1	Les langages	17
2.3.1.1	Le langage PHP	17
2.3.1.2	Le langage Ruby	17
2.3.1.3	Le langage Python	17
2.3.1.4	Choix du langage	18
2.3.2	Les framework	18
2.3.2.1	Laravel	
2.3.2.2	, ,	
2.3.2.3	Choix du framework	
2.4	Analyse et conception	
2.4.1	Analyse du système	19
2 4 2		4.0

2.4.2.1	Présentation d'UML	20
2.4.2.1.	1 Le diagramme de contexte	20
2.4.2.1.	2 Le diagramme de cas d'utilisation	21
2.4.2.1.	3 Le diagramme de classe	22
2.4.2.1.	4 Le diagramme de déploiement	24
2.4.2.1.	5 Déploiement de WEBPARTNER	25
III Mise e	n œuvre de l'application	28
3.1	Module 1 : Super Admin	
3.1.1	Diagramme de cas d'utilisation	
3.1.2	Maguettage avec Figma	
3.1.3	Réalisation	
3.2	Module 2 : Grossiste et Distributeur	
3.2.1	Diagramme de cas d'utilisation	32
3.2.2	Maquettage avec Figma	33
3.2.2.1	Maquettes du grossiste	33
3.2.2.2	Maquettes du distributeur	35
3.2.3	Réalisation	35
3.3	Module 3: Marchand	39
3.3.1	Diagramme de cas d'utilisation	39
3.3.2	Maquettes du marchand	40
3.3.3	Réalisation	41
Conclusio	n générale	45
Bibliogra	phie	46
Webogra	phie	47
Table des	matières	48