

ECOLE SUPERIEURE PRIVEE DES TECHNOLOGIES D'INFORMATION ET DE MANAGEMENT DE NABEUL



Mémoire de Fin d'Etudes

SUJET

Création d'une plateforme de gestion des projets

**Présenté en vue de l'obtention du titre
D'INGENIEUR EN INFORMATIQUE**

Elaboré par
Wael Chorfan

Encadrant Académique
Mme Safa Fennia

Encadrant Professionnel
Mr Bilel Boummijel



J'autorise l'étudiant à faire le dépôt de son rapport de stage en vue d'une soutenance.

Encadrant professionnel, Mr **Bilel Boumnijel**

Cachet et signature



J'autorise l'étudiant à faire le dépôt de son rapport de stage en vue d'une soutenance.

Encadrant académique, Mme, **Safa Fennia**

Cachet et signature

Table des matières

I Aspects Théoriques	4
1 Cadre géngérale du projet	5
1.1 Introduction	5
1.2 Contexte du travail	5
1.3 Présentation de l'organisme d'accueil	5
1.4 Contexte du projet	6
1.5 Business intelligence	7
1.6 Etude de l'existant	7
1.6.1 Asana	7
1.6.2 Jira	7
1.7 Problématique	8
1.8 Solution proposée	9
1.9 Etude méthodologique	10
1.9.1 Choix de la méthodologie	10
1.9.2 Présentation de la méthodologie SCRUM	10
1.9.3 Planning du projet	11
1.9.4 Diagramme de Gantt	13
2 Analyse et Spécifications besoins	14
2.1 Introduction	14
2.2 Les acteurs du système	14
2.3 Spécifications besoins	15
2.3.1 Besoins fonctionnels	15
2.3.2 Besoins non fonctionnels	16
2.4 Diagramme de cas d'utilisation générale	17
2.5 Backlog de planning	18
2.5.1 Les fonctionnalités du Backlog	18
2.5.2 Planification des sprints	19

II Aspects Pratiques	20
3 Release 1	21
3.1 Introduction	21
3.2 Sprint 1	22
3.2.1 Analyse	22
3.2.2 Conception	22
3.2.3 Schéma	22
3.2.4 Test	23
3.3 Sprint 2	26
3.3.1 Analyse	26
3.3.2 Conception	26
3.3.3 Test	27
3.3.4 Consultation carte géographique des clients	30
3.4 Sprint 3	30
3.5 Sprint 4	38
3.5.1 Conception	38
3.5.2 Test	39
3.6 Structure de l'application	40
3.7 Conception de la base de données	40
3.7.1 Modèle du liason des données	40
3.7.2 Diagramme de classes	42
3.8 Diagramme de déploiement	44
3.8.1 Diagramme de déploiement	44
3.9 Diagramme d'activité	45
4 Release 2	49
4.1 Introduction	49
4.2 Analyse et conception	50
4.2.1 Analyse	50
4.2.2 Conception	50
4.3 But de l'étude décisionnelle	51
4.4 Elaboration des rapports sur Power BI	51
4.5 Partie ETL	57
4.5.1 Intégration	57
4.5.2 Shémas obtenus	60
4.5.3 La consultation des rapports	63

5 Clôture du projet	66
5.1 Introduction	66
5.2 Implémentation et structure	66
5.3 Application en ligne	67
5.4 Architecture 3 tiers	68
5.5 Environnement de développement	68
5.5.1 Environnement matériel	68
5.5.2 Environnement logiciel	68
5.6 Outils et technologies	70

Table des figures

1.1	Logo de Cherchini.	6
1.2	Triangle d'or.	8
1.3	Architecture générale.	9
1.4	Logo de Cherchini.	11
1.5	Diagramme de Gantt.	13
2.1	Diagramme de cas d'utilisation générale.	17
3.1	Gérer un membre.	22
3.2	Gestion des membres.1.	24
3.3	Gestion des membres.2.	24
3.4	Gestion des membres.3.	25
3.5	Gérer un client.	26
3.6	Création d'une client.	27
3.7	Gestion des clients.1.	28
3.8	Gestion des clients.2.	29
3.9	Carte géographique.	30
3.10	Gérer un projet.	31
3.11	Gérer une tâche.	32
3.12	Création d'un projet.	33
3.13	Création d'une tâche.	34
3.14	Gestion de projet.1.	35
3.15	Gestion de projet.2.	36
3.16	Gestion de projet.3.	36
3.17	Gestion de projet.4.	37
3.18	Gestion de projet.5.	37
3.19	Gestion de projet.6.	38
3.20	Authentification.	39
3.21	Accueil.	39

3.22 Structure de l'application web.	40
3.23 Diagramme de classes.	42
3.24 Diagramme de déploiement.	45
3.25 Diagramme d'activité pour un membre.	48
4.1 Gérer un projet.	50
4.2 Consultation des rapports.	51
4.3 Ajustement des noms.	52
4.4 Ajustement de la durée totale des projets.	52
4.5 Coût total par projet.	53
4.6 Coût total par catégorie de projet.	54
4.7 Coût total par client.	55
4.8 Coût total par client en chiffres.	55
4.9 Durée total par projet (en jours).1	56
4.10 Durée total par projet (en jours).2	56
4.11 Extraction des coûts.1.	57
4.12 Extraction des coûts.2.	58
4.13 Extraction des coûts.3.	59
4.14 Extraction des durées.	60
4.15 Shéma initial.	61
4.16 Coût par projet	63
4.17 Coût par client	64
4.18 Coût par catégorie	64
4.19 Durée par projet.1.	65
4.20 Durée par projet.2.	65
5.1 phpmyadmin.	69
5.2 Wamp.	69
5.3 Visual Studio Code.	69
5.4 Staruml.	70
5.5 Node js.	70
5.6 Express js.	71
5.7 Postman.	71
5.8 Vue js.	72
5.9 MySQL.	72
5.10 EJS.	73
5.11 Highcharts.	73

Liste des tableaux

1.1	Planning du projet	12
2.1	Rôles des acteurs.	15
2.2	Product Backlog	19
2.3	Planification des sprints	19
3.1	Members.	23
3.2	Customers	28
3.3	Projects.	35
3.4	Relations	43
4.1	Shéma de la table des faits.	61
4.2	Contenu de la table des faits.	62

Dédicaces

Qu'il me soit permis au seuil de ce modeste mémoire d'exprimer ma plus profonde reconnaissance à : A ma mère,

"Tu m'as donné la vie, la tendresse, et le courage pour réussir. Tout ce que je peux t'offrir ne pourra pas exprimer l'amour et la reconnaissance que je porte. Je ne voudrais pas être sans toi.

En témoignage, je t'offre ce modeste travail pour te remercier pour tes sacrifices et pour l'affection dont tu m'as toujours entourée."

A mon père,

"L'épaule solide, l'œil attentif, compréhensif et la personne la plus digne de mon estime et mon respect. Aucune dédicace ne saurait exprimer mes sentiments. Que Dieu te préserve et te procure santé et longue vie."

A mes sœurs,

"Vous êtes les meilleurs du monde. Je vous souhaite de la joie et du succès pour les années à venir.

Remerciements

C'est avec un grand plaisir que je réserve ces lignes en signe de gratitude et de reconnaissance à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

Nous tenons à exprimer nos remerciements les plus vifs en premier lieu à ceux qui nous font l'honneur de participer et de siéger dans ce jury et d'être les rapporteurs de ce travail.

Nous remercions particulièrement Monsieur Bilel Boumnijel pour avoir eu l'amabilité de nous accueillir au sein de l'entreprise "Cherchini" et de nous avoir procuré ses précieux conseils tout au long de ce projet.

Nous tenons à remercier Madame Safé Fennia, enseignante du ITBS, pour son suivi, son aide ainsi que ses conseils tout au long de ce mémoire, ainsi que les enseignants de l'ITBS

Enfin, nos remerciements s'adressent aussi à tous nos enseignants pour la qualité de l'enseignement qu'ils nous ont prodigué durant nos études.

Avant-Propos

Cette étude entre dans le cadre de la préparation d'un mémoire de fin d'études en vue de l'obtention du diplôme d'Ingénieur en Informatique, au sein de L'Ecole Supérieure Privée de Technologie et de Management « ITBS ». C'est ainsi que nous avons eu l'occasion de préparer notre projet de fin d'étude intitulé: « Création d'une plateforme de gestion des projets » proposé par l'entreprise « Cherchini ».

Ce projet est un apport très bénéfique quant au perfectionnement des connaissances de l'étudiant dans le domaine informatique et pour avoir l'opportunité d'appliquer ses connaissances théoriques acquises tout au long de son cursus universitaire dans le cadre professionnel.

Mots clés : nodejs , mysql, javascript, html, bootstrap ,vuejs , vs code , project management..

Abbréviations

UML : Unified Modeling Language

JS : JavaScript

BD : Base de données

BI : Business Intelligence

ORM : Object Relational Mapping

JWT : Json Web Tokens

EJS : Embedded JavaScript

Introduction générale

Il ne fait désormais plus aucun doute que l'informatique est la révolution la plus importante et la plus innovante qui a marqué la vie de l'humanité moderne. En effet, les logiciels informatiques proposent maintenant des solutions à tous les problèmes de la vie, aussi bien dans des domaines professionnels que pour des applications personnelles. Et leurs méthodes de conception et de développement ont vu l'avènement d'autant de technologies qui facilitent leur mise en place et leurs donnent des possibilités et des fonctionnalités de plus en plus étendues.

Une start-up représente une des organisations ayant des ressources et des activités dont la gestion nécessite une application informatique. Ainsi, l'objectif de notre projet est de réaliser une application informatique interactive, fiable, conviviale et facile à intégrer dans l'environnement de travail, assurant la gestion des systèmes d'informations compte tenu des besoins exprimés. Cette application vise essentiellement à diminuer la complexité des traitements ainsi que le temps perdu lors de la gestion des projets, en particulier.

Dans le présent rapport, nous présentons en détail les étapes que nous avons suivies pour réaliser notre application. Ce rapport comporte quatre chapitres qui sont organisés comme suit :

Chapitre 1 :

Intitulé étude préalable qui consiste à souligner le contexte de projet, présenter l'organisme d'accueil ainsi que la thématique de projet dans laquelle on va citer la problématique, les solutions existantes et la solution proposée en passant par la description de l'architecture que nous avons adopté pour l'application en illustrant nos choix techniques. Ensuite nous finissons par le choix de la méthodologie du travail(SCRUM) et le planning du projet.

Chapitre 2 :

Présente une étape primordiale, les spécifications des besoins, la modélisation et l'étude conceptuelle. C'est à ce niveau que nous avons évoqué l'aspect conceptuel de notre application en commençant par spécifier les besoins et les acteurs de systèmes et en passant par la précision des éléments du "Product Backlog"

Chapitre 3 : Ce chapitre est consacré à la réalisation de release 1 ,qui est principalement la réalisation de l'application web .On va détailler l'analyse qui consiste à présenter le diagramme de cas d'utilisation, ,la conception qui consiste à présenter le diagramme de séquence, et nous passons par les tests correspondantes de chaque sprint.

Ainsi pour une meilleur explication de l'application web, nous présentons les diagrammes d'activité et de déploiement. **Chapitre 4 :**

Ce chapitre est consacré à la réalisation de release 2 ,qui est principalement la réalisation de la partie BI (Business Intelligence). On va détailler l'analyse ,la conception et nous passons par les tests correspondantes de sprint tout en s'aideant des interfaces graphiques.

Lors de l'étude décisive nous avons modélisé notre base de données afin d'être capable de l'exploiter et de la transformer en des rapports significatifs.

Chapitre 5 : Pour finir nous enchaînons avec le chapitre de clôture qui est consacré à la présentation de l'environnement matériel et logiciel utilisé pour la réalisation de notre application, en premier lieu. En second lieu, nous présentons les choix techniques adoptés avec bien sûr une description des choix ergonomiques adoptés.

Première partie

Aspects Théoriques

Chapitre 1

Cadre générale du projet

1.1 Introduction

Au cours de ce chapitre, nous allons nous intéresser par la présentation du contexte du travail ainsi que la présentation de Start up Cherchini pour laquelle ce travail a été réalisé. Puis on passe par la thématique du projet dans laquelle on va indiquer la problématique, citer des produits existants sur le marché, spécifier les besoins et indiquer la solution proposée. Ainsi qu'on va décrire la méthodologie avec laquelle on a pu concevoir et réaliser notre projet, et on finit par la présentation de l'architecture de projet tout en expliquant les choix techniques et les outils nécessaires.

1.2 Contexte du travail

Ce stage s'inscrit dans le cadre d'un projet de fin d'études pour l'obtention du diplôme Ingénieur en informatique De l'Ecole Supérieure Privée des Technologies de l'Information et de Management de Nabeul.

Notre stage a été effectué au sein de Start up Cherchini. Le sujet est intitulé «Création d'une plateforme de gestion des projets»

1.3 Présentation de l'organisme d'accueil

Cherchini.tn est une agence de communication, créée en 2017, qui vous aide à définir vos objectifs afin de vous fournir la solution web la plus adaptée à vos

besoins. Cherchini.tn cherche souvent d'être à la pointe de l'innovation par sa création graphique (logos, chartes graphiques, bannières publicitaires,..), création web, web design et référencement des sites web.

En outre, Cherchini.tn vous offre des solutions de vente en ligne, son objectif est de mettre en relation les clients et les vendeurs dans le but de réaliser de très bonnes affaires tout en bénéficiant de ses expertises.

Site Web : <https://cherchini.tn/>



FIGURE 1.1 – Logo de Cherchini.

1.4 Contexte du projet

Ce stage s'inscrit dans le cadre d'un projet de fin d'études pour l'obtention du diplôme Ingénieur en informatique De l'Ecole Supérieure Privée des Technologies de l'Information et de Management de Nabeul. Notre stage a été effectué au sein de Start up Cherchini. Le sujet est intitulé «Création d'une plateforme de gestion des projets»

1.5 Business intelligence

L'informatique décisionnelle, aussi appelée business intelligence(BI), désigne un ensemble de méthodes, de moyens et d'outils informatiques utilisés pour piloter une entreprise et aider à la prise de décision : tableaux de bord, rapports analytiques et prospectifs.

1.6 Etude de l'existant

Il existe sur le marché plusieurs plateformes de gestion des projets , et parmi les outils les plus connues on trouve ASANA et JIRA .

1.6.1 Asana

Il s'agit d'une plate-forme robuste qui sert à vos équipes de rester concentrées sur les objectifs, les projets et les tâches quotidiennes . Voici les fonctionnalités les plus importantes qu'elle offre :

- Tableau de bord permet de visualiser facilement votre travail.
- Voir la grande image. Cluez votre timing en visualisant les travaux sur un calendrier. Repérez facilement les trous et les chevauchements dans votre horaire et effectuez rapidement les ajustements nécessaires.
- Gardez une trace de ce qui est le plus important.
- Pas besoin de réinventer la roue. Transformez les processus courants en modèles que votre équipe entière peut utiliser pour que les projets se déroulent sans heurts à chaque fois.
- Partagez des informations avec les bonnes personnes. Rendez les équipes et les projets privés afin de créer un espace sécurisé pour les travaux sensibles.
- Partagez des informations avec les bonnes personnes. Rendez les équipes et les projets privés afin de créer un espace sécurisé pour les travaux sensibles.

1.6.2 Jira

Qui est l'un des meilleurs outils de gestion de produit qui comprend :

- Le suivi et la gestion des problèmes
- La gestion des produits

- Un Tableau de bord configurable
- Des Rapport sur l'avancement du projet
- Support des méthodologies Scrum & Kanban

1.7 Problématique

Toute entreprise aspire à se conformer au triangle d'or (Délai, Cout, Qualité) autour duquel tournent tous les projets professionnelles.

Dans le but d'organiser ses projets, Cherchini veut réaliser son propre outil de gestion des projets, en fait, l'absence d'un tel outil demeurent inaperçue pour les premières années d'une nouvelle entreprise et surtout pour les petites ou moyenne entreprises, mais évidemment, cela va engendrer au futur une mal organisation et une perte des informations importantes, et en conséquence une perte de temps et d'argent.

On remarque alors que les outils existants sont assez sophistiqués. Ainsi que ces deux outils, ils existent plusieurs concurrents qui offrent plusieurs manipulations et possibilités. Les fonctionnalités sont payantes pour chacun de ces outils et Cherchini n'a pas besoin de tous ces fonctionnalités d'une part, et d'autre part elle voudrait être capable d'étendre certaines fonctionnalités et de les réduire selon le besoin .Par exemple, ajouter une carte géographique qui indique la géolocalisation de ses partenaires dans cette plateforme . Pour cela, Cherchini voudrais avoir son propre outil de gestion de projet et désire réaliser un service d'informatique décisionnelle (Business Intelligence) ,ce service doit donner une vue générale sur le déroulement des projets selon les couts et les délais à suivre.

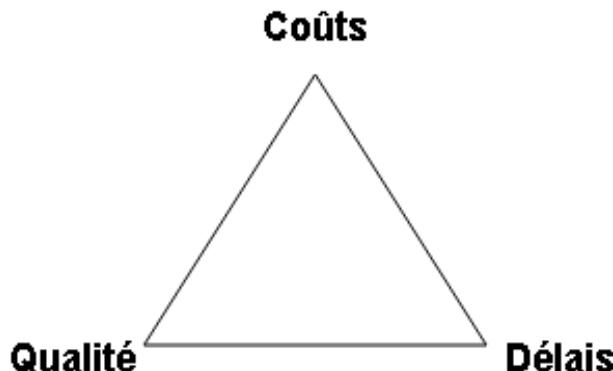


FIGURE 1.2 – Triangle d'or.

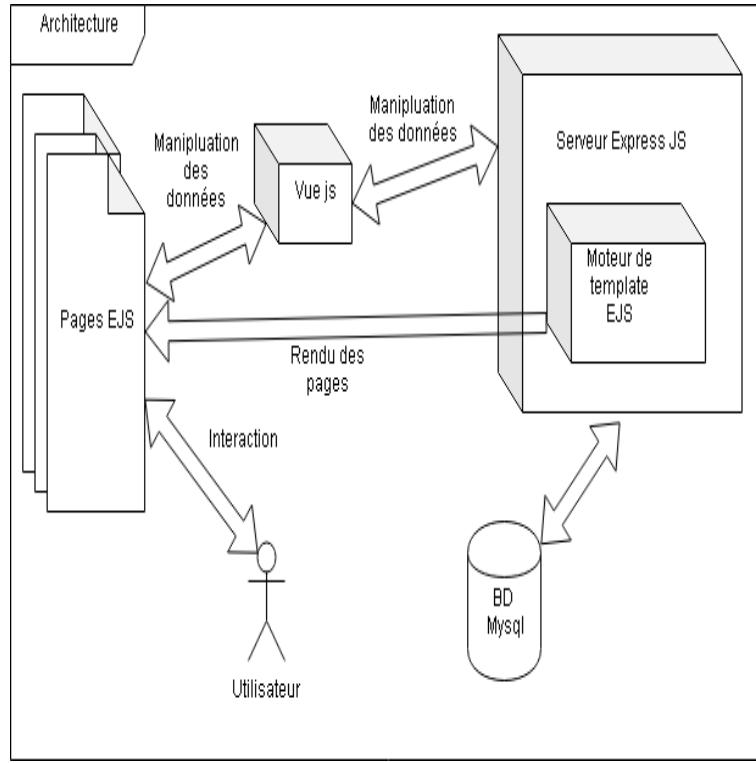


FIGURE 1.3 – Architecture générale.

1.8 Solution proposée

Pour répondre aux besoins fonctionnels, nous avons construit une application web en se basant sur l'architecture suivante.

Comme nous avons indiqué, pour être capable d'étendre les fonctionnalités selon le besoin en évitant la complexité des outils existantes et les couts de paiement pour chaque fonctionnalité, on a adopté des technologies open sources et gratuits, et voici les technologies qu'on a utilisé :

- Pour la base de données : MYSQL
- Pour le serveur et API REST : Express JS ,NODE JS
- Moteur de Template (engine view) : EJS
- Pour les pages web : HTML et Vue js
- Pour l'authentification : JWT
- Pour la carte de géolocalisation des clients : LeafLet maps
- Pour le diagramme de Gantt et les rapport : Highcharts

1.9 Etude méthodologique

1.9.1 Choix de la méthodologie

Dans la plupart des projets, on a besoin de suivre un processus qui définit QUI fait QUOI, QUAND et COMMENT pour être capable de :

- Atteindre un certain objectif.
- Construire des modèles d'un ou de plusieurs systèmes.
- Gérer le cycle de vie du projet de A à Z.
- Organiser le projet
- Gérer les risques
- Obtenir de manière répétitive des produits de qualité constante

Le choix entre une méthode et une autre, dépend de la nature du projet et de sa taille. Pour des projets de petite taille et dont le domaine est maîtrisé, par exemple, un cycle de vie en cascade s'avère largement su Sant. Lorsqu'il s'agit d'un projet où les données ne sont pas réunies dès le départ, où les besoins sont incomplets voire floues, il faut s'orienter vers une méthode itérative ou orientées prototypes. Parmi les méthodes itératives, nous pouvons distinguer les méthodes agiles largement utilisées de nos jours à travers le monde. Une méthode agile est menée dans un esprit collaboratif et s'adapte aux approches incrémentales. Elle engendre des produits de haute qualité tout en tenant compte de l'évolution des besoins du client. La nature de projet qui doit être évolutif et dont tous les besoins n'ont pas encore été totalement identifiés, nous a orientées vers une méthode de type agile et plus particulièrement SCRUM.

1.9.2 Présentation de la méthodologie SCRUM

Scrum signifie mêlée au rugby. Il exploite les valeurs et l'esprit du rugby et les adapte aux projets de développement. Comme le pack lors d'un ballon porté au rugby, l'équipe chargée du développement de travaille de façon collective, soudée vers un objectif précis. Comme un demi de mêlée, le Scrum-Master est le responsable du projet qui oriente les membres de l'équipe dans la bonne direction, assure l'environnement de travail et chercher à améliorer la productivité et le savoir faire de don équipe. Scrum se base sur la théorie du contrôle empirique de processus. L'empirisme mentionne que les connaissances proviennent de l'expérience et d'une prise de décision basée sur des faits connus. Scrum utilise une approche itérative et incrémentale pour opti-

miser la prédictibilité et contrôler le risque.

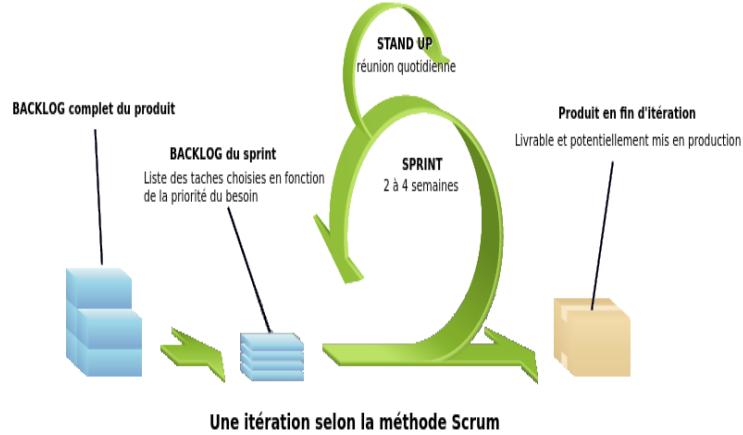


FIGURE 1.4 – Logo de Cherchini.

Comme nous pouvons le remarquer dans cette figure, pour mettre en place la méthode SCRUM, il faut tout d'abord définir les différentes fonctionnalités de notre application qui forment le Backlog du produit. Ensuite, nous procédons à la planification du sprint pour définir le plan détaillé d'une itération. Les sprints durent généralement deux à quatre semaines. Durant un sprint, il y a toujours des réunions quotidiennes entre les différents collaborateurs du projet pour présenter l'état d'avancement des différentes tâches en cours, les difficultés rencontrées ainsi que les tâches restantes à réaliser. Une fois le produit partiel est prêt, nous vérifions la conformité de ce qui a été fait durant le sprint et nous pouvons alors l'améliorer en procédant à l'étape de rétrospective.

1.9.3 Planning du projet

Le tableau suivant résume les étapes que nous avons suivies :

Backlog	Sprint	Durée
Etape Initiale	Recherche sur les nouvelles technologies qu'on va utiliser	3 Semaines
	Formation sur les langages	3 Semaines
	Rédaction du chapitre présentation de cadre générale	2 jours
	Création des diagrammes (Diagramme de classe, Diagramme de séquence, Diagramme de cas d'utilisation)	3 Semaines
	Rédaction du chapitre spécification des besoins	2 jours
Réunion		
Etape 2	Création des premières interfaces de l'application(Gestion des membres ,clients ,projet ,tâches)	1 Semaine
	Rédaction du chapitre conception	2 jours
	Création diagramme de Gantt	1 jour
	Maintenance et test de la fonctionnalité gestion de projet	2 jours
	Développement de l'interface Gannt et de la carte géographique	2 jours
Réunion		
Etape 3	Développement de L'interface membre	1 Semaine
	Développement de la partie Authentification	1 Semaine
Réunion		
Etape 4	Développement de la partie ETL	3 jours
	Ajouter les rapports à l'interface administrateur	1 jour
Réunion		
Etape finale	Dernières retouches sur le design de l'application	3 jours
	Mises à jour du rapport	3 jours

TABLE 1.1 – Planning du projet

1.9.4 Diagramme de Gantt

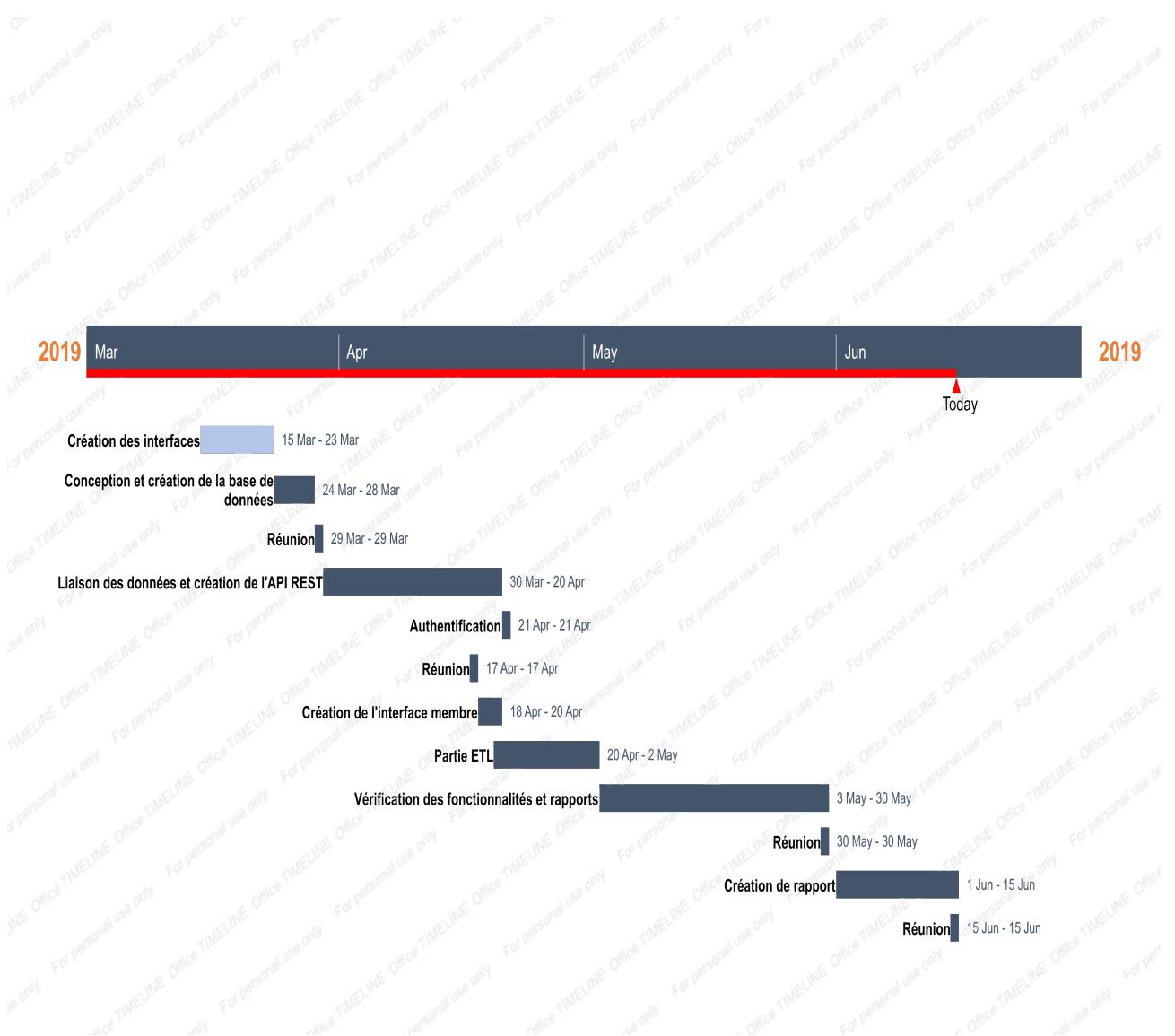


FIGURE 1.5 – Diagramme de Gantt.

Chapitre 2

Analyse et Spécifications besoins

2.1 Introduction

Cette partie consiste en une étape analytique dans laquelle nous allons recenser et factoriser les besoins des utilisateurs de l'application. Ceci est fortement lié à l'étude préalable menée au Cours du premier chapitre. Pour ce faire cette phase doit répondre aux questions suivantes : Quels sont les besoins fonctionnels de l'application ? Quelles sont les contraintes qui doivent être prises en considération ?

2.2 Les acteurs du système

C'est une entité externe qui agit sur le système (opérateur, autre système, ...). Il peut consulter Ou modifier l'état du système. En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin. Le principal acteur de système :

- **L'administrateur** : entité externe principale. Son rôle est qui a le droit de gérer un projet (créer, modifier, supprimer). Aussi son rôle et de gérer l'affectation des tâches aux membres correspondants.
- **Le membre** : entité externe secondaire, il peut se connecter pour consulter la tâche en cours que l'administrateur lui a effectué et la marquer comme terminée ou non.

Ce tableau présente les rôles selon les acteurs :

	Administrateur	Membre
Authentification	X	X
Création d'un projet	X	
Ajout d'une tâche	X	X
Modifier l'état d'une tâche	X	
Ajout d'un client/membre	X	
Consultation des rapports	X	
Consulter la carte géographique des clients	X	X

TABLE 2.1 – Rôles des acteurs.

2.3 Spécifications besoins

2.3.1 Besoins fonctionnels

Le besoin primordiale de notre application et de permettre à l'administrateur de « Cherchini » de gérer les projets et ceci consiste à :

Gérer les projets

- La création d'un projet.
- La modification d'un projet.
- La suppression d'un projet.
- Consulter les projets et les tâche et les détails correspondants.
- Affecter les membres correspondants à chaque projet.

Gérer les tâche

- La création d'une tâche.
- La modification d'une tâche.
- La suppression d'une tâche.
- Consulter les tâche et les détails correspondants.
- Affecter le membre correspondant à chaque tâche.

Gérer les membres

- La création d'un membre.
- La modification d'un membre.

- La suppression d'un membre.
- Consulter les membres et leurs détails correspondants.
- Affecter les membres correspondants à chaque projet.

Gérer les clients

- La création d'un .
- La modification d'un client.
- La suppression d'un client.
- Consulter les clients et leurs détails correspondants.

Suivre le déroulement des projets

- Suivre le travail des équipes en consultant le diagramme Gantt pour chaque projet.
- Consulter les rapports des coûts et les durées selon les projets et les clients .
- Consulter la carte géographique des géolocalisations des .

2.3.2 Besoins non fonctionnels

Les besoins non fonctionnels spécifient les propriétés du système afin de garantir la Cohérence, la confidentialité et l'intégrité des données. Le système doit être fiable : la validité de l'application. Réutilisabilité : aptitude de site à être utilisé en tout ou en partie dans de nouvelles applications.

La performance d'exécution

Le temps d'exécution du système doit être minimal pour ne pas gêner l'utilisateur. Ce temps dépend de la complexité du code implémenté, du serveur d'application utilisé, du débit de la ligne de connexion et de la conception de la base de données.

La sécurité

Le système doit respecter un niveau de sécurité élevé afin de garantir la confidentialité de l'accès des membres

L'ergonomie

L'interface de cette application doit être ergonome, conviviale et voire même apte à aider l'utilisateur à mieux gérer son espace de travail.

2.4 Diagramme de cas d'utilisation générale

Le diagramme de cas d'utilisation permet de décrire l'interaction entre l'acteur et le système. Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système. Un acteur et un cas d'utilisation sont mis en relation par une association représentée par une ligne. Le but principal du diagramme du cas d'utilisation est de définir le système du point de vue des utilisateurs et de définir les limites précises du système en utilisant une notation très simple

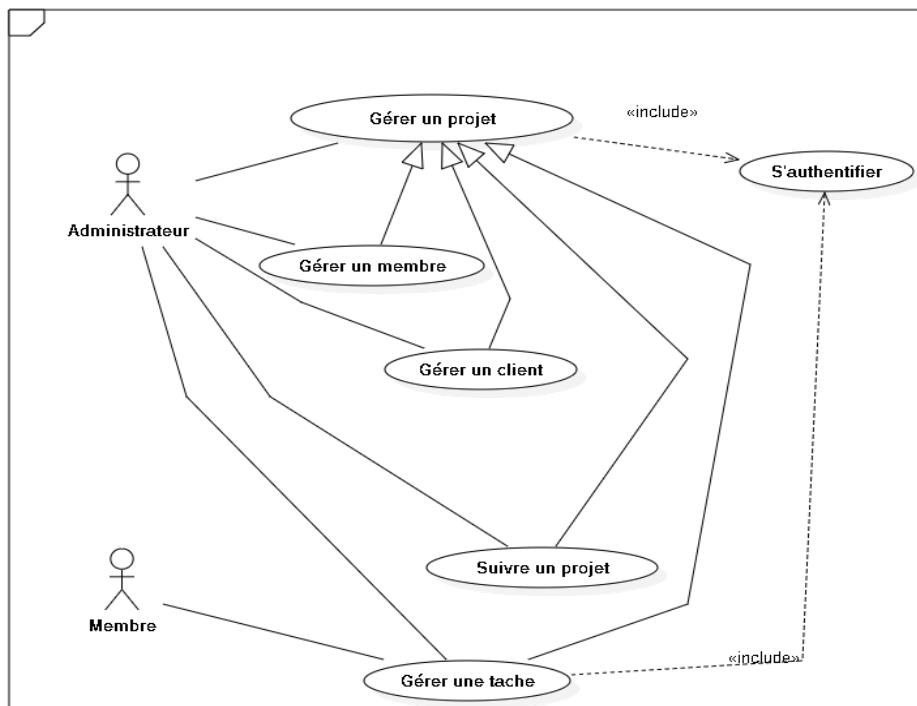


FIGURE 2.1 – Diagramme de cas d'utilisation générale.

2.5 Backlog de planning

Après avoir défini les acteurs des systèmes et les différentes interactions nous pouvons maintenant définir notre Product Backlog puis nous précisons la planification des sprints.

2.5.1 Les fonctionnalités du Backlog

Le Backlog est un artefact très important dans SCRUM. C'est l'ensemble des caractéristiques fonctionnelles ou techniques qui constituent le produit souhaité. Nous allons les décrire en détails dans le tableau qui suit :

Fonctionnalité	Acteur	Description
Gérer un projet	Administrateur	L'administrateur peut gérer un projet et ses tâches correspondantes d'affectation
Mettre à jour un Projet	Administrateur	L'administrateur peut changer les détails du projet ainsi que l'affectation des membres au projet
Créer ,Modifier , Supprimer un membre	Administrateur	L'administrateur peut manipuler les données des membres
Créer ,Modifier , Supprimer un client	Administrateur	L'administrateur peut manipuler les données des clients
Consulter les rapports	Administrateur	L'administrateur peut accéder aux rapports
Consulter les coordonnées des clients sur la carte géographique	Administrateur	L'administrateur peut accéder aux coordonnées géographiques des clients
Changer l'état et la progression approximative de ses tâches	Membre	Le membre peut changer ses tâches courantes selon l'avancement.

TABLE 2.2 – Product Backlog

2.5.2 Planification des sprints

Release 1 (L'application de gestion de projets)	Gestion des membres	14 jours
	Gestion des clients	5 jours
	Gestion des projets	20 jours
	Authentification	2 jours
	Interfaces membre	3 jours
Release 2 (La partie informatique décisionnelle)	Affichage des rapports	15 jours

TABLE 2.3 – Planification des sprints

Deuxième partie

Aspects Pratiques

Chapitre 3

Release 1

3.1 Introduction

Le terme release désigne ici une version de notre application constituée d'une suite d'itérations qui se terminent quand les incréments de ces derniers construisent un produit présentant suffisamment de valeur à l'utilisateur.

Dans ce chapitre, nous nous intéressons principalement par l'application web ,et nous allons nous intéresser à la première release dans laquelle nous traitons en détails chacun des cas d'utilisation préalablement présentés. Ainsi ,selon le sprint ,on précisons l'analyse qui consiste à illustrer le diagramme de cas d'utilisation ,la conception pour présenter le diagramme de séquence et le schéma de base de données et nous finissons par le test en présentant les interfaces correspondantes.

Pour chaque sprint on va présenter le cas d'utilisation,la conception qui consistre à présenter le diagramme de séquence et nous finissons par l'illustration du schéma de la base de données des modules correspondants.

Ainsi ,contrairement aux cycles de développement séquentiel, avec la méthodologie agile, le test n'est pas une phase qui se déroule après la fin de développement. En effet, les tests seront intégrés dès le début du premier sprint jusqu'à la livraison du produit final.

Enfin ,nous illustrons la méthode de liason de données que nous avons suivi ainsi nous présentons le diagramme de classes ,le diagramme d'activité et le diagramme de déploiement pour mieux expliquer notre application.

3.2 Sprint 1

3.2.1 Analyse

Diagramme de cas d'utilisation "Gérer un membre"

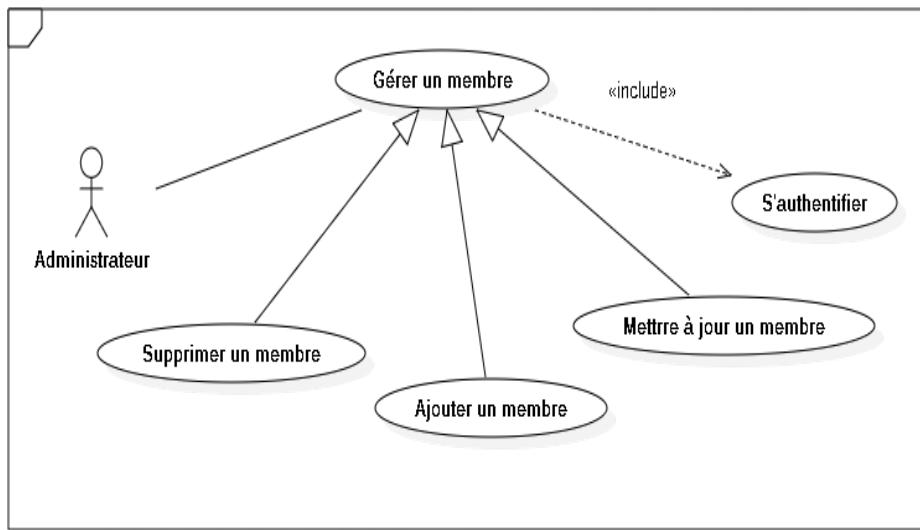


FIGURE 3.1 – Gérer un membre.

3.2.2 Conception

Le diagramme de séquence indique l'interaction entre plusieurs acteurs afin d'expliquer le déroulement des différents scénario entre les différents éléments du projet.

Le scénario « Crédation d'un membre »

Le diagramme de séquence « Ajout d'une tâche » présente le séquencement des interactions entre Administrateur, Application et Base de données (BD).

3.2.3 Schéma

La table < members >

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
login	varchar(30)	YES		NULL	
password	varchar(15)	YES		NULL	
firstname	varchar(50)	YES		NULL	
lastname	varchar(30)	YES		NULL	
email	varchar(100)	YES		NULL	
hiredate	datetime	YES		CURRENT_TIMESTAMP	
color	varchar(6)	YES		NULL	
projects.id	int(11)	YES	MUL	NULL	
pname	varchar(50)	YES		NULL	

TABLE 3.1 – Members.

3.2.4 Test

Dans ce premier sprint, nous testons la fonctionnalité « Gestion des membres» qui affiche, ajoute, supprime les membres de la base de données. Ci-dessus nous ajoutons des captures écran du sprint réalisé.

Lors de la création d'un utilisateur , un email contenant son mot de passe lui sera envoyé, ainsi si on modifie les paramètres d'utilisateur , un nouveau mot de passe lui sera envoyé sur la nouvelle adresse email .

The screenshot shows a dark-themed user interface for managing members. At the top, there is a navigation bar with tabs: Home, Projects, Members, Live-Map, and Customers. Below the navigation bar is a green button labeled "Add new member +". A search bar with the placeholder "Search..." is positioned below the button. The main area displays three member profiles in cards:

- AM**: Profile picture, first name Alexaa, last name McCartney, email field empty.
- GH**: Profile picture, first name Garyy, last name Harvard, email field empty.
- JH**: Profile picture, first name Johnny, last name Harveu, email field empty.

Each profile card has a red "X" icon in the top right corner.

FIGURE 3.2 – Gestion des membres.1.

The screenshot shows the same dark-themed interface as Figure 3.2, but with a modal window open in the center. The modal is titled "New member" and contains three input fields: "First name", "Last name", and "Email". Below the input fields are three buttons: "Close", "Clear", and "Confirm". The "Confirm" button is highlighted with a green border. In the background, the three existing member profiles (AM, GH, JH) are visible, dimmed to indicate they are not the active focus.

FIGURE 3.3 – Gestion des membres.2.

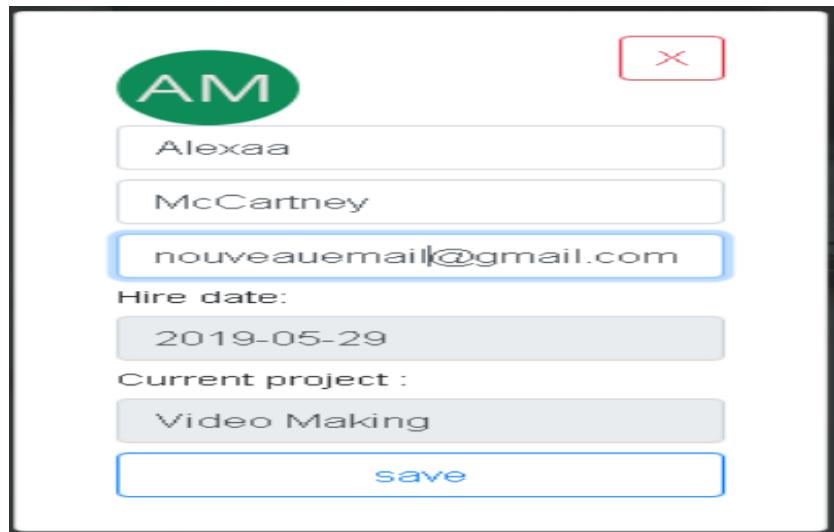


FIGURE 3.4 – Gestion des membres.3.

3.3 Sprint 2

3.3.1 Analyse

Diagramme de cas d'utilisation "Gérer un client"

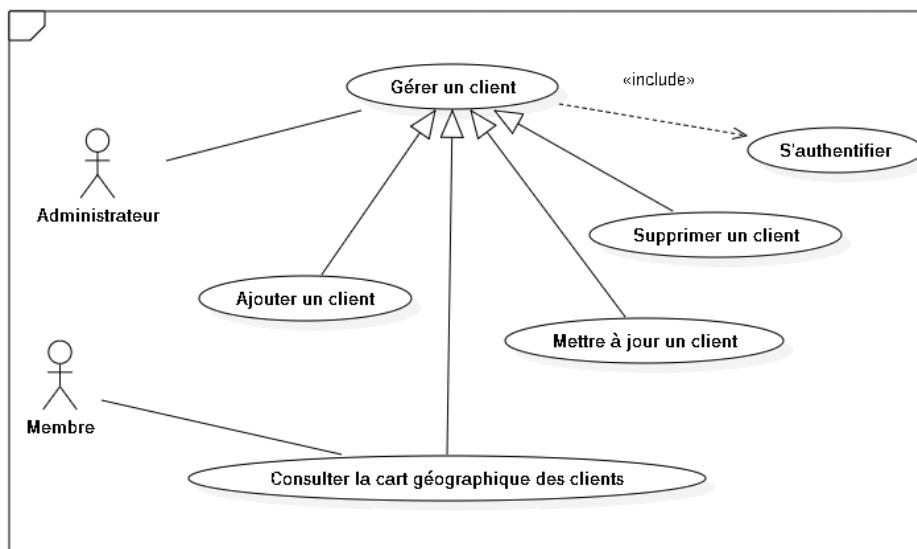


FIGURE 3.5 – Gérer un client.

3.3.2 Conception

Le scénario « Crédit d'un client »

Le diagramme de séquence « Ajout d'une tâche » présente le séquencement des interactions entre Administrateur, Application et Base de données (BD).

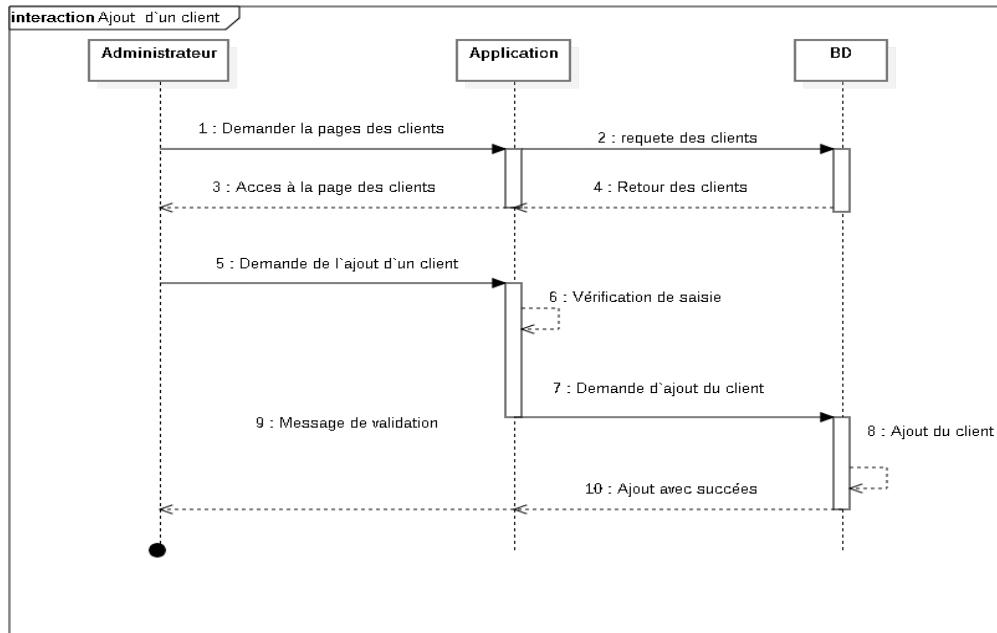


FIGURE 3.6 – Création d'une client.

Le scénario « Consultation de la carte géographique des clients»

Le diagramme de séquence « Ajout d'une tâche » présente le séquencement des interactions entre Administrateur, Application et Base de données (BD).

Schéma

La table < customers >

3.3.3 Test

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	varchar(45)	YES		NULL	
phone	varchar(45)	YES		NULL	
address	varchar(1000)	YES		NULL	
email	varchar(50)	YES		NULL	
longitude	double	YES		NULL	
latitude	double	YES		NULL	
subscriptionDate	datetime	YES		CURRENT_TIMESTAMP	

TABLE 3.2 – Customers

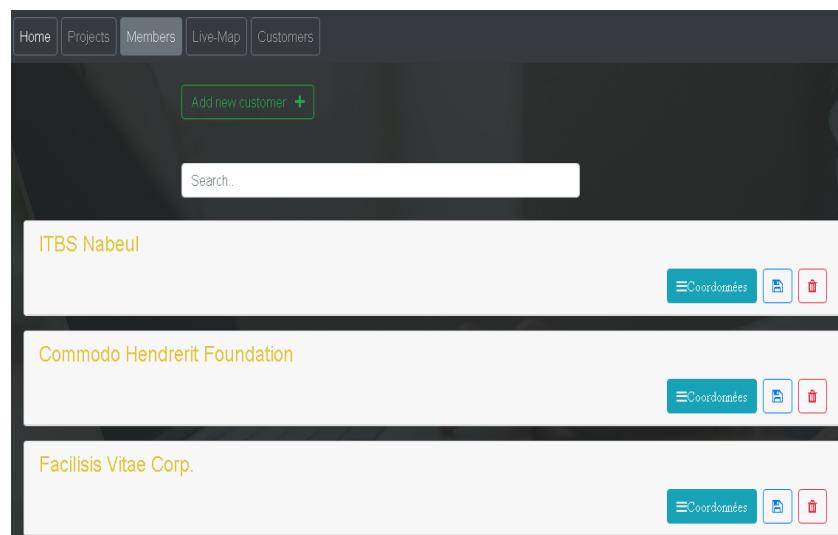


FIGURE 3.7 – Gestion des clients.1.

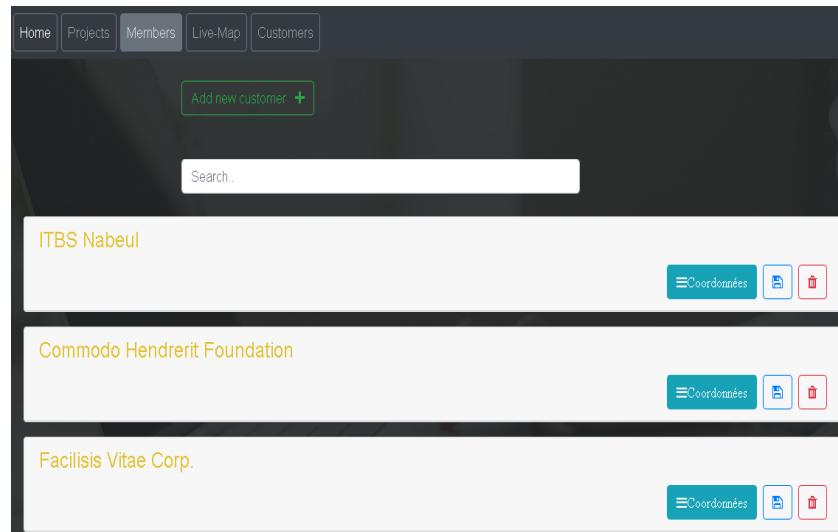


FIGURE 3.8 – Gestion des clients.2.

3.3.4 Consultation carte géographique des clients

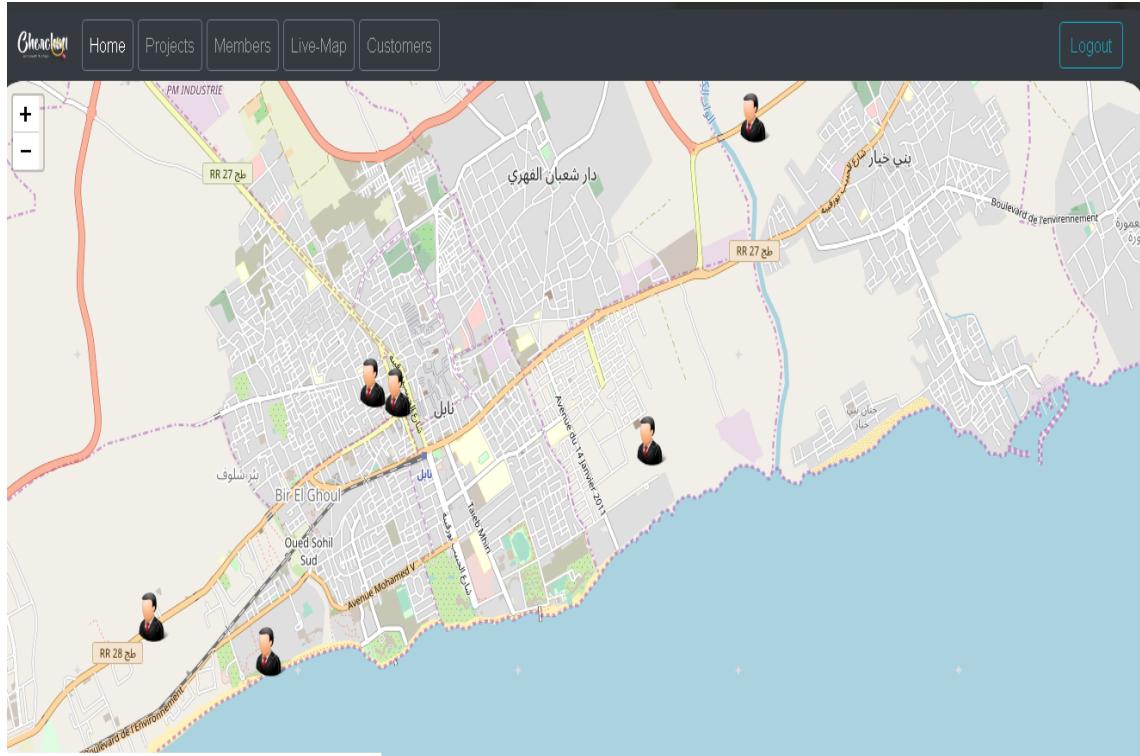


FIGURE 3.9 – Carte géographique.

3.4 Sprint 3

Lors de ce sprint , nous allons raffiner le cas d'utilisation "gestion du projet" . Détailler le cas d'utilisation < "Gestion du projet " > revient à détailler ses sous cas d'utilisation à savoir : • Gestion des tâches. • Gestion des projets

Analyse

Diagramme de cas d'utilisation "Gérer un projet"

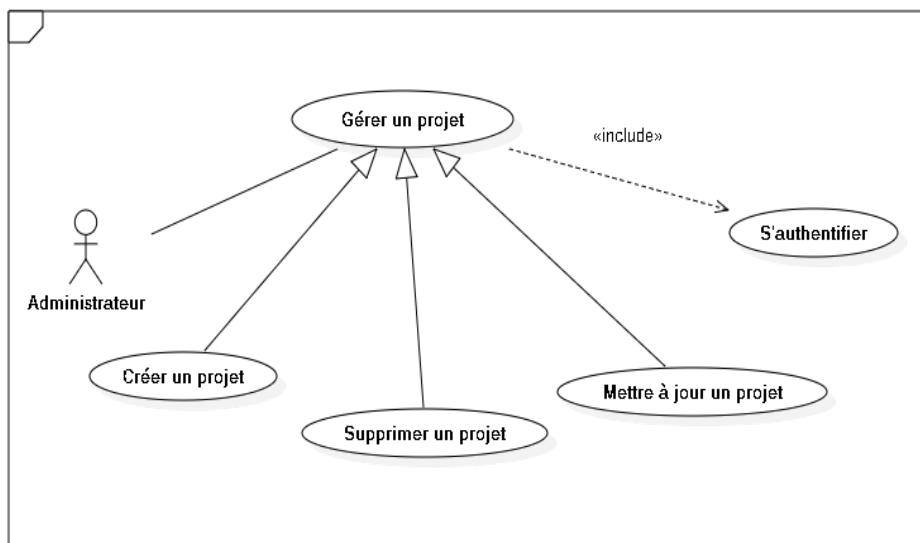


FIGURE 3.10 – Gérer un projet.

Diagramme de cas d'utilisation "Gérer une tâche"

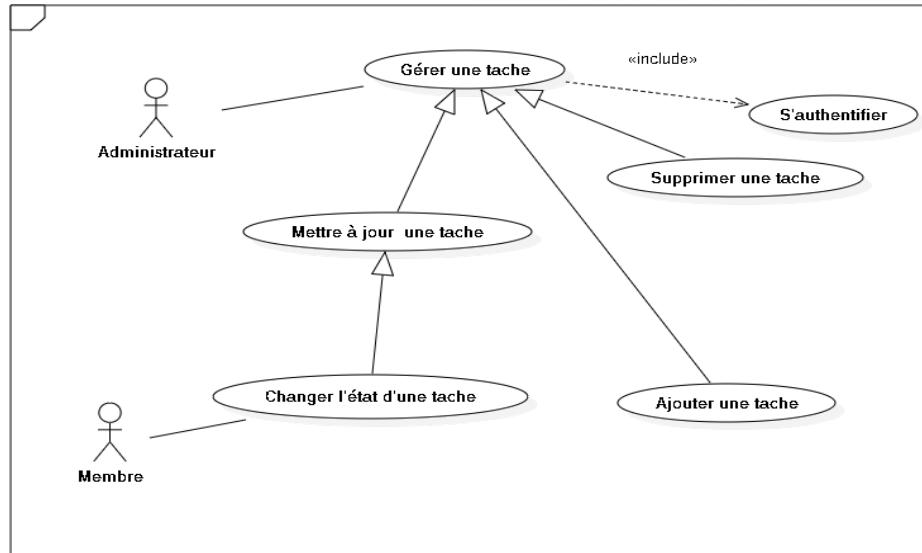


FIGURE 3.11 – Gérer une tâche.

Conception

Le scénario « Crédation d'un projet »

Le diagramme de séquence « Ajout d'un projet » présente un séquencement des interactions entre Administrateur, Application et Base de données (BD)

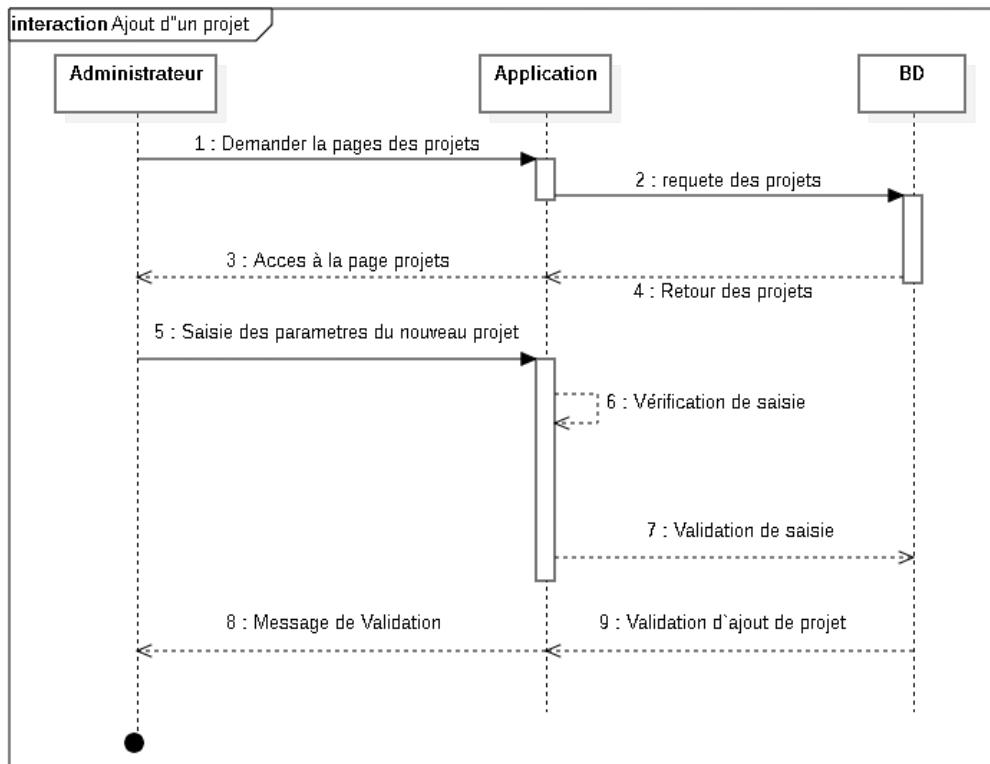


FIGURE 3.12 – Création d'un projet.

Le scénario « Crédation d'une tâche»

Le diagramme de séquence « Ajout d'une tâche » présente le séquencement des interactions entre Administrateur, Application et Base de données (BD).

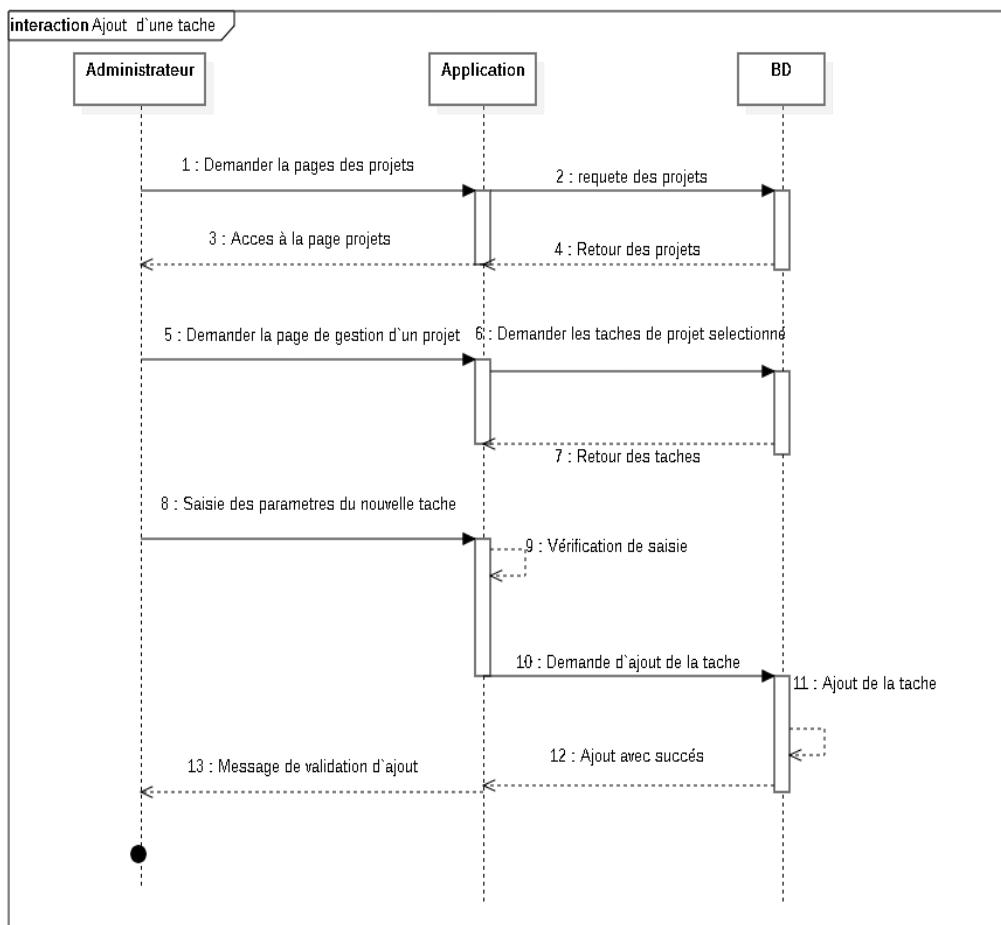


FIGURE 3.13 – Crédation d'une tâche.

Schémas

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	varchar(45)	NO		NULL	
description	varchar(1000)	YES		NULL	
start	date	NO		NULL	
end	date	NO		NULL	
category	varchar(60)	YES		NULL	
customers_id	int(11)	YES	MUL	NULL	

TABLE 3.3 – Projects.

Test

C'est l'interface principale de notre application

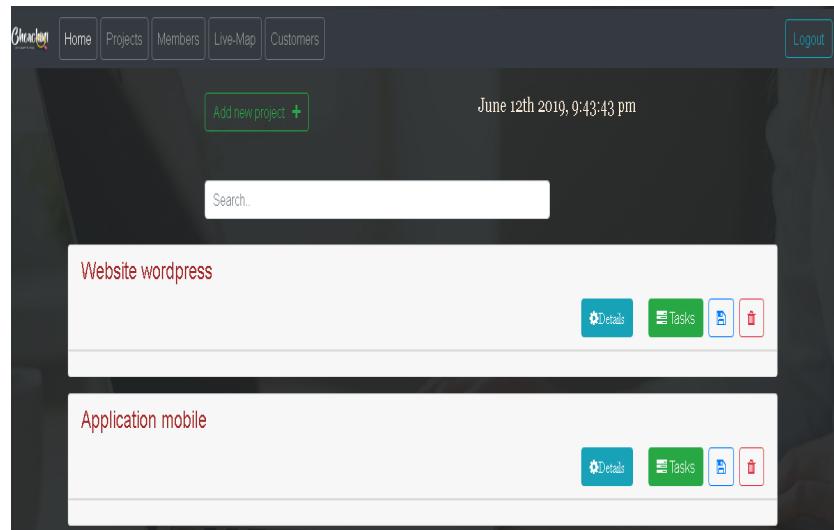


FIGURE 3.14 – Gestion de projet.1.

Nous ajoutons un nouveau projet et l'assignons un ou plusieurs membres.

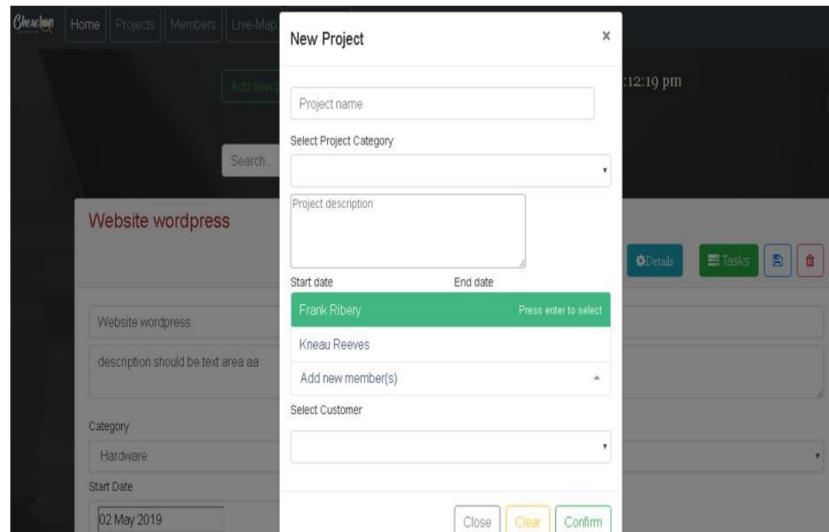


FIGURE 3.15 – Gestion de projet.2.

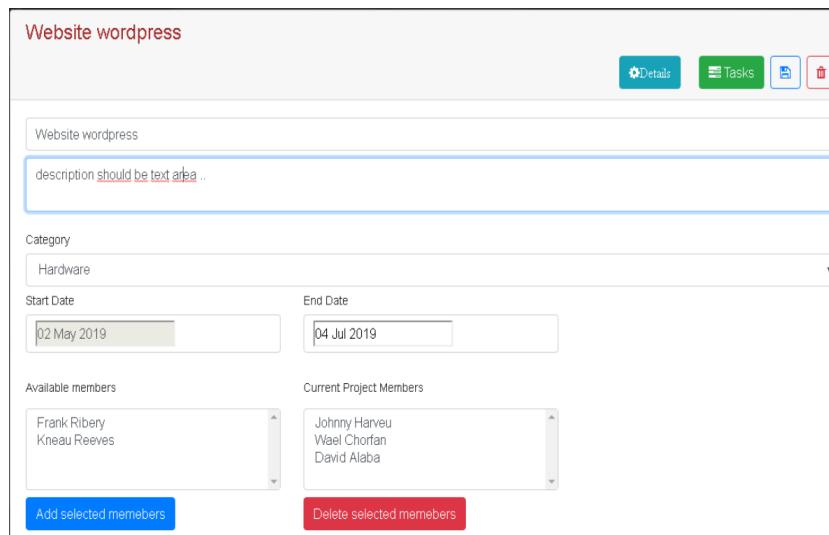


FIGURE 3.16 – Gestion de projet.3.

Pour chaque projet en cliquant sur le buttons Details ,on peut modifier ses détails ,on peut séléctionner plusieurs membres et les ajouter . On ne peut pas modifier les dates de début et de fin puisque les dates des taches devraient etre pénibles à changer tache par tache .

En cliquant sur le button < Tasks > d'un certain projet , l'administrateur est amenée à l'interface de gestion des taches correspondantes dans laquelle il peut suivre la progression des taches ,consulter le diagramme de gannt dynamique et modifier les paramètres des taches .

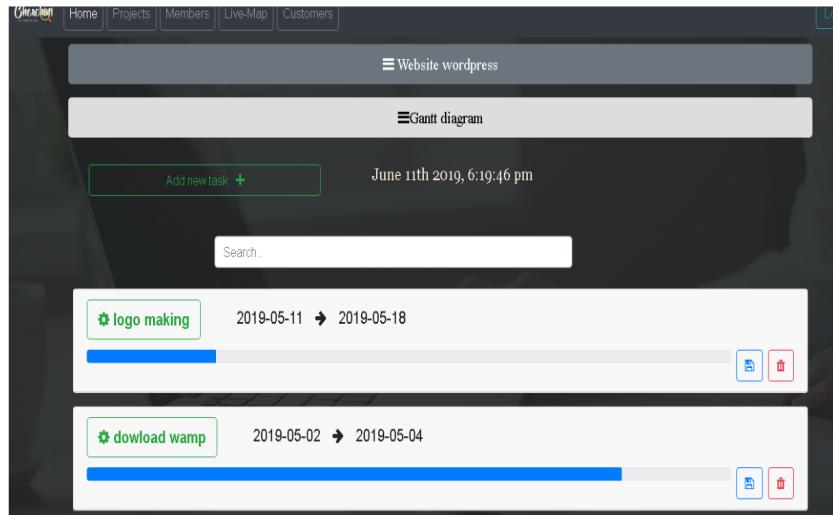


FIGURE 3.17 – Gestion de projet.4.

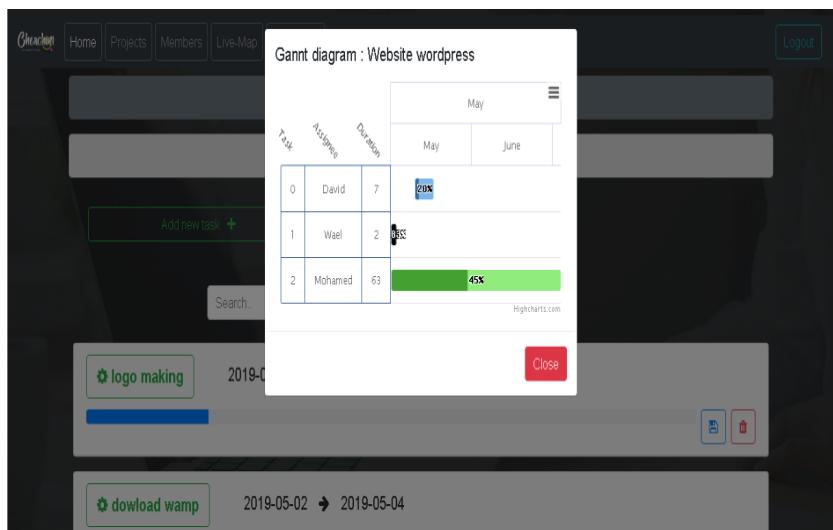


FIGURE 3.18 – Gestion de projet.5.

The screenshot shows a project management interface for a task named "logo making". The task details are as follows:

- Name:** logo making
- Description:** add logo to webpage
- Start Date:** 2019-05-11
- End Date:** 2019-05-18
- Assignee:** David Alaba (selected in dropdown)
- Status:** To Do
- Price:** TND 75
- Start Date:** 11 May 2019
- End Date:** 18 May 2019

FIGURE 3.19 – Gestion de projet.6.

3.5 Sprint 4

3.5.1 Conception

Le scénario « Authentification » Ce schéma présente le même scénario pour l'administrateur et un simple membre :

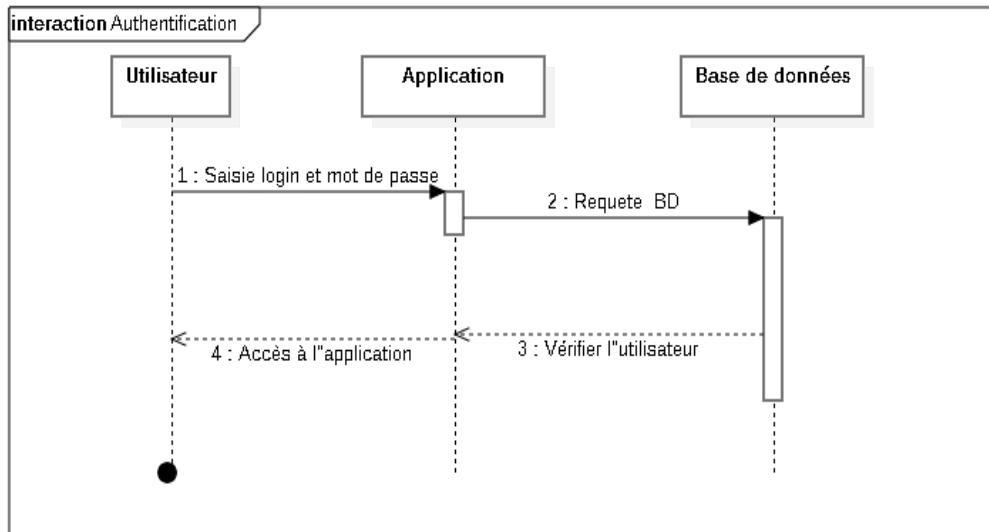


FIGURE 3.20 – Authentification.

3.5.2 Test

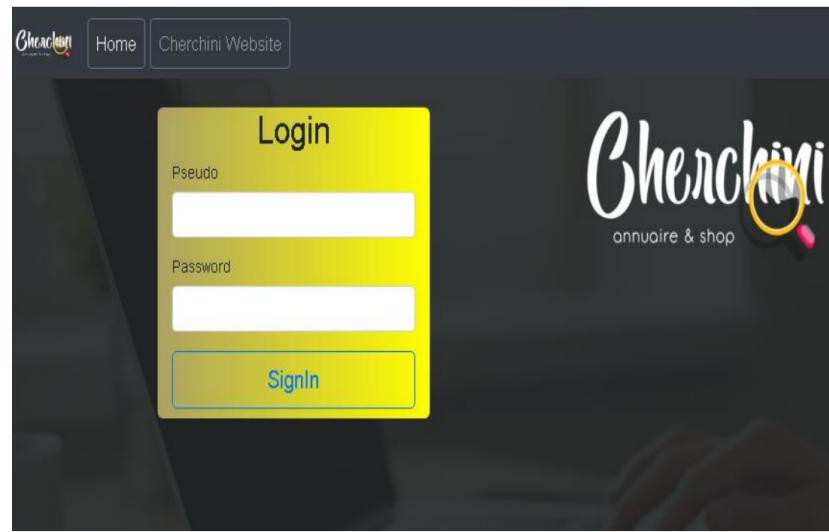


FIGURE 3.21 – Accueil.

3.6 Structure de l'application

Le schéma suivant présente la structure de notre application

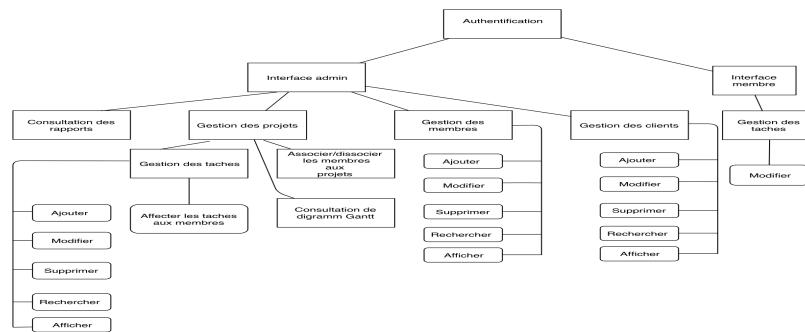


FIGURE 3.22 – Structure de l'application web.

3.7 Conception de la base de données

3.7.1 Modèle du liason des données

L'architecture de notre application nous implique à créer un modèle physique des données , et nous avons pas été besoin d'un model conceptuel logique ,puisque on a liée directement les données à l'application par le biais d'un pilote de connexion et pas par un ORM , et ceci est le diagramme Entités –Relations de la base de données qui est en interaction avec l'application web . Choix de la méthodologie de conception : La liaison par des ORM ,ou posséde des avantages bienque des inconvénient , parmi ses avantages :

- **La portabilité :** ORM est utilisé pour que vous écriviez votre structure une seule fois et la couche ORM générera l'instruction finale adaptée au SGBD configuré. C'est un excellent avantage, car une opération simple, telle que limit, est ajoutée sous la forme "limit 0,100" à la fin

de l'instruction select dans MySQL, alors qu'elle est "select top 100 from table" dans MS SQL.

- **Imbrication de données** : en cas de relations, la couche ORM extraira automatiquement les données pour vous.
- **Langage unique** : vous ne connaissez pas le langage SQL pour traiter la base de données uniquement avec votre langage de développement. Ajouter revient à modifier : la plupart des couches ORM traitent l'ajout de nouvelles données (insertion SQL) et la mise à jour des données (SQL Update) de la même manière, ce qui facilite grandement l'écriture et la maintenance du code.
- **Imbrication de données** : en cas de relations, la couche ORM extraira automatiquement les données pour vous.

Et parmi les inconvénients de l'ORM on trouve :

- **La complexité des requêtes** : certaines couches ORM ont des limitations, en particulier lors de l'exécution de requêtes. Vous serez donc parfois obligé d'écrire en SQL brut.
- **Lenteur** : si vous comparez les performances entre l'écriture de SQL brut ou l'utilisation d'ORM, vous trouverez le brut beaucoup plus rapidement car il n'y a pas de couche de traduction.
- **Réglage** : si vous connaissez bien le langage SQL et votre SGBD par défaut, vous pouvez utiliser vos connaissances pour accélérer les requêtes, mais ce n'est pas la même chose avec ORM.
- **Configuration** : si vous travaillez dans un projet Big Data et que vous n'êtes pas satisfait de la performance, vous vous retrouverez en train d'étudier la couche ORM afin de pouvoir minimiser les occurrences du SGBD.

3.7.2 Diagramme de classes

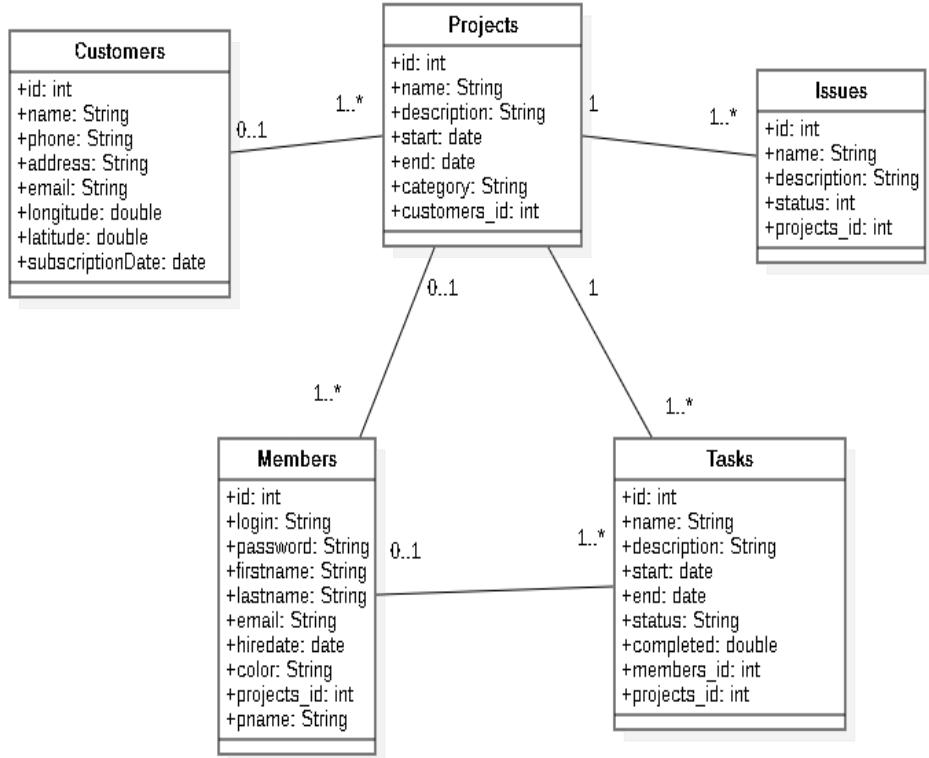


FIGURE 3.23 – Diagramme de classes.

Tableau explicatif

Afin de clarifier ce schéma en explicitant les différents relations pour les entités ,nous avons choisi de les mettre dans un tableau descriptif .

Nom relation	Entité E1	Entité E2	Relation(E1 :E2)	Relation(E2 :E1)
Projects -Tasks	Projects	Tasks	1 : N	1 : 1
			Non-identifiée	identifiée
Projects -Members	Projects	Members	1 : N	1 : 1
			Non-identifiée	Non-identifiée
Members-Tasks	Members	Tasks	1 : N	1 : 1
			Non-identifiée	Non-identifiée
Projects-Customers	Projects	Customers	1 : 1	1 : N
			Non-identifiée	Non-identifiée

TABLE 3.4 – Relations

Les relations peuvent être expliquées par les règles suivantes :

- Une relation non identifiée 1 :N entre une entité E1 et une autre E2 :
Pour E1 il existe ou il n'existe pas une ou plusieurs entités E2
- Une relation non identifiée 1 :1 entre une entité E1 et une autre E2 :
Pour E1 il existe ou il n'existe pas une entité E2
- Une relation identifiée 1 :N entre une entité E1 et une autre E2 : Pour E1 il existe une ou plusieurs entités E2
- Une relation identifiée 1 :1 entre une entité E1 et une autre E2 : Pour E1 il existe une entité E2

3.8 Diagramme de déploiement

3.8.1 Diagramme de déploiement

- **Le Client :** C'est le navigateur web, il permet aux utilisateurs d'accéder au serveur, c'est d'interface à l'utilisateur.
- **Le serveur web :** C'est le serveur principal qui abrite les différents composants logiciels de notre application. Il assure la gestion des connexions et des requêtes du client ainsi que aussi la distribution et rendu (rendering) des pages EJS. Cet élément contient principalement un environnement d'exécution qui est le framework javascript Node js sur lequel est déployé l'application Web(Express JS).
- **La base de données :** Est exploité par le avec le serveur wamp. C'est le composant qui s'occupe du stockage et de la gestion des données. La communication des données entre l'application est la base de données est assurée par le pilote (driver) de mysql pour Node js.

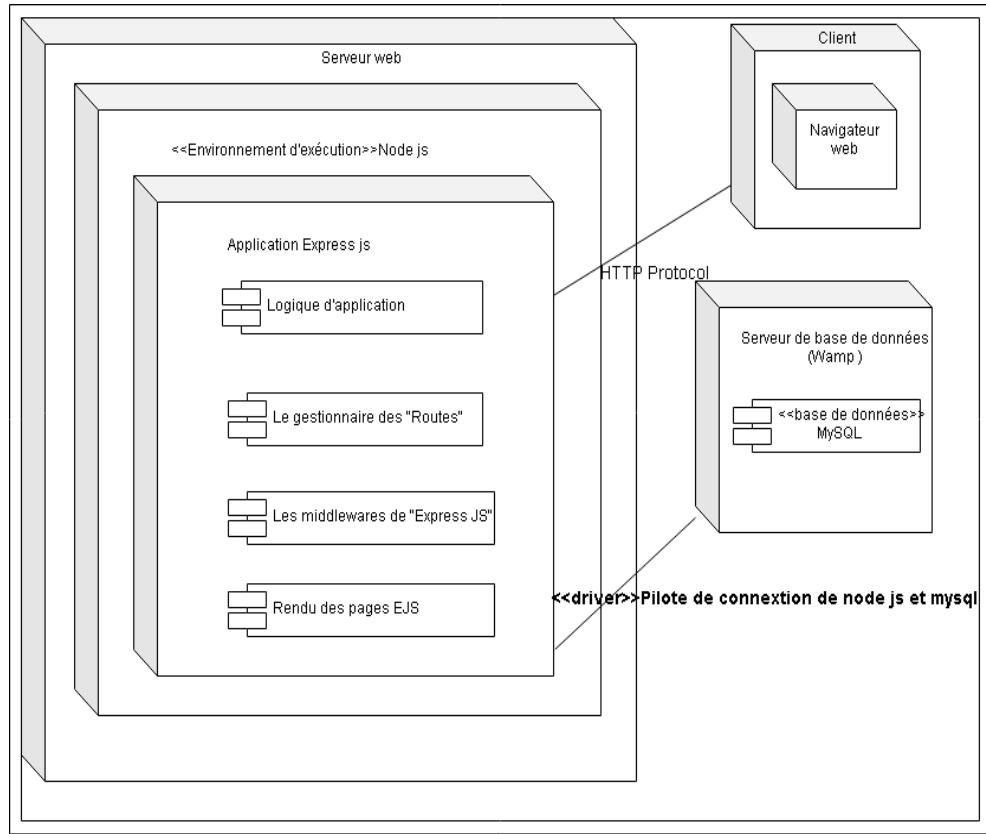


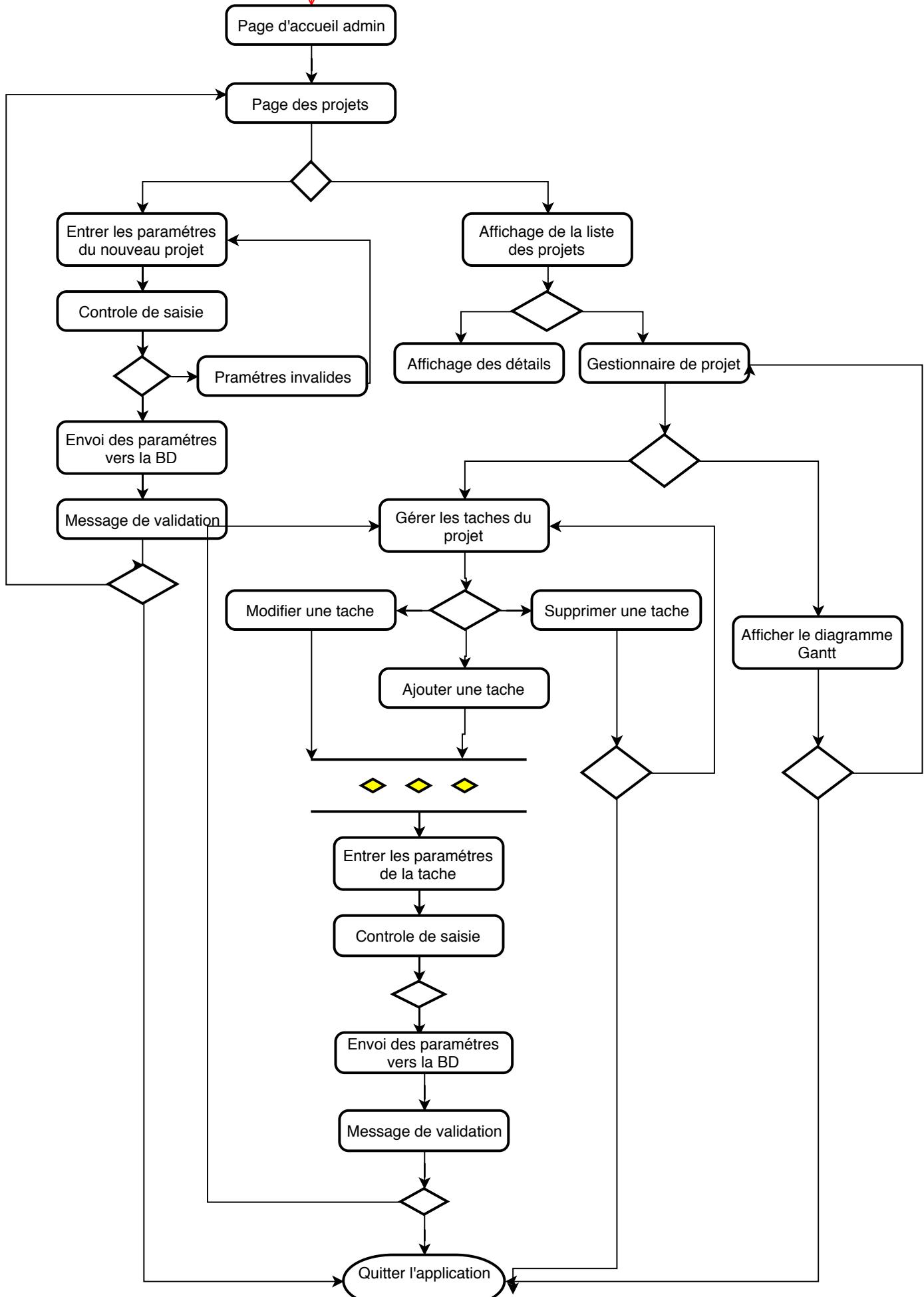
FIGURE 3.24 – Diagramme de déploiement.

3.9 Diagramme d'activité

Le diagramme d'activité nous permet de décrire les traitements. Il est particulièrement adapté à la modélisation du cheminement de flots de contrôle et de flots de données. Il permet ainsi de représenter le comportement d'une méthode ou le déroulement d'un cas d'utilisation dans un graphe. Les diagrammes d'activités sont très proches des diagrammes d'états-transitions dans leur présentation, mais leur interprétation est différente. Une activité représente une exécution d'un mécanisme, un déroulement d'étapes séquentielles. Le passage d'une activité vers une autre est matérialisé par une transition. Les transitions sont déclenchées par la fin d'une activité et provoquent le début immédiat d'une autre. Dans la page suivante , nous présentons notre

diagramme d'activité qui illustre le déroulement séquentiel des traitements accomplis par l'administrateur afin de créer et gérer un projet.

Diagramme d'activité pour l'administrateur



Ainsi , la figure suivante représente le diagramme d'activité pour un membre :

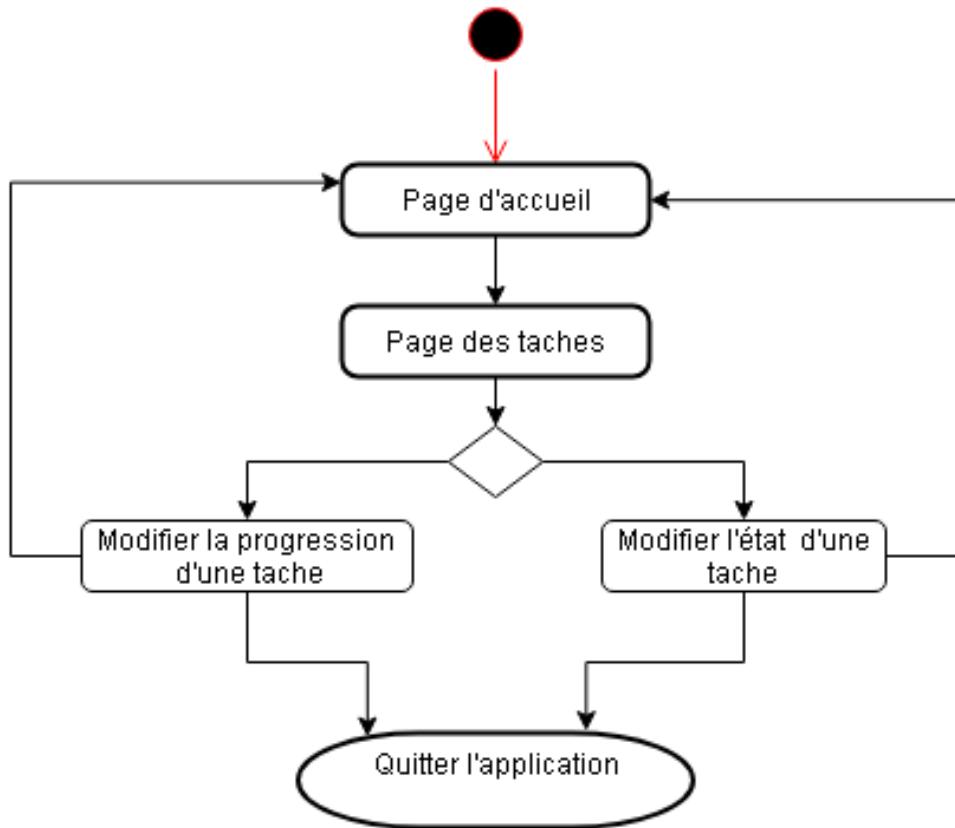


FIGURE 3.25 – Diagramme d'activité pour un membre.

Chapitre 4

Release 2

4.1 Introduction

Dans ce chapitre, il s'agit d'un release qui consiste en un seul sprint qui est la consultation des rapports , afin de pouvoir représenter ces rapports ,nous avons besoins d'abord de determiner les axes de notre étude décisionnelle , puis nous allons élaborer et tester les rapports en utilisant l'outil Power BI ensuite nous illustrons la partie ETL(Extract Trasform Load) à l'aide du l'outil Talend, et nous finissons par la présentation des rapports obtenues.

4.2 Analyse et conception

4.2.1 Analyse

Diagramme de cas d'utilisation "Consultation des rapports"

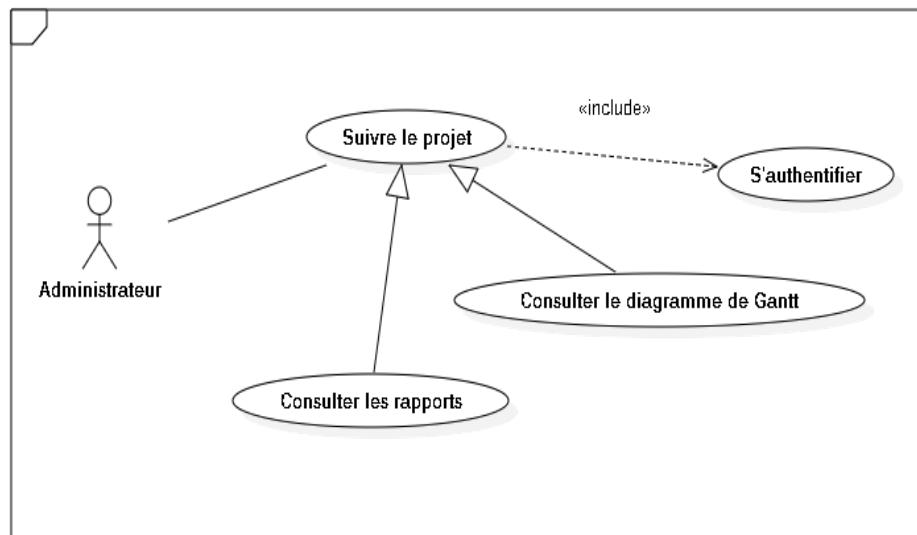


FIGURE 4.1 – Gérer un projet.

4.2.2 Conception

Le scénario « Consultation des rapports »

Le diagramme de séquence « Ajout d'une tâche » présente le séquencement des interactions entre Administrateur, Application et Base de données (BD).

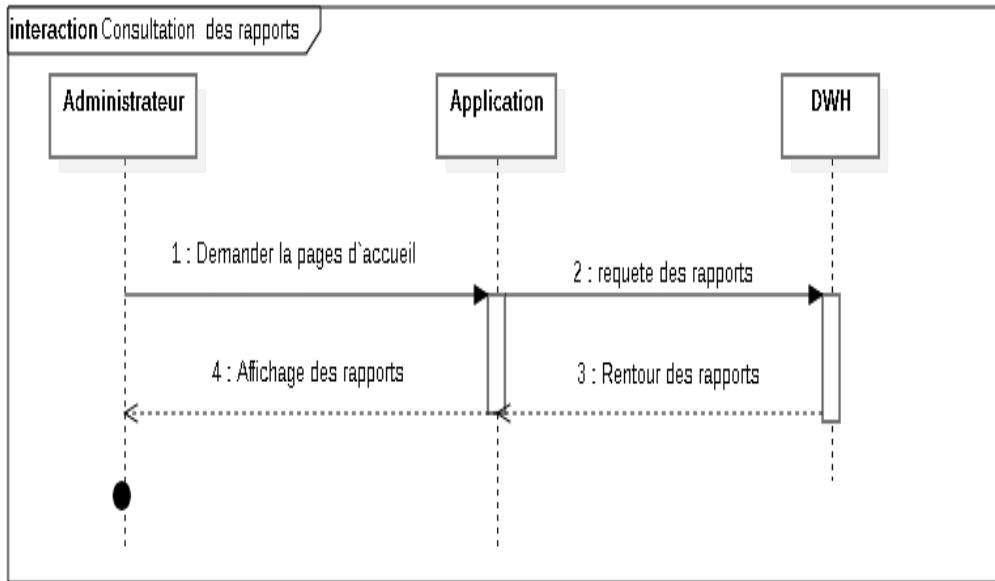


FIGURE 4.2 – Consultation des rapports.

4.3 But de l'étude décisionnelle

Notre étude décisionnelle portera sur :

- Le coût total de chaque projet
- Le coût total des projets pour chaque client
- Le coût total par catégorie des projets
- La durée totale du chaque projet

4.4 Elaboration des rapports sur Power BI

Préparation

Nous avons besoin d'ajuster quelques champs pour être adéquates à être exploitées . Par exemple dans la table membre on trouve "firstname " et "lastname" Et on doit les concaténer pour dans une hiérarchie pour avoir des rapports lisibles à propos des membres .

name = CONCATENATE('pfe_r1 members'[firstname] & " ", 'pfe_r1 members'[lastname])								
id	login	password	firstname	lastname	email	hiredate	color	projects_id
0						6/6/2019 11:19:49 PM		92 Website wordpress
25 AM	liigdhv	Alexaa	McCartney			5/29/2019 03:51:54 PM	0D8C58	94 Video Making
26 GH	zefnocy	Garry	Harvard			5/29/2019 03:51:54 PM	5D8A58	94 Video Making
27 JH	xovzezf	Johnny	Harveu			5/29/2019 03:51:54 PM	0D1058	92 Website wordpress
41 MS	lywizgm	Mohamed	Salah			6/6/2019 02:47:03 AM	Qbc7c	93 Application mobile
53 WC	0001	Wael	Chorfan	chorfanwael@gmail.com		6/6/2019 10:51:08 PM	c310d	92 Website wordpress
54 admin	0000	Admin	Admin			1/1/2017 10:51:08 PM	c310d	Admin Admin
55 DA	xscgonqjm	David	Alaba			6/10/2019 12:41:49 PM	97572e	92 Website wordpress
56 FR	qnpe	Frank	Ribery			6/10/2019 12:41:58 PM	33e4db	Frank Ribery
57 KR	uptchijl	Kneau	Reeves			6/10/2019 12:42:15 PM	9236d	Kneau Reeves

FIGURE 4.3 – Ajustement des noms.

durationInDays = DATEDIFF(['pfe_r1 projects'[start],'pfe_r1 projects'[end],DAY)							
id	name	description	start	end	category	customers_id	durationInDays
92 Website wordpress	description should be text area aa		Thursday, May 02, 2019	Thursday, July 04, 2019	Hardware	1	63
93 Application mobile	création d'une application mobile		Sunday, June 09, 2019	Wednesday, July 10, 2019	Software	5	31
94 Video Making	random desc		Saturday, August 17, 2019	Wednesday, November 06, 2019	Software	1	81
95 Create geolocalisation system	desc random		Wednesday, August 14, 2019	Tuesday, September 10, 2019	Hardware	6	27
96 Make reservation Website	augue id ante dictum cursus.		Tuesday, September 24, 2019	Tuesday, December 17, 2019	Software	8	84

FIGURE 4.4 – Ajustement de la durée totale des projets.

Reporting sur Power BI

Pour tester les rapports et la validité des données , nous avons utilisé l'outil Power BI pour nous donner une impression sur les rapports qu'on doit intégrer à notre application afin de les visulaisier.

Pour ce fait on a crée les rapports correspondants à l'étude décisionnelle :
Les deux axes importants de l'étude sont les couts et la durée :

— Les coûts

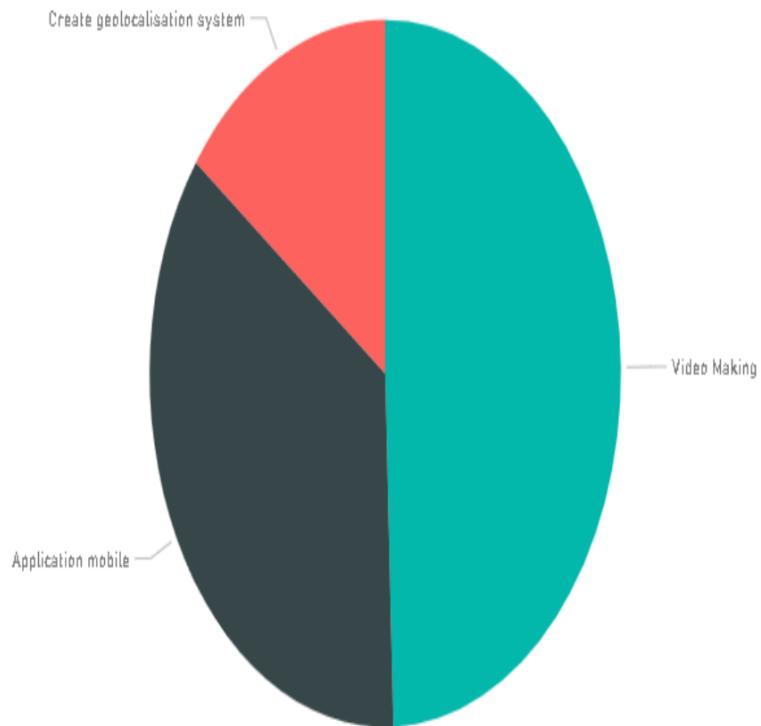


FIGURE 4.5 – Coût total par projet.

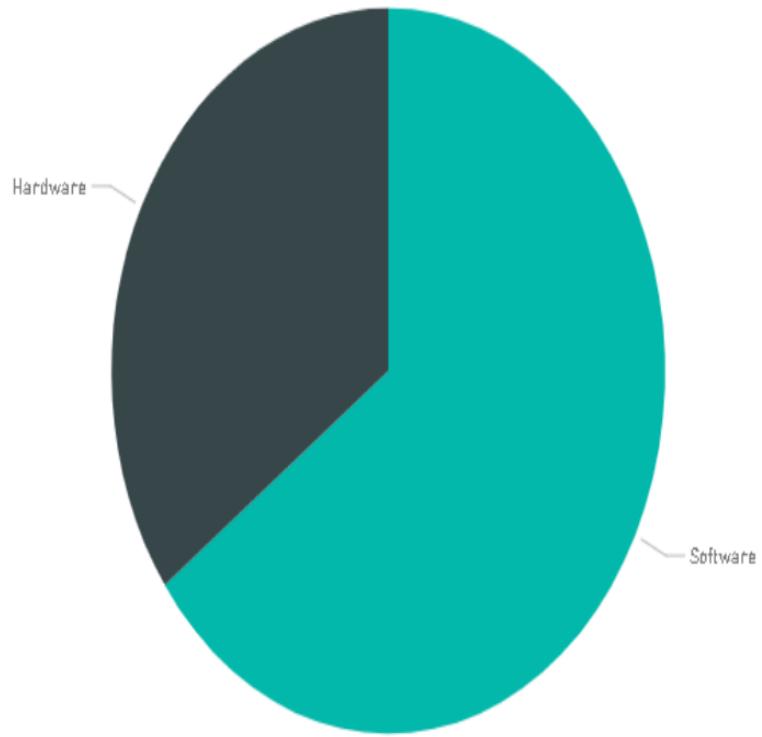


FIGURE 4.6 – Coût total par catégorie de projet.

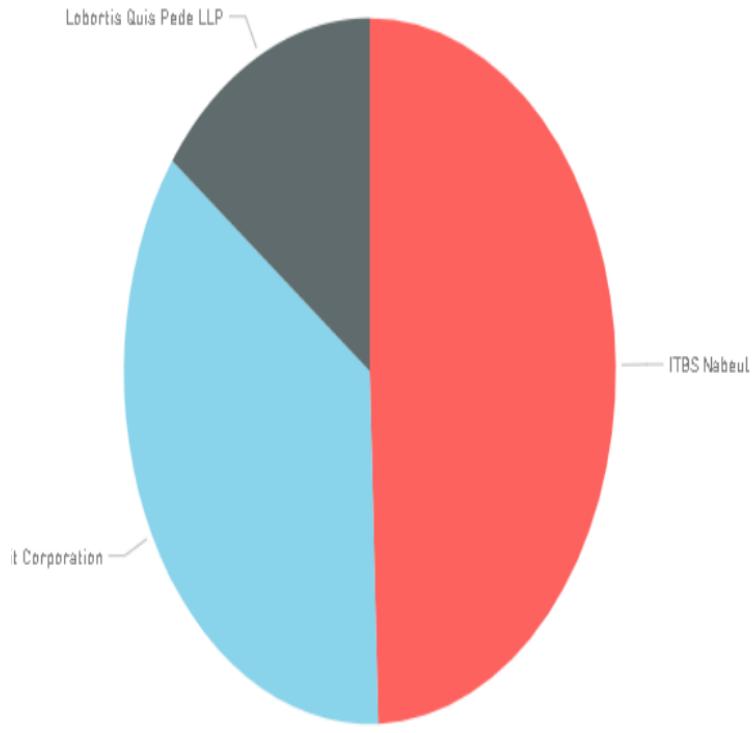


FIGURE 4.7 – Coût total par client.

name	price
ITBS Nabeul	500.00
Proin Velit Corporation	361.00
Lobortis Quis Pede LLP	150.00

FIGURE 4.8 – Coût total par client en chiffres.

— La durée

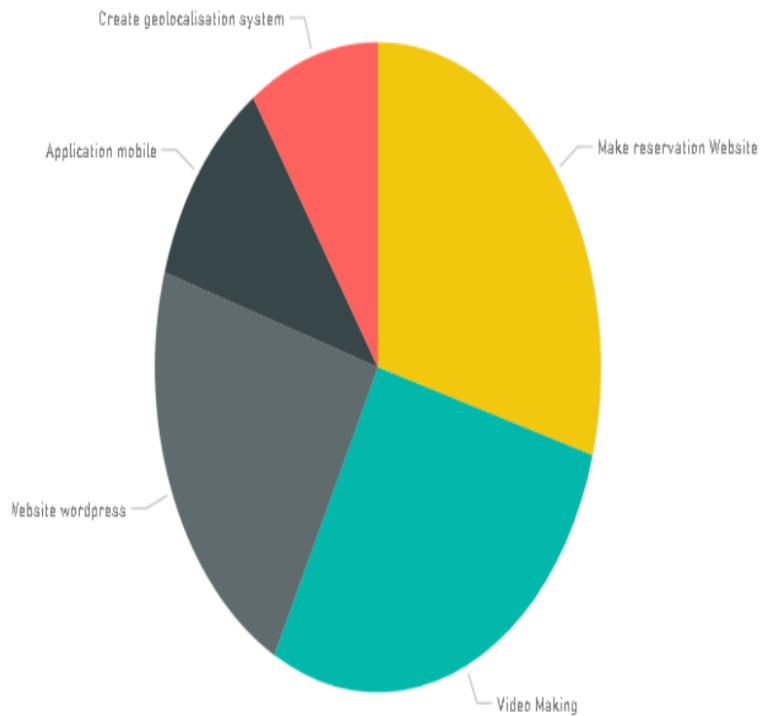


FIGURE 4.9 – Durée total par projet (en jours).1

name	durationInDays
Make reservation Website	84
Video Making	81
Website wordpress	63
Application mobile	31
Create geolocalisation system	27

FIGURE 4.10 – Durée total par projet (en jours).2

4.5 Partie ETL

Pour avoir des rapports sur l'application web ,on doit avoir les données sous une certaine format , afin de satisfaire le besoin de la consultation du suivi des projets selon le temps et le cout. Ainsi les données correspondantes sont collectées, modifiées et chargées dans un entrepôt de données assurant ainsi la phase ETL.

Nous avons utilisé l'outil talend pour charger les données nécessaires . Puisque nous avons les données ordonnées et intégrées ,nous n'avons pas été besoin d'une ODS (Operational data source) nous passons directement à construire notre table de faits

4.5.1 Intégration

Pour obtenir les coûts nous avons calculé le prix par chaque projet ,catégorie ,et nom du client.

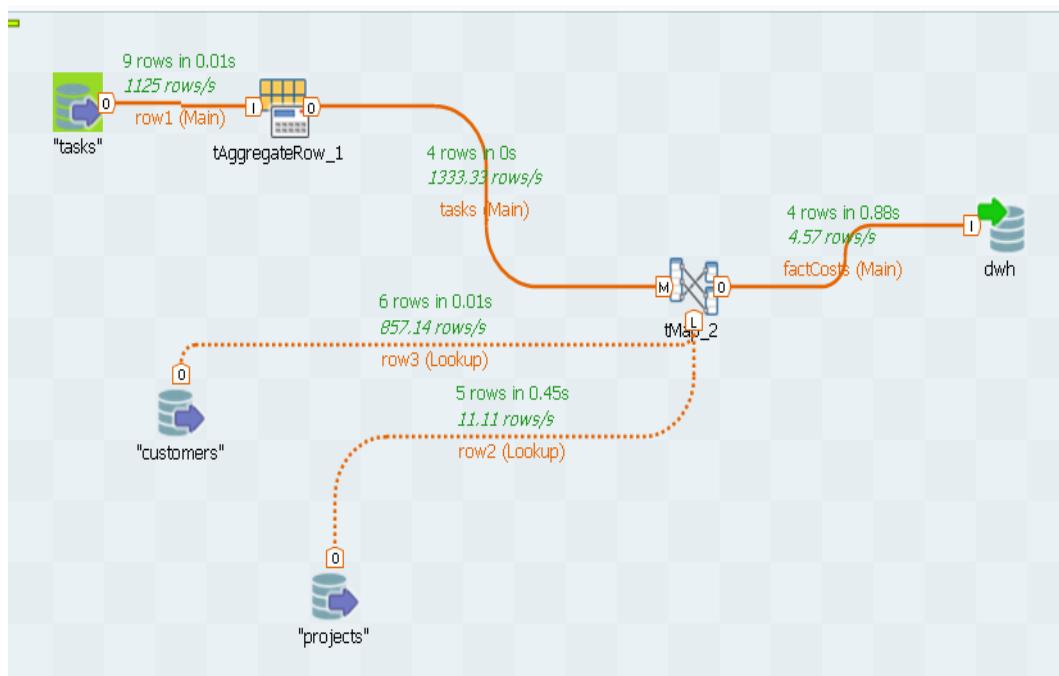


FIGURE 4.11 – Extraction des coûts.1.

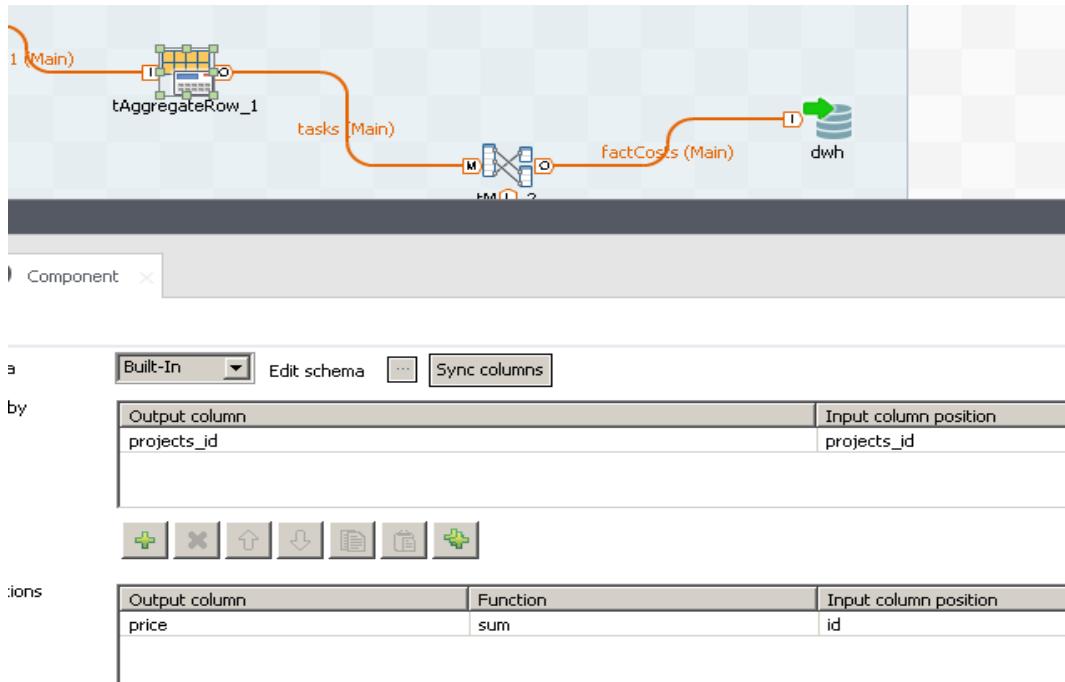


FIGURE 4.12 – Extraction des coûts.2.

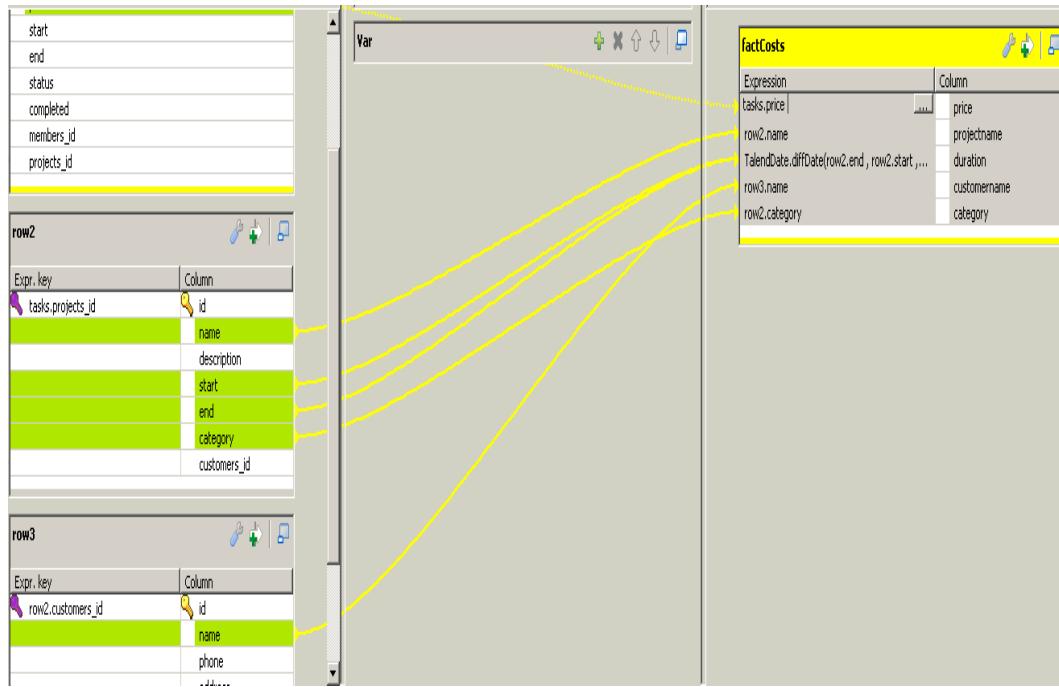


FIGURE 4.13 – Extraction des coûts.3.

Pour obtenir la durée du chaque projet en jours , nous avons calculé pour chaque projet la différence en jours .

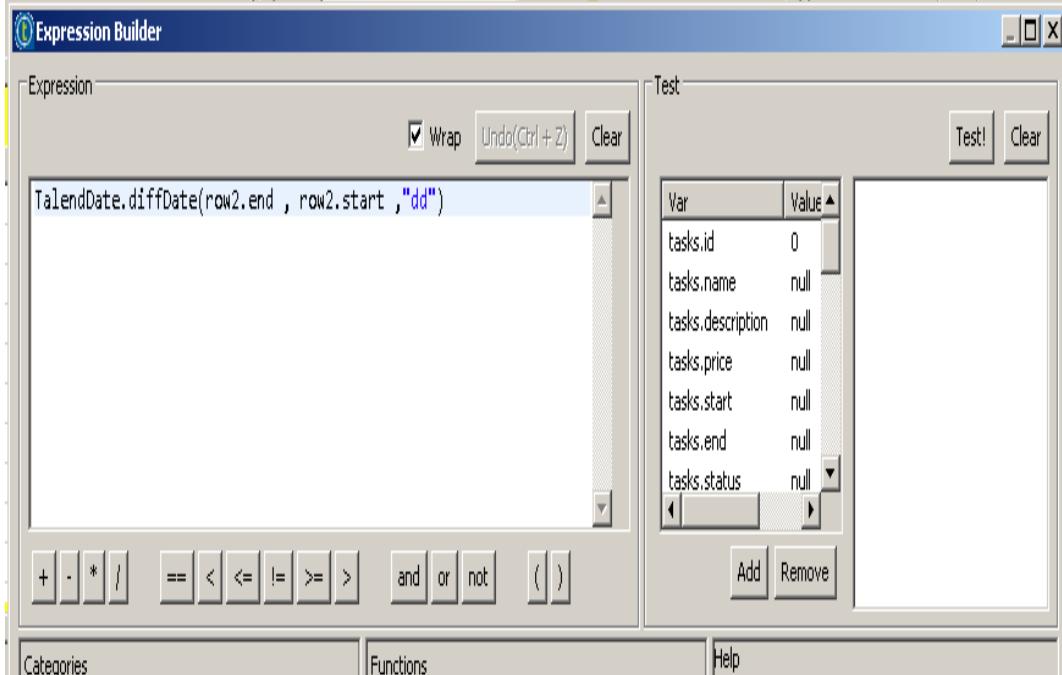


FIGURE 4.14 – Extraction des durées.

4.5.2 Shémas obtenus

A partir du schéma de la figure 4.13, on a obtenu la table fact selon ce schéma

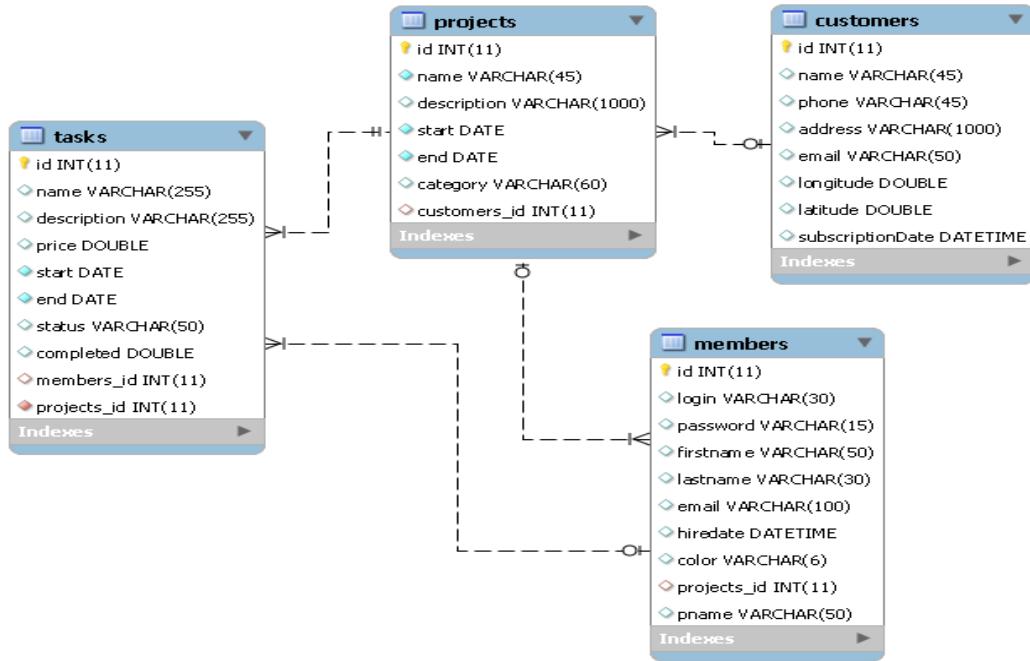


FIGURE 4.15 – Shéma initial.

Field	Type	Null	Key	Default	Extra
price	double(22,0)	YES		NULL	
projectname	varchar(45)	NO		NULL	
duration	bigint(10)	NO		NULL	
customername	varchar(45)	YES		NULL	
category	varchar(60)	YES		NULL	

TABLE 4.1 – Shéma de la table des faits.

les données obtenues sont comme suit :

price	projectname	duration	customername	category
66	Website wordpress	63	ITBS Nabeul	Hardware
121	Application mobile	31	Proin Velit Corporation	Software
81	Video Making	81	ITBS Nabeul	Software
43	Create geolocalisation system	27	Lobortis Quis Pede LLP	Hardware

TABLE 4.2 – Contenu de la table des faits.

Test

Si l'administrateur parvient à se connecter il trouve cette interface d'accueil où il trouvera les rapports qui décrivent des statistiques primordiales au déroulement des projets donc les interfaces et les fonctionnalités disponibles pour l'admin sont :

4.5.3 La consultation des rapports

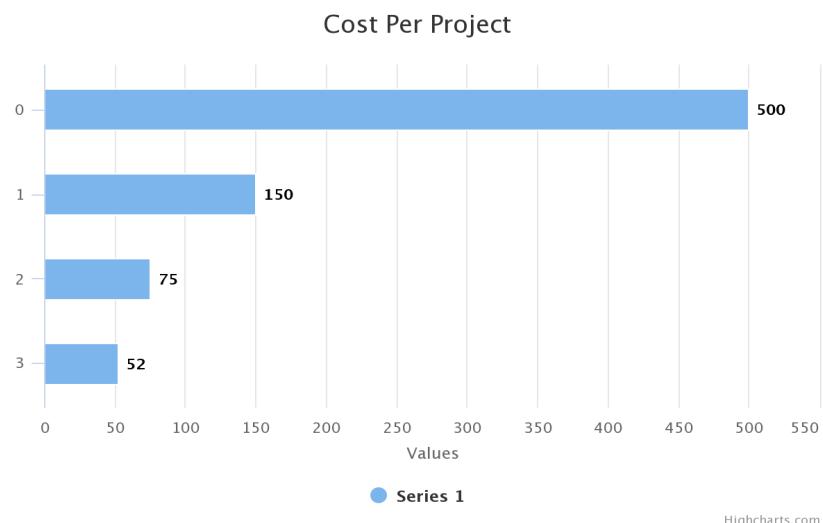


FIGURE 4.16 – Coût par projet

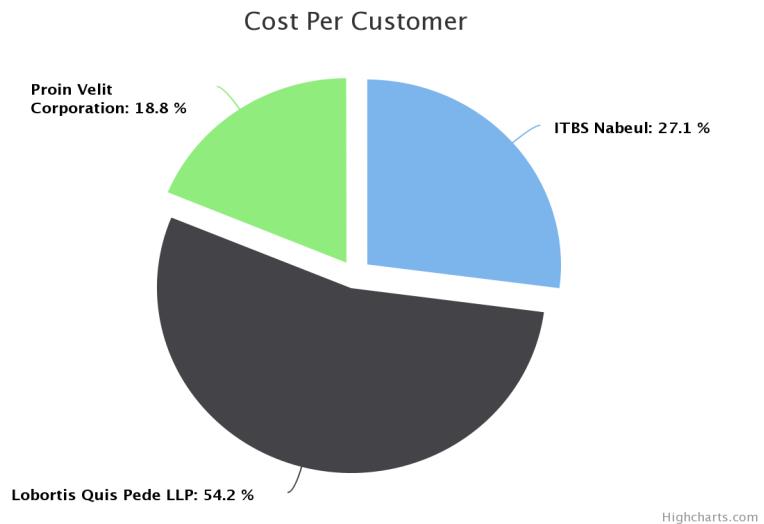


FIGURE 4.17 – Coût par client

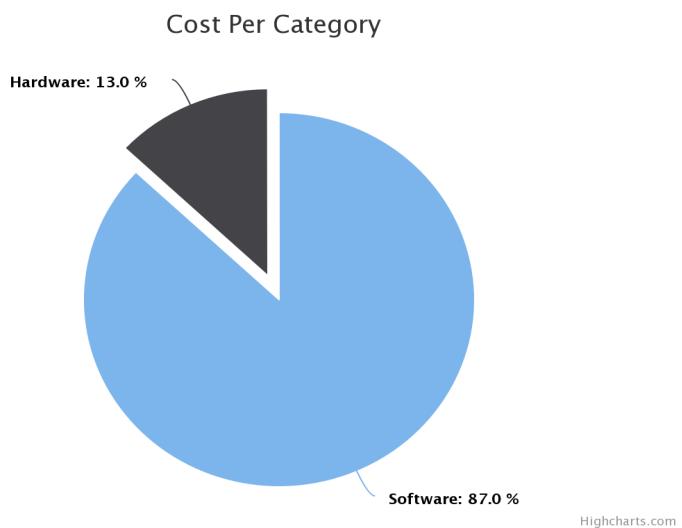


FIGURE 4.18 – Coût par catégorie

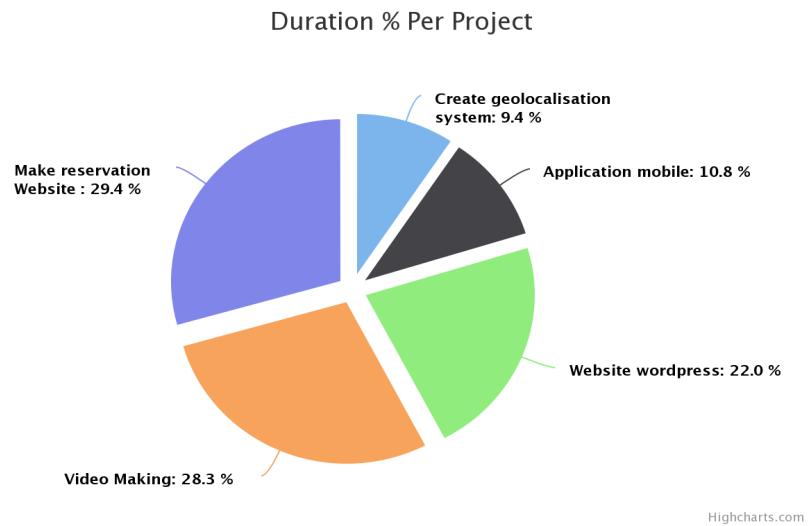


FIGURE 4.19 – Durée par projet.1.

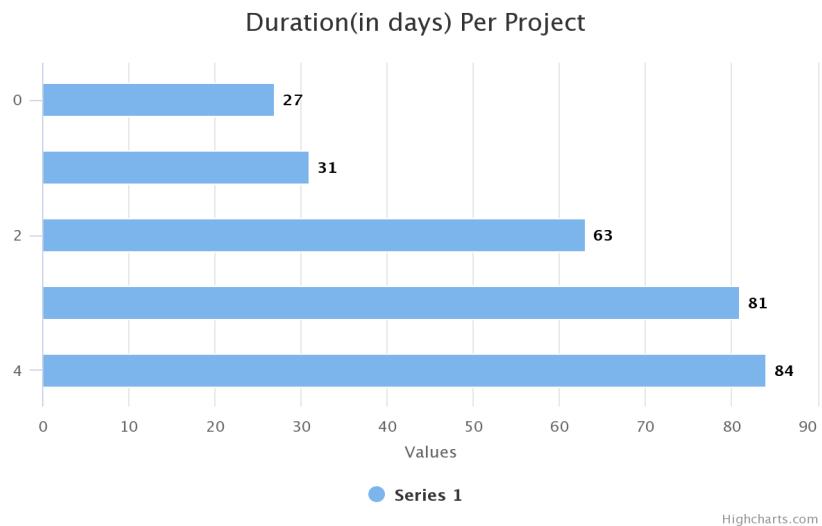


FIGURE 4.20 – Durée par projet.2.

Chapitre 5

Clôture du projet

5.1 Introduction

La phase clôture ou de fermeture est la dernière phase dans le cycle de développement d'un logiciel avec Scrum. Cette phase est souvent appelé sprint de stabilisation. Les tâches effectuées pendant cette phase prédéfinies, et ils dépendent fortement du type de projet. Pour notre projet, ce chapitre sera consacré pour la présentation des langages et outils de programmation utilisés pour la réalisation de nos deux applications.

5.2 Implémentation et structure

Enfin nous décrivons les étapes globales suivies lors de la réalisation de ce projet. Les étapes étaient :

- Créer le site web statique (front -end)
- Créer une application express js
- Création du base de données et liaison des données par le driver node js de mysql
- Créer un rest api à l'aide de express js

- Intégration de front-end avec le back-end
- Ajout des modules supplémentaire (Authentification avec jwt et gestion des roles utilisateur et administrateur)
 - Hébergement en ligne de la base de données
 - Hébergement en ligne de l'application web

5.3 Application en ligne

Vous pouvez trouverz l'application sur le lien [13].

Pour accéder en tant qu'administrateur veuilez utiliser :

- **Pseudo** :admin
- **Password** :0000

Pour accéder en tant que membre 'Wael Chorfan' par exemple veuillez utiliser :

- **Pseudo** :WC
- **Password** :0001

5.4 Architecture 3 tiers

Notre projet est caractérisé par son architecture 3 tiers qui inclus un modèle MVC (Modèle Vue Contrôleur) • Machine : accès et mise à jour des données, • Back-End express Server : contient l'application, • MySql : gestion de données et mise à jour.

5.5 Environnement de développement

5.5.1 Environnement matériel

Pour la réalisation du projet, nous avons utilisé un ordinateur portable pour le développement ayant les caractéristiques suivantes :

- Modèle : Asus XJ550
- Processeur : i7 2.6GHz
- Disque Dur : 1To
- Systèmes d'exploitation : Windows 7
- Mémoire : 8Go

5.5.2 Environnement logiciel

L'environnement logiciel utilisé pour réaliser notre projet est comme suit :

- MySQL

C'est un système de gestion de bases de données relationnelles. Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde.

Nous avons utilisé wamp (phpmyadmin)[5] en local et remote mysql pour l'hébergement en ligne[6].



FIGURE 5.1 – phpmyadmin.



FIGURE 5.2 – Wamp.

- Visual Studio Code [1]

est un éditeur de code extensible développé par Microsoft pour Windows, Linux et OS X. Nous avons utilisé visuel code pour écrire le code de l'application.



FIGURE 5.3 – Visual Studio Code.

- StarUML [14]

StarUML est un logiciel de modélisation UML, cédé comme open source par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL. Nous avons fait les diagrammes avec cette technologie.



FIGURE 5.4 – Staruml.

5.6 Outils et technologies

Node js [2]

Framework javascript ,nous l'avons utilisé pour créer le serveur web . Il offre la rapidité de ,la performance et la modularité.

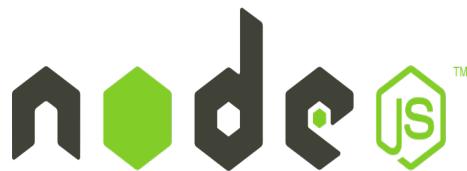


FIGURE 5.5 – Node js.

Express js : [3]

C'est un framework pour construire des applications web basées sur Node js . Il sert à créer l'application ,il est en relation avec la base de données par

le biais de driver mysql et en relation avec les modules web par le moteur de vues EJS.



FIGURE 5.6 – Express js.

Postman : [16]

Est le seul environnement de développement d'API complet utilisé par plus de 7 millions de développeurs et 200 000 entreprises dans le monde cela rend le développement d'API plus rapide, plus facile et plus performant.



FIGURE 5.7 – Postman.

Vue js : [11]

Framework javascript front-end utilisé pour la programmation et la manipulation des actions, entrées et sorties des différents modules .



FIGURE 5.8 – Vue js.

Mysql : [12]

Langage de la base de données relationnelle utilisée. e.EJS moteur de vue d'express js , ce type permet l'intercommunication entre les modules web et le serveur .



FIGURE 5.9 – MySQL.

EJS : 10]

moteur de vue d'express js , ce type permet l'intercommunication entre les modules web et le serveur .

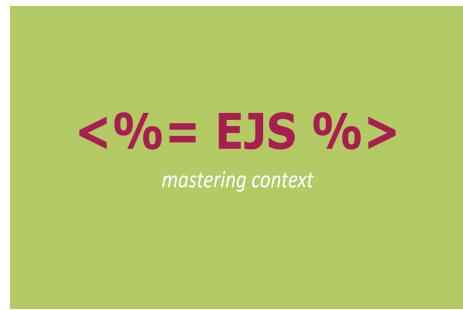


FIGURE 5.10 – EJS.

Highcharts : [15]
L'outil de <> reporting <> sur les pages web.



FIGURE 5.11 – Highcharts.

Conclusion et Perspectives

Notre projet consiste à la conception et la réalisation d'une plateforme web destiné pour la gestion des projets. Contrairement à la majorité des travaux existants sur le marché qui nécessitent un effort de configuration considérable, nous avons réalisé un système qui permet à la fois de gérer les projets et générer des rapports pour avoir une idée générale sur le déroulement des projets en cours.

En ce qui concerne la démarche, nous avons en premier lieu effectué une phase d'étude des différents outils existants. En deuxième lieu nous avons spécifié notre application pour discerner les fonctionnalités .En troisième lieu, nous avons procédé à sa conception ainsi qu'aux choix technologiques pour sa réalisation. Enfin, nous l'avons mise en œuvre. Toutes les fonctionnalités décrites dans le cahier des spécifications fonctionnelles ont été développées et validées. Néanmoins, notre projet pourra être amélioré par l'ajout d'autre fonctionnalités comme :

- La gestion par heure et non par date seulement.
- L'ajout des comptes d'utilisateurs pour les clients afin qu'il suivent l'avancement de leurs projets.
- La suivie des erreurs et des problèmes qui sera une conséquence de la gestion des tickets lancées par les clients.

Il est important à noter que la réalisation de ce projet nous a été bénéfique sur tous les plans. Sur le plan technique, ce projet nous a été une bonne occasion pour découvrir et maîtriser la technologie node js et vue js , d'approfondir nos connaissances sur le plan des nouvelles technologies de développement web et d' hébergement des applications en ligne avec un compte gratuit limité sur la plateforme « heroku ».

Sur le plan humain, ce projet a été une véritable occasion de vivre de près l'expérience du travail au sein d'une société, qui exige la ponctualité et l'intégration dans un groupe de travail.

Ce qui nous a permis d'améliorer nos capacités de communication et de nous adapter à la vie professionnelle. Nous avons fait de notre mieux pour bien laisser une bonne impression sur notre discipline, nos qualités et nos compétences techniques vis à vis du staff technique de start-up Cherchini.tn et présenter un travail à la hauteur de la formation qui nous avons eu au sein de l'ITBS.

Bibliographie

- [1] <https://code.visualstudio.com/download>
- [2] <https://nodejs.org/en/>
- [3] <https://www.npmjs.com>
- [4] <http://www.wampserver.com>
- [5] <https://www.phpmyadmin.net>
- [6] <https://remotemysql.com>
- [7] <https://www.heroku.com>
- [8] <https://getbootstrap.com/docs/4.3/getting-started/introduction/>
- [9] <https://stackoverflow.com/>
- [10] <https://ejs.co/#docs>
- [11] <https://vuejs.org/v2/guide/>
- [12] <https://dev.mysql.com/doc/refman/5.7/en/>
- [13] <https://cherchini-project.herokuapp.com>
- [14] <http://staruml.io/>
- [15] <https://www.highcharts.com> [16] <https://www.getpostman.com/>

RESUMÉ

C'est dans le cadre de l'obtention du titre d'ingénieur en informatique ,ce projet réalisé pour Cherchini.tn consiste à développer une application Web qui représente un outil de gestion de projets. Le chapitre 1 définit le contexte du projet. Le chapitre 2 s'intègre à l'analyse de besoin de notre projet. Le chapitre 3 est consacré à la conception. Le chapitre 4 illustre la réalisation du notre système. Il donne également la possibilité de suivre les coûts et la durée des projets dans un tableau de bord.

Mots clés : WebServer,Database , REST API , ETL ,BI

ABSTRACT

It is in the context of obtaining the title of engineer in computer science, this project realized with Cherchini.tn consists in developing a web application for project management , it also gives the possibility to follow projects costs and duration within a dashboard . Chapter 1 defines the context of the project. Chapter 2 is interested in the analysis of need of our project. Chapter 3 is dedicated to design. Chapter 4 is reserved for the realization of our system.

Key words : Serveur Web,Base de données , REST API , ETL ,BI