

PFE

PROJET DE FIN D'ETUDES

Thème : Développement d'un progiciel BeeOne Dédié aux entreprises du secteurs agricole

Entreprise

AGRIDATA-Consulting

Filière

Génie Informatique

Etudiant

Fatima EL MOURTAKI

Encadrant

Mr. Mohamed Salim ELBAZZI

Superviseur

Mr. Nabil AYOUB

Année universitaire : 2021-2022

DÉDICACE

À mes très chers parents

À mon cher frère et ma chère sœur

À tous mes professeurs qui ne cessent jamais de m'encourager et

me faire confiance tout au long de mes études,

à ceux qui attendent mon succès et gardent toujours le contacte

et le soutien malgré tous.

À ma famille À tous personne chère. Je dédie ce travail.

REMERCIEMENTS

Avant tout remerciement, louange à Dieu.

Au terme de ce travail je remercie spécialement Monsieur Mohamed Salim ELBAZZI pour son encadrement pédagogique et ses efforts avec nous durant ce stage. Nous remercions vivement nos professeurs et encadrants pédagogiques au sein de l'École Polytechnique d'Agadir. Nous remercions également les membres du jury pour avoir accepté d'évaluer ce travail, avec certainement beaucoup d'intérêt et rigueur. Merci professeurs pour votre attention habituelle. Je tiens à remercier l'entreprise AGRIDATA CONSULTING MAROC SA de m'avoir accueilli durant mes 4 mois du stage de fin d'études. Et mes sincères remerciements vont également à Madame Altaf EL BLIDI pour sa disponibilité et ses conseils durant ce stage. Je tiens à remercier Monsieur ALAMI Soufiane et Monsieur EL KASSEL Youssef de nous avoir donné la chance pour effectuer notre stage de fin d'études au sein D'AGRIDATA CONSULTING MAROC SA, et Monsieur Nabil AYOUB pour son encadrement, ses disponibilités, ses accompagnements, et ses orientations tout au long du projet. Nos remerciements s'adressent aussi à l'ensemble du corps professoral et administratif de l'École Polytechnique d'Agadir pour l'effort qu'ils fournissent afin de nous garantir une bonne formation et à l'Universiapolis pour tous les services offerts. Je voudrais aussi exprimer ma reconnaissance à tous mes professeurs qui m'ont enseigné tout au long de mes études. Enfin, pour tous ceux qui ont, de près ou de loin, contribué à la réalisation de ce travail, trouvez en ces mots toute ma profonde gratitude.

L'AVANT-PROPOS

Le stage de fin d'études est une expérience pour valider et mettre en pratique les savoirs et les savoir-faire de l'étudiant, en plus de l'évaluation des qualités humaines de l'étudiant surtout sa capacité à travailler en équipe pour réussir une tâche.

Le présent travail est réalisé dans le cadre du stage de fin d'études, ce dernier s'inscrit dans la démarche pédagogique de l'École Polytechnique d'Agadir, créée en 2004, qui a pour vocation la formation des cadres hautement qualifiés dans les domaines liés à l'ingénierie. En tant qu'étudiante à ladite école en troisième année cycle ingénieur Génie informatique spécialisé en Développement logiciel, j'ai passé quatre mois de stage à la société D'AGRIDATA CONSULTING MAROC SA, à la cellule métier de développement de l'ERP Bee One, pour travailler notre projet de fin d'études. Le choix de cette société est raisonné par le fait qu'en tant que future ingénieur en développement logiciel, j'ai la curiosité de découvrir en réalité le domaine des systèmes d'information et logiciels, et les différents métiers qui lui sont reliés. En plus je note une expérience dans une telle société forte dans ce domaine et riche en termes d'expérience développée, et du travail effectué. À travers ce rapport, nous allons essayer de présenter la synthèse de la mission réalisée au cours du stage de fin de formation au sein de l'entreprise AGRIDATA CONSULTING MAROC SA. La mission a porté sur le développement de l'ERP Bee One dans le but de pouvoir une gestion technique et économique pour la production agricole.

TABLE DES MATIÈRES

INTRODUCTION GENERALE	12
CHAPITRE 1 :	14
Contexte du stage	14
2-1- Histoire	15
2-2- Fiche technique	15
2-3- Fiche d'identité	16
2-4- La stratégie de l'entreprise	17
2-5- Les moyens de l'entreprise	17
2-6- Pôle d'activité AGRIDATA consulting	18
2-7- Les produits :	19
2-8- Le choix de l'entreprise	19
CHAPITRE 2 :	21
Diagnostic, Problématique et Analyse	21
2-1- Objectifs	22
2-2- Problématique et cahier des charges	22
3-2-1 Contexte et définition du problème	22
3-2-2 Besoins fonctionnels	23
3-2-3 Besoins non fonctionnels	24
2-3- Élément de risque	24
2-4- Facteur de succès	25
4-1- Choix de la méthodologie	25
4-2- Présentation de la méthodologie SCRUM	26
4-3- Organisation et application de la méthode	28
4-4- Planning du projet	29
4-5- Estimation des durées des tâches et de la durée totale du projet	30
4-6- Enchaînement des tâches	31
4-7- Les outils de gestion de projet utilisé dans l'entreprise	32
4-7-1 L'outil Slack	32
4-7-2 L'outil Clickup	32
CHAPITRE 3 :	34
Recherche bibliographique	34
2-1- Angular	36
2-2- Node.js	36

2-3-	API REST	37
2-4-	Postman	38
2-5-	sass	38
2-6-	Amchart 5	38
2-7-	Base de données : SGBD	38
2-8-	PowerAMC	39
2-9-	VScode	39
2-10-	Git	39
2-11-	GitLab	39
CHAPITRE 4 :		41
Ingénierie, Développement et Méthodologie		41
2-1-	Méthodologie de développement	42
2-2-	Architecture globale du projet	43
3-1-	Choix du modèle de conception	44
3-1-1-	Diagramme de cas d'utilisation	45
3-1-1-1-	Cas d'utilisation général	45
3-1-1-2-	Cas d'utilisation : Gestion des profils	46
3-1-1-3-	Cas d'utilisation : Consultation des parcelles	47
3-1-1-4-	Cas d'utilisation : Suivi de la main d'œuvre.	48
3-1-1-5-	Cas d'utilisation : Réalisation d'un suivi technique	48
3-1-1-	Diagramme de séquences :	49
3-1-2-1	Diagramme de séquences : L'Authentification	49
3-1-2-2	Diagramme de séquences : La gestion des profils	51
3-1-2-3	Diagramme de séquences : Suivi des parcelles agricole	52
3-1-2-	Diagramme de classe :	53
CHAPITRE 5 :		56
Mise en œuvre, Déploiement et implémentation		56
2-1-	Les interfaces graphiques du système	57
CONCLUSION GENERALE		71
BIBLIOGRAPHIE		72
WEBOGRAPHIE		72
ANNEXES		73

LISTE DES FIGURES

Figure 1	Logo de l'entreprise AGRIDATA-consulting	16
Figure 2	Fiche d'identité de l'entreprise AGRIDATA-consulting	17
Figure 3	Organigramme de l'entreprise AGRIDATA-consulting	19
Figure 4	Principe fonctionnel de la méthode agile Scrum	28
Figure 5	Équipe SCRUM de BEEOne	29
Figure 6	Organigramme basé sur le découpage WBS	30
Figure 7	Diagramme de Gantt	31
Figure 8	Mode SaaS	35
Figure 9	Architecture Ngrx du projet	43
Figure 10:	Cas d'utilisation générale	46
Figure 11	Cas d'utilisation : Gestion des profils	47
Figure 12	Cas d'utilisation : consultation des parcelles	47
Figure 13:	Cas d'utilisation : Suivi de la main d'œuvre	48
Figure 14:	Cas d'utilisation Réalisation d'un suivi technique	49
Figure 15:	Diagramme de séquence Authentification	50
Figure 16:	Diagramme de séquence la gestion des profils	51
Figure 17	Diagramme de séquence suivi des parcelles agricole	53
Figure 18:	Diagramme de classe Gestion de la main d'œuvre	54
Figure 19	Interface d'authentification	57
Figure 20	Interface Tableau de consultation des profils	58
Figure 21:	Formulaire Ajout d'un utilisateur	58
Figure 22:	Menu de gestion des profils	58
Figure 23:	Menu principale de BeeOne	60
Figure 24:	Menu secondaire de BeeOne	60
Figure 25:	Interface de consultation des personnels	60
Figure 26:	Formulaire d'ajout des personnels	61
Figure 27:	Interface de gestion des présence	61
Figure 28:	Formulaire de pointage des ouvriers	62
Figure 29:	Interface de suivi du temps	62
Figure 30:	Formulaire de suivi du temps	63
Figure 31:	paramétrage présence obligatoire	63
Figure 32:	Dashboard : Indicateur de performance	64
Figure 33:	graphique linéaire, moyen de l'effectif mensuel par mois	64
Figure 34:	graphique en barre, meilleur 10 opération agricole par effectif total	64
Figure 35:	diagramme circulaire, ratio sexe des ouvriers	65
Figure 36 :	diagramme circulaire : la structure d'âge des ouvriers	65
Figure 37:	interface des produits de la récolte	66
Figure 38:	Formulaire d'ajout des produits récoltés	67
Figure 39:	interface consultation détails d'un produit	67
Figure 40:	interface génération des QR code	68

LISTE DES TABLEAUX

Tableau 1 Fiche d'identité d'AGRIDATA	16
Tableau 2 Identification des risques de projet	23
Tableau 3 Ordonnancement des tâches	30

LISTE DES ABREVIATIONS

UML	Unified Modeling Language
NTIC	Nouvelle Technologies de l'information et de la Communication
SaaS	Software as a Service
IT	Information technology
LGA	Logiciel de gestion agricole
ERP	Enterprise Resource Planning
CDC	Cahier des charges
IHM	Interface Homme machine
SGBD	Système de gestion de base de données
PO	Product Owner
US	User Story
JWT	Json Web Token
WBS	Work Breakdown Structure
Sass	Syntactically Awesome Style Sheets
HTML	HyperText Markup Langage
OPT	Organigramme des tâches du projet
SPA	Single Page Application
JSON	JavaScript Object Notation
Ngrx	Angular Reactive Extensions
API	Application Programming Interfaces
REST	Representational State Transfer
SQL	Structured Query Language
CRUD	Create Read Update Delete
KPI	Key Performance Indicator
RH	Ressource Humaine

RÉSUMÉ

Le travail que nous présentons dans ce rapport de stage de fin d'études s'intègre dans une problématique générale du développement du progiciel BeeOne dédié aux entreprises du secteur agricole.

Le but principal est de mettre à la disposition des entreprises agricoles des solutions qui vont les servir dans la gestion de leur activité dans le secteur agricole, tout en appliquant notre savoir et savoir-faire. Durant la réalisation de cette solution, nous avons adopté la méthode agile SCRUM, UML pour la partie conception. Dans la phase de développement, nous avons utilisé le Framework Angular, le serveur web NodeJs, TypeSCRIPT, et Amcharts pour la visualisation des données. Pour le stockage des données, il est réalisé à l'aide du SGBD Sql Server.

Mot clés : Secteur Agricole, Projet informatique, graphes/Charts, amcharts, analyse et visualisation des données, prise de décision, Angular, NodeJs

ABSTRACT

This work presented in this internship report is part of a broader issue of development of the BeeOne software ERP (Enterprise Resource Planning) dedicated to companies in the agricultural sector

The main goal is to make available to the agricultural company some solutions, which will serve the management of its business process, while applying our knowledge and known-how.

During the implementation of this solution, we adopted the agile SCRUM method and the UML for design method. In the development phase, we used Angular framework and nodeJs as a web server. Moreover, I used amcharts for data visualization and analysis. At the end for data storage, it's carried out using the Sql Server DBMS.

Keywords: Agriculture sector, IT project, graphs/charts, amcharts, data visualization and analysis, decision making, Angular, nodeJs...

INTRODUCTION GENERALE

Le plan Maroc vert lancé en avril 2004 par le gouvernement marocain vise à considérer l'agriculture comme l'un des piliers majeurs dans le développement durable qui mènera à maintenir l'économie du pays toute entière et soutenir la population.

Par conséquent, comme tous les autres secteurs professionnels, l'agriculture a commencé à intégrer l'informatique dans ses méthodes d'exploitation depuis quelques années. Aujourd'hui, l'utilisation de logiciels de gestion devient indispensable, notamment lorsqu'il s'agit des très grosses exploitations, dépassant parfois les 10 millions d'hectares. Parmi ces outils, on retrouve plusieurs types de logiciels qui visent à améliorer ses rendements et ses performances. Certains lui facilitent la gestion quotidienne des ressources matérielles et humaines en les dégageant des soucis administratifs. Il gagne donc en temps et forcément en argent. Ce sont des logiciels qui sont programmés pour gérer plusieurs tâches dans le domaine de l'agriculture en général. Avec des données précises, ils peuvent fournir une multitude d'informations utiles pour accroître la productivité de l'exploitation et mener des études et des plans prévisionnels plus corrects.

En revanche, les solutions de gestion dédiées au secteur agricole sont absentes du marché. La majorité des solutions proposées sont adaptées au secteur de l'industrie et ne prennent pas en considération les besoins du métier agricole. Comment alors une entreprise agricole pourra-t-elle suivre en temps réel la production de sa ferme ? Par quel moyen pourra-t-elle collecter les données de base pour estimer des plans prévisionnels ? Comment gérer sa main-d'œuvre ? et finalement comment gérer efficacement toutes les données techniques et économiques afin d'assurer le suivi opérationnel de l'activité agricole ?

En réponse à ces questions et dans la perspective de digitaliser et d'intégrer les NTIC dans l'agriculture pour optimiser la production et l'exploitation agricole, l'entreprise AGRIDATA Consulting -qui est une société éditeur intégrateur de solutions de gestion web et mobile-propose une solution web dédiée aux entreprises opérantes dans le secteur agricole et agroalimentaire. Il s'agit de suivre et répondre aux besoins de ses clients en termes de performance et de profit et leur faciliter le suivi et la gestion de plusieurs aspects liés au domaine agricole et d'autres liés à tout ce qui est

administratif, et d'autre part, la digitalisation et l'informatisation des données, dans le but de fournir des informations pertinentes et de les manifester sous forme de présentations visuelles et graphiques qui serviront à la prise de décision et à l'élaboration des plans prévisionnels.

Dans cette optique, AGRIDATA a conçu et développé un progiciel servant comme plateforme de gestion complète du pôle agricole. Désormais totalement indispensable pour les entreprises agricoles qui exploitent plus de 10 millions d'hectares. Cet ERP de gestion intégré contient 12 modules (achat, vente, stock, budget, frais généraux, immobilisation, main d'œuvre...), il est en mode SaaS. Son but est d'assurer une gestion technique et économique de la production agricole. Il permet un suivi opérationnel de toutes les activités. D'ailleurs il intègre un moteur analytique performant, traçabilité et performance technique et un contrôle budgétaire ainsi qu'un suivi de la main d'œuvre.

Au long du stage, nous avons conçu le système en tranchant les acteurs principaux en deux protagonistes. Il s'agit du manager qui sera le gérant et l'administrateur du système, et les utilisateurs qui seront des techniciens. Le but du système est de permettre une gestion administrative et un accès global dédié au manager tandis que le technicien pourra visualiser ses domaines d'expertise à travers des tableaux et des graphes. Ces derniers sont construits à l'aide d'une collecte des informations stockés dans la base de données. Par conséquent, les techniciens se trouvent face à des graphes lisibles dont ils peuvent bénéficier pour mener proprement leurs études et leurs plans provisionnels.

Le plan adopté pour ce rapport commence tout d'abord par une introduction servant à donner un périmètre thématique du projet. Ensuite, nous passerons à une analyse fonctionnelle à travers l'élaboration du cahier de charges et la spécification des besoins fonctionnels et non fonctionnels. Une fois le besoin bien exprimé, nous allons enchaîner sur la partie d'étude conceptuelle, à savoir la modélisation UML du projet. Cette dernière sera sous forme d'un diagramme de cas d'utilisations général et d'autres qui seront détaillés, accompagnés d'une description des différents scénarios ainsi que la modélisation chronologique à l'aide des diagrammes de séquences. Pour clore la conception, nous allons dresser le diagramme de classe qui sera le reflet de l'architecture des tables relationnelles en base de données.

Finalement, nous présentons la partie réalisation et la mise en œuvre de la solution faisant objet livrable du projet de fin d'étude.

CHAPITRE 1 :

Contexte du stage

I. INTRODUCTION

Ce premier chapitre illustre le contexte général du projet tout en commençant par la présentation de l'entreprise d'accueil AGRIDATA, son domaine d'activité et son organigramme.

II. PRÉSENTATION DE L'ENTREPRISE

2-1- Histoire

AGRIDATA CONSULTING MAROC SA est une boîte éditrice de solutions logicielles dédiées au secteur agricole et agroalimentaire, elle a comme missions, l'aide de l'agriculture pour être plus rentable, plus compétitive et plus respectueuse de l'environnement grâce à un meilleur usage de l'information. Elle a été créée en fin de 2006 par deux fondateurs. Une année plus tard, la société a lancé la première version de l'ERP LGA (Logiciel de gestion agricole), qui est sous la dénomination actuelle de Bee One. Elle participe au développement de l'agriculture marocaine à travers trois piliers majeurs :

La promotion des TIC (Technologies de l'Information et de la Communication) surtout que la société AGRIDATA CONSULTING MAROC SA est un concepteur, développeur et intégrateur de solutions informatiques destinées au secteur agricole et agroalimentaire.

En outre, la société organise des formations et des séminaires en vue de sensibiliser les différents acteurs aux défis futurs d'organisation et d'efficacité.

En plus des études de projets, à travers lesquelles la société offre un accompagnement et l'audit aux agriculteurs afin de faire profiter le monde agricole de meilleures pratiques de gestion.



Figure 1 Logo de l'entreprise

2-2- Fiche technique

Le tableau suivant présente la fiche technique à savoir les différentes informations de la société AGRIDATA CONSULTING MAROC SA.

Tableau 1 Fiche d'identité d'AGRIDATA

Forme juridique	Société anonyme	
Dirigeants	Mr. ALAMI Soufiane Mr. KASSEL Youssef Mr. LAKHDISSI Mouhsine	
Raison Social	AGRIDATA consulting	
Siège social	2ème étage, Immeuble Aziza, Avenue Hassan II, Agadir	
Capital social	2 079 900 DHS	
Nombre de personnel	25	
Activité	SSII (Société de Services et d'ingénierie en Informatique)	
Contact	Tel	+212 528 82 84 44
	Fax	+ 212528 82 59 49
	Email	contact@agridata-consulting.com
	Web site	www.agrida-consulting.com

La source : réalisée par nos soins après une recherche d'information en interne de l'entreprise.

2-3- Fiche d'identité

Ayant environ 15 années d'expérience dans le domaine agricole, AGRIDATA n'a en aucun moment cessé d'investir dans ses ressources informatiques pour produire des solutions digitales dédiées à ses clients. En effet, AGRIDATA met en œuvre plusieurs atouts informatiques sous forme de plateformes web. Ces dernières sont destinées aux clients agricoles qui s'en serviront pour améliorer leur



Figure 2fiche d'identité de l'entreprise

performance et leur rendement. Il s'agit aussi de la structuration du savoir-faire en termes d'administration et de management. Ceci va permettre à AGRIDATA de se procurer d'un rôle indispensable dans l'évolution de l'agriculture que ce soit au niveau national ou à l'échelle africaine.

2-4- La stratégie de l'entreprise

La stratégie globale de l'entreprise est de mobiliser ses ressources matérielles, logicielles, personnelles dans le but de fournir des solutions durables, pertinentes et optimales. En effet, le profit de l'entreprise est la mise en œuvre des moyens stratégiques qui vont lui permettre de conquérir et de fidéliser ses clients. Par ailleurs, pour comprendre la stratégie générale de l'entreprise AGRIDATA, nous allons citer sa mission, sa vision et son objectif :

- La mission : la politique de l'entreprise est non seulement la digitalisation et l'externalisation des exploitations agricoles mais la création de la valeur ajoutée et s'appuyer sur les concepts afin de les transformer en solution efficace et par conséquent convaincre le secteur agricole des bénéfices de la transformation digitale qui va impacter positivement leurs attentes et leurs profits.
- La vision : AGRIDATA possède une équipe riche en compétences technique et de gestion confirmés dans les domaines informatique, agronomie et finances. Cette diversité de profils est susceptible de répondre aux besoins de ses clients et d'apporter des changements nécessaires à l'évolution.
- L'objectif : élargir son marché de travail, inclure des nouveaux collaborateurs, mettre en valeur l'agriculture marocain à l'échelle nationale aussi bien qu'internationale. Ainsi que le développement d'un progiciel SaaS commercialisable et rentable à l'international et spécialement à Saudia Arabia.

2-5- Les moyens de l'entreprise

AGRIDATA est dotée des moyens qui lui permettent d'atteindre ses finalités en termes de performance et profits, parmi ces moyens nous trouvons :

- La situation géographique de l'entreprise permet à l'entreprise d'exercer son activité en toute flexibilité et de couvrir et d'accroître la proximité avec ses clients, de surcroît AGRIDATA dispose de deux bureaux : un pour couvrir la région de sud qui est à agadir, et l'autre à Casablanca.
- Les ressources matérielle et logicielles : Fourniture de bureau, équipement informatique, salle de réunion et de formation...
- Les ressources humaines : l'entreprise est dotée d'environ 25 employés

- Les directeurs associés
- Les responsables commerciaux
- Les contrôleurs de gestion
- Des ingénieurs informatiques
- Des analystes de données
- Des ingénieurs agronome
- Les comptables
- Une assistante

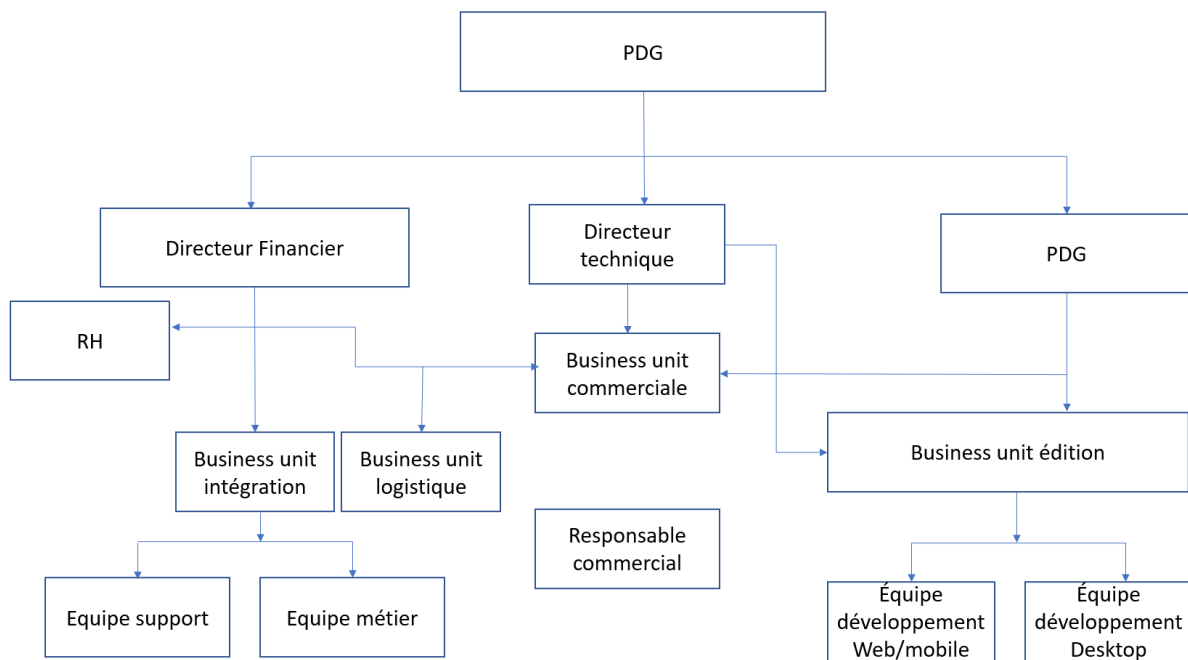


Figure 3 Organigramme de l'entreprise AGRIDATA consulting

2-6- Pôle d'activité AGRIDATA consulting

La nécessité des systèmes d'information pour la gestion des exploitations agricoles devient un atout, pour mettre en valeur les données des entreprises clientes opérantes dans le domaine agricole et agroalimentaire. De ce fait, il faut collecter les données, les traiter, les stocker et les diffuser. Ce processus permet de fournir des informations pertinentes qui vont être exploitées dans la prise de la décision. Par conséquent, Le cabinet AGRIDATA Consulting et un cabinet IT dont les domaines d'activité sont principalement :

- Le développement informatique pour fournir des solutions au secteur agricole et agroalimentaire et intégrer les NTIC dans le secteur agricole à travers l'édition des progiciels de gestion intégré, des systèmes d'information...
- Proposition une large palette de solutions innovants

- Conseil, formation et étude.
- Accompagnement sur l'ensemble du cycle de vie du projet, de la spécification à la recette.

2-7- Les produits :

La société a contribué à la création d'une large gamme de produits de suite LGA. Cette solution technique a pour intérêt de permettre à leurs clients de gérer leur exploitation agricole, parmi ces produits nous citons :



BEE COOP est une plateforme pour la gestion de l'agrégation, c'est une solution qui assure le suivi technique, administratif et financier ainsi qu'une maîtrise des processus avec accès rapide à l'information.



BEE LITE est un outil multilingue, intuitif et facile à manipuler, dédié pour la gestion de trésorerie des petits agriculteurs. Doté d'un assistant intelligent pour bien piloter la ferme et assurer un suivi complet des activités agricoles.



BEE OBS est une solution mobile pour faciliter l'échange et la prise de décision. Il fournit également une vue cartographique pour faciliter la lecture et l'analyse des données. Cette application permet un suivi des opérations par exemple des apports en eau et en engrais, un suivi phytosanitaire pour traiter les ravageurs et les maladies.



BEE AGRI est un réseau social et une plateforme communautaire destinée aux dirigeants de l'agriculture pour partager et échanger leurs expériences liées aux opérations techniques de conduites. Le but de l'application est d'améliorer leurs connaissances sur la technique de traitement contre les maladies et les ravageurs en faisant appel à des certifiés par AGRIDATA.

2-8- Le choix de l'entreprise

Étant reconnue au niveau des entreprises agricoles, AGRIDATA est toujours considérée l'une des destinations les plus préférées des experts en informatique. Ce n'est pas juste parce que le secteur agricole leur présente un sujet challengeant assez complexe dans son côté métier et dans sa conception mais aussi ils sont amenés à se doter d'une certaine détermination à monter en compétences en tout ce qui est nouvelle technologie. AGRIDATA compte parmi ses salariés, une variété de profils IT qui sont spécialisés dans les nouvelles technologies. Par conséquent, c'est un choix pertinent et une opportunité à saisir pour gagner en expérience fonctionnelle (métier complexe) et technique (Nouvelles technologies).

III. CONCLUSION

Dans ce chapitre introductif, nous avons présenté l'organisme d'accueil en termes d'activités et d'organisation. En général, l'entreprise s'implique dans un environnement où il y a des opportunités importantes.

CHAPITRE 2 :

Diagnostic, Problématique et Analyse

I. INTRODUCTION

Après une prise de connaissance de l'entreprise en général, nous avons entamé dans ce chapitre vers la problématique et l'analyse du projet à savoir l'élaboration du cahier de charges et la spécification des besoins fonctionnels et non fonctionnels issus des besoins exprimés par les clients de l'entreprise pour aboutir à la concrétisation des dits objectifs.

II. PRÉSENTATION DU PROJET

2-1- Objectifs

Pour assurer un bon déroulement au niveau de l'échange des informations et augmenter le rendement et l'efficacité des services offerts aux clients de la société AGRIDATA. Le projet vise à concevoir et à mettre en place l'ERP BeeOne, dédiée aux secteurs agricoles national et international.

La problématique de gestion du pôle agricole s'étale sur plusieurs domaines, il contient 12 modules (achat, vente, stock, budget, frais généraux, immobilisation, main d'œuvre...) dans le but de pouvoir une gestion technique et économique de la production agricole. Cet ERP permet un suivi opérationnel de toutes les activités. D'ailleurs il intègre un moteur analytique performant, traçabilité et performance technique et un contrôle budgétaire ainsi qu'un suivi de la main d'œuvre.

2-2- Problématique et cahier des charges

Nous allons associer à cette partie une analyse et spécification des besoins à travers l'élaboration d'un CDC. Il est considéré comme un document qui explique tous les besoins, les attentes et les contraintes d'une application ou projet. Afin que ces derniers soient assimilés par l'ensemble des acteurs inclus dans le projet.

3-2-1 Contexte et définition du problème

Lors du lancement de leurs activités, les entreprises en général se trouvent dans une situation souvent fragile qui nécessite des efforts colossaux pour construire et parvenir à des bases solides. Ces dernières représenteront une force motrice et un garant de la continuité dans un périmètre de rivalité avec les autres concurrents.

En particulier, une entreprise agricole travaille sur du vivant. Par conséquent, elle se trouve face à une responsabilité plus particulière. Il s'agit des exigences de la société. En effet, pour chaque activité agricole, il est indispensable de faire un suivi pour les achats, les ventes, les stocks, les immobilisations, le budget, la main d'œuvre, ainsi que tous les aspects de gestion d'une exploitation

agricole. En outre, ces exigences demandent un certain niveau de réflexion et d'analyse de danger, de risque et même une analyse comptable et budgétaire.

Pour ce faire, l'agro-industrie doit toujours maîtriser et améliorer ses performances tout en assurant une gestion et une maîtrise optimales de ses ressources. L'exploitation d'une ferme nécessite donc à la fois la maîtrise des techniques de production agricole, des opérations comptables et des outils d'analyse, ainsi que dans un environnement bien géré.

Par conséquent, les agriculteurs doivent être accompagnés par des outils et des gestionnaires tout au long du cycle de vie de leur entreprise. Ces atouts reposent sur la collecte des données nécessaires pour mener de multiples analyses des situations très diverses qu'ils peuvent rencontrer au quotidien.

C'est dans ce cadre que s'inscrit l'objet du projet de fin d'études, visant à promouvoir et protéger le cycle de vie des organismes agricoles à travers un ERP nommé BEEONE afin de suivre toutes les activités qui les concernent.

3-2-2 Besoins fonctionnels

La solution établie prend en compte deux types d'acteurs protagonistes à savoir : L'admin qui sera représenté par les managers et les décideurs du pôle agricole.

Les besoins fonctionnels ciblés pour ces acteurs sont, D'une part le manager :

- Le manager pourrait s'authentifier via son login et mot de passe pour accéder à son espace administrateur, Il est responsable de la gestion des accès des utilisateurs (techniciens) et les attribue leurs rôles. Ainsi que le manager pourrait affecter des domaines agricoles (fermes) à chaque technicien.

D'une autre part, les techniciens représentent un deuxième hiérarchique quoiqu'ils aient une variété de spécialités. Cette variété s'étale sur la RH, comptabilité, analyse budgétaire ...etc.

De ce fait, la solution permettrait à chaque type de technicien de bénéficier des fonctionnalités suivantes :

- Il pourrait s'authentifier à son espace personnel via son login et son mot de passe, afin de visualiser les pages auxquelles le manager lui a donné accès.
- Le technicien pourrait accéder aux domaines agricoles (fermes) auxquels il est affecté.
- Le technicien pourrait facilement visualiser des graphes représentant son domaine d'expertise.

3-2-3 Besoins non fonctionnels

- L'extensibilité : L'application devra être extensible, c'est-à-dire qu'on pourra y ajouter plusieurs fonctionnalités
- La sécurité : L'application devra être sécurisée en fournissant à chaque interlocuteur un identifiant et un mot de passe
- L'interface : Avoir une application qui respecte les principes des interfaces Homme/Machine (IHM) tel que l'ergonomie et la fiabilité.
- La performance : l'application devra réagir dans les meilleurs délais à chaque opération exécutée par l'utilisateur.
- La convivialité : l'application devra être simple et facile à manipuler.

2-3- Élément de risque

Afin d'assurer un bon déroulement du projet, nous avons établi des mesures préventives à travers l'identification des risques, qui peuvent être prévues pendant la réalisation du projet. Que ceux-ci soient liés au non-respect de délai prévisionnel ou de la qualité.

De ce fait, nous avons créé un tableau qui regroupe les risques susceptibles d'échouer le projet et les solutions pour y remédier.

Tableau 2 Identification des risques de projet

Risque	Impact	Gravité	Solution
Mauvaise planification du projet	Le non-respect de délais Retard dans l'avancement du projet	Elevé	La bonne utilisation des outils et méthode de gestion de projets
Piloter le projet sans aucune visibilité par les collaborateurs	Retard dans l'avancement du projet Pas de suivi en temps réel	Elevé	Organiser des réunions pour faire le point sur l'état d'avancement du projet et penser aux autres axes stratégiques. Il est préférable de faire un point tous les matins

Perte des fichiers et document lié au code et au projet	Charge, coût, délai	Moyen	Conserver le projet après chaque clôture en utilisant soit gitlab, cloud ou autre support
Utilisation des nouvelles technologies (Angular, Node.js,)	Une action qui nécessite beaucoup de temps	Elevé	Être performant vis-à-vis l'auto-formation
Mauvaise analyse et expression des besoins	Les résultats ne sont pas conformes aux attentes	Elevé	Commencer par élaboration d'un CDC Réaliser attentivement une conception du projet en utilisant les diagrammes Faire des points avec les collaborateurs du projet avant de passer au codage

2-4- Facteur de succès

D'après l'identification des risques, il se prélevé que la réussite de l'application est liée à plusieurs facteurs de succès. Parmi lesquels on trouve :

- Planification et ordonnancement correct des tâches.
- Assurer une meilleure communication et collaboration des différents partenaires (acteur technique, acteur métier, directeur, client) associé au projet.
- Fluidité et respect des tâches dans les délais fixés auparavant.
- Utilisation des nouvelles technologies avec une disponibilité de la documentation

III. MÉTHODOLOGIE DE GESTION DE PROJET

4-1- Choix de la méthodologie

Comme c'est le cas dans chaque projet informatique, il est indispensable d'adopter une méthodologie de travail. Cette dernière fournit tout un ensemble de procédures et d'outils servant à l'orchestration et la répartition des tâches au niveau des employés.

Le choix de la méthode de travail dépend du périmètre d'un projet à un autre. A l'instar d'un projet de petite taille, un cycle de vie en cascade est certainement suffisant. D'autre part, lorsqu'il s'agit d'un projet compliqué, autrement dit, s'il nécessite un ensemble de processus pour qu'il soit établi. Par exemple, les données sont mal collectées dès le départ ou les besoins sont mal spécifiés et non clairs, dans ce cas il faut appliquer une méthode itérative.

Parmi les méthodes itératives les plus répandues et utilisées aujourd'hui, ce sont les méthodes Agiles. Cette méthode s'oriente vers un esprit collaboratif et conforme aux approches incrémentales. La méthode Agile regroupe un ensemble d'outils pertinents tout en prenant en considération la progression des besoins du client.

La méthode AGILE assure un bon déroulement de projet, surtout au niveau de la communication avec le client et une meilleure visibilité du projet. Elle permet aussi de prélever au plus tôt possible les problèmes et les risques, permettant de déclencher des actions correctives et préventives.

Dans le cas de notre projet, nous avons choisi la méthode Agile plus particulièrement SCRUM qui répond au périmètre de l'application.

4-2- Présentation de la méthodologie SCRUM

Le principe de la méthodologie SCRUM est de réaliser un projet de façon incrémentale. Ce Framework de gestion de projet a pour objectif d'améliorer la productivité de son équipe.

Toutes les quatre semaines, le client reçoit une partie de logiciel fonctionnel après chaque itération. Grâce à ces dernières, plus on se rapproche de la fin du projet, plus l'application est complète et performante.

La présente figure illustre le périmètre fonctionnel de la méthode agile SCRUM. On y trouve en premier temps les acteurs protagonistes responsables sur les processus agiles. Parmi ces acteurs on trouve le PO qui est le Product Owner. Il s'occupe de la rédaction des spécifications fonctionnelles et des User stories qui seront traitées ensuite par l'équipe de développement. Le PO est toujours en contact avec la partie métier, notamment les stakeholders.

Le processus agile regroupe un ensemble de cérémonies indispensables. En effet, après avoir rédigé les Users Stories, le PO devrait assister à une cérémonie de macro chiffrage organisée par le SCRUM master. L'équipe des développeurs y participe aussi pour évaluer les complexités techniques de la réalisation des Users Stories.

Après l'étude de la vélocité de l'équipe et la disponibilité des environnements de développement, tous les acteurs se mettent dans une réunion appelée cérémonie d'engagement dans laquelle l'équipe s'engage à réaliser des US en tenant compte de sa capacité et de son savoir-faire.

La réalisation des points engagés se fait en un sprint ou ce qu'on appelle une itération. Cette dernière dure de deux à trois semaines. Elle est caractérisée par une réunion matinale régulière appelée Daily qui dure au plus 10 minutes et dans laquelle les membres de l'équipe évoquent leurs avancements et déplacent les US qui sont finies dans le SCRUM BOARD pour donner à toute l'équipe une visibilité sur l'avancement du sprint. A la fin du sprint, l'équipe est amenée à déployer son travail sur un environnement de recette pour préparer la cérémonie de démonstration. Durant cette cérémonie, le PO valide les points réalisés s'ils correspondent parfaitement aux points engagés au préalable avant le démarrage du sprint.

Toutefois, il est possible que le travail de l'équipe ne soit pas totalement fini vu qu'il y a toujours des risques de stabilité des environnements de développement. Dans ce cas le PO se charge de créer une US des reliquats pour se rattraper dans la prochaine itération. Après la démonstration, le SCRUM master organise une cérémonie rétrospective dans laquelle il organise des ICEBREAKER pour créer de l'ambiance et donne la possibilité de débattre l'ensemble des problèmes rencontrés lors du sprint pour les remédier ou pour proposer de meilleures façons de travailler pour un rendement meilleur.

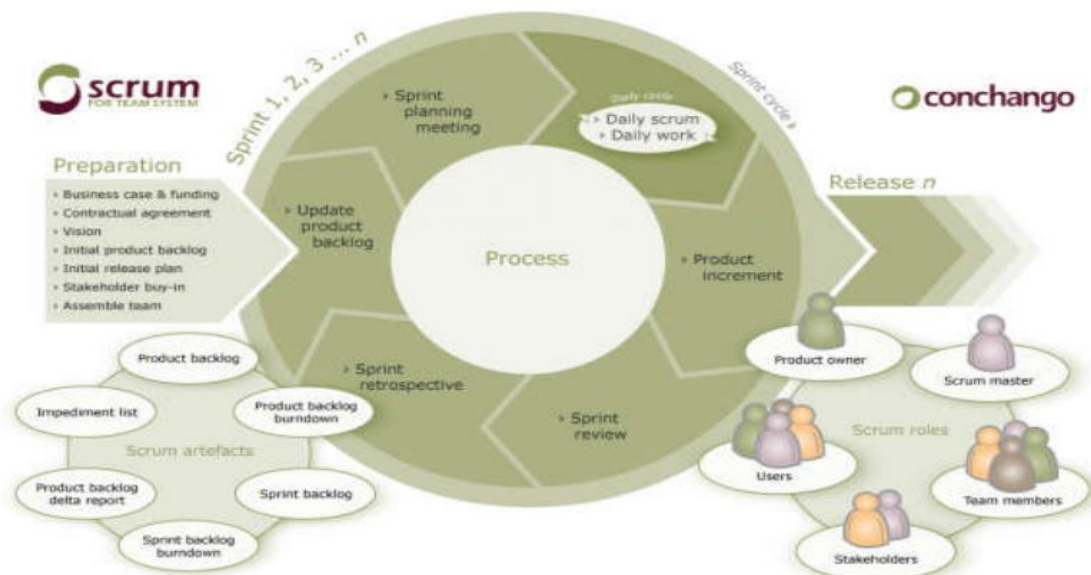


Figure 4 Principe fonctionnel de la méthode agile Scrum

4-3- Organisation et application de la méthode

Comme c'est le cas dans chaque projet informatique, il est indispensable d'adopter une méthodologie de travail. Cette dernière fournit tout un ensemble de procédures et d'outils servant à l'orchestration et la répartition des tâches au niveau des employés.

AGRIDATA a adopté la méthode agile SCRUM pour organiser le travail au sein de ses équipes. Il s'agit en occurrence des réunions matinales appelées "Daily Meeting" lors desquelles on présente ce qui a été fait, ce qui est en cours et ce qui va être fait dans la journée. Le SCRUM master est responsable de l'animation de cette réunion qui dure à peu près 10 minutes.

Toute l'équipe organise son travail dans un SCRUM BOARD dans lequel on trouve toutes les (user stories) concernant les spécifications fournies par le PO. Ce dernier est le responsable de la rédaction des US accompagnées des règles métiers et des spécifications fonctionnelles.

Le travail au niveau de l'équipe de développement est organisé grâce à l'outil GIT sur la plateforme gitlab. L'ensemble de l'équipe fait des commits et des push réguliers pour monter la version de l'application et pour en garder celle qui est la plus stable.

Les cérémonies de macro chiffrage et celles des engagements sont animées avant le démarrage de chaque itération. Elles sont organisées avec une présence requise du client. Quant aux itérations, elles durent deux semaines selon le contenu des US rédigées par le PO. Dans notre application, nous pouvons distinguer les rôles suivants :



Figure 5 Équipe SCRUM de BEEOne

Élaboré par nos soins

4-4- Planning du projet

La phase de planification du projet représente l'activité qui consiste à ordonnancer les tâches du projet, à estimer leurs coûts surtout en termes de temps et de délai, et à répartir les tâches selon les profils inclus à leur réalisation.

L'échec d'un projet informatique est forcément dû soit à la négligence de la planification du projet qui affecte négativement son bon déroulement. Par conséquent, le projet n'aura jamais une stabilité fonctionnelle ce qui cause des erreurs préjudiciables au client et à l'équipe de travail. La planification s'avère une étape incontournable de Management du projet permettant de :

- Définir les travaux à accomplir
- Répartition des tâches
- Fixer les objectifs
- Estimer le résultat
- Diminuer les risques (dépassement des délais).
- Être conscient de l'état d'avancement du projet

Pour ce faire, nous avons adopté une décomposition dite Organigramme des tâches du projet (OTP) ou (WBS acronyme anglais de Work Breakdown Structure). Il s'agit d'un découpage hiérarchique des tâches et activités du travail, que l'équipe doit respecter pour atteindre les objectifs du projet et produire des livrables.

La structure de WBS est composée de plusieurs éléments, chacun d'eux correspond à une tâche ou à un ensemble de tâches du projet. Le WBS considère le projet lui-même comme un premier élément. Il reçoit en premier temps le nom du projet, et par la suite les autres éléments sont organisés en dessous pour représenter chaque élément du projet. La figure suivante présente le WBS de notre projet :

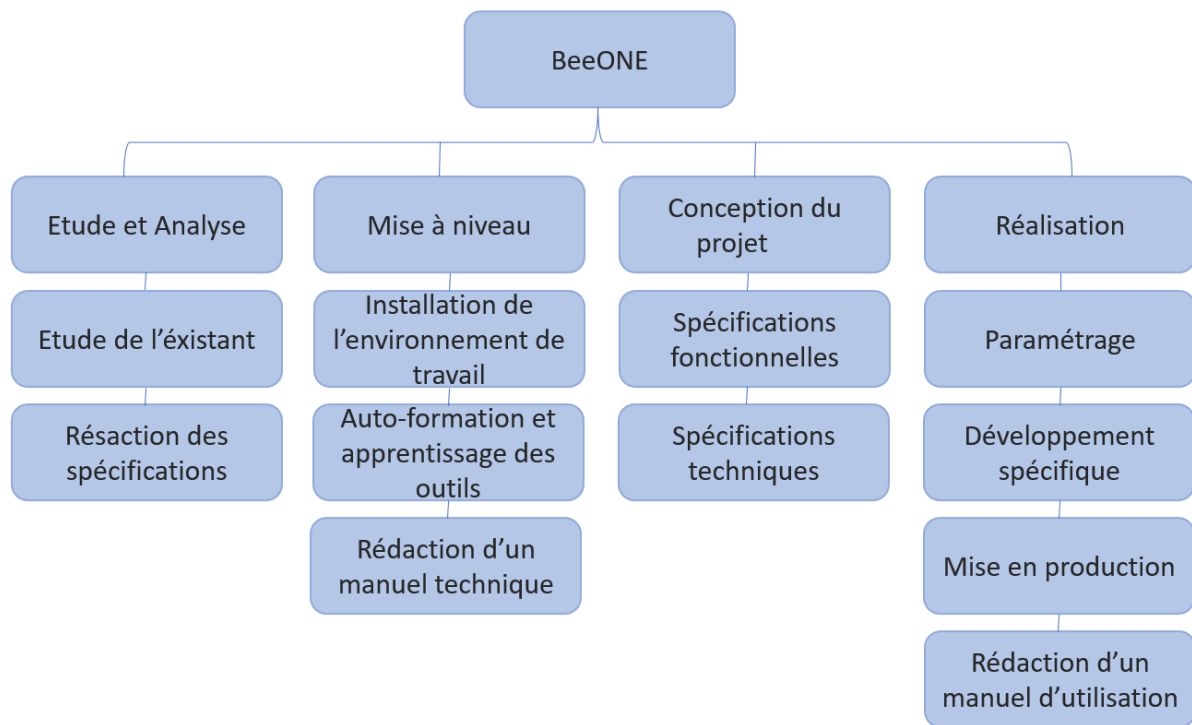


Figure 6 Organigramme basé sur le découpage WBS

Elaboré Par nos soins

4-5- Estimation des durées des tâches et de la durée totale du projet

Dans notre projet, nous avons estimé de réaliser notre application dans une durée de 4 mois. Dans la suite, nous allons détailler chaque étape mentionnée dans l'organigramme ci-dessous qui montre le planning adopté pour répondre au besoin de l'entreprise. Les phases du cycle de vie du projet sont les suivantes :

- Etude et analyse : nous avons consacré cette phase à la découverte de l'organisme d'accueil et à la détermination des objectifs à réaliser, tout en s'appuyant sur la spécification des besoins en élaborant un cahier des charges
- Mise à niveau : cette étape a pour objectif de préparer l'environnement de travail, en commençant par le choix des technologies et des langages de programmations, et faire des auto-formations si nécessaire, dans notre cas, il s'agit de deux langages de programmation (Angular et node.js). Le but est de définir le contexte dans lequel celui-ci sera utilisé.
- Conception du système : il s'agit de créer une illustration informatique des éléments du monde réel, sans se préoccuper de leur implémentation et indépendamment du langage de programmation.
- Réalisation : c'est l'implémentation et la mise en place de l'application, le codage des fonctionnalités.

4-6- Enchaînement des tâches

Le diagramme de GanTT est une technique utilisée en gestion de projet qui montre l'ordonnancement des tâches et des m Nous avons illustré Dans les figures qui suivent le planning adopté, pour répondre attentivement aux besoins de l'entreprise à travers l'élaboration de diagramme de GanTT.

Tableau 3Ordonnancement des tâches

	Task Name	Duration	Start	Finish	Predecessors
1	Mise en place de 3 modules du BeeOne ERP	88 days	22-03-01	22-06-30	
2	Etudes et analyses	9 days	22-03-01	22-03-11	
3	Mise à niveau	9 days	22-03-14	22-03-24	2
4	Modélisation et conception	7 days	22-03-25	22-04-04	3
5	Préparation de l'environnement	3 days	22-04-05	22-04-07	4
6	Codage et développement	60 days	22-04-08	22-06-30	5
7	Rédaction de rapport	82 days	22-03-08	22-06-29	2SS+5 days

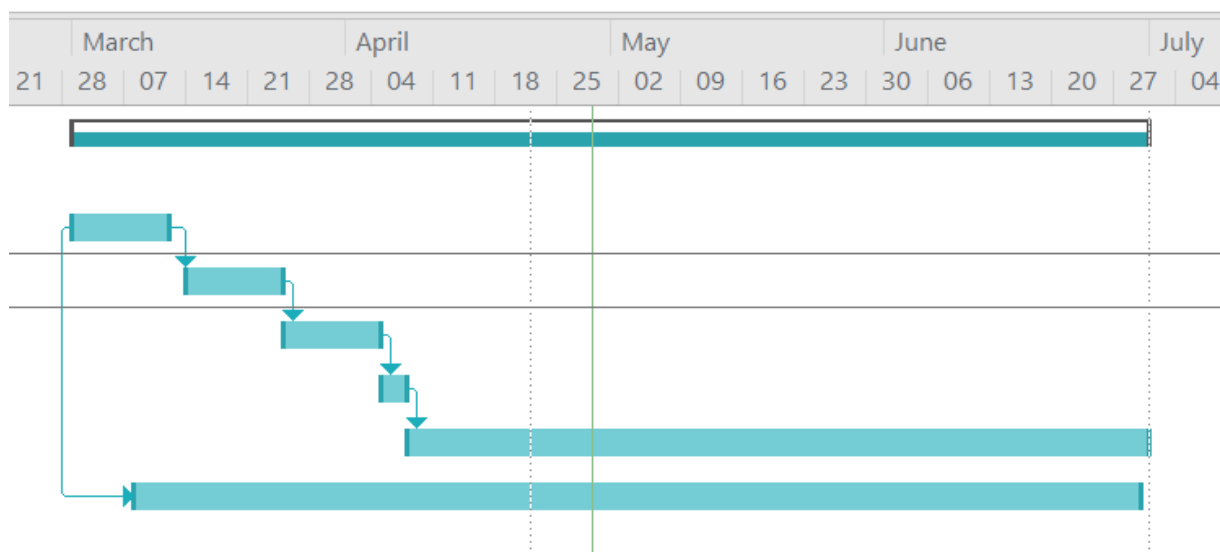


Figure 7Diagramme de Gantt

Dans cette partie introductive, nous avons présenté l'organisme d'accueil en termes d'activités et d'organisation. Nous avons aussi spécifié le contexte et le but du projet et nous avons élaboré un cahier de charge contenant toutes les spécifications ou exigences exprimées par les entreprises clientes.

Ainsi que, nous avons décrit en détail la méthodologie adoptée pour structurer le travail. Finalement, nous avons présenté le planning à respecter pour bien maîtriser les ressources allouées au projet.

4-7- Les outils de gestion de projet utilisé dans l'entreprise

Dans les lignes qui suivent nous allons présenter quelques outils que le service concerné par notre projet utilise dans la réalisation de ces projets :

4-7-1 L'outil Slack



L'outil de communication Slack qui est une plateforme de messagerie instantanée ainsi qu'un logiciel de gestion de projet, il appartient au groupe Salesforce.[1]

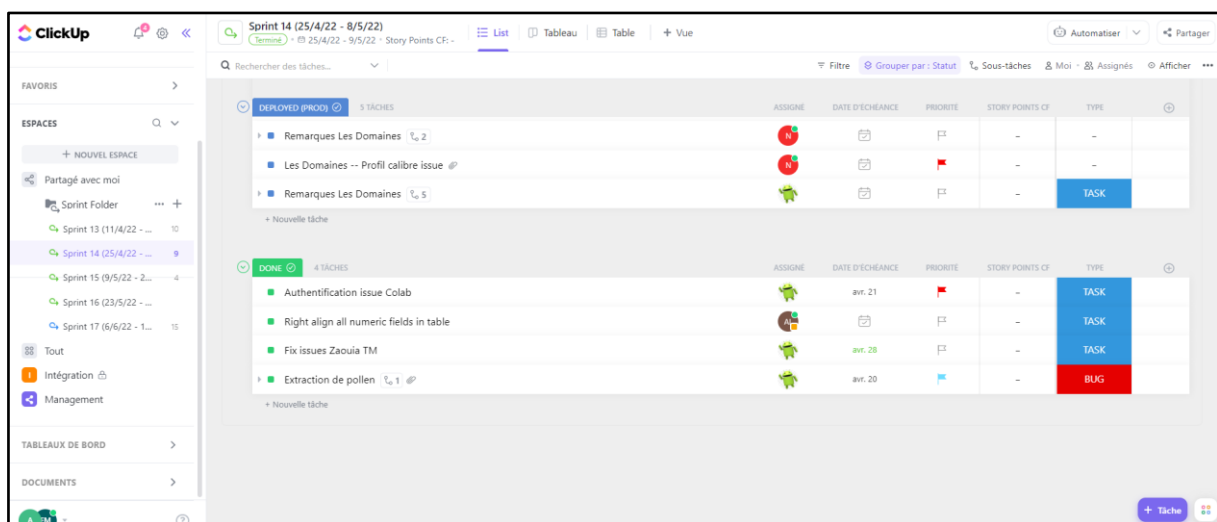
Cet outil permet la communication instantanée entre les membres de différentes équipes qui travaillent dans le cadre du projet BeeOne. Ces équipes sont actuellement divisées sur deux bureaux, l'un de l'équipe métier et commercial, et l'autre de l'équipe technique. Et il permet aussi le partage des fichiers. En plus il contient un moteur de recherche pour faciliter la recherche des projets ou des fichiers. En effet il existe une application mobile de cette solution. Cet outil est payant mais il existe une version gratuite mais des fonctionnalités limitées.

4-7-2 L'outil Clickup



Clickup permet de faciliter la gestion des projets en équipe grâce à une interface unique qui centralise toutes les fonctions collaboratives [2].

Cet outil est basé sur une organisation des projets en sprint listant des cartes, chacune représentant des tâches. Les cartes sont assignables à des utilisateurs et le product owner suit la progression en temps réel de ces tâches. La figure suivante présente une interface du logiciel Clickup:



Dans le menu à gauche nous avons un dossier du sprint en cours ainsi que les sprints déjà effectués, de plus pour chaque tâche nous avons son état que ce soit To do, in progress, done, deployed et blocked.

IV. CONCLUSION

Au début de ce chapitre, nous avons spécifié le contexte et le but du projet, et nous avons élaboré un cahier de charge contenant toutes les spécifications ou exigences exprimées par les entreprises clientes. Ainsi que, nous avons décrit en détail la méthodologie adoptée pour structurer le travail. Et finalement, nous avons présenté le planning à respecter pour bien maîtriser les ressources allouées au projet. Dans le chapitre suivant, nous enchaînerons sur la recherche bibliographique sur les outils utilisés pour réaliser le projet.

CHAPITRE 3 :

Recherche bibliographique

I. INTRODUCTION

Dans ce chapitre nous allons commencer par définir l'environnement et les outils de travail utilisés pour répondre aux besoins exprimés dans la partie d'élaboration du cahier de charges et la spécification de besoins fonctionnels et non fonctionnels pour aboutir à une version stable de la solution.

II. CHOIX DU MODE SAAS

Le logiciel en tant que service SaaS est un moyen de fournir des applications en tant que service sur Internet. Il élimine le besoin d'installer et de maintenir des logiciels et est facilement accessible via Internet, ce qui vous libère de la gestion complexe des logiciels et du matériel.



Figure 8 Mode SaaS

<https://www.cegid.com/fr/faq/quest-ce-que-le-saas/>

Parmi les caractéristiques du mode SaaS, on trouve :

Accessibilité : étant donné que le logiciel est hébergé en externe par le fournisseur, il est facile de modifier les plans d'utilisation sans préavis. De plus, l'utilisation basée sur le Web permet aux abonnés d'accéder facilement au logiciel de n'importe où avec une connexion Internet.

Gestion opérationnelle : Pas besoin d'installer, de mettre à jour les appareils ou de gestion traditionnelle des licences.

Rentabilité : Pas de frais de matériel initiaux et des méthodes de paiement flexibles comme les modèles de paiement à l'utilisation.

Évolutivité : Solution facile à adapter à l'évolution des besoins.

Stockage des données : Les systèmes de données sont sauvegardés dans le cloud.

Analyse : Accès aux outils de reporting et d'intelligence des données.

Renforcement de la sécurité : Les fournisseurs de SaaS investissent massivement dans les technologies et l'expertise en matière de sécurité.

Notre entreprise d'accueil a choisi cette solution car au lieu de prendre en charge la maintenance ou embaucher un personnel pour gérer tout, il suffit de payer un abonnement et c'est le fournisseur de service qui s'occupe de toute l'infrastructure, et le déploiement du service. Les applications sont accessibles à partir de divers dispositifs clients par le biais d'une interface client léger, telle qu'un navigateur web (par exemple, une messagerie électronique basée sur le web), ou d'une interface de programme. Le consommateur ne gère ni ne contrôle l'infrastructure cloud, y compris le réseau, les serveurs, les systèmes d'exploitation, le stockage ou même les capacités des applications individuelles, à l'exception peut-être de paramètres de configuration limités propres à l'utilisateur.

III. CHOIX TECHNOLOGIQUE POUR LE DÉVELOPPEMENT

2-1- Angular



Créé par GOOGLE, Angular est un Framework TypeScript utilisé pour construire des interfaces graphiques étendant le HTML pour le rendre dynamique. Il possède un ensemble d'outils et de bibliothèques JavaScript permettant de construire le cadre le mieux adapté au développement d'une application.

Framework caractérisé par :

- Data-binding bidirectionnel : permet le transfert des données entre la vue statique en HTML et le contrôleur qui présente la partie logique. Ce transfert se fait à travers un objet spécifique appelé Scope. Le Scope permet le transfert des modèles (Base de données) et Query (Données dynamiques) qui déterminent l'état des vues HTML (Composants graphiques dont l'état dépend de la partie logique).
- Routing : Angular est considéré comme SPA Autrement dit, une application développée en Angular contient plusieurs SPA. La navigation entre ces SPA est garantie par la fonctionnalité du ROUTING. Le point fort de cette fonctionnalité se manifeste lors du passage d'une page à l'autre sans besoin de recharger l'application toute entière.

2-2- Node.js



Node.js est une plateforme d'exécution construite à base du moteur JavaScript Compatible avec la version du navigateur Chrome (Client). Il est souvent utilisé pour développer des services côté serveur appelés API.

Le Non-blocking est l'un des points forts du Node JS, il s'agit de la possibilité d'une compilation répétitive au fur et à mesure de l'avancement du développement. De ce fait, le développeur n'est plus obligé de recharger son JAR ou son

TOMCAT (serveur) pour tester une nouvelle modification dans son code. Il pourra juste recharger son client pour voir ses modifications.

Node.js est très extensible, il est écrit et s'utilise en JavaScript. Ce langage est initialement réservé aux programmes exécutés par les navigateurs, mais Node en a fait un langage utilisable également côté serveur. Cela permet, avec la connaissance d'un seul langage, d'écrire des programmes pour le client (qui s'exécutent dans le navigateur Internet) et pour le serveur (qui s'exécutent dans le serveur). Enfin, la popularité de Node vient aussi de sa rapidité d'exécution. Ceci est lié à son architecture événementielle, qui le rend différent des autres types de serveurs.

2-3- API REST

Une API REST est une interface de programmation d'applications qui adhère aux contraintes du style architectural REST et permet l'interaction avec des services web RESTful. Les réseaux interconnectés constituent le web. Un service web est un ensemble de protocoles et de normes ouverts utilisés pour échanger des données entre des applications client-serveur. Les services web qui suivent l'architecture REST sont connus sous le nom de services web RESTful[7].

REST s'agit essentiellement d'un style architectural pour la conception d'applications en réseau. REST s'appuie sur un protocole client-serveur sans état et dans presque tous les cas, il s'agit de HTTP. Au départ, les programmeurs s'appuyaient sur SOAP pour mettre en œuvre l'API dans les services Web, mais ces dernières années, REST est devenu le choix des programmeurs en raison de sa simplicité et de son évolutivité.

REST a été conçu pour traiter les objets du côté serveur comme des ressources pouvant être créées, mises à jour et supprimées. REST peut être utilisé par pratiquement tous les langages de programmation. Voyons maintenant les avantages de l'utilisation de REST.

Il est important d'utiliser le type REST des API car il est :

- **Facile à apprendre et mettre en œuvre** : REST utilise les méthodes HTTP pour la communication et la plupart d'entre nous sont familiers avec les verbes HTTP tels que GET, POST, PUT ou DELETE. Ces méthodes sont explicites quant à leur fonction, ce qui rend REST facile à apprendre. En ce qui concerne l'architecture, REST est basé sur l'architecture client-serveur où le client est complètement séparé du serveur. Cette caractéristique très cool permet à l'équipe du développeur de travailler sur les deux extrémités de manière indépendante. Nous n'avons pas à nous soucier de ce qui a été codé du côté client ou de la manière dont le serveur est assemblé. Cette fonctionnalité donne l'opportunité de travailler sur plusieurs projets liés au développement.
- **Scalabilité** : C'est l'une des meilleures caractéristiques qui font que REST se distingue des autres. La séparation du client et du serveur dans l'architecture REST permet aux développeurs de faire évoluer les applications plus facilement
- **Stateless** : C'est l'une des caractéristiques uniques de REST. Il s'agit d'un système sans état du côté du serveur, ce qui signifie que le serveur ne stocke aucune des demandes ou réponses

précédentes. Chaque demande sera traitée et servie indépendamment, et il n'aura rien à faire avec la précédente.

- **Format d'échange de données plus rapide** : C'est un autre grand avantage de l'utilisation de REST. Dans les API REST, le format JSON est utilisé pour échanger les données. Par rapport à XML, JSON est plus petit en taille et beaucoup plus compact. En outre, il peut être analysé plus rapidement que le XML.
- **Flexibilité et portabilité** : En REST, il est facile de mettre à jour les données dans la base de données à tout moment. Il offre la flexibilité de répondre à de nombreux clients demandant différents types de données (XML, JSON, etc.). Tout ce que vous avez à faire est de demander les types de données en utilisant l'en-tête Accept. REST renverra la réponse en fonction des types de données spécifiés.

2-4- Postman



Postman est un client API qui permet aux développeurs de créer, partager, tester et consigner. Pour ce faire, il permet aux utilisateurs de créer et d'enregistrer des requêtes HTTP/s simples et complexes, ainsi que de lire leurs réponses.[3]

2-5- sass



Sass est un langage de feuille de style qui est compilé en CSS. Il vous permet d'utiliser des variables, des règles imbriquées, des mixins, des fonctions et bien d'autres choses encore, le tout avec une syntaxe entièrement compatible avec CSS. Sass permet de bien organiser les feuilles de style volumineuses et facilite le partage de la conception au sein d'un même projet ou entre plusieurs projets.[4]

2-6- Amchart 5



AmCharts est un ensemble de bibliothèques de visualisation de données basées sur JavaScript qui comprend des types de graphiques réguliers tels que Serial, Pie, etc. ainsi que des types avancés tels que Stock Charts et Maps. Dans notre système nous avons utilisé la quatrième version de Amcharts, car il est écrit en TypeScript et compatible avec Angular. [5]

Il se caractérise par sa réactivité. Il permet de remplacer et de modifier dynamiquement certains paramètres ou fonctionnalités du graphique en fonction de l'espace réellement disponible. Pour les types de graphes, il existe les charts pour les graphiques et les Maps pour les cartes, et les stocks charts pour les graphiques financiers.

2-7- Base de données : SGBD

SQL Server est un système de gestion de base de données utilisé pour stocker et distribuer des informations dans une base de données tout en garantissant sa qualité, sa pérennité et sa



confidentialité. Pour assurer la sécurité de la base de données, SQL Server fournit le chiffrement et la sauvegarde des données.[6]

Microsoft SQL Server est une plate-forme de données d'entreprise permettant de gérer et stocker dans des bases de données tout type d'information :

- Données structurées : données relationnelles par exemple.
- Données non structurées : documents, images, ...
- SQL Server intègre par défaut des outils de gestion, d'administration et de développement de bases de données. – Déploiement par un setup, mise en œuvre et administration par des interfaces graphiques intuitives.
- Gestion avancée de la sécurité en offrant deux modes d'authentification (Authentification Windows et Authentification SQL Server).
- Prise en compte des spécificités des projets décisionnels.
- Coût relativement moins cher par rapport aux autres SGBD du marché.

2-8- PowerAMC



PowerAMC est un logiciel de modélisation sophistiqué pour modéliser une plateforme UML rapide, flexible et fonctionnelle. Il contient des atouts de description conceptuelle de la solution informatique. En effet, il est utilisé pour dresser des diagrammes que ceux-ci soient des cas d'utilisation, des séquences, des activités ou des diagrammes de classes.[7]

2-9- VScode



VScode est un éditeur de texte dédié au développement des applications informatiques et formaté en une multiplicité de langages de programmations. Il se distingue des autres éditeurs grâce à son interface maniable et ses diverses fonctionnalités. De plus, il définit des colorations syntaxiques vis-à-vis des langages de programmation utilisés.

2-10- Git



Git est un client qui nous permet de nous connecter à des systèmes de contrôle de version distribués tels que github et gitlab. Il est optionnel du point de vue du Angular CLI. Il faut l'installer pour télécharger un projet angular dans un dépôt git.[8]

2-11- GitLab

gitlab est une plateforme web d'hébergement de code et de suivi des problèmes basée sur le système de contrôle de version git. Publié pour la première fois en 2011, il n'a cessé de croître et d'évoluer au



fil des ans, ajoutant de nouvelles fonctionnalités et capacités, et s'est transformé en un outil unique pour la productivité agile. bien qu'il soit détenu et géré par gitlab. Qui dirige la direction du projet, le cœur de gitlab est un logiciel open source avec plus de 2000 contributeurs distincts à ce jour.

gitlab est construit au-dessus de git afin que les utilisateurs qui contribuent à un projet aient une copie du projet téléchargée, vérifiée ou clonée sur leur ordinateur local. Il fournit une interface web pour gérer la plupart des flux de travail les plus avancés de git et recommande un flux de travail pour interagir avec git afin d'obtenir la meilleure productivité, efficacité et facilité d'utilisation.

IV. CONCLUSION

Pour conclure ce chapitre, maîtriser l'information signifie la comprendre, la contrôler et la sécuriser. Le rôle de l'analyse de données est de faciliter cette maîtrise grâce à ses différents outils et méthodes. Elle permet de corréler les informations entre elles pour que toute la chaîne de l'information soit comprise par les décideurs.

CHAPITRE 4 :

Ingénierie, Développement et Méthodologie

I. INTRODUCTION

La phase de développement est l'étape initiale de la réalisation et de la mise en œuvre de notre projet. Elle représente une étape primordiale de réflexion dans le cycle de développement logiciel après la phase de l'analyse et de la spécification. Dans ce chapitre, nous allons présenter en détails la modélisation de notre solution, et la conception du projet à travers les diagrammes UML à savoir le diagramme des cas d'utilisation, le diagramme de classes et les diagrammes de séquence.

II. MODÉLISATION

2-1- Méthodologie de développement

Durant notre projet, nous avons conçu que la plateforme adoptera l'architecture NgRX. Ce design pattern est l'acronyme de Angular Réactive Extension (NgRX) c'est un groupe de bibliothèques inspirées du pattern Redux. Comme son nom l'indique, NgRx est écrit spécifiquement pour l'application Angular en tant que solution de gestion d'État. Il assure une organisation architecturale de notre application en séparant :

- Les stores : est l'élément clé de l'ensemble du processus de gestion de l'État (State). Il facilite l'interaction entre les composants et cet état (State).
- Les States (état) : elle est composée des données reçues par les appels API, des entrées utilisateur, etc.
- Les Actions : Une action est une instruction que vous envoyez au store, éventuellement accompagnée de métadonnées (payload). En fonction du type d'action, le store décide des opérations à exécuter.
- Les Reducers : ils sont chargés de modifier l'état et de renvoyer un nouvel objet d'état avec les modifications. Les réducteurs prennent deux paramètres, l'état actuel et l'action.
- Les Selectors : Les sélecteurs sont des fonctions pures utilisées pour obtenir des éléments de l'état du store.
- Les Effets : vous permettent d'effectuer des effets secondaires lorsqu'une action est envoyée au store.

La figure suivante montre l'approche de l'architecture NgRX qui est basé sur quatres composants :

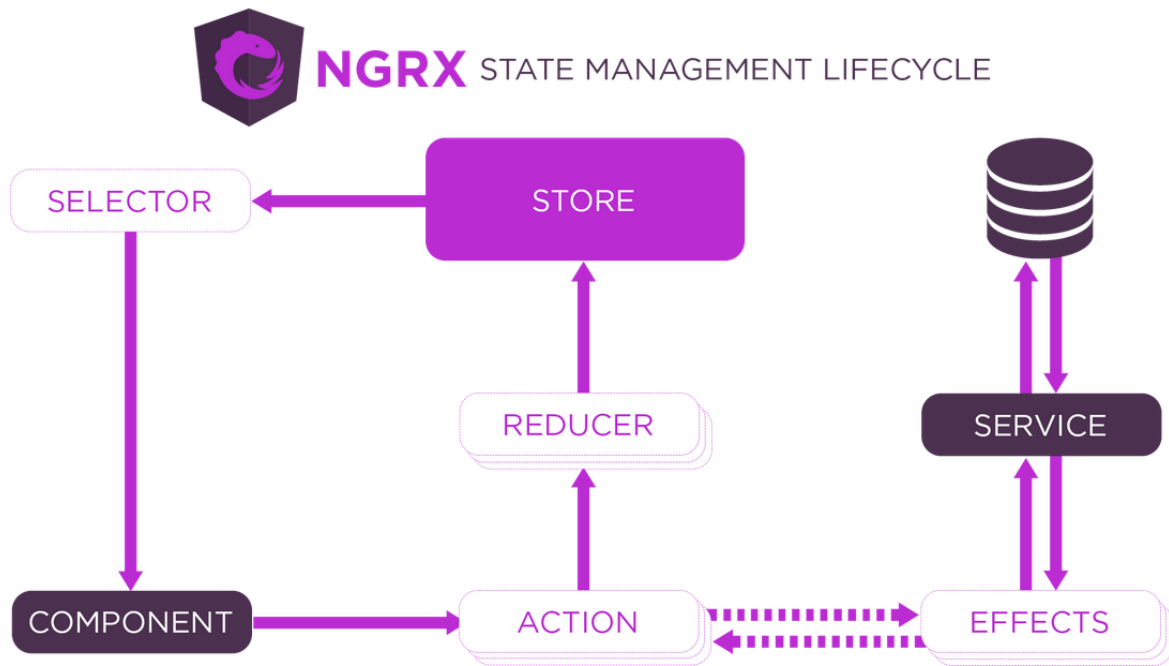


Figure 9 Architecture NgRx du projet

Source : <https://devtobecurious.fr/a-la-dcouverte-de-ngrx-avec-angular-9-etape-1/>

2-2- Architecture globale du projet

L'architecture d'une application web et plus généralement d'un projet informatique correspond à l'ensemble des choix techniques qui structurent le projet : modèle de conception (« design pattern », lié au « Framework » choisi), choix et définition de la base de données, choix du langage de programmation et aussi mais surtout la structure générale de l'application (telle que nous la définissons, à savoir par exemple les interactions entre les différentes fonctionnalités / modules de l'application).

L'architecture d'un progiciel et plus généralement d'un projet informatique correspond à une collection des choix technique qui architecturant le projet à savoir :

- Le modèle de conception (Framework)
- La définition et la structure de la base de données
- Les langages de programmation
- La structure générale de l'application qui sera défini par nous-même dont le but de présenter les interactions entre les différentes fonctionnalités et module de l'application

Dans notre cas l'architecture de notre projet sera basée sur le modèle ngrx avec 3 composants distinctes :

La base de données (SQL Server)

Le serveur web (Node.js)

Le client web (ANGULAR ...)

La base de données est le noyau de notre projet. Cette partie contient toutes les données essentielles de notre application. Cette base de données sera en liaison avec la partie serveur web qui interagit avec ses données. Seule la partie serveur web aura la possibilité de communiquer avec la base de données. Le serveur assure la transmission des données pour les navigateurs.

III. CONCEPTION

La conception constitue la phase primordiale dans le cycle de développement d'un projet. Elle permet de comprendre facilement et sans prérequis les interactions entre les acteurs et les fonctionnalités du système et donne une vue globale sur le projet.

3-1- Choix du modèle de conception

Afin de bien mener la conception de notre application, nous avons adopté une approche objet qui s'organise autour de quatre principes fondamentaux. C'est une démarche :

- Itérative et incrémentale.
- Dirigée par les besoins du client et des utilisateurs.
- Centrée sur l'architecture du logiciel.
- Basée sur la description des actions et des informations dans une seule entité.

Notre choix repose donc sur l'utilisation de la notation UML qui est une démarche qui se base sur l'approche objet, en se basant sur ses représentations visuelles à travers les diagrammes, permettant de résoudre et modéliser un projet d'une façon standard. Ce langage est un référentiel en termes de modélisation objet. De nos jours, UML possède treize diagrammes qui sont classés en deux catégories (dynamique et statique).

Dans ce cadre, nous allons commencer par les diagrammes de cas d'utilisation (Use Case) qui permettent de donner une vue globale sur l'application. En deuxième lieu, nous allons présenter la

catégorie statique représentée par le diagramme de classes. Finalement, afin de bien mener la chronologie de nos opérations, nous avons réalisé des diagrammes de séquences.

3-1-1- Diagramme de cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision statique et globale du comportement fonctionnel d'un système logiciel. Commençons par les cas d'utilisation. Nous allons d'abord identifier les acteurs, puis les cas d'utilisation pour finir avec des diagrammes.

3-1-1-1- Cas d'utilisation général

Dans ce diagramme de cas d'utilisation nous décrivons les acteurs du système et la hiérarchie existante entre ses derniers ainsi que les différents sous cas d'utilisation de chaque acteur :

Identification des acteurs :

Au niveau de cette section, nous présentons les différents acteurs susceptibles d'interagir avec le système, mais tout d'abord, nous donnons une définition du concept acteur. Dans cette partie, nous allons présenter les acteurs qui vont interagir avec notre application. Mais avant d'entamer à la description nous allons définir le concept acteur :

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié.

Dans notre cas, on distingue deux acteurs protagonistes de notre système. Il s'agit de :

- Manager : C'est l'acteur responsable de la gestion globale de l'application. Il est responsable sur les droits d'accès et les permissions qu'ils attribuent aux différents utilisateurs du système. Il fait l'ajout, la modification et la suppression de ces utilisateurs tout en limitant leurs tableaux de bord à ceux qui sont dans le domaine de l'expertise de l'utilisateur. En plus de son intervention en RH, le manager a accès à toutes les vues de l'application. De ce fait, il pourra mener des études et faire des statistiques et des plans prévisionnels en se basant sur les données fournies dans l'application.
- Technicien : Il bénéficie des droits d'accès qui lui sont attribués par son manager. Cet acteur constitue une abstraction qui englobe plusieurs types de techniciens. En effet, on trouve par exemple un gestionnaire d'actif qui a accès à toutes les pages orientées assets management. On trouve aussi des techniciens RH responsables du suivi des suivis de la main d'œuvre. Chaque type de technicien dispose de son propre tableau de bord dans l'application. Il est

rempli de données qui lui serviront comme atouts accessibles pour mener une étude précise relative à son domaine d'intervention.

De ce fait, le diagramme de cas d'utilisation suivant, représente l'ensemble des cas d'utilisations les plus élémentaires de notre système :

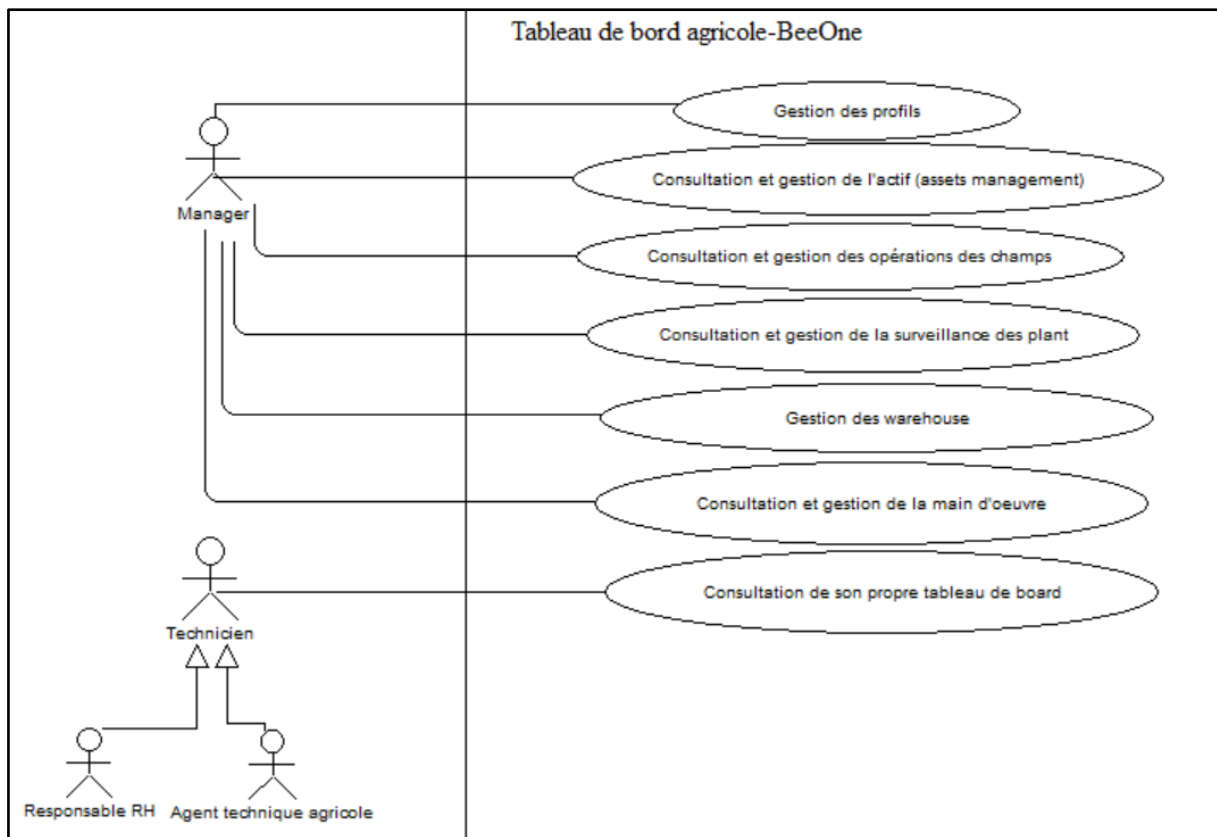


Figure 10: Cas d'utilisation générale

Dans ce diagramme on trouve les acteurs qu'on avait déjà prédéfinis. En effet, le manager s'occupe de la gestion RH des différents profils qui utilisent l'application. En plus, il pourra consulter n'importe quelle vue du système et pourra mener des analyses diverses telles que celles portées sur les données de l'analyse globale des exploitations agricoles à savoir la distribution géographique des parcelles ou des fermes et le suivi de la main d'œuvre à l'aide des modules de pointage, de paie et de transport.

D'autre part, le technicien pourra consulter son tableau de bord pour visualiser l'ensemble des données qui lui permettront de faire des analyses budgétaires, techniques... etc.

3-1-1-2- Cas d'utilisation : Gestion des profils

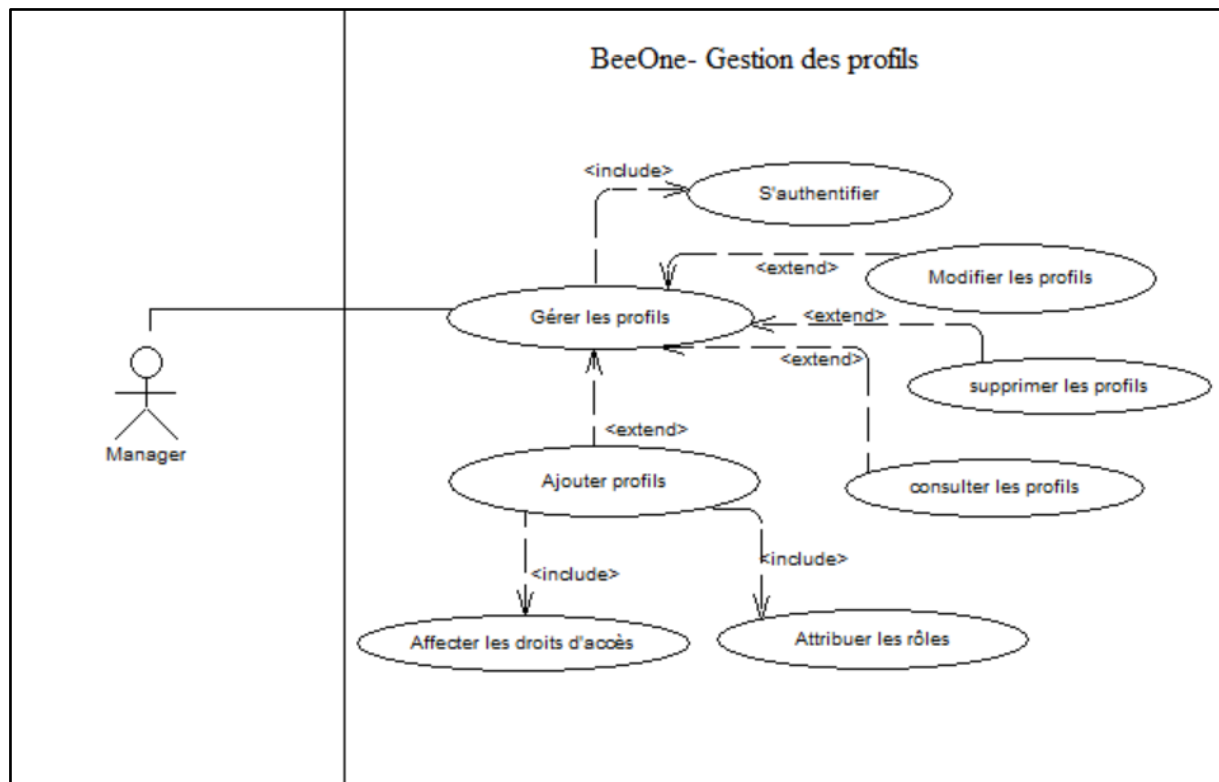


Figure 11 Cas d'utilisation : Gestion des profils

Dans ce diagramme on présente le manager en tant qu'acteur principal de la gestion des profils. En effet, il s'agit de la possibilité d'ajouter, modifier ou supprimer les profils tout en leur attribuant des droits d'accès aux différentes pages par des rôles prédéfinis. Il pourra aussi consulter son tableau de bord après la saisie du login et mot de passe dans la mire de l'authentification.

3-1-1-3- Cas d'utilisation : Consultation des parcelles

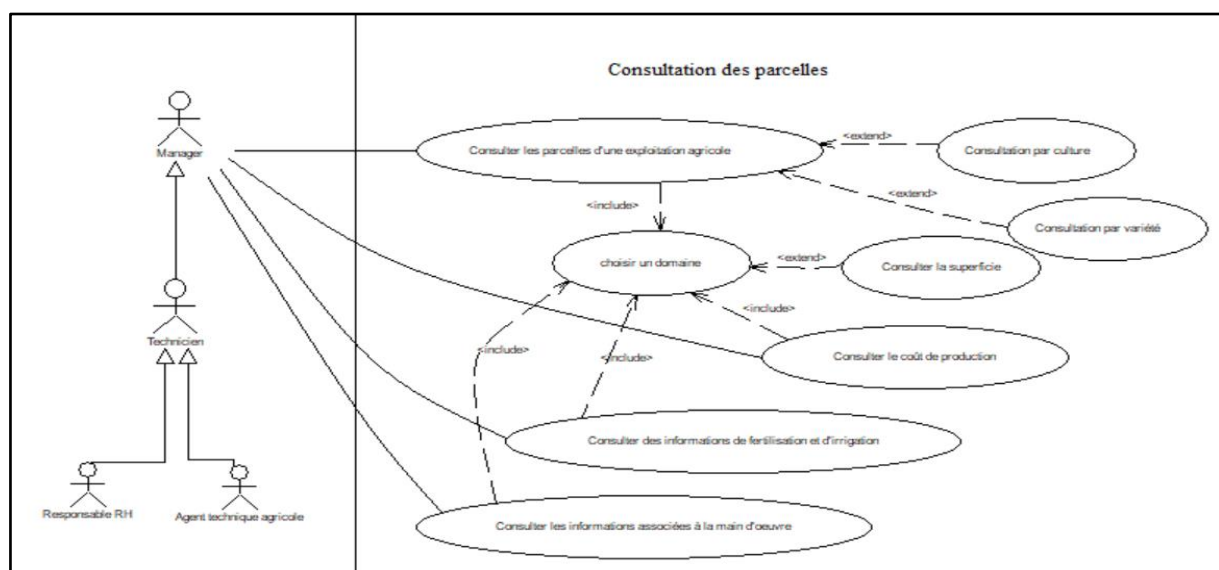


Figure 12 Cas d'utilisation : consultation des parcelles

Les acteurs concernés dans ce cas d'utilisation sont : le manager et les techniciens de types : responsable RH et agent technique agricole. Il s'agit en premier temps de consulter les parcelles associées à une certaine exploitation agricole. Ceci en mettant en disposition des filtres de recherche qui se basent sur des critères tels que la variété de la parcelle et sa culture.

La notion du choix du domaine consiste à spécifier l'entreprise ou le groupe auquel appartient la parcelle. Autrement dit, un domaine c'est un partenaire de l'entreprise AGRIDATA. Ce partenaire dispose de plusieurs fermes ou parcelles qui seront consultées par un technicien spécialisé (Des fois le manager lui-même) pour en étudier la superficie, le coût de production et toute information concernant à la répartition de la main d'œuvre.

3-1-1-4- Cas d'utilisation : Suivi de la main d'œuvre.

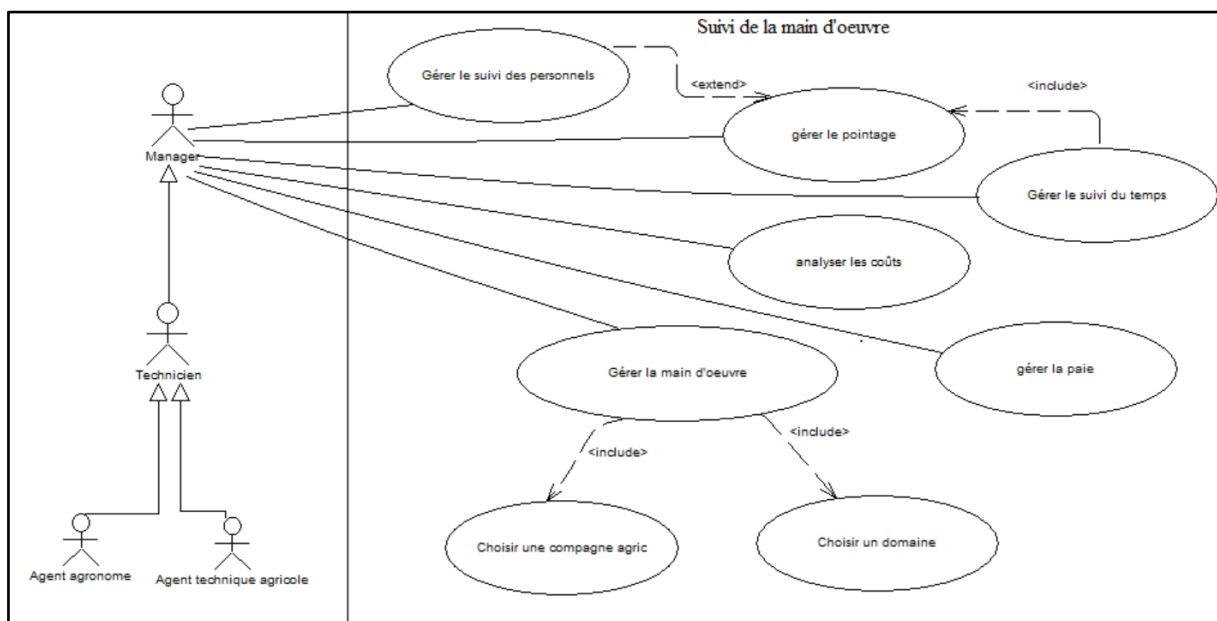


Figure 13: Cas d'utilisation : Suivi de la main d'œuvre

Les acteurs principaux responsables sur ce cas d'utilisation sont le manager et les techniciens de type RH ou agent technique agricole.

Dans cette partie, les acteurs sont amenés à gérer la main-d'œuvre pour une campagne agricole associée à un domaine précis. Il s'agit de gérer le personnel de la ferme, le pointage des ouvriers, le suivi du temps ainsi que l'analyse des coûts et de la paie.

3-1-1-5- Cas d'utilisation : Réalisation d'un suivi technique

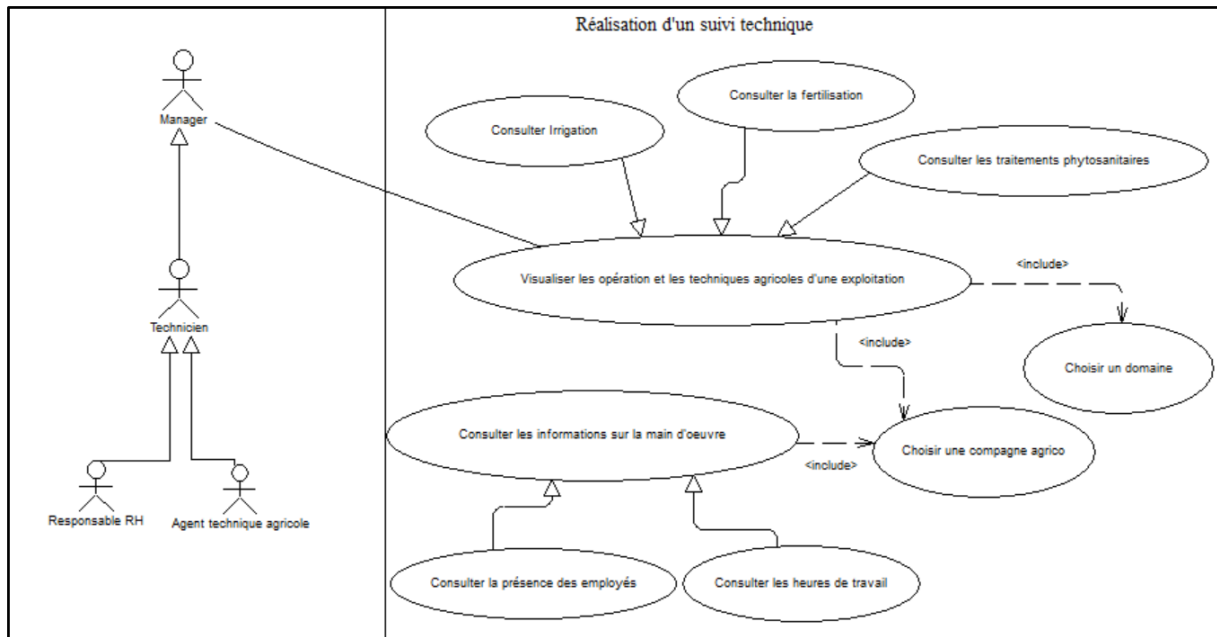


Figure 14: Cas d'utilisation Réalisation d'un suivi technique

Dans cette partie, le manager, le RH et l'agent technique agricole présentent les principaux acteurs responsables sur la réalisation d'un suivi technique d'une exploitation agricole. En effet, ils pourront consulter des informations sur la main-d'œuvre à savoir tout ce qui a un rapport avec les heures de travail des employés. En outre, ils ont la possibilité d'accéder à des données descriptives de l'irrigation et la fertilisation ainsi que les traitements phytosanitaires. Tout cela décrit un domaine précis dans une campagne agricole prédéfinie.

3-1-1- Diagramme de séquences :

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML.

Les diagrammes de séquences permettent de décrire COMMENT les éléments du système interagissent entre eux et avec les acteurs :

- Les objets au cœur d'un système interagissent en s'échangeant des messages.
- Les acteurs interagissent avec le système au moyen d'IHM (Interface Homme Machine).
- Nous présentons dans cette partie les diagrammes les plus importants de notre système à savoir :
- La gestion des profils
- L'authentification
- Le suivi agricole

3-1-2-1 Diagramme de séquences : L'Authentification

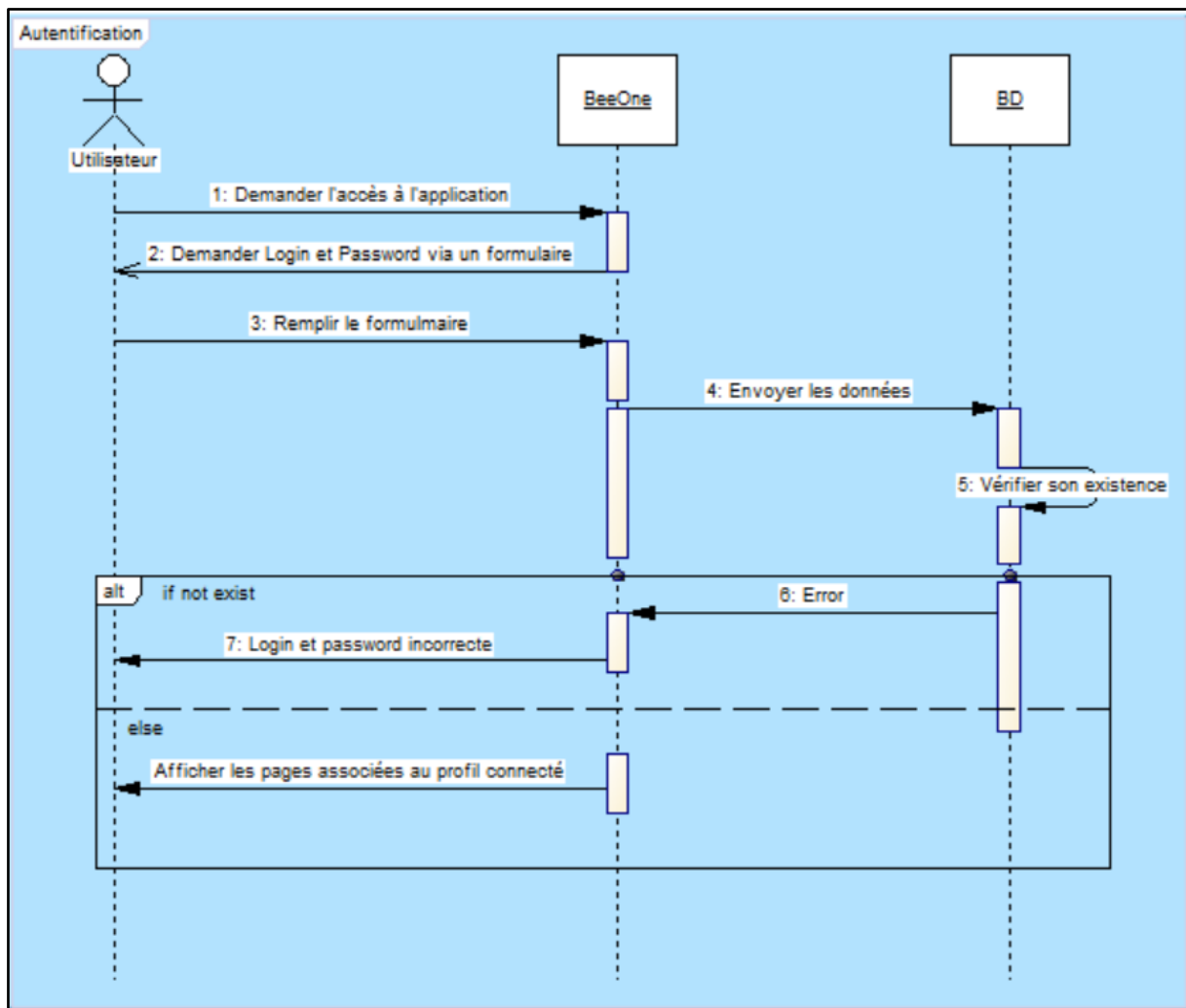


Figure 15: Diagramme de séquence Authentification

Description des objets du diagramme :

- Utilisateur : Administrateur ou technicien qui a le droit d'accéder à l'ERP via un navigateur internet. Il a déjà des données à saisir (login & mot de passe) pour accéder à son espace et réaliser les opérations souhaitées.
- BeeOne : C'est le système responsable sur le traitement des requêtes http envoyées par l'entité utilisateur.
- Base de données : C'est l'espace de stockage qui englobe toutes les données et les informations des utilisateurs.

Description du scénario :

La figure ci-dessus illustre le scénario d'authentification. Il commence après que l'utilisateur choisit de lancer une requête. Le système s'occupe de l'exécution de tâches planifiées :

- L'utilisateur insère son login ainsi que son mot de passe.
- Le système procède à la comparaison de l'identifiant et du mot de passe à ceux stockés dans la base de données. Un traitement s'effectue en Backend pour assurer la sécurité de chaque compte employé.
- Si les identifiants sont équivalents à ceux présents dans la base de données, l'utilisateur accède à son tableau de bord.
- Sinon, l'utilisateur reçoit un message d'erreur et on lui demande de ressaisir ses données.

3-1-2-2 Diagramme de séquences : La gestion des profils

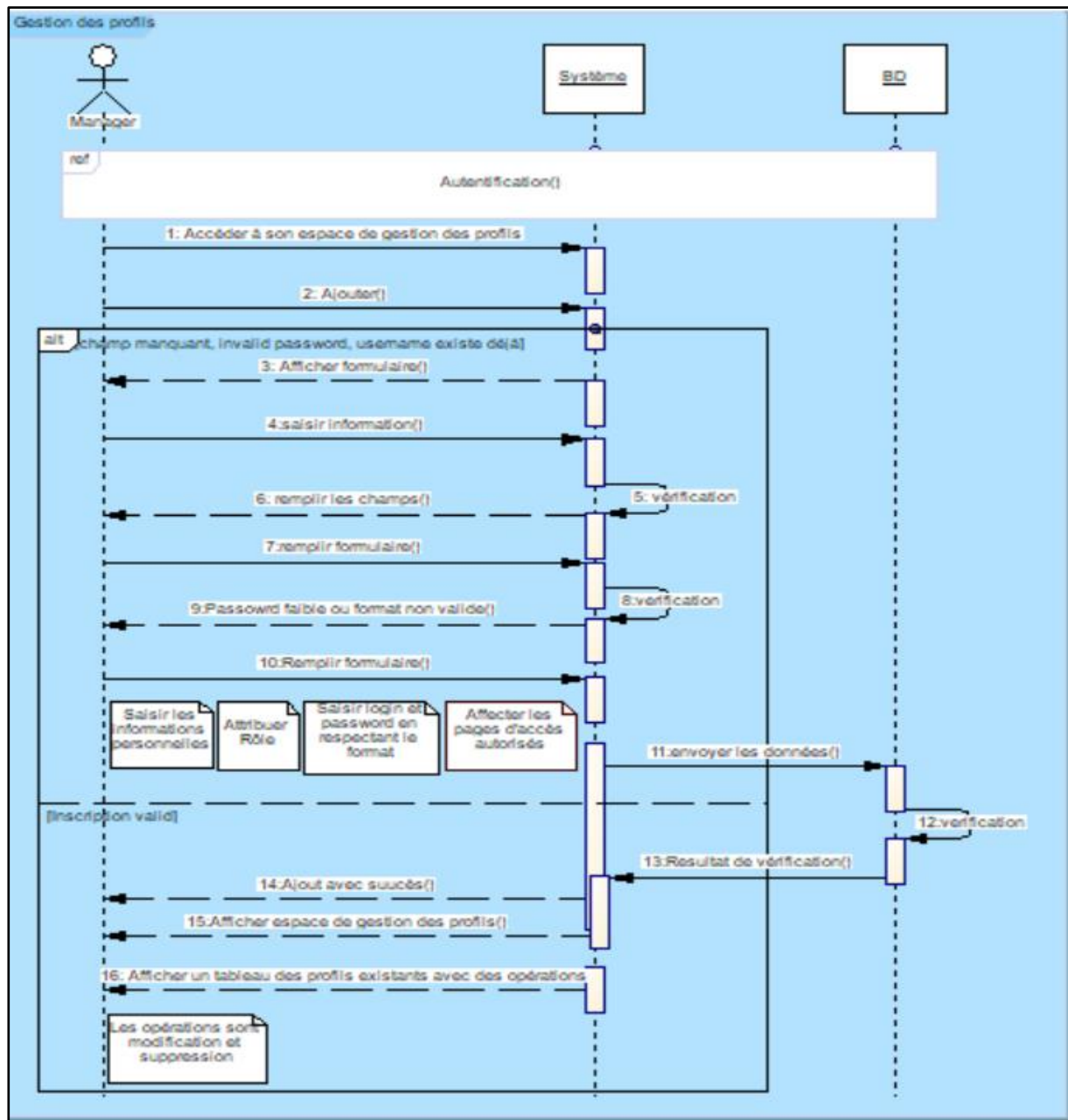


Figure 16: Diagramme de séquence la gestion des profils

Description des objets du diagramme :

- Manager : Administrateur qui accède à l'application via un navigateur internet. Il a déjà des données à saisir (login et mot de passe) pour accéder à son Dashboard, et réaliser les opérations souhaitées.
- Système : C'est le système responsable sur le traitement des requêtes http envoyées par l'entité utilisateur.
- Base de données: C'est l'espace de stockage qui englobe toutes les données et les informations des utilisateurs.

Description du scénario:

Le manager étant responsable sur la gestion des profils fait des actions d'ajout, de modification et de suppression. Pour ajouter un profil, il doit passer par un formulaire de saisie des données relatives au nouveau profil qu'il souhaite inscrire dans le système. Ce formulaire demande plusieurs informations personnelles y compris le login et le mot de passe.

Le manager est amené à attribuer des rôles à chaque utilisateur qu'il crée pour lui donner des droits d'accès. Avant d'enregistrer le nouvel utilisateur, une vérification s'effectue au niveau du formulaire pour contrôler les patterns de saisie pour chaque champ ainsi qu'une vérification dans la base de données pour allouer un espace dynamique en mémoire pour l'ajout. Le manager est redirigé après le succès de l'opération vers la page de gestion des profils, qui lui présentera un tableau avec tous les enregistrements existants en base de données avec la possibilité de suppression et de modification ainsi que le blocage des comptes des utilisateurs.

3-1-2-3 Diagramme de séquences : Suivi des parcelles agricole

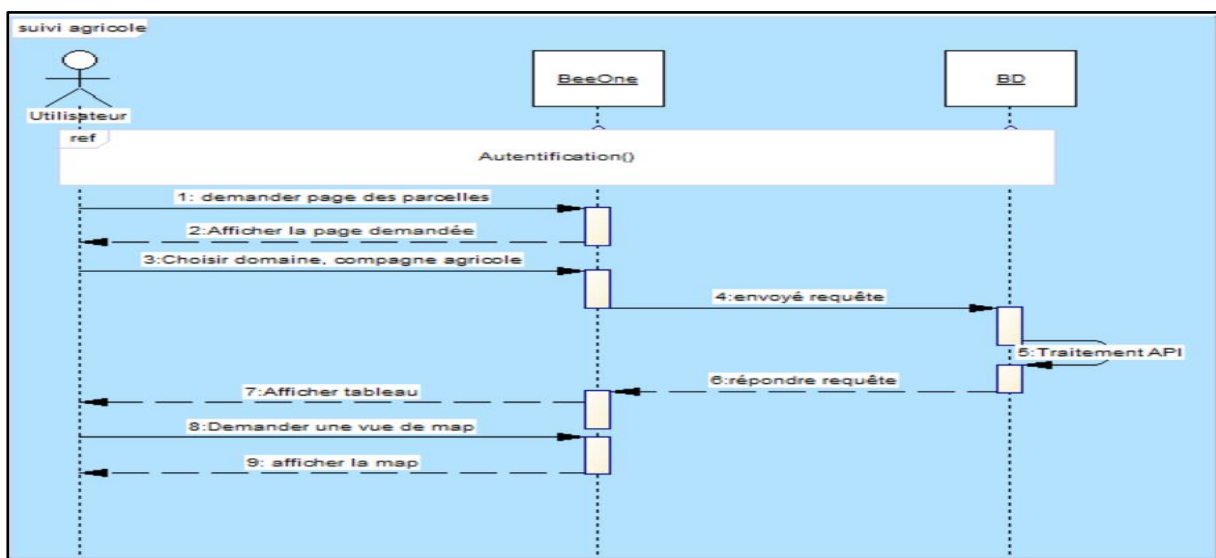


Figure 17 Diagramme de séquence suivie des parcelles agricole

Description des objets du diagramme :

Utilisateur : Administrateur ou technicien qui accède à l'ERP via un navigateur internet. Il a déjà des données à saisir (login et password) pour accéder à son espace et réaliser les opérations souhaitées.

BeeOne : c'est le système responsable sur le traitement des requêtes http envoyées par l'entité utilisateur.

Base de données : C'est l'espace de stockage qui englobe toutes les données et les informations des utilisateurs.

Description du scénario :

Avant toute opération, l'authentification est indispensable pour interagir avec tous les processus de l'application. En effet, après avoir rempli la mire de connexion avec succès, l'utilisateur demande la page de suivi des parcelles agricoles. Il sera amené à faire des filtres pour envoyer des requêtes précises à savoir le domaine, la compagnie agricole, etc. Une fois que l'utilisateur choisit ses filtres, les entrées de la requête définissent l'API qui sera appelé par le serveur. Cette API retourne une réponse JSON qui sera exploitée pour charger les données qui dépend du filtre choisi par l'utilisateur.

3-1-2- Diagramme de classe :

Le diagramme de classe permet de s'approcher à la structure de la base de données du système en indiquant la structure des objets (classes) qui les composent et les liens structurels entre eux.

On a extrait un ensemble d'entités et de dépendances, cela a été traduit par UML en ce diagramme de classes :

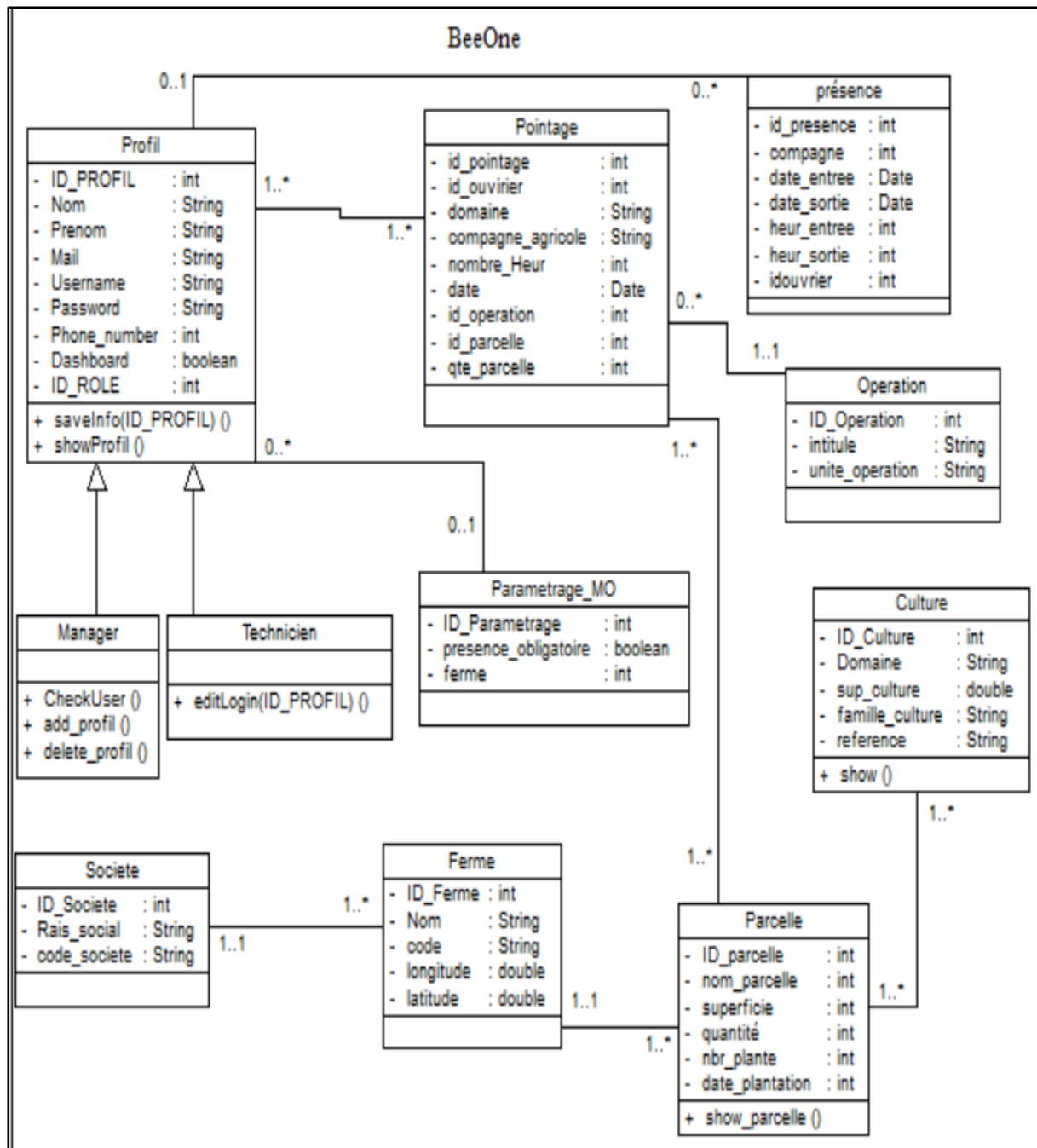


Figure 18: Diagramme de classe Gestion de la main d'œuvre

Le diagramme de classe dans cette figure présente les classes protagonistes issues du schéma de la base de données et spécialement le modèle de gestion de la main d'œuvre. Il s'agit en premier temps de la classe profil qui regroupe, en héritage, le manager et le technicien. Elle contient les informations relatives à chaque profil présent dans la base de données. Quant à la classe pointage, elle est consultée par les profils qui ont l'accès de gérer ou visualiser le pointage. On les représente le nombre d'heures de travail, la ferme et la parcelle dont il est affecté l'employée... etc. Chaque instance de la classe profile pourrait paramétrer si la présence est obligatoire ou non, afin de faire un suivi du pointage. Toutes ces classes concernent une certaine parcelle. Cette dernière appartient à une ferme et elle est caractérisée par sa culture.

IV. CONCLUSION

Dans ce chapitre, nous avons détaillé la partie développement du projet en commençant par l'étude des différents cas d'utilisations de notre système, ensuite l'identification des acteurs qui interagissent avec le système.

CHAPITRE 5 :

Mise en œuvre, Déploiement et implémentation

I. INTRODUCTION

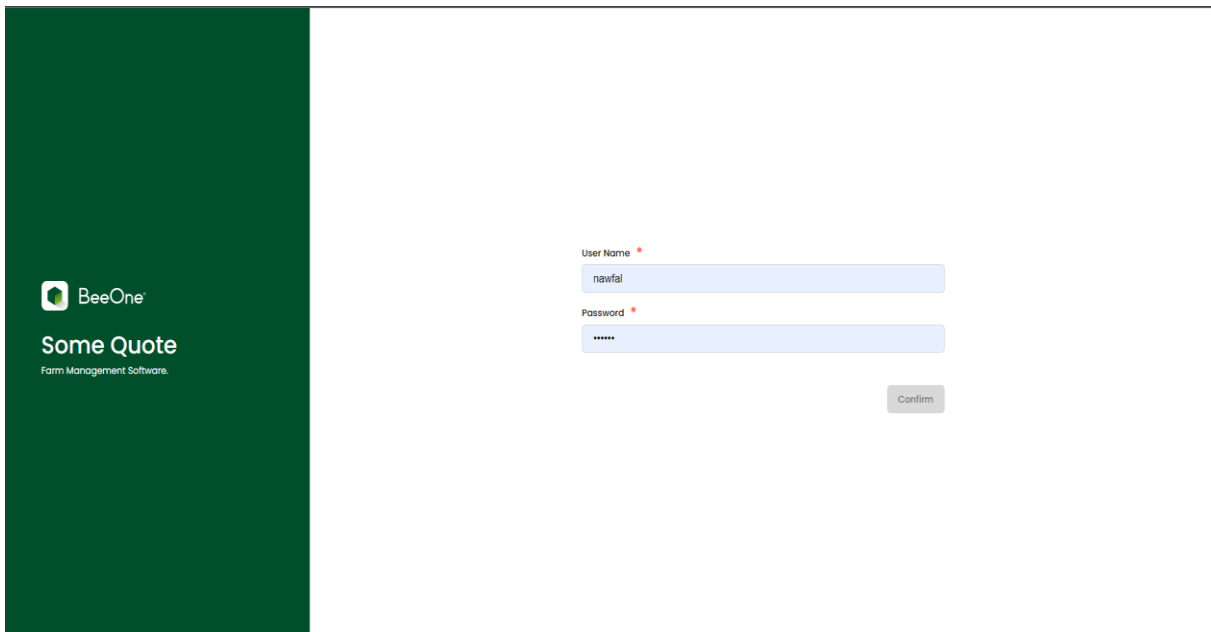
Dans le chapitre précédent, nous avons détaillés la phase de conception de notre projet, et nous présentons par la suite la dernière partie de notre rapport : c'est la phase d'implantation qui comprend la conversion de notre modèle conceptuel aux interfaces qui composent notre système.

II. RÉALISATION

Comme déjà mentionné, AGRIDATA a choisi la solution SaaS pour son projet BeeOne. Tout au long de ces mois de stage, nous avons passé par différentes étapes, à savoir la phase d'étude et analyse pour clarifier les buts principaux du projet, ensuite la modélisation et la conception que nous avons établies dans durant le chapitre précédent sous format des différents diagrammes. Nous présentons par la suite la phase de réalisation de notre système, c'est la démarche du projet qui implique la réalisation des interfaces et leurs interactions avec la base de données à l'aide des APIs.

2-1- Les interfaces graphiques du système

2-1-1- L'authentification

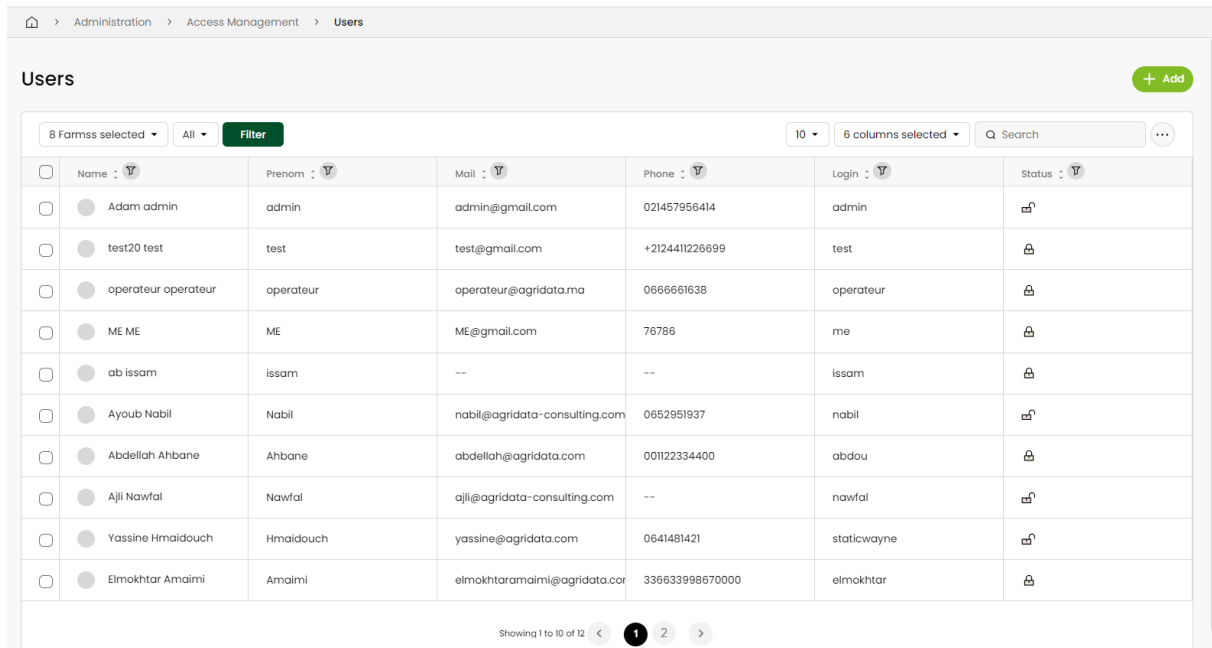


*Figure 19*Interface d'authentification

Pour garder le système sécurisé, il est indispensable de passer par la page d'authentification afin d'interagir avec les différentes autres interfaces du système. Il s'agit de la première étape effectuée par les acteurs qui auront le droit d'utiliser le système. Ces acteurs devront saisir leur nom d'utilisateur et mot de passe qui est déjà créé par l'administrateur. Si les informations saisies sont valides l'utilisateur serait dirigé vers la page en respectant les droits d'accès prédéfinis.

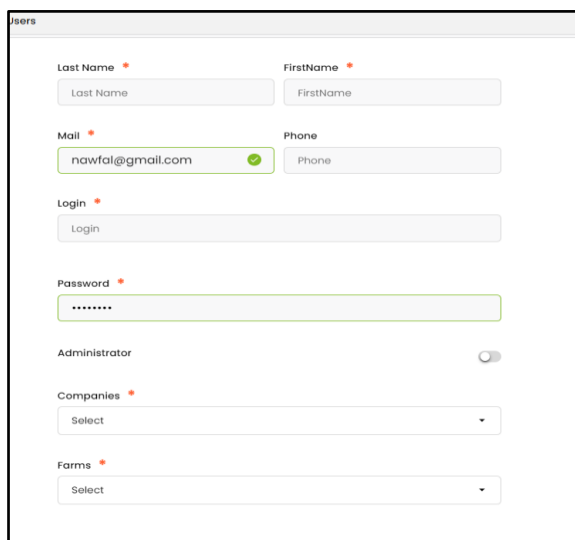
Le système utilise aussi la JWT pour assurer l'authentification en toute sécurité, c'est l'échange des informations d'une manière sécurisée. C'est une technique utilisée pour le mécanisme d'authentification stateless ce qui signifie que le maintien de la session est sur le côté client au lieu de stocker les sessions sur le serveur.

2-1-2- Gestion des profils



	Name	Prenom	Mail	Phone	Login	Status
<input type="checkbox"/>	Adam admin	admin	admin@gmail.com	021457956414	admin	
<input type="checkbox"/>	test20 test	test	test@gmail.com	+212441226699	test	
<input type="checkbox"/>	operateur operateur	operateur	operateur@agridata.ma	0666661638	operateur	
<input type="checkbox"/>	ME ME	ME	ME@gmail.com	76786	me	
<input type="checkbox"/>	ab issam	issam	--	--	issam	
<input type="checkbox"/>	Ayoub Nabil	Nabil	nabil@agridata-consulting.com	0652951937	nabil	
<input type="checkbox"/>	Abdellah Ahbane	Ahbane	abdellah@agridata.com	001122334400	abdou	
<input type="checkbox"/>	Ajli Nawfal	Nawfal	ajli@agridata-consulting.com	--	nawfal	
<input type="checkbox"/>	Yassine Hmaidouch	Hmaidouch	yassine@agridata.com	0641481421	staticwayne	
<input type="checkbox"/>	Elmokhtar Amaimi	Amaimi	elmokhtaramaimi@agridata.com	336633998670000	elmokhtar	

Figure 20 Interface Tableau de consultation des profils



Last Name *
Last Name

First Name *
First Name

Mail *
nawfal@gmail.com

Phone
Phone

Login *
Login

Password *
Password

Administrator
☐

Companies *
Select

Farms *
Select

Figure 21 : Formulaire d'ajout d'un utilisateur

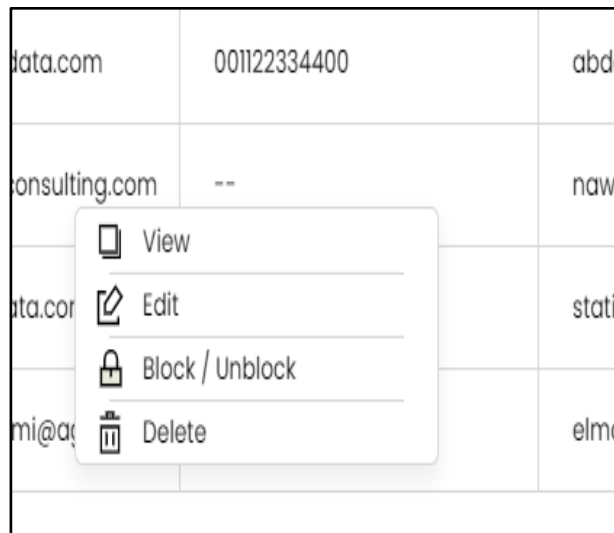


Figure 22: Menu de gestion des profils

Le manager en tant qu'acteur principal a le droit de consulter la table qui regroupe tous les profils de l'application. Il est capable d'ajouter un nouveau profil, le modifier ou le supprimer. Il s'agit alors d'une simple opération CRUD effectuée par le manager. Il pourrait aussi voir les détails d'un utilisateur avec la possibilité de bloquer ou débloquer leur accès.

2-1-3- La barre de menu

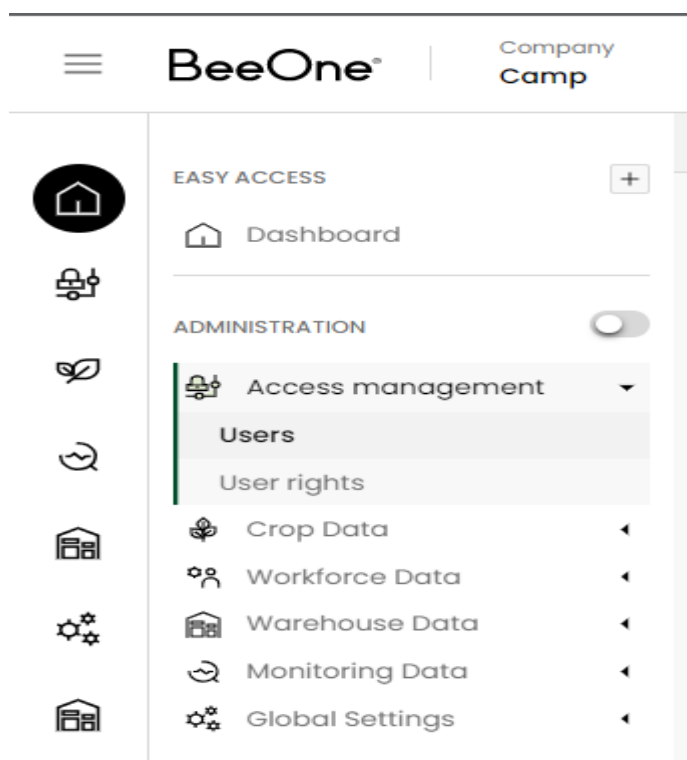


Figure 23 : Menu principal de BeeOne

Le menu principal regroupe l'ensemble des accès pour l'utilisateur connecté. Les icons qui sont à gauche représentent les sept modules réalisés de l'ERP BeeOne jusqu'à présent. Ces modules sont les suivants: l'administration, gestion de l'actif, field operation, surveillance, entrepôt de données et la gestion de la main d'œuvre, (il reste la gestion des rendements, ventes, approvisionnement, coûts, budgets et rapport) ils sont en cours de réalisation afin de compléter les 12 modules.

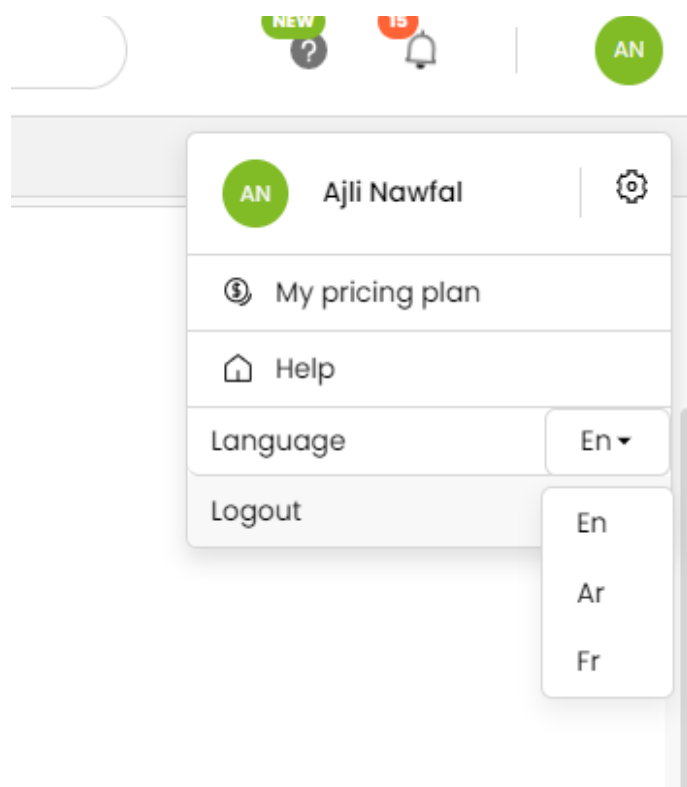


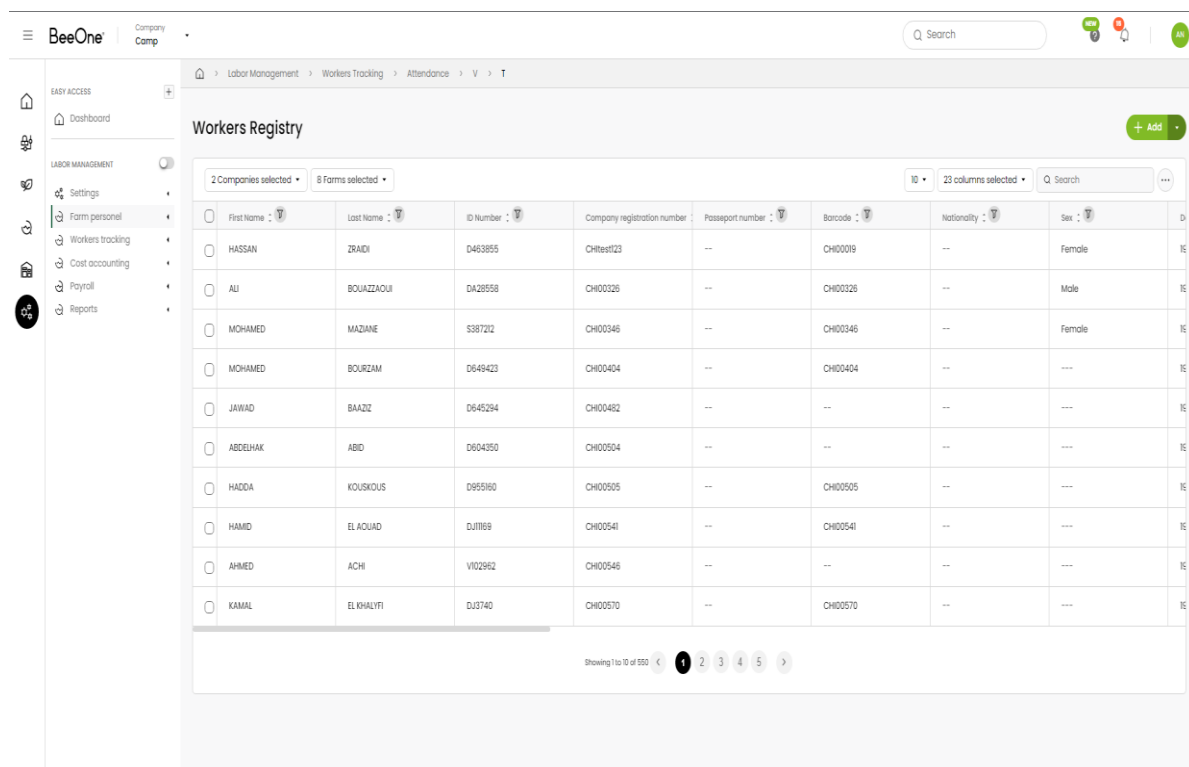
Figure 22: Menu secondaire de BeeOne

L'utilisateur connecté pourrait changer le langage du système, et consulter les informations concernant son profil.

Ainsi qu'un bouton Help qui affiche des informations aux nouveaux utilisateurs pour leur faciliter la compréhension des différentes fonctionnalités du système.

et finalement un Logout pour se déconnecter du système.

2-1-4- Gestion du personnel de la ferme



	First Name :	Last Name :	ID Number :	Company registration number :	Passport number :	Barcode :	Nationality :	Sex :	Date of Birth :
<input type="checkbox"/>	HASSAN	ZRAIDI	D463855	CHtest123	--	CH00019	--	Female	15
<input type="checkbox"/>	ALI	BOUAZZAOUI	DA29558	CH00326	--	CH00326	--	Male	15
<input type="checkbox"/>	MCHAMED	MAZIANE	S387212	CH00346	--	CH00346	--	Female	15
<input type="checkbox"/>	MCHAMED	BOURZAM	D649423	CH00404	--	CH00404	--	---	15
<input type="checkbox"/>	JAWAD	BAAZIZ	D645294	CH00482	--	--	--	---	15
<input type="checkbox"/>	ABDELHAK	ABID	D604350	CH00504	--	--	--	---	15
<input type="checkbox"/>	HADDA	KOUSKOUS	D955960	CH00505	--	CH00505	--	---	15
<input type="checkbox"/>	HAMD	EL AOUAD	DJ11819	CH00541	--	CH00541	--	---	15
<input type="checkbox"/>	AHMED	ACHI	V102962	CH00546	--	--	--	---	15
<input type="checkbox"/>	KAMAL	EL KHALYFI	DJ3740	CH00570	--	CH00570	--	---	15

Figure 25: Interface de consultation des personnels

Cette interface représente un tableau de consultation des employées qui sont enregistrées de la part de leur manager. Chaque employée est affectée à une ferme qui se trouve dans une campagne. Il est obligatoire de saisir toute information concernant l'employé à savoir son identification (les identifiants, le code barre ...), sa fonction agricole (rôle, date d'embauche...), et la paie (salaire de base et la banque..) ainsi que la ferme et la campagne dans laquelle il appartient. Les figures ci-dessous présentent le formulaire à remplir pour la gestion du personnel de la ferme :

Workers Registry

Step 1
Identification

First Name *

First Name

Last Name *

Last Name

Company registration number *

Company registration number

ID Number *

ID Number

Passeport number

Passeport number

Barcode

Barcode

Nationality

Step 2
Farm function

Function +

Select

Category +

Select

Hiring date *

Hiring date

Active *

☒

Formation Phyto *

☒ Yes ☐ No

Observations

Observations

Step 3
Payroll

Base salary *

Base salary

Bank +

Select

Step 4
Companies & Farms

Companies *

Select

Farms *

Select

Main farm *

Select

Figure 26: Formulaire d'ajout des personnels

2-1-5- Suivi de la présence

Form	Registration number	First name	Last name	Entry date	Exit date	Entry hour	Exit hour	hour number	Creation Date	Created by
<input type="checkbox"/> FERME AAC	CH18_0000	PI	NI	20/01/2021	--	--	--	0	--	--
<input type="checkbox"/> FERME AAC	CH18_0000	PI	NI	08/02/2021	--	--	--	0	--	--
<input type="checkbox"/> FERME AAC	CH18_0000	PI	NI	28/02/2021	--	--	--	0	--	--
<input type="checkbox"/> FERME 2	CH18_0000	PI	NI	03/06/2021	--	--	--	0	--	--
<input type="checkbox"/> FERME 2	CH18_0000	PI	NI	22/06/2021	--	--	--	0	--	--
<input type="checkbox"/> FERME AAC	CH0251	RADOUANE	OURAHOU	20/08/2020	--	--	00:00:00	0	--	--
<input type="checkbox"/> FERME AAC	CH0251	RADOUANE	OURAHOU	05/08/2020	05/08/2020	--	16:00:00	8	--	--
<input type="checkbox"/> FERME AAC	CH0251	RADOUANE	OURAHOU	11/08/2020	--	--	00:00:00	0	--	--
<input type="checkbox"/> FERME AAC	CH0251	RADOUANE	OURAHOU	20/01/2021	--	--	--	0	--	--
<input type="checkbox"/> FERME AAC	CH0251234	PI	NI	28/01/2021	--	--	--	0	--	--

Figure 27: Interface de gestion des présence

La figure ci-dessus représente l'interface de suivi de présence, c'est la gestion précise de la présence des employés. En premier lieu, le manager devrait insérer la date et l'heure d'entrée et de sortie de chaque employée, cette solution est efficace pour le traitement de la paie surtout si l'effectif important est dispersé sur différents sites.

L'utilisateur pourrait visualiser un Dashboard sous forme de widgets qui représentent les jours travaillés, les heures supplémentaires ainsi que le moyen de l'effectif mensuel ce qui rend le traitement de la paie plus facile, plus transparent et plus précis.

La figure ci-dessous représente le formulaire d'ajout des présences :

Farm *

Select

Check *

☒ Check in ☐ Check out

Date * **Time ***

22/06/2022 23:44

Workers *

Select

Figure 28: Formulaire de pointage des ouvriers

L'utilisateur devrait sélectionner la ferme dans laquelle il appartient les ouvriers concernés, tout d'abord, s'il voulait enregistrer l'entrée il suffit de cliquer sur le checkbox 'check-in' pour leur afficher les ouvriers qui n'ont pas fait encore l'entrée, ensuite il saisit la date et l'heure de leur entrée. Si l'ouvrier a déjà une entrée, l'utilisateur clique sur le checkbox 'check-out' (il affiche les ouvriers qui ont déjà fait une entrée) pour leur affecter la date et l'heure de sortie.

2-1-6- Suivi du temps

BeeOne Company Camp

Search

LABOR MANAGEMENT

Dashboard

Settings

Farm personnel

Workers tracking

Cost accounting

Payroll

Reports

Time tracking

Table view

8 Farms selected

20 Operations selected

01/06/2020 - 14/06/2022

Filter

10

7 columns selected

Q Search

	Form :	Date :	Operation :	Headcount :	Hours :	Creation Date :	Created by :
<input type="checkbox"/>	FERME AAC	11/06/2021	Alimentation cartons vides		1	0	--
<input type="checkbox"/>	FERME AAC	11/06/2021	Entretien pivots		1	10	--
<input type="checkbox"/>	FERME AAC	12/06/2021	Chisel		2	0	--
<input type="checkbox"/>	FERME AAC	12/06/2021	Chisel		1	0	--
<input type="checkbox"/>	FERME AAC	31/06/2021	Confection des buttes		1	0	--
<input type="checkbox"/>	FERME AAC	02/06/2021	Confection des billons		1	0	--
<input type="checkbox"/>	FERME AAC	02/06/2021	Confection des buttes		1	0	--
<input type="checkbox"/>	FERME AAC	05/06/2021	Analyses		1	5	--
<input type="checkbox"/>	FERME AAC	06/12/2021	Capotaillage		2	16	06/12/2021
<input type="checkbox"/>	FERME AAC	06/12/2021	Fauchage mécanique		8	64	06/12/2021

Showing 1 to 10 of 20

Figure 29: Interface de suivi du temps

Cette interface représente la gestion du “Time tracking” ou suivi du temps, c’est une solution importante dans la gestion de la main d’œuvre, il fait généralement référence à la manière dont la campagne enregistre les heures de travail des employés. L’utilisateur pourrait filtrer le tableau de consultation par ferme et opération agricole pour leur faciliter la visualisation. La figure ci-dessous présente le formulaire de saisie du time tracking :

Step 1
Operation and Plots

Form *

Farm *

24/06/2022

Date *

Operation *

Irrigation

Plots *

cocalico12.7FOUR7

Table input

Crop Plot ;	Area ;	Amount ;
cocalico12.7FOUR7	12	3

Step 2
Workers

Workers *

(CHIO0505) KOUSSOUS HADDA

Table input

Registration Number ;	Name ;	Base salary ;	Amount ;	Hours worked ;
CHIO0505	KOUSSOUS HADDA	76.7		3

Figure 30: Formulaire de suivi du temps

Le formulaire de suivi du temps est divisé en deux parties, d’une part nous avons “opérations et parcelle”. L’utilisateur devrait choisir la ferme afin de leur afficher les opérations et les parcelles concernées, ensuite il saisit le coût de cette parcelle. D’une autre part, nous avons la partie des ouvriers. La sélection de ces derniers se base sur un paramétrage de la présence, c’est-à-dire pour chaque ferme, il faut cocher si la présence est obligatoire ou non. Puis, l’utilisateur saisit les heures travaillées sur chaque opération. Cette mesure est utilisée pour s’assurer que les employés sont rémunérés en conséquence.

farms *

FERME AAC

Attendance required

Figure 31: paramétrage présence obligatoire

2-1-7- Dashboard de gestion de la main d'œuvre :

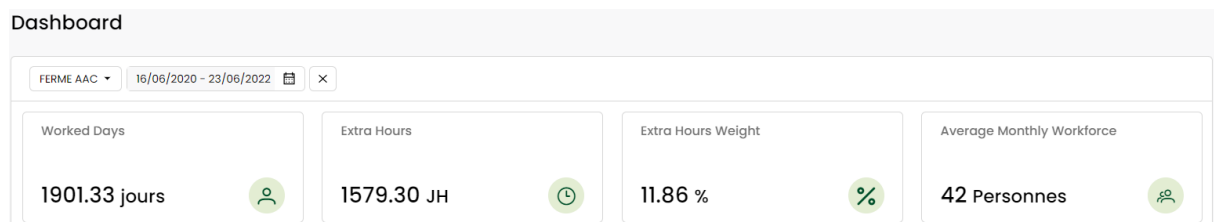


Figure 32: Dashboard : Indicateur de performance

Les widgets ci-dessus représentent les mesures des kpis pour les jours travaillés, les heures supplémentaires et leur poids, ainsi que le moyen de l'effectif mensuel.

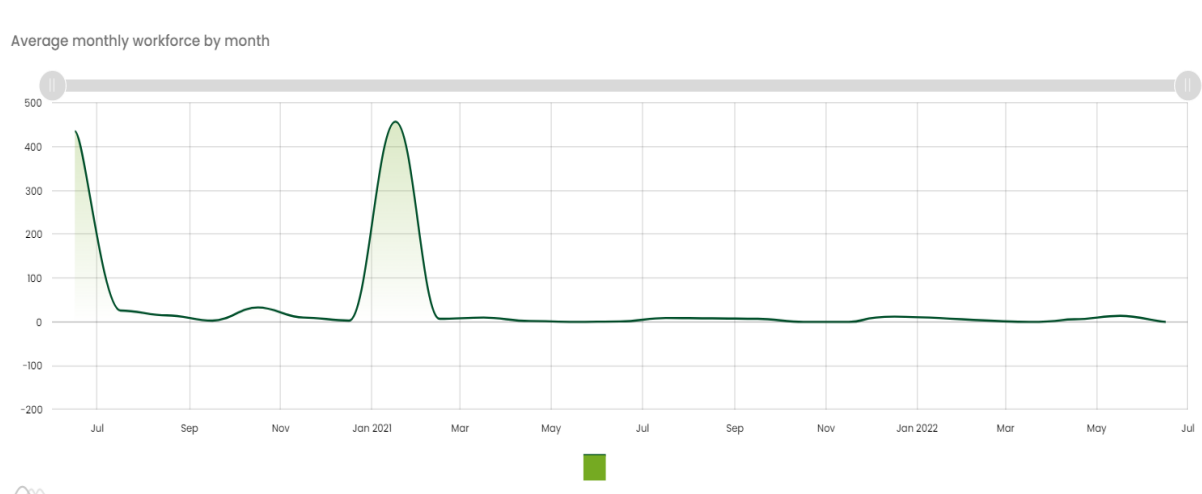


Figure 33: graphique linéaire, moyen de l'effectif mensuel par mois

Le graphique linéaire ci-dessus représente le moyen de l'effectif mensuel par mois.

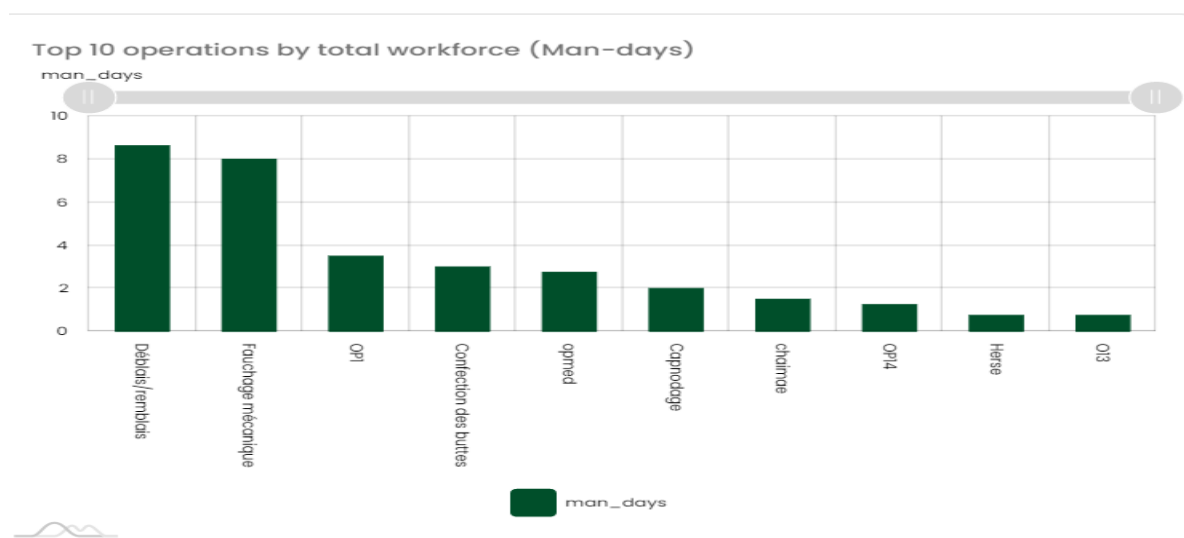


Figure 34: graphique en barre, meilleur 10 opération agricole par effectif total

Le graphique en barre ci-dessus représente les dix meilleures opérations par l'effectif total man-days (en prenant la quantité de main d'œuvre nécessaire pour effectuer ces opérations).

Cette analyse permet à l'expert qui lui est associé d'étudier des données relatives aux ressources exploitées dans certaines opérations. Il s'agit d'avoir une vision globale sur l'effectif dédié à chaque opération avec le nombre d'heures travaillées qui vient d'être mentionné. L'acteur ayant accès à cet onglet pourrait visualiser toutes les données concernées. Il donnerait une vision sur le besoin de recrutement pour combler un besoin dans une période précise et maintenir une certaine balance entre le budget disponible et le coût total des ressources humaines. La main d'œuvre est caractérisée par une vitesse de récolte qui évalue son rendement et sa productivité

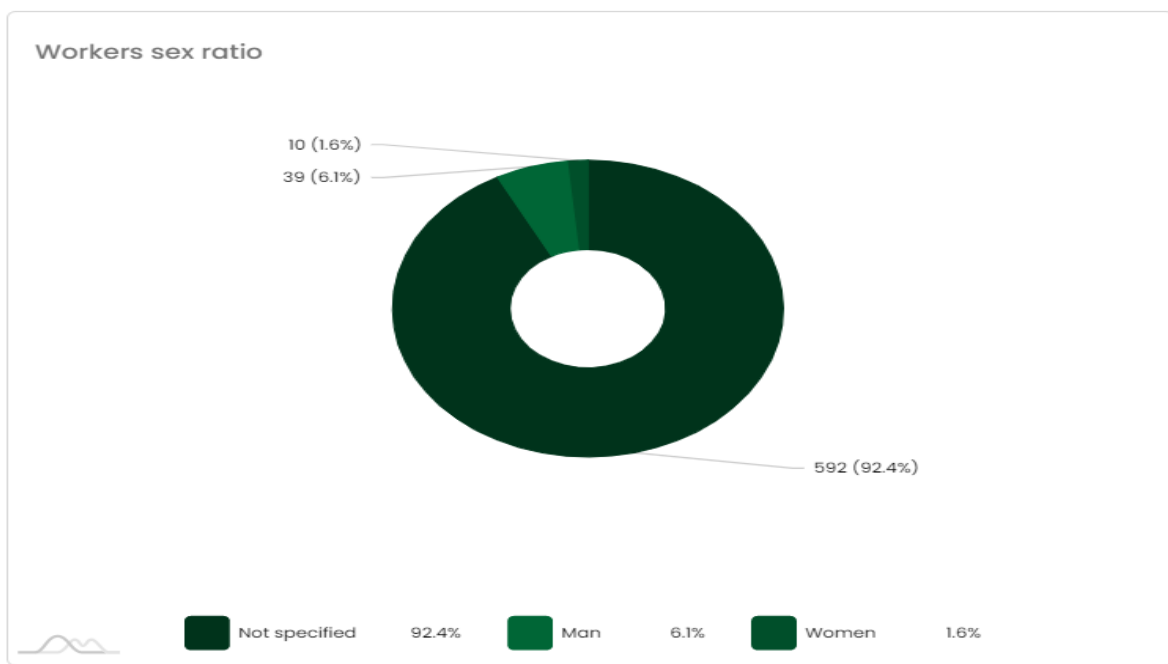


Figure 35: diagramme circulaire, ratio sexe des ouvriers

Cette interface représente le diagramme circulaire du ratio-sexe des employés: (1.6% des femmes et 6.1% des hommes, et 92% n'ont pas spécifiés leurs sexe).

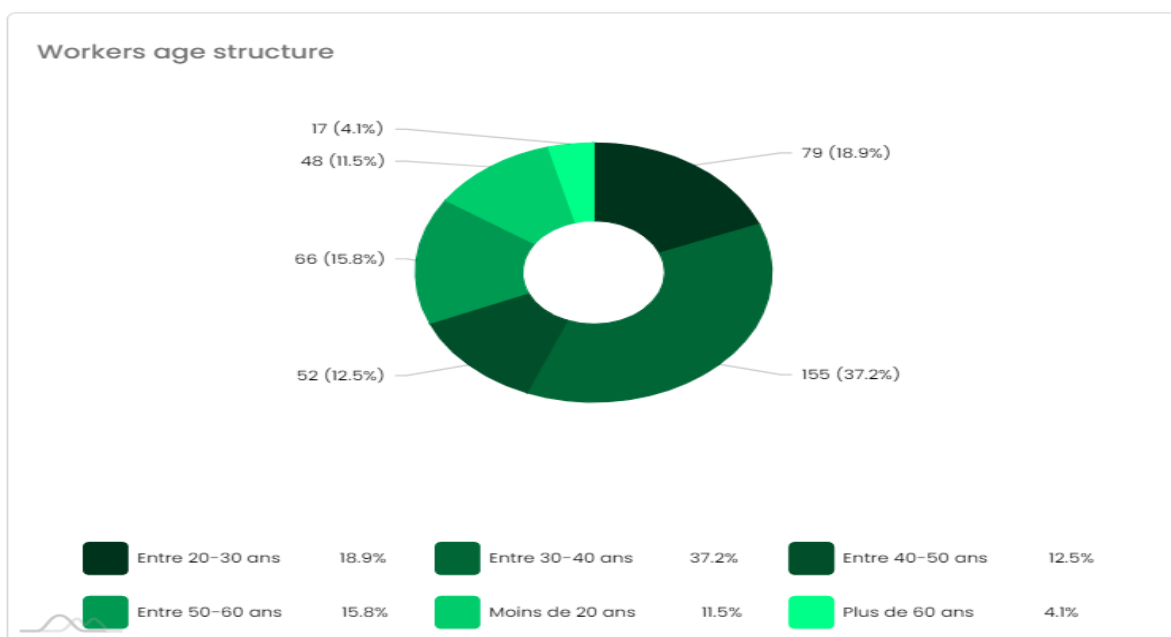


Figure 36 : diagramme circulaire : la structure d'âge des ouvriers

Le diagramme circulaire ci-dessus représente la structure d'âge des ouvriers.

L'analyse de la main d'œuvre permet une vision globale sur l'ensemble des renseignements concernant les ouvriers et l'avancement dans leurs tâches grâce au suivi des horaires et la gestion des heures travaillées. Nous avons commencé par le suivi des personnels pour gérer leurs dossiers, ensuite la gestion de la présence et du temps passé sur chaque opération, il nous reste le suivi de la récolte, l'analyse des coûts et de la paie (ils sont en cours de développement) pour clôturer le module de gestion de la main d'œuvre.

Cette analyse permet à l'expert qui lui est associé d'étudier des données relatives aux ressources exploitées dans certaines opérations. Il s'agit d'avoir une vision globale sur l'effectif dédié à chaque opération avec le nombre d'heures travaillées qui vient d'être mentionné. L'acteur ayant accès à cet onglet pourrait visualiser toutes les données concernées. Il donnerait une vision sur le besoin de recrutement pour combler un besoin dans une période précise et maintenir une certaine balance entre le budget disponible et le coût total des ressources humaines. La main-d'œuvre est caractérisée par une vitesse de récolte qui évalue son rendement et sa productivité.

2-1-8- Produit de récolte

Harvest Produce

+ Add

Table view

Produce barcodes

FERME AAC

Crop

Variety

X

Filter

10

6 columns selected

Q Search

...

<input type="checkbox"/>	Produce reference :	Produce name :	Produce type :	Storable :	Creation date	Created by :
<input type="checkbox"/>	PF111010067	Mandarine Nadorcott Vrac tout v	Secondary	Not Storable	--	
<input type="checkbox"/>	PF111010068	Clémentaine Ain taoudate Vrac	Secondary	Storable	--	
<input type="checkbox"/>	PF111010069	Clémentaine Azemmour Vrac to	Secondary	Storable	--	
<input type="checkbox"/>	PF111010071	Clémentaine Cadoux Vrac tout v	Secondary	Storable	--	
<input type="checkbox"/>	PF111010072	Clémentaine Carte noire Vrac to	Secondary	Storable	--	
<input type="checkbox"/>	PF111010073	Citron Citron Vrac tout venant	Secondary	Not Storable	--	
<input type="checkbox"/>	PF111010075	Clémentaine Commune Vrac to	Secondary	Storable	--	
<input type="checkbox"/>	PF111010076	Clémentaine Ezbal Vrac tout ven	Secondary	Storable	--	
<input type="checkbox"/>	PF111010077	Clémentaine Fortuna Vrac tout v	Secondary	Storable	--	

Figure 37: interface des produits de la récolte

Cette interface représente le tableau de consultation des produits de la récolte. En cliquant sur le bouton add en haut à droite, il permettrait d'ajouter un nouveau produit avec les informations nécessaires à savoir la référence, le nom, le type...etc. toutes ces informations seraient filtrées que ce soit par la ferme ou par la culture et la variété.

La figure ci-dessous présente le formulaire d'ajout d'un produit de récolte :

Produce reference *

Prod33

Produce name *

prod33

Variety *

AMANDA, AMAZONITE

Produce type *

☒ Primary

☐ Secondary

Storable *

☐ Storable

☒ Not Storable

Description

prod33

Average Unit Weight

Unit *

Select

Kg

Average weight (Kg) *

Average weight (Kg)

4

+

×

Figure 38: Formulaire d'ajout des produits récoltés

Dans le formulaire nous avons saisi les informations sur le produit en ajoutant aussi le poids moyen d'une unité et la figure suivante présente les détails du produit que nous venons d'enregistrer avec la possibilité de modifier ces données ou les supprimer.

The screenshot shows a mobile application interface for product details. At the top, there are navigation icons and buttons for 'Delete' and 'Edit'. Below this, a header section displays 'CREATION DATE Jun 24, 2022' and 'CREATED BY Ajli Nawfal'. The main content area is divided into two sections: 'Informations' and 'Associated Varieties'. The 'Informations' section contains a table with product details. The 'Associated Varieties' section contains a table with two rows of variety data. Below this, there is a section for 'Associated Units' with a table showing unit weight.

Informations	
Produce reference	Prod33
Produce name	prod33
Produce type	Primary
Storable	Not Storable

Associated Varieties 2	
Nom	Reference
AMANDA	11207002
AMAZONITE	11207003

Associated Units 1	
Unit	Average Weight (Kg)
Kg	4

Figure 39: interface consultation détails d'un produit

The screenshot shows a mobile application interface for generating a QR code. The title is 'QR Code'. There are two input fields: 'Unite Count' with a value of '1' and a green checkmark, and 'Barcode Type' with two radio buttons: 'Unique' (selected) and 'Duplicate'. Below these fields is a 'Preview Qr Code' button. The preview area shows a QR code and the following text: 'Farm : FERME AAC', 'Crop Plot : 13021 MELANGE OLIVIER', 'Unit : CAISSE', and 'Date : June 14th 2022, 3:20 Pm'. At the bottom right, there are 'Cancel' and 'Print' buttons.

Unite Count *

1

Barcode Type *

☒ Unique ☐ Duplicate

Preview Qr Code

Farm : FERME AAC
Crop Plot : 13021 MELANGE OLIVIER
Unit : CAISSE
Date : June 14th 2022, 3:20 Pm

Cancel Print

Figure 40: interface génération des QR code

La figure ci-dessus présente un aperçu d'une fiche qui contient le QR code et des informations sur le produit récolté, il serait imprimé et utiliser pour le packaging de ces produits. Cette technique permettrait aux agriculteurs d'améliorer l'efficacité de la chaîne d'approvisionnement.

Les produits récoltés tel que les fruits et légumes...etc. peuvent être dotés de cette étiquette pour qu'à chaque fois qu'elle soit scannée, il permettrait de récupérer toutes les informations pertinentes telle que la date de la récolte, l'origine, la ferme, l'unité...etc.

III. CONCLUSION

Dans ce chapitre nous avons détaillé la partie réalisation du projet en illustrant les différentes interfaces réalisées jusqu'à présent de la solution. C'est la partie de déploiement d'une version du progiciel dans la Saudia Arabia pour tester les retours clients et attendre leur feedback pour améliorer des fonctionnalités en cas de besoin. Il est temps d'authentifier le produit avec les spécifications du cahier de charges.

CONCLUSION GENERALE

Au terme de ce travail, nous avons été capable de mettre en œuvre cet ERP BeeOne qui est une solution SaaS digitale dédiée aux clients du secteur agricole de l'entreprise AgriDATA-consulting, nationale et internationale. Ceci en se basant sur des Frameworks du développement, à savoir Angular Et express et sur l'intégration d'outils graphiques tels que Amchart.

Cet ERP considéré comme une nouvelle vision dans le monde agricole est une solution de gestion de l'exploitation agricole consistant à optimiser et gérer les opérations agricoles et les activités de production. Le logiciel aide à automatiser les activités agricoles telles que la gestion des dossiers, le stockage des données, le suivi et l'analyse des activités agricoles, la gestion de la production et des horaires de travail...etc.

Durant ce travail, il était primordial de participer à la partie fonctionnelle à travers les diverses réunions. Ceci avait facilité la compréhension des règles métier et du périmètre agricole des besoins exprimés. De ce fait, nous étions capables de mener une analyse des besoins et de réaliser une conception détaillée et de choisir convenablement les outils et techniques avant d'arriver à la partie de réalisation et mise en œuvre.

La mission de ce projet était bénéfique, dans la mesure où elle permet de se familiariser avec un environnement professionnel du travail, de mettre en pratique les acquis durant la formation pédagogique à l'école ainsi que nos acquis novateurs en analyse, développement et programmation pour atteindre les objectifs et réaliser ce présent projet. Il est efficace d'utiliser un progiciel unique qui englobe douze modules pour toute la gestion de l'activité concerné, au lieu d'utiliser une multitude de plateformes qui laissent l'information liée à un projet dispatchée sur plusieurs outils.

D'ailleurs nous avons présenté dans un premier lieu le cahier de charges et l'analyse des besoins, ensuite, la modélisation et la conception qui nous ont permis d'avoir une vision claire sur l'ensemble des interactions du système. Enfin la réalisation des différentes interfaces.

Nous avons pensé à l'utilisation du langage UML, qui a été intéressant dans le niveau des données, car il nous a permis de présenter plusieurs diagrammes dans un soucis de clarté, grâce à l'avantage de la multi vie, nous avons pu décrire les différentes fonctionnalités de l'ERP BeeOne qui répondre à notre problématique

BIBLIOGRAPHIE

Sarrion, Eric(2022) *Programmation avec Node.js, Express.js et MongoDB : JavaScript côté serveur Ed.*

Bojinov, Valentin (2018) *RESTFUL WEB API Design with Node.js 1° Ed 3*

Bassem El Haddad, Julien Oger "Scrum, de la théorie à la pratique" Initiation. Perfectionnement. Agilité.

Claudy Aubry "Scrum pour une pratique vivante de l'agilité" 5e édition DUNOD.

Joseph Gabay, David Gabay(2008) "UML 2 ANALYSE ET CONCEPTION" Dunod, Paris

Noring, Christoffer (2018) *Architecting Angular Applications with Redux, Rxjs and NgRx*

WEBOGRAPHIE

<https://agridata-consulting.com/> (consulter le 15 mars 2022)

[1] <https://slack.com/intl/fr-ma/> (consulter le 08 avril 2022)

[2]<https://www.blogdumoderateur.com/tools/clickup/> (consulter le 08 avril 2022)

<https://www.w3schools.com/angular/> (Consulté le 10 avril 2022)

[3] <https://www.postman.com/> (consulter le 20 avril 2022)

[4] <https://sass-lang.com/> (consulter le 15 mai 2022)

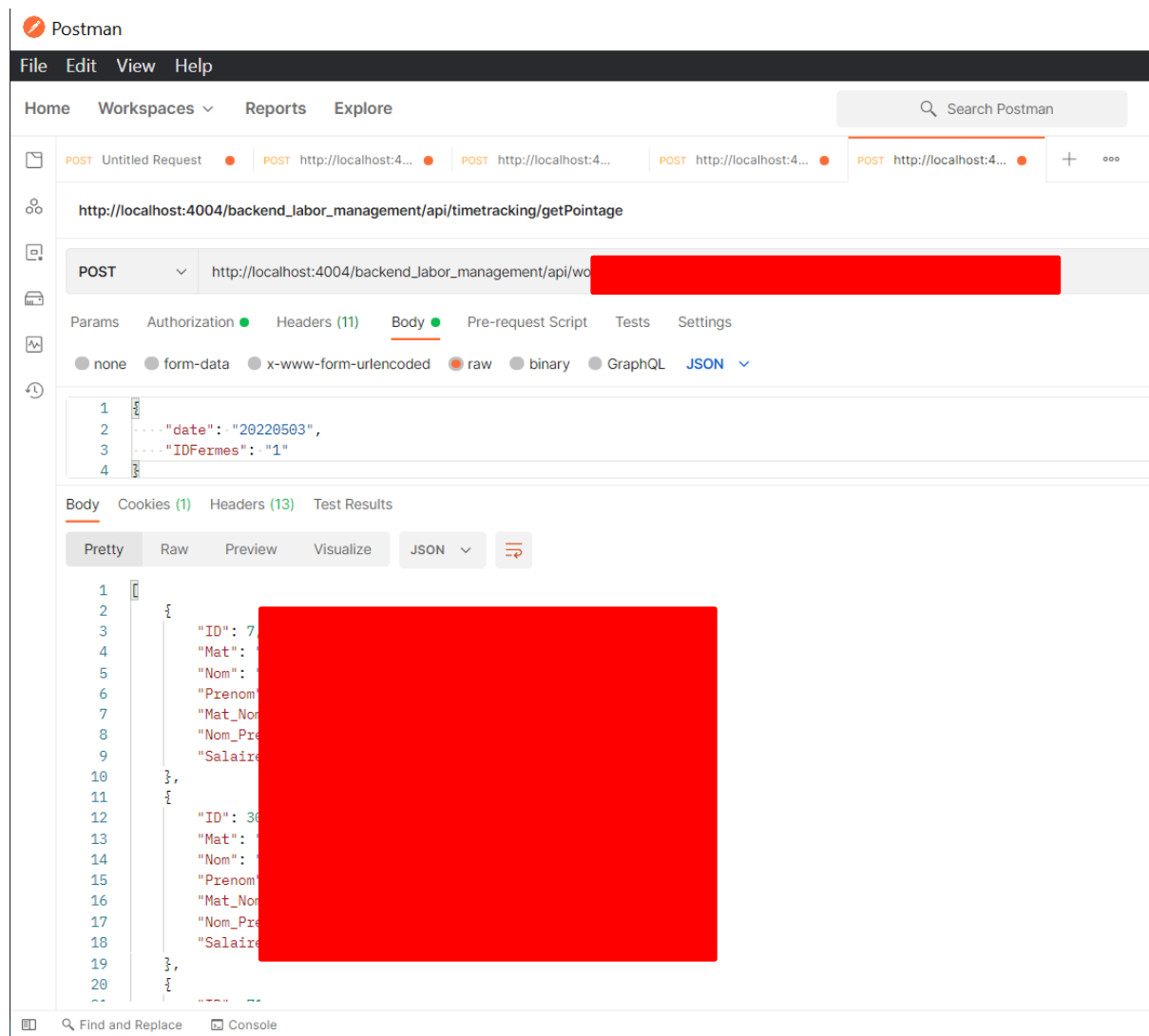
[5] <https://www.amcharts.com/docs/v5/> (consulter le 12 juin 2022)

[6] <https://sql.sh/sgbd/sql-server/> (consulter le 10 avril 2022)

[7] <https://www.geeksforgeeks.org/why-rest-api-is-important-to-learn/?ref=lbp>

(Consulter le 20 juin 2022)

ANNEXES



Annexe 1 : Le fichier JSON généré après l'appel de l'API sur Postman

Annexe 2: Préparation du backend, les requêtes à exécuter

```

server > lga-service > presence.service > JS presence.controller.js > asyncForEach
1  "use strict";
2
3  const { log } = require("console");
4
5  var _ = require("lodash"),
6      path = require("path"),
7      query = require(path.resolve("./server/query")),
8      crypto = require("crypto"),
9      moment = require("moment"),
10     TABLE = "Presence";
11
12  async function controlData() {
13      for (let index = 0; index < array.length; index++) {
14          await callback(array[index], index, array);
15      }
16  }
17  // controller
18
19  async function controlData(
20      // ...
21      // ...
22      // ...
23      // ...
24      // ...
25      // ...
26      // ...
27      // ...
28  ) {
29      var msg_retour = "";
30      msg_retour = {
31          message: "OK",
32          description: "OK",
33      };
34      var data_manque = "";
35      data_manque = ouvriers ? 1 : 0;
36      if (data_manque == 0) {
37          msg_retour = {
38              message: "ajout echoue",
39              description: "Ouvriers introuvable",
40          };
41          return msg_retour;
42      }
43      data_manque = IDFermes ? 1 : 0;
44      if (data_manque == 0) {
45          msg_retour = {

```

```

1  "use strict";
2
3  var express = require("express");
4  var path = require("path");
5  var presenceController = require(path.resolve(
6    "serve
7  ));
8
9  var router = express.Router();
10
11  router.use(function timeLog(req, res, next) {
12    var today = new Date();
13    var dd = today.getDate();
14    var mm = today.getMonth() + 1;
15    var yyyy = today.getFullYear();
16    var hh = today.getHours();
17    var min = today.getMinutes();
18    if (dd < 10) {
19      dd = "0" + dd;
20    }
21
22    if (mm < 10) {
23      mm = "0" + mm;
24    }
25
26    if (hh < 10) {
27      hh = "0" + hh;
28    }
29
30    if (min < 10) {
31      min = "0" + min;
32    }
33    console.log("Date : ", dd + "/" + mm + "/" + yyyy + " - " + hh + ":" + min);
34    next();
35  });
36  router.post("/", p
37  router.post("/view
38  router.post("/deta
39  router.post("/list
40  router.post("/list
41  router.post("/list
42
43  router.post("/crea
44  router.post("/upda
45  router.post("/dele

```

```

198  msg_retour = {
199    message: "ajout echoue",
200    description: "ID presence introuvable",
201  };
202  return msg_retour;
203  }
204  if (typeE == 2 || (typeE == 1 && ES == 2)) {
205    var queryString = "";
206    if (ES == 1) {
207      var queryString = 'select CONVERT(varchar, Presence.Date_sortie, 23) AS
208    } else if (ES == 2) {
209      var queryString = 'select CONVERT(varchar, Presence.Date_entree, 23) AS
210    }
211    let data = await query.asyncQuery(queryString);
212    if (data.length != 0) {
213      var dateE = data[0].Date;
214      var heureE = data[0].Heure;
215      if (ES == 2) {
216        if (
217          moment(dateE).format("YYYY-MM-DD") > moment(date).format("YYYY-MM-DD")
218        ) {
219          msg_retour = {
220            message: "ajout echoue",
221            description:
222              "La date de sortie doit être supérieure ou égale à la date d'entrée",
223          };
224          return msg_retour;
225        }
226      }

```