



REPUBLIQUE DU BENIN

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE



UNIVERSITE D'ABOMEY CALAVI (UAC)

ECOLE NATIONALE D'ECONOMIE APPLIQUEE ET DE MANAGEMENT (ENEAM)

MEMOIRE DE FIN DE FORMATION POUR L'OBTENTION DU DIPLOME DE
TECHNICIEN SUPERIEUR (DTS)

Filière : Informatique de Gestion (IG)

Spécialité : Analyse Informatique
et Programmation (AIP)

THEME

Mise en place d'un système de management d'idées dans
les organismes privés et publics au Bénin

Réalisé par : Elognissè Eude Emmanuel ASSOGBA BALLE

Sous la direction de

Maître de stage

M. Hermann ADENSI
Directeur Général de Digicix

Directeur de mémoire

Dr. Maurice Comlan
Enseignant à l'ENEAM

ANNEE ACADEMIQUE : 2017 - 2018

DEDICACES

Je dédie ce mémoire de fin de stage à mes parents Sylvain et Delphine ASSOGBA
BALLE

REMERCIEMENTS

Ce travail est le fruit des efforts conjugués de bienveillantes personnes à qui nous devons toute notre reconnaissance et gratitude. Nos remerciements vont particulièrement à l'endroit de :

- ❖ Mme Rosaline D. WOROU HOUNDEKON, Directrice de l'ENEAM ;
- ❖ M. Théophile K. DAGBA, PhD., Directeur-adjoint, Chargé des affaires académiques de l'ENEAM ;
- ❖ M. Maurice COMLAN, mon encadreur qui, malgré ses multiples occupations, a accepté de diriger ce travail et y a accordé une attention particulière ;
- ❖ M. Hermann ADINSI, mon maitre de stage ;
- ❖ Nos enseignants, qui ont su nous inculquer le goût du travail bien fait ;
- ❖ Enfin, tous ceux qui, de près ou de loin ont contribué à la réalisation ce travail.

RESUME

Dans ce mémoire, nous réalisons une application de gestion des idées (boîte à idées virtuelle) qui améliore le système de boîte à idées actuellement en place (boîte à idées physique) dans la plupart des entreprises privées et publiques du Bénin. Ce système associe les clients, les employés ainsi que les dirigeants de l'entreprise autour d'une même perspective, celle de l'amélioration des performances de leur unité de production. Ce nouveau système de boîte à idées intègre un module de récompense des meilleures propositions d'idées suivant des critères définis par l'entreprise. Ceci peut susciter chez les employés beaucoup plus de motivation et d'engagement dans les activités de l'entreprise. Avec ce système, les employés pourront entre autres :

- ✓ suivre leurs idées et plaintes en temps réel et
- ✓ voter pour l'application d'une idée donnée.

Ce système de boîte à idées virtuelle permet également aux clients de se prononcer sur la qualité des produits et /ou services qui leurs sont proposés. A l'entreprise alors, d'identifier les problèmes en interne qui pourraient expliquer une démotivation des salariés ou une baisse de la productivité.

Mots clés : idées, boîte à idées, innovation, management d'idées, suggestions.

ABSTRACT

In this thesis we intend to realize an idea management application (virtual ideas box) that improves the idea box system currently in place (physical ideas box) in most private and public companies in Benin. This system associates customers, employees as well as managers of the company around the same perspective, that of improving the performance of their production unit. This new idea box system integrates a reward module of the best ideas proposals according to criteria defined by the company. This can create a lot more motivation and commitment among employees in the company's activities. With this system, employees will be able to:

- ✓ follow their ideas and complaints in real time and
- ✓ vote for the application of a given idea.

This virtual idea box system also allows customers to express their opinion on the quality of the products and / or services offered to them and the company, to identify internal problems that could explain a discouragement of employees or a decrease in productivity.

Key words: ideas, ideas box, innovation, idea management, suggestions.

LISTE DES TABLEAUX

<i>Tableau 1: Identification des acteurs et des cas d'utilisations du SMI.....</i>	<i>16</i>
<i>Tableau 2: Dictionnaire de données</i>	<i>23</i>

LISTE DES FIGURES

Figure 1 : organigramme de Digicix.....	5
Figure 2: Digramme de cas d'utilisation du SMI	17
Figure 3: Diagramme de classes du SMI	27
Figure 4 : Diagramme d'objets des classes Idees, Roles et Utilisateurs.....	28
Figure 5: Diagramme d'objets des classes Utilisateurs, Roles, Motifs et Plaintes	29
Figure 6: Diagramme d'objets des classes Utilisateurs, Roles, Idees, Types, Reagir et Voter.....	30
Figure 7: Diagramme de séquences du cas « Proposer une idée »	31
Figure 8: Digramme de séquences du cas « Envoyer une plainte »	32
Figure 9 : Diagramme de séquences du cas « Voter une idée ».....	33
Figure 10: Diagramme d'état transition du cas « Proposer une idée »	34
Figure 11: Diagramme d'état transition du cas « Envoyer une plainte »	35
Figure 12: Diagramme d'état de transition du cas « Voter une idée »	36
Figure 13: Interface de connexion.....	42
Figure 14 : Interface de suivi des idées pour le rôle employé	43
Figure 15 : Interface de suivi des plaintes pour le rôle employé	44
Figure 16: Interface de suivi des statistiques pour le rôle employé	45
Figure 17 : Interface de gestion des idées pour les rôles admin et responsable	51
Figure 18 : Interface de gestion des plaintes pour les rôles admin et responsable	52

LISTE DES SIGLES ET ABREVIATIONS

- **API** : Application Programming Interface
- **BD** : Base de données
- **CSS** : Cascading Style Sheets
- **ENEAM** : Ecole Nationale d'Economie Appliquée et de Management
- **HTML** : HyperText Markup Language
- **HTTP** : HyperText Transfert Protocole
- **JSON** : JavaScript Object Notation
- **JWT** : Json Web Token
- **MVC** : Modèle Vue Contrôleur
- **NTIC** : Nouvelles Technologies de L'Information et de la Communication
- **PHP** : Php Hypertext Preprocessor
- **REST** : REpresentational State Transfer
- **SGBD** : Système de Gestion de Base de Données
- **SMI** : Système de Management d'idées
- **SQL** : Structured Query Language
- **UML** : Unified Modeling Language
- **Xss** : Cross-site scripting

TABLE DES MATIERES

INTRODUCTION	1
Chapitre I : Présentation du contexte de l'étude	3
1.1. Présentation de Digicix.....	4
1.1.1 Historique et Mission	4
1.1.2 Organigramme.....	4
1.1.3 Déroulement du stage.....	5
1.2. Etude préliminaire.....	6
1.2.1. Présentation de l'existant.....	6
1.2.2. Critique de l'existant	7
1.2.3. Approche de solution : une boîte à idées virtuelle	7
1.2.4. Objectifs globaux et spécifiques	8
Chapitre II : Analyse et Spécification des besoins.....	10
2.1 Spécification des besoins.....	11
2.1.1. Spécification des acteurs.....	11
2.1.2. Spécification fonctionnelle.....	11
2.1.3. Description fonctionnelle.....	12
2.1.4. Spécification non-fonctionnelle.....	14
2.2 Analyse orienté objet.....	15
2.2.1. Les cas d'utilisation.....	15
2.2.1.1. Identification des acteurs et des cas d'utilisation du SMI	15
2.2.1.2. Diagramme de cas d'utilisation.....	16
2.2.1.3. Description textuelle des cas d'utilisation	18
2.2.2. Les classes	23
2.2.2.1. Dictionnaire de données du SMI.....	23
2.2.2.2. Règles du SMI	25

2.2.2.3. Diagramme de classes	26
2.2.3. Diagramme d'objets	27
2.2.4. Diagrammes de séquences	30
2.2.5. Diagrammes d'état transition	33
Chapitre III : Implémentation du système	37
3.1. Outils de développement utilisés	38
3.2. Sécurité de l'application	41
3.3. Travaux réalisés	41
CONCLUSION	47
PERSPECTIVE	48
BIBLIOGRAPHIE & WEBOGRAPHIE	49
ANNEXES	51

INTRODUCTION

Le concept de la boîte à idées n'est pas nouveau. Depuis de nombreuses années, les organismes l'utilisent pour permettre à leurs employés de soumettre leurs idées et de s'exprimer (Olivier Dutel,2013). Cependant, cette pratique, dans l'entreprise, n'est pas anodine (Olivier Dutel,2013). Elle montre une certaine volonté d'associer l'ensemble de ses collaborateurs à l'évolution de son entreprise et de valoriser l'innovation participative. Elle permet donc aux employés d'être initiateurs et acteurs dans la vie de l'entreprise. Ce qui est très motivant pour le salarié.

Le constat est que de nos jours ces boîtes à idées n'occupent plus cette place qui est la leur dans les entreprises. Cette sous-utilisation de la boîte à idées est due à plusieurs facteurs dont le principal est l'emploi archaïque fait de la boîte à idées à un moment où le numérique se répand dans nos vies et dans tous les secteurs d'activité. En outre, la boîte à idées dans une entreprise permet la réception des avis des salariés. Cependant, cette fonction de la boîte à idées ne permet pas aux clients de se prononcer sur les produits et ou services qui leurs sont fournis. Ce qui freine d'une manière ou d'une autre l'évolution normale de l'entreprise.

Ainsi, vient la nécessité de développer une solution permettant de régler l'intégralité de ces problèmes, pour satisfaire au mieux les clients et améliorer le rendement de l'entreprise tout en tenant compte des réalités du Bénin. C'est dans cette optique que s'inscrit notre stage pratique de trois (3) mois en programmation au sein de l'entreprise **Digicix** où nous avons développé une solution applicative sur la base du thème « **Mise en place d'un système de management d'idées dans les organismes privés et publics du Bénin** ».

Le présent mémoire structuré en trois (3) chapitres et expose le travail d'analyse et d'implémentation pour la mise en place du système. Le premier

chapitre aborde la présentation du contexte d'étude à travers la présentation de la structure d'accueil et la présentation du thème. Le deuxième chapitre expose l'analyse et la spécification des besoins effectués pour ledit système. Enfin le troisième chapitre présente l'aspect sécuritaire du système ainsi que les outils utilisés pour sa réalisation.

Chapitre I : Présentation du contexte de l'étude

1.1. Présentation de Digicix

1.1.1 Historique et Mission

Digicix est une entreprise franco-béninoise créée en 2012 sous la dénomination « Agence Genetic ». Elle est spécialisée dans les domaines de la transformation digitale et de la communication.

De façon plus spécifique, l'entreprise propose des services dans les domaines de la création de solutions web et mobile, de solutions de performances online (webmarketing), graphisme, communication médias et hors-médias.

L'entreprise Digicix est actuellement présente au Bénin et ambitionne, d'étendre ses activités dans la sous-région d'ici à la fin de cette année 2019.

1.1.2 Organigramme

L'entreprise Digicix est organisée en quatre (4) départements principaux, à savoir :

- La direction ;
- Le département web et mobile ;
- Le département Marketing & webmarketing ;
- Le département commercial.

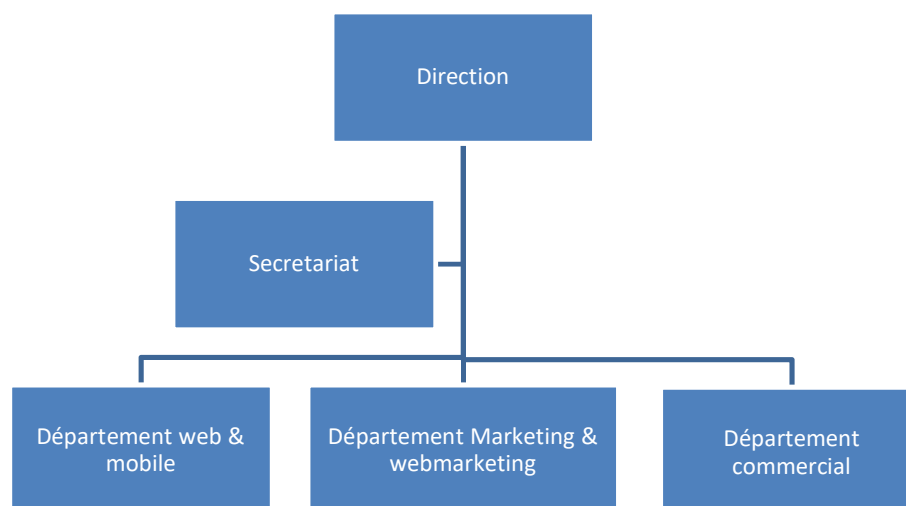


Figure 1 : organigramme de Digicix

1.1.3 Déroulement du stage

Au cours de notre stage, en dehors de notre application nous avons eu l'opportunité de participer à certains travaux au sein de l'entreprise en réalisant les tâches qui nous étaient confiées. Nous avons participé au développement d'une application de gestion de projet, tâche et suivi de rapport dénommé IKILEO. Nous avons également eu à faire diverses formations au nombre desquels on peut citer : l'utilisation du framework PHP Laravel, l'utilisation du framework frontend Angular, le versionnage de projet avec git, le développement en API REST ainsi que la gestion et la configuration d'un droplet (cas de DigitalOcean).

1.2. Etude préliminaire

1.2.1. Présentation de l'existant

La boîte à idées est un système organisé par lequel un salarié d'une entreprise peut, à tout moment, formuler par écrit une idée susceptible d'intéresser sa hiérarchie.

Dans les entreprises privées comme publiques du Bénin, la boîte à idées est représentée par un petit récipient à forme rigide (en carton, en bois ou en métal), munie d'une fente, dans laquelle l'employé(e) peut, à tout moment, glisser un papier sur lequel il aurait préalablement écrit son idée. La boîte à idées doit être placée à un point stratégique de l'entreprise et est destinée à recevoir les propositions (messages écrits au crayon sur des papiers de façon anonyme ou non) faites par les employés. Les idées proposées peuvent être des conseils, des recommandations, des suggestions, des plaintes ou des avis en tout autre genre.

La mise en place d'une boîte à idées dans une entreprise présente plusieurs avantages. C'est un outil de communication qui :

- favorise l'épanouissement et une implication plus accrue de ses employés ;
- augmente également la motivation et l'engagement des collaborateurs.
- constitue pour l'employeur un bon moyen d'innover, d'améliorer le fonctionnement de l'entreprise et de faire des gains grâce à des idées simples et efficaces ;
- aide à la prise de décisions et la conduite du changement ;
- permet enfin aux collaborateurs de faire parvenir à la hiérarchie des réalités du terrain susceptible d'améliorer les produits et services de l'entreprise pour une meilleure productivité.

Malgré ses nombreux avantages, le système de boîte à idées rencontre quelques insuffisances qui ne favorisent pas forcément son adoption. Ce système peut ne pas être la bienvenue dans certaines entreprises.

1.2.2. Critique de l'existant

La mise en place d'un système de boîte à idées a des limites. Lorsque celle-ci est physique, elle doit être stratégiquement placée à la portée de tous. Elle nécessite du temps, un crayon et du papier pour pouvoir faire parvenir son idée. Elle peut être encombrant. Elle ne peut être utilisée qu'aux heures de travail. Elle ne permet pas un suivi des idées proposées. Ce qui peut être décourageant.

Ce principe de boîte à idées dans les entreprises ne date pas d'aujourd'hui. Il existe depuis 1900 et a été mis en arrière-plan car beaucoup d'entreprises se sont rendues compte qu'il servait beaucoup aux revendications anonymes et surtout qu'il remontait des questions auxquelles on ne voulait pas répondre.

Malgré cet important problème rencontré dans l'utilisation de ce système, nous proposons une boîte à idées virtuelle qui procure les mêmes avantages qu'une boîte à idées physique avec moins de contraintes dans son utilisation et apporte beaucoup d'autres améliorations dans son fonctionnement.

1.2.3. Approche de solution : une boîte à idées virtuelle

Dans le but d'aider tous les organismes à se développer, nous proposons une solution qui rassemble tous les acteurs de la vie de l'entreprise dans l'atteinte d'un même objectif. Il s'agit d'une solution cloud où chaque entreprise pourra disposer d'un espace privé après avoir souscrit à un abonnement. Cette solution permet une implication plus importante de ses employés dans son système de prise de décision. Les employés étant aussi des acteurs de l'organisme et surtout ceux étant les plus confrontés aux diverses réalités du terrain (le monde en dehors

de l'entreprise), mais aussi à celles de l'organisme. Il est évident qu'ils sont bien placés pour donner leur avis sur les différentes décisions à prendre que ce soit pour l'organisme en lui-même ou pour ses relations avec l'extérieur. Cette solution permettra aussi aux clients de donner leur avis sur les produits et ou services qui leurs sont proposés, réduisant ainsi les mécontentements, les pertes de clients et autres. Le fait de réunir plusieurs compétences autour d'un même problème fait naître parfois des solutions au-delà des espérances. Une telle solution permettra d'atteindre efficacement les objectifs fixés.

1.2.4. Objectifs globaux et spécifiques

1.2.4.1. Objectifs globaux

La mise en place de cette solution à principalement pour but d'aider chaque entreprise à prospérer, innover et user pleinement des ressources à sa portée. Nous proposons une solution de boîte à idées virtuelle adaptée aux réalités de notre pays, favorisant ainsi l'innovation participative dans nos entreprises.

1.2.4.2. Objectifs spécifiques

Plus spécifiquement, les objectifs visés à travers la mise en place de cette solution sont :

- permettre à un employé de participer activement à la vie de l'entreprise à travers les débats, les propositions d'idées, les prises de décisions ;
- permettre à un employé de suivre l'évolution de ses idées proposées et ce en temps réel ;
- permettre à un client de se prononcer, de donner son avis et même de faire des propositions, toujours dans le but d'améliorer la qualité du service ;
- organiser facilement les votes ;

- avoir rapidement des statistiques ;
- repérer les forces vives de l'organisme ;
- s'assurer de répondre dans la mesure du possible aux interrogations et idées des collaborateurs ;
- tenir informer les clients des changements ou de nouveaux produits ;
- permettre à l'entreprise de gagner en clientèle, accroître son rendement, faire facilement de la publicité, éviter des pertes de temps inutiles et surtout d'améliorer la qualité des produits et ou services proposés.

Afin d'atteindre les objectifs précédemment énumérés, les utilisateurs du Système de Management d'Idées (SMI) pourront principalement selon leurs attributions :

- proposer des idées ;
- voter des idées postées ;
- commenter des idées postées ;
- envoyer des plaintes ;
- accéder aux données statistiques ;
- répondre aux plaintes.
- Appuyer une plainte

Chapitre II : Analyse et Spécification des besoins

2.1 Spécification des besoins

2.1.1. Spécification des acteurs

Un acteur est une personne, un matériel ou un logiciel qui interagit avec le système dans le but de réaliser une plus-value. Les acteurs en interaction avec notre système sont :

- l'employé
- le Client
- le Responsable
- l'Administrateur

2.1.2. Spécification fonctionnelle

Le système à mettre en place devra fournir certaines fonctionnalités. Elle doit permettre :

- ✓ aux clients connectés sur la plateforme
 - de proposer des idées
 - d'envoyer des plaintes
- ✓ à l'administrateur connecté sur la plateforme
 - d'envoyer des invitations
 - d'administrer tous les comptes utilisateurs aussi bien clients qu'employés
- ✓ aux employés connectés sur la plateforme
 - de proposer des idées
 - de voter des idées
 - de commenter des idées
 - de voir les récompenses proposées par l'entreprise

- d'envoyer des plaintes
 - d'appuyer une plainte
 - de consulter ses propres statistiques
- ✓ aux responsables connectés sur la plateforme
- de soumettre une idée à un vote
 - de valider ou rejeter une idée
 - de proposer des récompenses suivant des critères
 - de répondre à une plainte
 - de consulter les données statistiques

2.1.3. Description fonctionnelle

La mise en place du système de boîte à idées virtuelle doit prendre en compte les fonctionnalités du système de boîte à idées physique et, au-delà de cela, elle doit prendre en compte de nouvelles fonctionnalités dans le but de rendre le nouveau système plus performant et plus sécurisé. Le système à mettre en place permettra entre autres :

➤ d'inviter un utilisateur :

Cette fonctionnalité permet à l'administrateur d'envoyer une invitation par mail à un employé ou à un responsable pour rejoindre l'espace de l'entreprise. L'invitation une fois reçue par l'employé ou le responsable, ce dernier pourra rejoindre l'espace en fournissant les informations nécessaires pour la confirmation de son compte.

➤ de gérer les utilisateurs :

Cette fonctionnalité regroupe plusieurs autres sous fonctionnalités à savoir bloquer ou débloquer un compte, supprimer un compte, avoir certaines informations sur le détenteur d'un compte.

➤ de proposer une idée :

Cette fonctionnalité permet à un client ou à un employé de proposer une idée à l'entreprise.

➤ d'envoyer une plainte :

Cette fonctionnalité permet à un client ou un employé d'adresser une plainte à l'entreprise.

➤ de soumettre une idée à un vote :

Cette fonctionnalité permet à l'administrateur ou aux responsables de rendre une idée disponible pour un vote.

➤ de voter une idée :

Cette fonctionnalité permet à un employé d'approuver ou de rejeter une idée proposée.

➤ de commenter une idée :

Cette fonctionnalité permet à un employé de commenter une idée. Ceci peut être l'expression d'une incompréhension par rapport à l'idée proposée ou autre.

➤ de valider une idée :

Cette fonctionnalité permet à l'administrateur ou aux responsables de rendre une idée applicable.

➤ de répondre à une plainte :

Cette fonctionnalité permet à l'administrateur ou aux responsables d'indiquer au plaignant si sa plainte sera prise en considération ou non. Il s'agit d'une réponse (textuelle) envoyée suite à la plainte reçue.

➤ de proposer des récompenses :

Cette fonctionnalité permet à l'administrateur d'enregistrer des récompenses qui seront visibles par les employés. Ce qui pourrait constituer une motivation pour ces derniers.

➤ de consulter les statistiques :

Cette fonctionnalité permet aux utilisateurs d'apprécier, à un moment donné, tout le travail effectué grâce à des statistiques complètes.

2.1.4. Spécification non-fonctionnelle

Outre les besoins fondamentaux, notre système doit répondre aux critères suivants :

LA SÉCURITÉ ET L'INTÉGRITÉ DES DONNÉES : L'accès à l'application n'est permis qu'après une phase d'authentification sécurisée. Les informations d'authentification doivent être confidentielles. L'application doit garantir l'intégrité, la cohérence et la persistance des données.

LA RAPIDITÉ DE TRAITEMENT : En effet, vu le nombre important des transactions quotidiennes, il est impérativement nécessaire que la durée d'exécution des traitements s'approche le plus possible du temps réel.

LA PERFORMANCE : Notre logiciel doit être avant tout performant, c'est-à-dire à travers ses fonctionnalités, répondre à toutes les exigences des usagers d'une manière optimale.

LA CONVIVIALITÉ : Le futur logiciel doit être facile à utiliser. En effet, les interfaces utilisateurs doivent être conviviales, c'est-à-dire simples, ergonomiques et adaptées à l'utilisateur.

2.2 Analyse orienté objet

Pour l'analyse de notre système, nous avons choisi le langage de modélisation unifié généralement connu sous *Unified Modeling Language (UML)* en anglais. UML est un métalangage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.

UML 2.3 propose 14 types de diagrammes. Ces diagrammes sont répartis en deux types : les diagrammes comportementaux ou dynamiques et les diagrammes structurels ou statiques. Nous ne présenterons que 5 d'entre eux dans ce travail d'analyse. Il s'agit des diagrammes de cas d'utilisation, de classe, d'objet, de séquence et d'état transition.

2.2.1. Les cas d'utilisation

2.2.1.1. Identification des acteurs et des cas d'utilisation du SMI

Pour le bon fonctionnement du **SMI**, il est nécessaire de définir, pour chacun de ses acteurs, les actions à mener. Les divers acteurs intervenant dans le système ainsi que les actions qu'ils peuvent mener sont représentés dans le tableau ci-dessous.

Tableau 1: Identification des acteurs et des cas d'utilisations du SMI

Acteurs	Cas d'utilisation
Employé	<ul style="list-style-type: none"> -Proposer une idée -Envoyer une plainte -Consulter ses statistiques -Voir les récompenses -Voter des idées -Commenter des idées -Appuyer une plainte
Responsable	<ul style="list-style-type: none"> -Soumettre une idée à un vote -Valider ou rejeter une idée -Répondre aux plaintes -Accéder aux données statistiques -Proposer des récompenses
Administrateur	<ul style="list-style-type: none"> -Gérer les comptes - Inviter un employé
Client	<ul style="list-style-type: none"> -Proposer une idée -Envoyer une plainte

2.2.1.2. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation permet d'exprimer les besoins des utilisateurs du système. La figure suivante illustre le diagramme de cas d'utilisation du SMI.

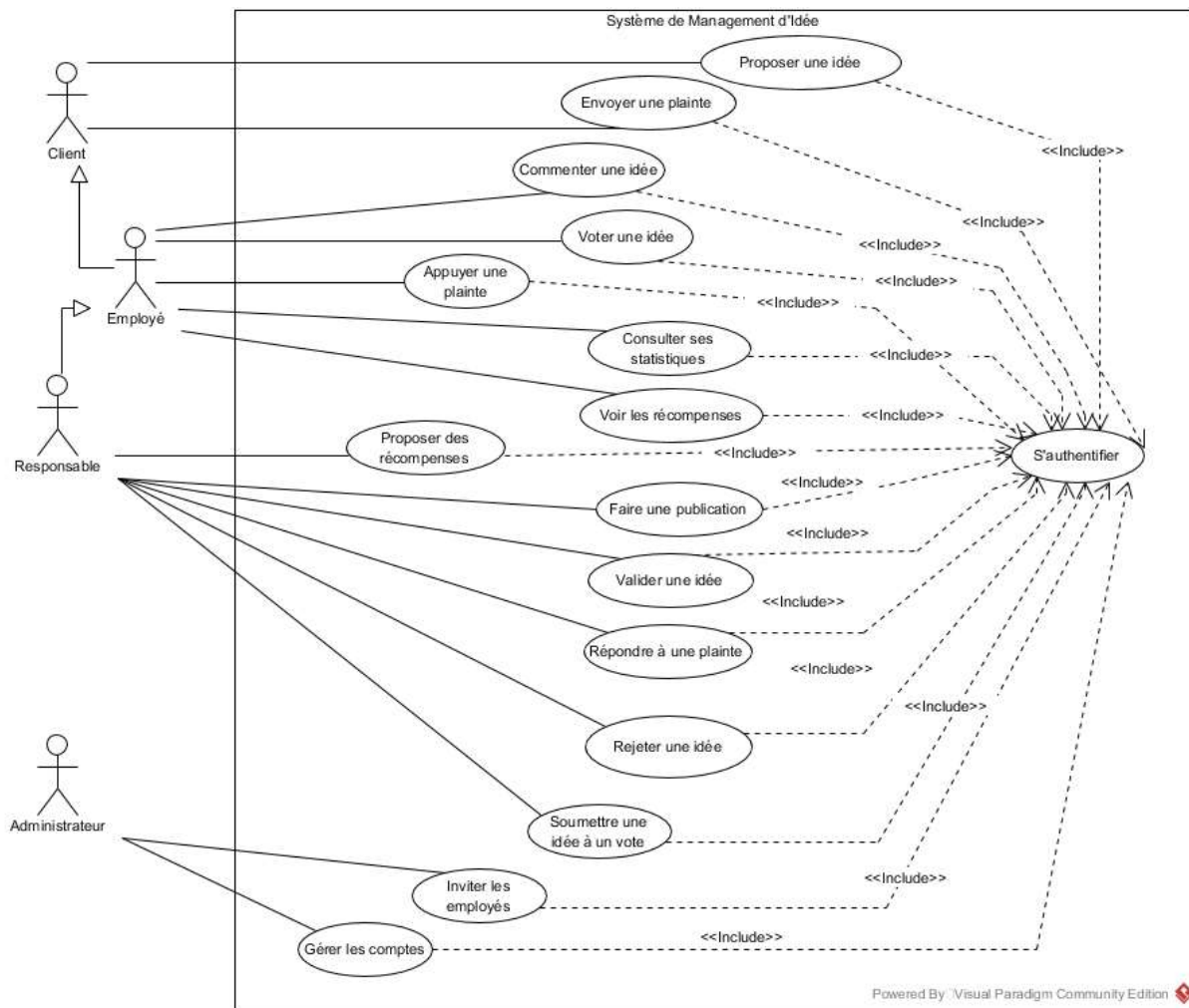


Figure 2: Digramme de cas d'utilisation du SMI

De l'analyse des fonctionnalités du système ressort le diagramme ci-dessus qui regroupe les relations entre les cas d'utilisation eux-mêmes et les relations entre les différents cas d'utilisation et les acteurs.

2.2.1.3. Description textuelle des cas d'utilisation

Pour une bonne conception du SMI, il faut la description de toutes les interactions qui auront lieu entre les utilisateurs et le système lors de son utilisation.

- Description textuelle du cas d'utilisation « **Proposer une idée** »
Interface Homme - Machine

Nouvelle idée

Objet

Contenu

Type

☐ Sous anonymat

Envoyer

Légende : ***** Zone de saisie

Sommaire d'identification

Titre : Proposer une idée

Résumé : Ce cas permet aux utilisateurs de faire des propositions d'idées

Acteurs : Employé ou Client

Date de Création : 20/12/2018

Date de mise à jour : 20/12/2018

Version : 1.0

Responsable : Emmanuel Assogba Balle

Préconditions : Acteur authentifié

Description des Scénarii

- **Scénario nominal**

1. L'acteur demande l'affichage du formulaire d'enregistrement des idées
2. Le système affiche le formulaire
3. L'acteur renseigne les informations demandées et valide
4. Le système vérifie la validité des informations renseignées
5. Le système enregistre l'idée dans la base de données
6. Le système renvoie le message d'enregistrement de l'idée avec succès

- **Scénario alternatif**

A1 : Un (ou plusieurs) champs non renseigné(s) ou invalide(s)

Le scénario A1 démarre au point 4 du scénario nominal.

Le point 5 devient « Le système indique à l'acteur que certains champs sont requis ou invalides »

Le scénario reprend au point 2 du scénario nominal

Postconditions : idée proposée

- Description textuelle du cas d'utilisation « **Envoyer une plainte** »

Interface Homme – Machine



Nouvelle plainte

Motif

Contenu

☐ Sous anonymat

Envoyer

Légende : ***** Zone de saisie

Sommaire d'identification

Titre : Envoyer une plainte

Résumé : Ce cas permet aux utilisateurs d'adresser une plainte à une entreprise

Acteurs : Employé ou Client

Date de Création : 20/12/2018

Date de mise à jour : 20/12/2018

Version : 1.0

Responsable : Emmanuel Assogba Balle

Préconditions : Acteur authentifié

Description des Scénarii

- **Scénario nominal**

1. L'acteur demande l'affichage du formulaire d'enregistrement des plaintes
2. Le système affiche le formulaire
3. L'acteur renseigne les informations demandées et valide
4. Le système vérifie la validité des informations renseignées
5. Le système enregistre la plainte dans la base de données
6. Le système renvoie le message d'envoi de la plainte avec succès

- **Scénario alternatif**

A1 : Un (ou plusieurs) champs non renseigné(s) ou invalide(s)

Le scénario A1 démarre au point 4 du scénario nominal.

Le point 5 devient « Le système indique à l'acteur que certains champs sont requis ou invalides »

Le scénario reprend au point 2 du scénario nominal

Postconditions : Plainte envoyée

- Description textuelle du cas d'utilisation « **Voter une idée** »

Interface Homme – Machine

The screenshot displays a web interface for voting on ideas. At the top, there are four tabs: 'Idées proposées', 'Espace vote' (which is active and highlighted in blue), 'Idées validées', and 'Idées rejetées'. Below the tabs, a proposal card is shown. The card has a timestamp '20 avril 2019 08 : 02' in the top right corner. The title of the proposal is 'Lancement 5G au Bénin'. The description reads: 'Etablir un système de gestion des investissements en Recherche & Développement similaire à d'autres systèmes dans le secteur ...'. Below the description, there are two buttons: a blue one labeled '5 Approuvées' and a red one labeled '2 rejetées'. At the bottom left of the card, there is a blue smiley face icon and the name 'Jack Tossou'.

Titre : Voter une idée

Résumé : Ce cas permet aux utilisateurs de voter des idées qu'ils trouvent innovantes

Acteurs : Employé ou Responsable

Date de Création : 20/12/2018

Date de mise à jour : 20/12/2018

Version : 1.0

Responsables : Emmanuel Assogba Balle

Préconditions : Acteur authentifié

Description des Scénarii

- **Scénario nominal**

1. Le système affiche la page des idées
2. L'utilisateur accède à l'espace vote
3. L'utilisateur choisit l'idée et clique sur « approuver » ou « rejeter »
4. Le système enrégistre le vote et incrémente le compteur du nombre d'approbation ou le nombre de rejet selon le choix effectué par l'utilisateur pour cette idée

- **Scénario alternatif**

A1 : L'utilisateur a déjà voté cette idée

Le scénario A1 démarre au point 3 du scénario nominal.

Le point 4 devient « Le système conserve le vote de l'utilisateur. »

Postconditions : Vote effectué

2.2.2. Les classes

2.2.2.1. Dictionnaire de données du SMI

Tableau 2: Dictionnaire de données

Termes	Attributs	Définitions	Types
	idUtilisateur	Identifiant d'un utilisateur	N
	Code	Code d'un utilisateur	AN
	Nom	Nom d'un utilisateur	A

Utilisateurs	Prenom	Prénom d'un utilisateur	A
	Contact	Numéro de téléphone d'un utilisateur	N
	Email	Adresse mail d'un utilisateur	AN
	Password	Mot de passe d'un utilisateur	AN
	dateNaiss	Date de naissance d'un utilisateur	Date
	Adresse	Adresse d'un utilisateur	A
	Sexe	Sexe d'un utilisateur	
	Fonction	Fonction occupée par un utilisateur dans la vie professionnelle	A
	Avatar	Photo de profil d'un utilisateur	A
Reagir	idReaction	Identifiant d'une réaction (vote ou commentaire)	N
	Date	Indique la date d'enregistrement d'une réaction	Date
Voter	Statut	Indique si le vote est une approbation ou un rejet de l'idée	Bool
Commenter	Contenu	Contenu ou message d'un commentaire	A
Idees	idIdee	Identifiant d'une idée	N
	Titre	Titre d'une idée	A
	Contenu	Description d'une idée	A
	Statut	Indique si une idée est valide ou non	Bool
	Anonyme	Indique si une idée doit être présentée comme anonyme ou non	Bool
	Type	Type d'une idée (suggestion, conseil, recommandation, etc.)	A
	dateCreation	Indique la date de création d'une idée	Date
	Etat	Indique pour une idée si elle est en attente de vote, si le vote est en cours ou si le vote est clôturé	N
	Priorite	Permet d'associer une priorité à une idée	A
	Rappel	Permet d'associer un rappel à une idée	A
Type	idType	Identifiant d'un type	A
	Libelle	Libellé du type	A
	Statut	Indique si un type est actif ou non	Bool
	VuPar	Indique les utilisateurs pour lesquels ce type sera proposé	
Plaintes	idPlainte	Identifiant d'une plainte	N
	Contenu	Indique le contenu de la plainte	A
	Statut	Indique si le plaignant a été satisfait ou non	Bool

	Anonyme	Indique si une plainte doit être présentée comme étant anonyme ou non	Bool
	dateCreation	Indique la date de création de la plainte	Date
	Priorite	Permet d'associer une priorité à une plainte	A
	Rappel	Permet d'associer un rappel à une idée	A
Motifs	idMotif	Identifiant d'un motif	N
	Libelle	Libellé d'un motif de plainte	A
Repondre	idReponse	Identifiant d'une réponse	N
	Contenu	Contenu d'une réponse à une plainte	A
Entreprises	idEntreprise	Identifiant d'une entreprise	N
	Code	Code d'identification d'une entreprise	AN
	Nom	Nom d'une entreprise	A
	Adresse	Adresse d'une entreprise	AN
	Email	Adresse mail d'une entreprise	AN
	Contact	Numéro de téléphone d'une entreprise	AN
Recompenses	idRecompense	Identifiant d'une récompense	N
	Criteres	Critère d'obtention d'une récompense	A
	Libelle	Indique le libellé de la récompense à obtenir	A

Légende : **A** pour **Alphabétique**, **N** pour **Numérique**, **AN** pour **Alphanumérique**, **Bool** pour **Booléen**

2.2.2.2. Règles du SMI

Les règles de domaine ci-après régissent le fonctionnement de notre système :

RD1 : un utilisateur peut être soit un client, un employé, un responsable ou l'administrateur

RD2 : un utilisateur appartient à une entreprise

RD3 : une entreprise a plusieurs utilisateurs

RD4 : un client ou un employé poste un ou plusieurs avis

RD5 : un avis est posté par un seul client ou employé

RD6 : un avis est soit une idée soit une plainte

RD7 : un est adressé à une et une seule entreprise

RD8 : une entreprise reçoit plusieurs avis

RD9 : une plainte à un seul motif

RD10 : plusieurs plaintes peuvent avoir le même motif

RD11 : une idée est d'un type donné

RD12 : plusieurs idées peuvent être du même type

RD13 : une entreprise propose plusieurs récompenses

RD14 : une récompense est proposée par une seule entreprise

RD15 : un responsable répond à une plainte

RD16 : un ou plusieurs responsable envoie une ou plusieurs réponses pour une même plainte

RD17 : une réponse est envoyée par un seul responsable pour une plainte donnée

RD18 : un employé réagit par rapport à un avis

RD19 : une réaction est soit un commentaire soit un vote

RD 20 : un employé commente plusieurs fois une même idée

RD21 : un commentaire vient d'un seul employé

RD22 : un employé vote une idée qu'une seule fois

2.2.2.3. Diagramme de classes

Le diagramme de classe décrit clairement la structure du système en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets. Les classes issues de l'analyse du système sont regroupés dans le diagramme ci-dessous avec les relations qui les lient.

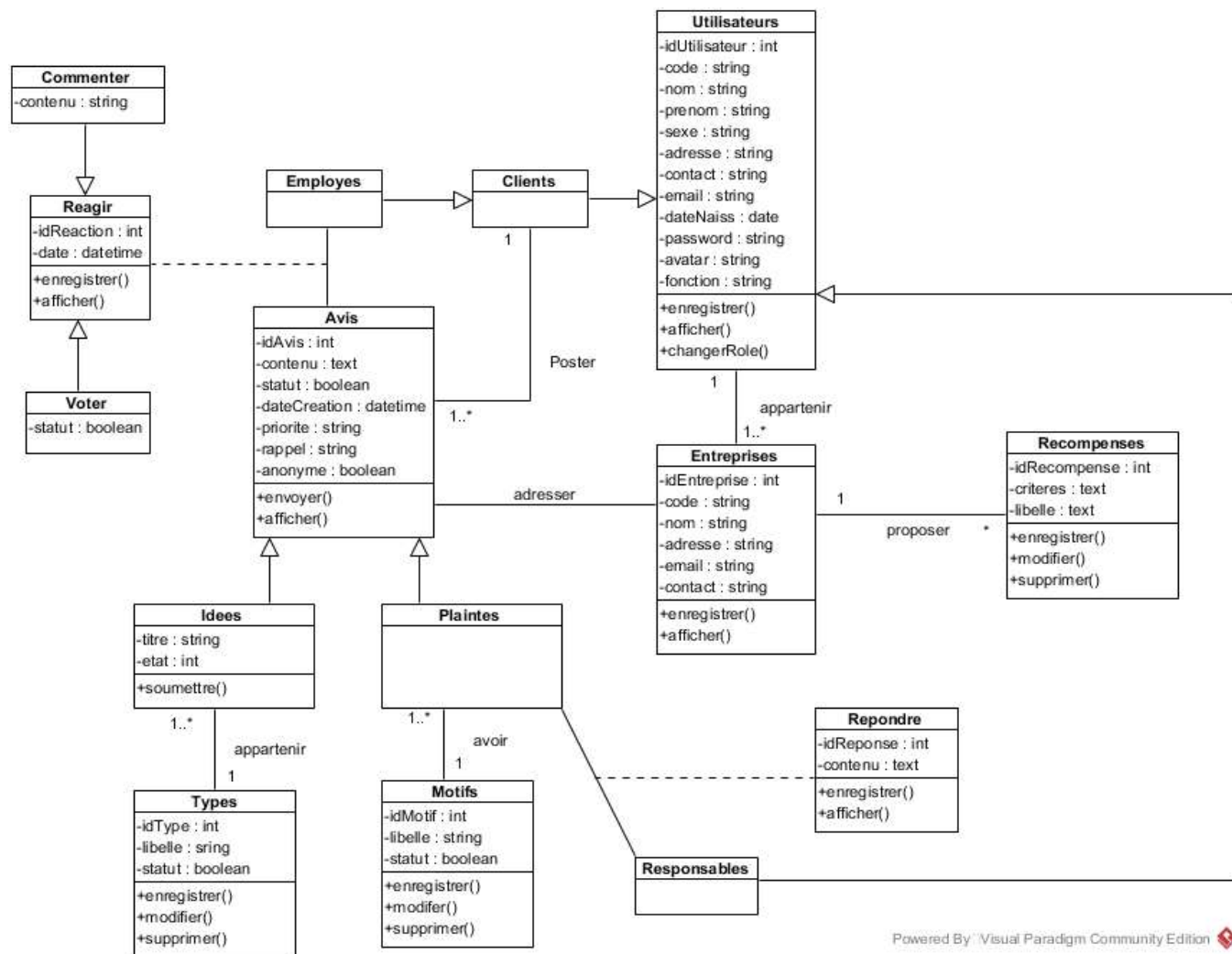


Figure 3: Diagramme de classes du SMI

2.2.3. Diagramme d'objets

Un diagramme d'objet permet d'avoir une vue d'ensemble du système. Il représente une instance spécifique du diagramme de classe à un moment précis.

Voici illustrés ci-dessous quelques diagrammes d'objet du SMI.

○ Diagramme d'objet des classes Idees,Types et Employes

Le diagramme d'objet ci-après illustre qu'un « employé » poste une ou plusieurs Idées.

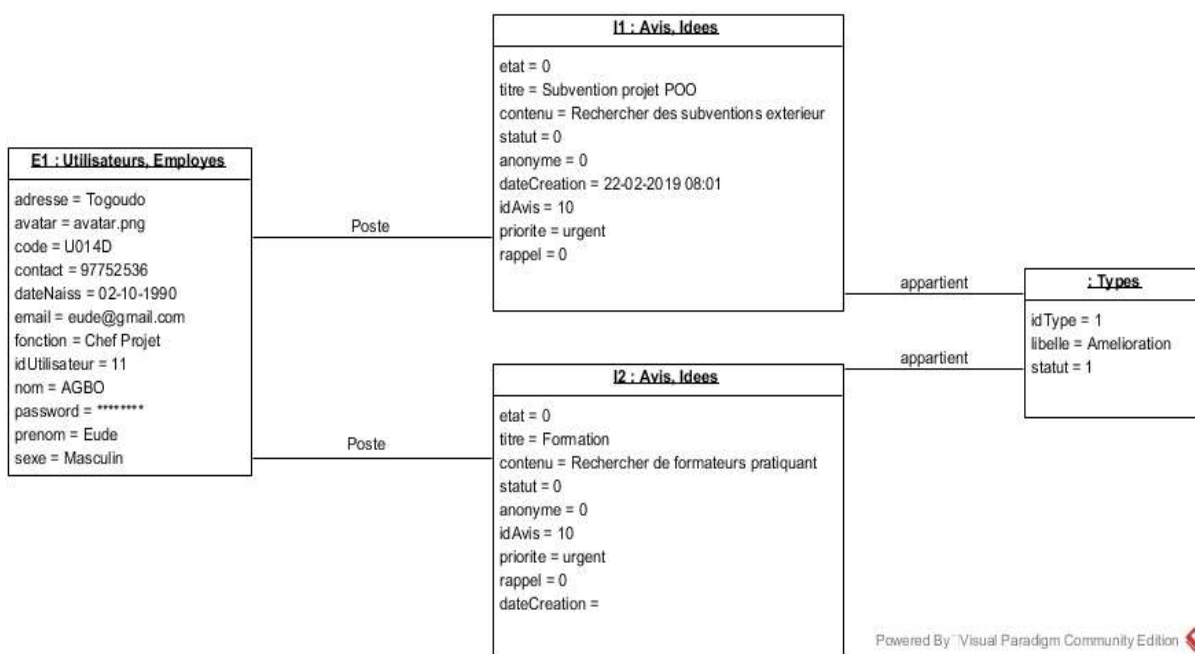
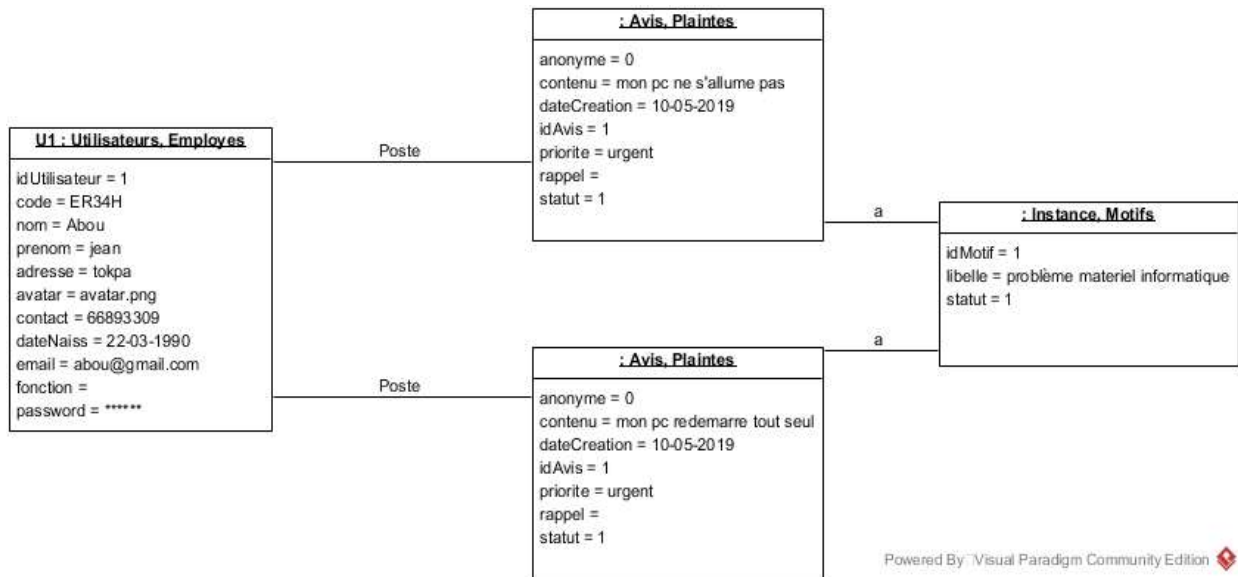


Figure 4 : Diagramme d'objets des classes Idees,Types et Employes

- Diagramme d'objet des classes Plaintes,Motifs et Employes Le diagramme d'objet ci-après illustre qu'un utilisateur envoie une ou



plusieurs plaintes pour un motif bien précis.

Figure 5: Diagramme d'objets des classes Plaintes,Motifs et Employes

○ Diagramme d'objet des classes Employes, Idees, Types et Voter

Le diagramme d'objet ci-après illustre qu'un utilisateur ayant le rôle « employé » vote une idée en cours de vote.

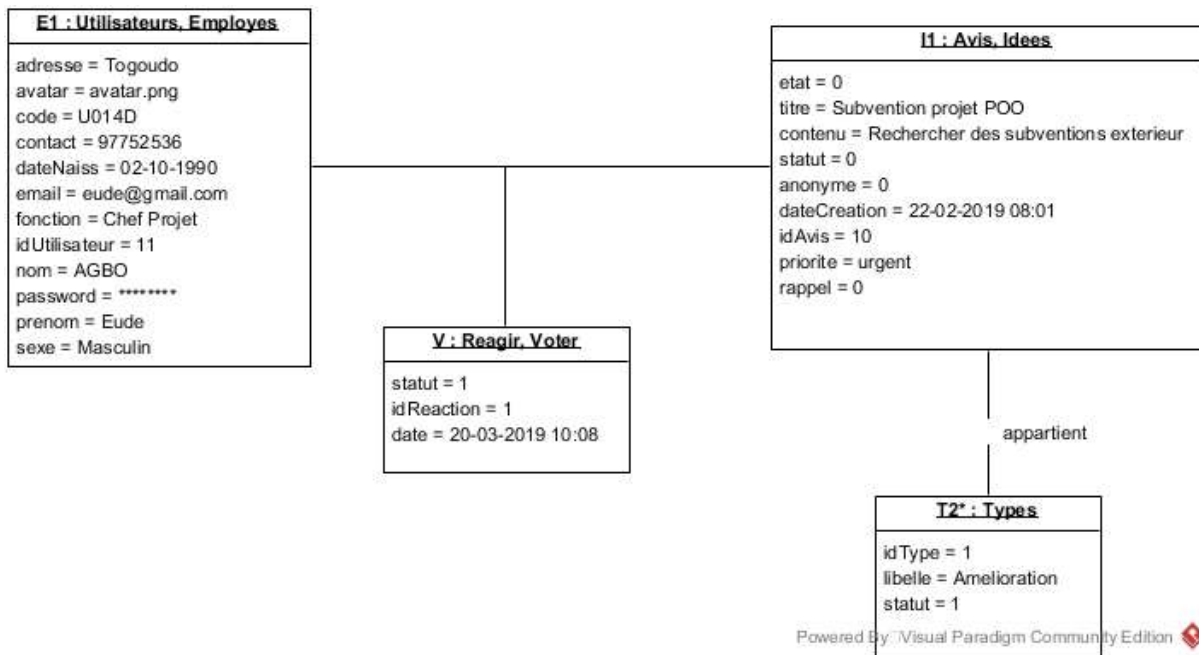


Figure 6: Diagramme d'objets des classes Utilisateurs, Roles, Idees, Types, Reagir et Voter

2.2.4. Diagrammes de séquences

Un diagramme de séquence est diagramme d'interaction. Il décrit comment et dans quel ordre plusieurs objets fonctionnent ensemble.

Les diagrammes ci-dessous donnent une représentation graphique des interactions décrites plus haut entre utilisateurs et système.

- Diagramme de séquences du cas « Proposer une idée »

Le diagramme ci-dessous illustre le processus d'exécution du Cas « Proposer une idée ». Ainsi, pour proposer une idée, l'utilisateur demande au système l'affichage du formulaire de proposition d'idées puis renseigne les

informations demandées ; ensuite le système vérifie si les informations fournies sont correctes, dans quel cas il les envoie vers la Base de données (BD) pour la sauvegarde ; puis le système retourne à l'utilisateur un message de succès ou d'échec selon que la sauvegarde a été effective ou non.

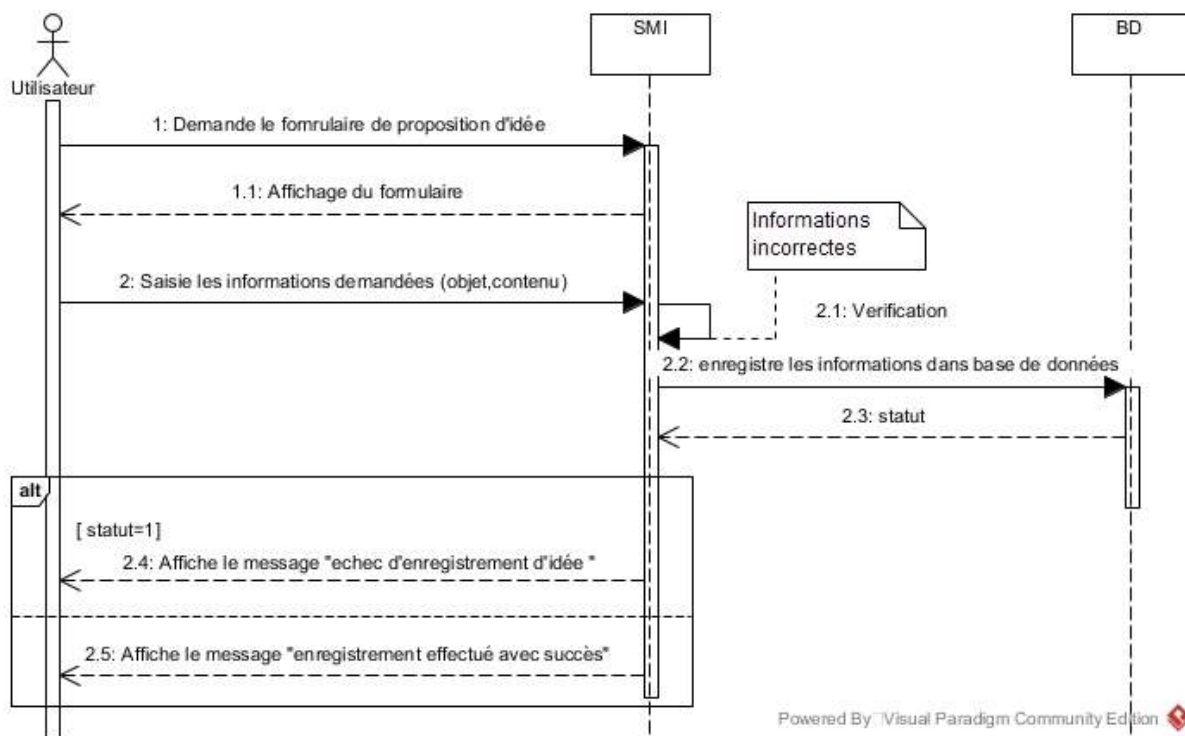


Figure 7: Diagramme de séquences du cas « Proposer une idée »

- Diagramme de séquences du cas « Envoyer une plainte »

Le diagramme ci-dessous illustre le processus d'exécution du Cas « Envoyer une plainte ». Ainsi pour envoyer une plainte l'utilisateur demande au système l'affichage du formulaire d'envoi de plainte puis renseigne les informations demandées ; ensuite le système vérifie si les informations fournis sont correctes, dans quel cas il les envoie vers la BD pour la sauvegarde ; puis le système retourne

à l'utilisateur un message de succès ou d'échec selon que la sauvegarde a été effective ou non.

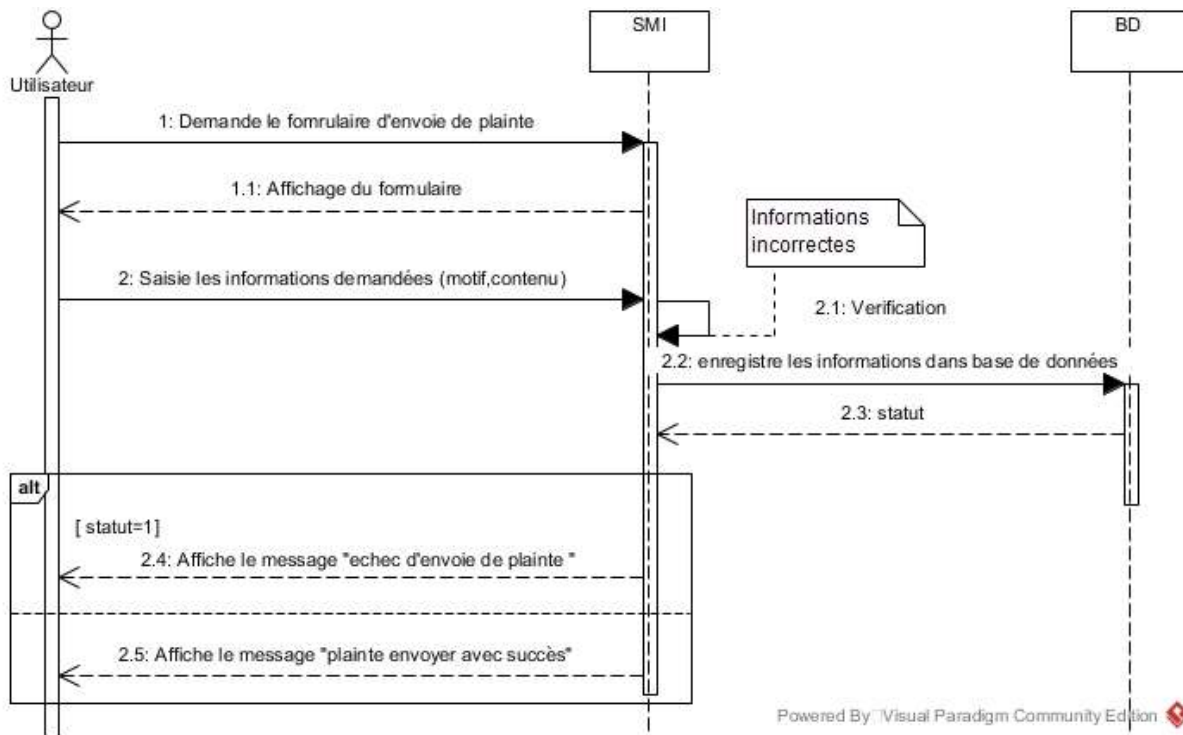


Figure 8: Digramme de séquences du cas « Envoyer une plainte »

- Diagramme de séquences du cas « Voter une idée »

Le diagramme ci-dessous illustre le processus d'exécution du Cas « Voter une idée ». Ainsi pour voter une idée l'utilisateur demande au système l'affichage de la page des idées puis accède à l'espace vote ; ensuite, il recherche l'idée à voter. Si l'utilisateur avait déjà voté cette idée alors il ne pourrait effectuer aucune action pour changer ce vote. Par contre si l'utilisateur n'avait pas encore voté pour cette idée alors il aurait le choix entre "approuver l'idée" et "rejeter l'idée ". Suite au choix effectué le compteur correspondant au nombre d'idées approuvées se verra incrémenter ou celui correspondant au nombre d'idées rejetées se verra décrémenter.

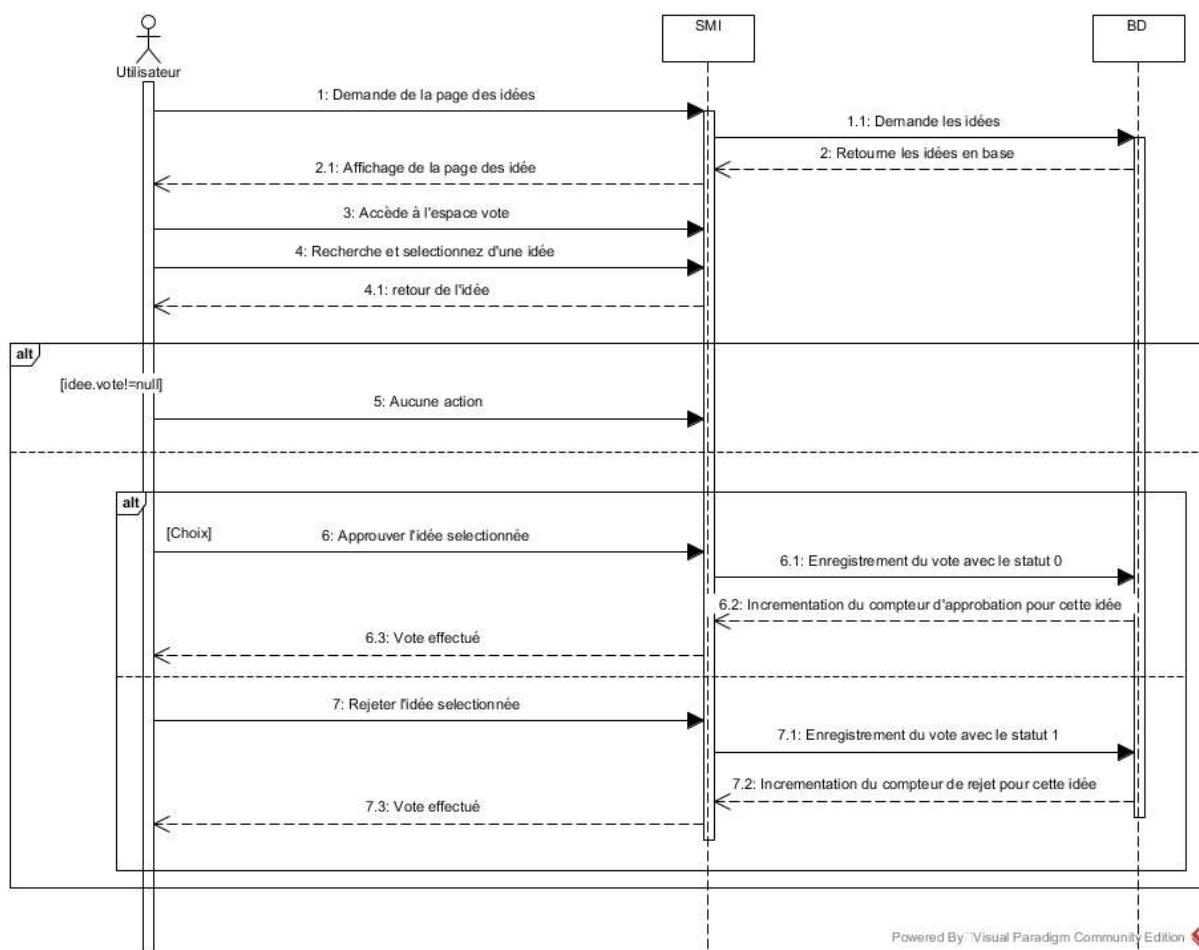


Figure 9 : Diagramme de séquences du cas « Voter une idée »

2.2.5. Diagrammes d'état transition

Un diagramme d'état transition est diagramme comportemental qui représente les transitions entre divers objets. Les états ici correspondent aux différentes combinaisons d'informations qu'un objet peut contenir. Ce diagramme présente un automate à état finis. Il permet ainsi donc de décrire les changements d'état d'un objet ou d'un composant.

Les diagrammes d'état transition ci-dessous expliquent précisément de quelle façon les actions sont exécutées lors des transitions entre les différents états ; ces

états correspondent aux différentes situations se produisant lors de la réalisation de chaque cas d'utilisation.

Voici illustré ci-dessous quelques diagrammes d'état transition du SMI.

- Diagramme d'état transition du cas « Proposer une idée »

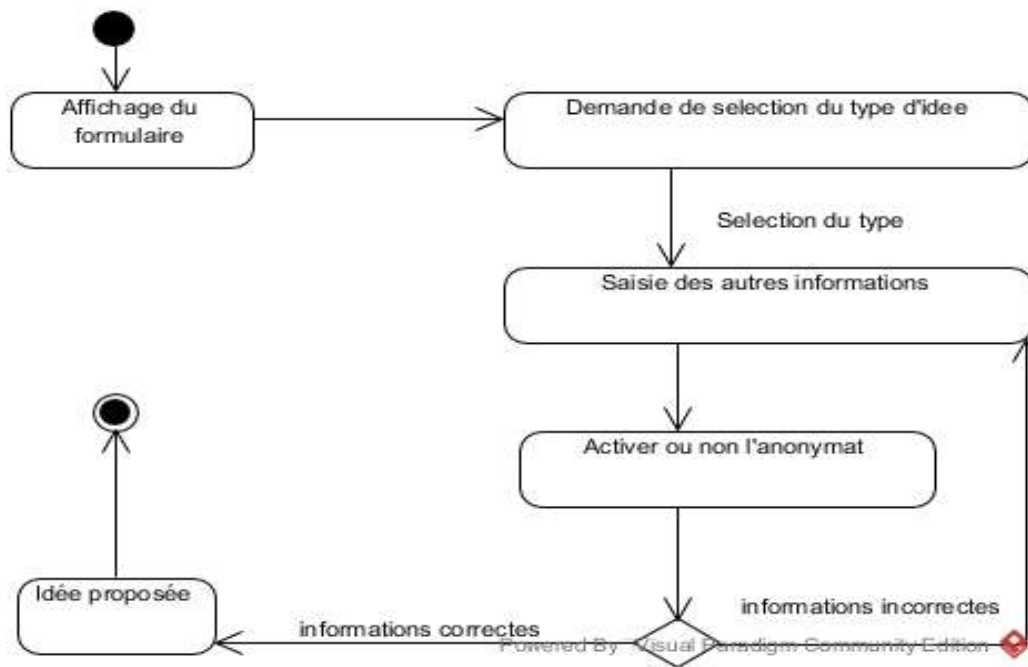


Figure 10: Diagramme d'état transition du cas « Proposer une idée »

- Diagramme d'état transition du cas « Envoyer une plainte »

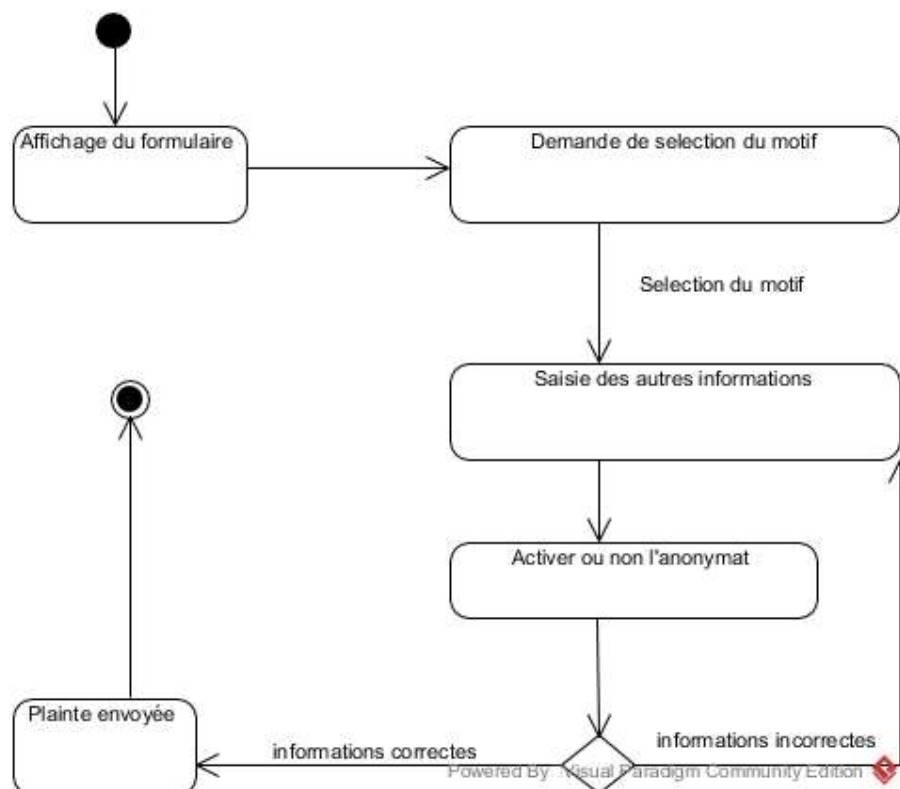


Figure 11: Diagramme d'état transition du cas « Envoyer une plainte »

- Diagramme d'état de transition du cas « Voter une idée »

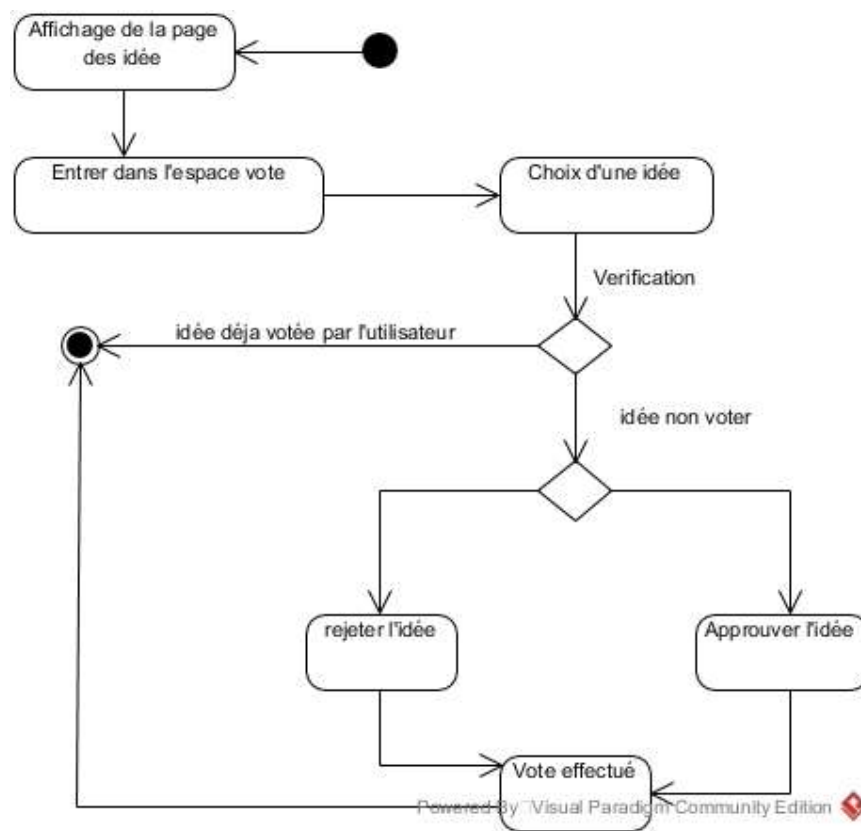


Figure 12: Diagramme d'état de transition du cas « Voter une idée »

Chapitre III : Implémentation du système

3.1. Outils de développement utilisés

❖ HTML

Le **HTML** (HyperText Markup Language) a été créé pour les besoins du web, par Tim Bernes Lee physicien au Conseil Européens pour la Recherche Nucléaire (CERN). Depuis il n'a cessé d'évoluer et en est à sa cinquième version (Html5). Très pratique, le Html5 nous a permis de concevoir nos pages web.

❖ CSS

Le **CSS** (Cascading Style Sheets) se dit en français feuilles de styles en cascades. Ce langage décrit la présentation des documents Html. Introduit par le World Wide Web Consortium(W3C) en 1990, CSS est bien pris en charge par les navigateurs justifiant ainsi notre choix vers ce langage pour la présentation des pages web.

❖ TypeScript

Le **TypeScript** est un langage de programmation orienté client développé par Microsoft. Il a une structure similaire au JavaScript qui est son langage de compilation. C'est le langage de programmation de toutes les versions d'Angular. Il offre beaucoup plus de possibilité. Il est plus flexible et permet en ce sens l'utilisation des classes et d'interfaces.

❖ Bootstrap

Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que

des extensions JavaScript en option. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement GitHub.

❖ Laravel

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet. Il permet grâce à ces nombreuses fonctionnalités, d'accentuer davantage l'effort sur la conception du système.

❖ Angular

Angular est un Framework frontend écrit en TypeScript et permettant de réaliser simplement des interfaces web riches et dynamiques.

Plus qu'une bibliothèque, la solution **Angular** propose un cadrage complet s'articulant autour de solides patrons de conceptions (design pattern). Il est basé sur le concept de l'architecture MVC (Model View Controller) qui permet de séparer les données, les vues et les différentes actions que l'on peut effectuer.

❖ APACHE HTTP SERVER

APACHE HTTP SERVER est un serveur créé et maintenu au sein de la fondation Apache. C'est le serveur HTTP le plus populaire du World Wide Web. Il sera utile pour intégrer les pages PHP.

❖ SGBD (Système de Gestion de Base de Données)

Pour pouvoir mieux gérer les données et les utilisateurs du futur système à mettre en place, le besoin d'un système de gestion s'est vite fait ressentir. La gestion de la base de données se fait grâce à un système appelé MySQL.

MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde. L'une des plus grandes raisons qui nous motive à l'utiliser est que MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multithread et multi-utilisateur.

❖ MVC

L'architecture *Modèle/View/Contrôleur* (MVC) est une façon d'organiser une interface graphique d'un programme. Elle consiste à distinguer trois entités distinctes qui sont, le *modèle*, la *vue* et le *contrôleur* ayant chacun un rôle précis dans l'interface.

Le **modèle** contient les données manipulées par le programme. Il assure la gestion de ces données et garantit leur intégrité. Dans le cas typique d'une base de données, c'est le modèle qui la contient.

La **vue** fait l'interface avec l'utilisateur. Sa première tâche est d'afficher les données qu'elle a récupérée auprès du modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, boutons, ...). Ses différents événements sont envoyés au contrôleur.

Le **contrôleur** est chargé de la synchronisation du modèle et de la vue. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle et ensuite avertit la vue que les données ont changé pour que celle-ci se mette à jour.

3.2. Sécurité de l'application

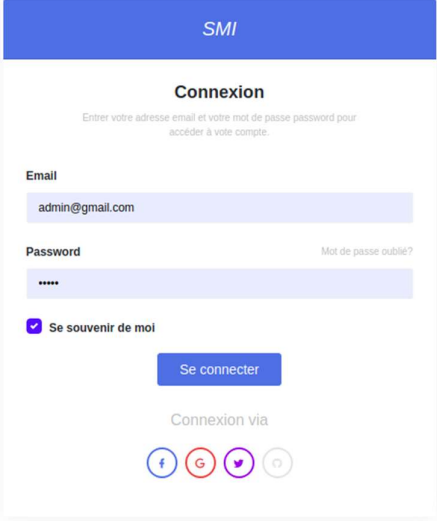
Compte tenu de l'importance et de la sensibilité des données traitées, il est nécessaire de s'assurer de la sécurité du système en prenant certaines mesures :

- L'accès à l'application requiert le renseignement d'un identifiant et d'un mot de passe permettant d'identifier l'acteur ;
- Les mots de passes sont cryptés grâce à l'algorithme de chiffrement **bcrypt** avant d'être stockés dans la base de données ;
- Les fonctionnalités réservées aux utilisateurs ayant un certain privilège sont cachées aux autres utilisateurs de sorte qu'ils en ignorent même l'existence ;
- Toutes les requêtes entrantes et sortantes du serveur passent par des middlewares qui sont chargés de vérifier si l'utilisateur possède le droit d'accéder à la page qu'il demande ; s'il ne les possède pas, il en est informé et l'accès lui est refusé ;
- L'utilisation du JWT (Json Web Token) pour encrypter les données sensibles échangées entre le client et le server ;

3.3. Travaux réalisés

Quelques captures qui montrent à quoi ressemble la version démo de notre SMI.

❖ Interface de connexion



The screenshot shows a web interface for a system named 'SMI'. At the top is a blue header with the text 'SMI'. Below it is a section titled 'Connexion' (Login). Under this title is a subtitle: 'Entrez votre adresse email et votre mot de passe password pour accéder à votre compte.' (Enter your email address and your password to access your account). There are two input fields: 'Email' with the value 'admin@gmail.com' and 'Password' with masked characters '*****'. To the right of the password field is a link 'Mot de passe oublié?' (Forgot password?). Below the password field is a checkbox labeled 'Se souvenir de moi' (Remember me) which is checked. A blue button labeled 'Se connecter' (Connect) is positioned below the checkbox. Underneath the button is the text 'Connexion via' (Login via) followed by four circular icons representing social media or third-party login options: Facebook, Google, Twitter, and a generic icon. At the bottom of the interface, there is a line of text: 'Vous êtes une entreprise ? Vous n'avez pas de compte. Rejoignez nous' (Are you a company? You don't have an account. Join us).

Figure 13: Interface de connexion

Description : L'utilisateur entre son email et son mot de passe afin de pouvoir se connecter à la plateforme.

❖ Interface de suivi des idées pour le rôle employé

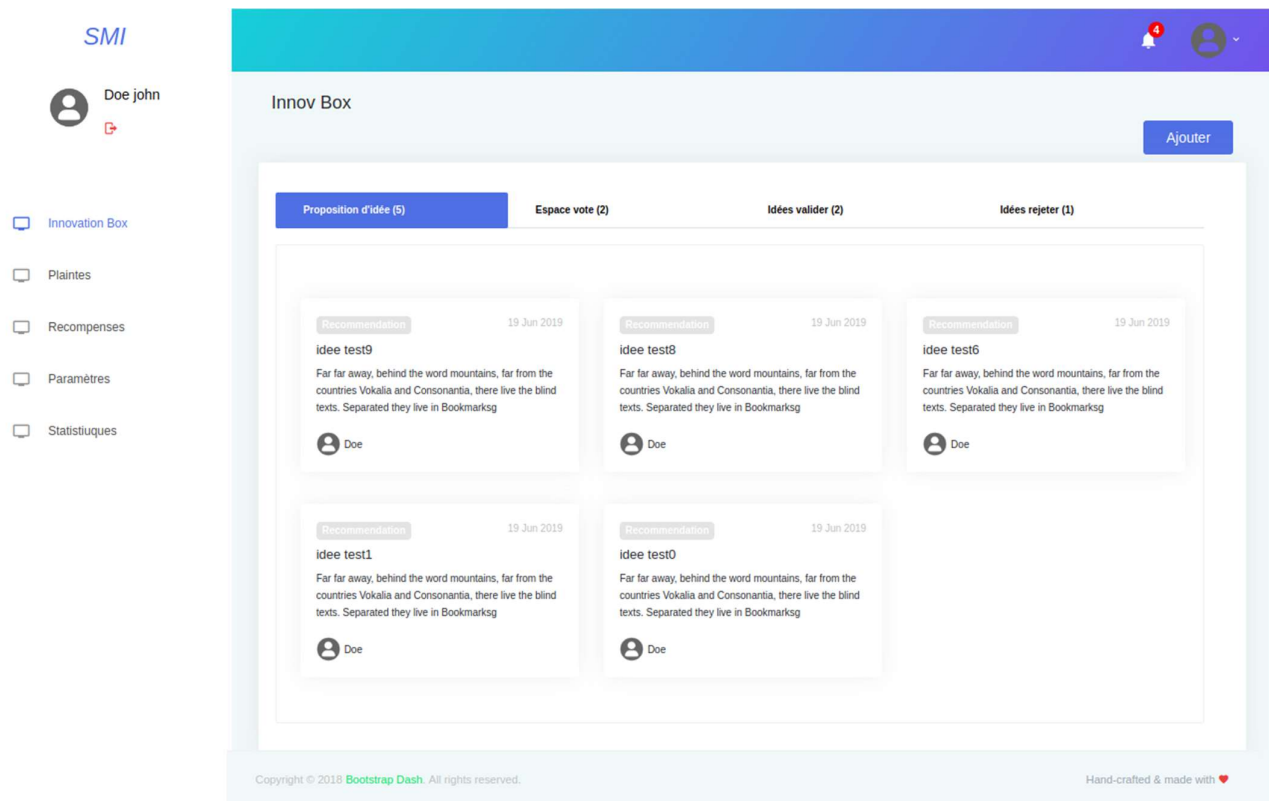


Figure 14 : Interface de suivi des idées pour le rôle **employé**

Description : L'utilisateur ici a accès à toutes les idées soumises par lui et par ses collaborateurs. Il suit en temps réel l'évolution de toutes les idées et peut également en proposer de nouvelle.

❖ Interface de suivi des plaintes pour le rôle **employé**

SMI

Doe john

Innovation Box

Plaintes

Recompenses

Paramètres

Statistiques

Plaintes

Ajouter

Mes plaintes (20)

Toutes les plaintes (20)

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Personnel incompétent

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Coût élevé du service

19 Jun 2019

Plainte Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts. Separated they live in Bo

0 réponses

Copyright © 2018 Bootstrap Dash. All rights reserved.

Hand-crafted & made with

Figure 15 : Interface de suivi des plaintes pour le rôle **employé**

Description : L'utilisateur ici a accès à ses propres plaintes et celles des autres utilisateurs. Il peut également ajouter de nouvelle plainte et aussi voir les réponses fournir par rapport à chaque plainte.

❖ Interface de suivi des statistiques pour le rôle **employé**

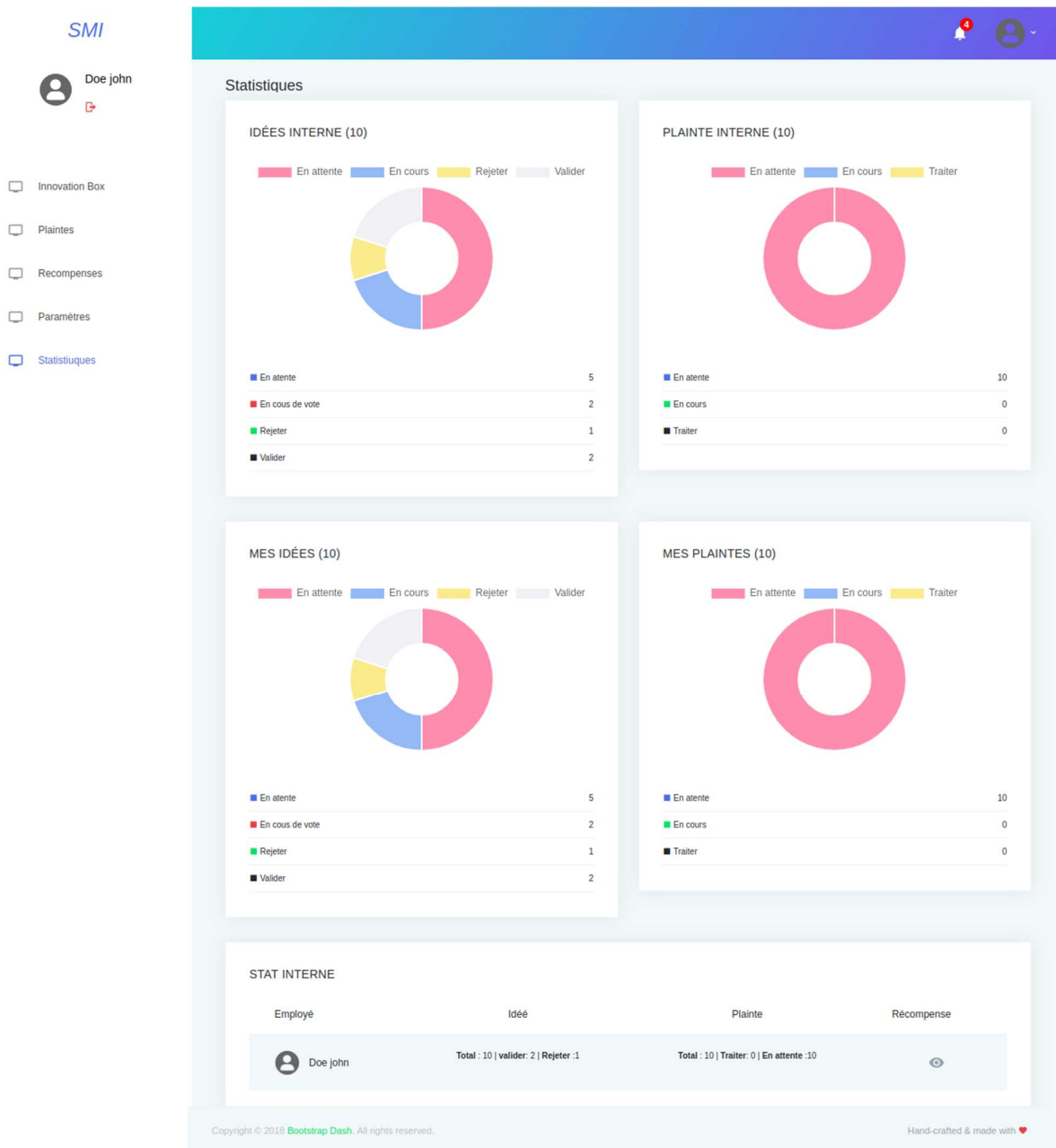


Figure 16: Interface de suivi des statistiques pour le rôle **employé**

Description : L'utilisateur peut voir ses statistiques complètes, sur la base de quoi il peut donc juger de ses performances. Il a aussi un aperçu pas très détaillé des performances de ses collègues.

CONCLUSION

Ce stage académique effectué à Digicix nous a permis d'une part de mettre en application les connaissances théoriques reçues au cours de notre formation de trois (03) ans en Analyse Informatique et Programmation (AIP) et d'autre part d'avoir un aperçu des réalités dans le monde professionnel.

Face aux contraintes autour du système de boîte à idées physique actuellement utilisé dans les entreprises au Bénin, nous avons développé un SMI qui est une virtualisation de ce système. Son but est d'aider les organismes rencontrant des difficultés dans leur fonctionnement et dans l'atteinte de leurs objectifs. C'est pour cela qu'il est basé sur l'innovation participative, une démarche de management structurée qui vise à stimuler et à favoriser l'émission, la mise en œuvre et la diffusion d'idées par l'ensemble du personnel d'un organisme.

Ce nouveau système de boîte à idées permet, entre autres, de poster des idées, d'envoyer des plaintes, d'organiser les votes, d'élaborer des statistiques basées sur de nombreux paramètres. Le travail d'analyse et de conception ayant aboutie au SMI est basé sur le langage de modélisation UML. L'utilisation de diverses méthodes de cryptages comme le bcrypt ou le JWT à divers niveaux pour garantir la confidentialité des données et assurer la sécurité du système.

PERSPECTIVE

Tout le système étant appelé à évoluer dans le temps, des améliorations peuvent être apportées à ce travail afin de le rendre plus utile. On pourrait penser au développement d'une plateforme mobile spécialement pour permettre aux clients d'interagir beaucoup plus facilement avec l'entreprise. Cette plateforme complémentaire intégrerait un fil d'actualité des nouveaux produits ou services de l'entreprise pour tenir informé les clients, avec la possibilité de donner leur avis directement par rapport tel produit ou tel service. Cette plateforme intégrera également un module de récompense orienté client pour primer les meilleures idées ou critiques. La nouvelle version de notre système prendra aussi en compte les insuffisances qui auraient été relevées dans l'utilisation de la première version.

BIBLIOGRAPHIE & WEBOGRAPHIE

- 4 RAISONS D'UTILISER LA BOÎTE À IDÉES 2.0 DANS VOTRE ENTREPRISE, MAENA BOUGHEZALA (Disponible à <https://www.oipsolutions.com/fr/4-raisons-dutiliser-boite-a-idees-2-0-entreprise/>, consulté le 02 novembre 2018)
- Olivier Dutel, Comprendre les intérêts de la boîte à idée (Disponible à <http://les-temps-changent.com/comprendre-les-interets-de-la-boite-a-idee/>, consulté le 02 novembre 2018)
- Mettre en place une boîte à idée dans l'entreprise (Disponible à <https://www.petite-entreprise.net/P-2592-81-G1-mettre-en-place-une-boite-a-idee-dans-l-entreprise.html>, consulté le 03 novembre 2018)
- Comment réinventer la boîte à idée dans l'entreprise (Disponible à https://lentreprise.lexpress.fr/rh-management/comment-reinventer-la-boite-a-idees-dans-l-entreprise_1523939.html, consulté le 10 novembre 2018)
- Boîtes à idées avantages et inconvénients (Disponible à <https://www.etudier.com/dissertations/Boites-%C3%A0-Id%C3%A9es-Avantages-Et-Inconv%C3%A9nients/79496929.html>, consulté le 15 novembre 2018)
- Les intérêts des boîtes à idées : amélioration continue, innovation, cohésion sociale... (Disponible à <http://www.motivationfactory.com/blog/actualite/interet-boite-a-idees-amelioration-continue-innovation-cohesion-sociale>, consulté le 20 décembre 2018)
- Taylor Otwell, *Documentation officielle du framework Laravel*. (Disponible à <https://laravel.com/docs/5.7> , consulté le 20 décembre 2018)
- https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade , consulté le 27 décembre 2018
- Documentation officielle du framework Angular. (Disponible à <https://angular.io/docs>, consulté le 10 janvier 2019)

- Openclassroom, Développez des applications web avec angular (Disponible à <https://openclassrooms.com/fr/courses/4668271-developpez-des-applications-web-avec-angular>, consulté le 10 janvier 2019)
- Connor Leech , Construisez l'authentification dans votre API Laravel avec JWT (Disponible à <https://medium.com/employbl/build-authentication-into-your-laravel-api-with-json-web-tokens-jwt-cd223ace8d1a>, consulté le 12 janvier 2019)
- Jason Watmore's ,Tutoriel d'authentification avec angular 6 et JWT (Disponible à <http://jasonwatmore.com/post/2018/05/23/angular-6-jwt-authentication-example-tutorial>, consulté le 12 janvier 2019)
- Fred Lin , Uploader un fichier avec laravel 5 et angular 2 (Disponible à <http://flint.sg/2017/01/04/file-upload-with-angular-2-and-laravel-5/>, consulté le 15 janvier 2019)
- Daouda Diallo, Angular ACL (Disponible à <https://medium.com/code-divoire/angular-acl-9d1ce013312e>, consulté le 17 janvier 2019)
- https://fr.m.wikipedia.org/wiki/Hypertext_Markup_Language
- <https://www.scriptol.fr/programmation/typescript.php> , consulté le 30 janvier 2019
- <https://www.irif.fr/~carton/Enseignement/InterfacesGraphiques/MasterInfo/Cours/Swing/mvc.html>, consulté le 20 mai 2019

ANNEXES

1. Autres Captures d'écran du SMI

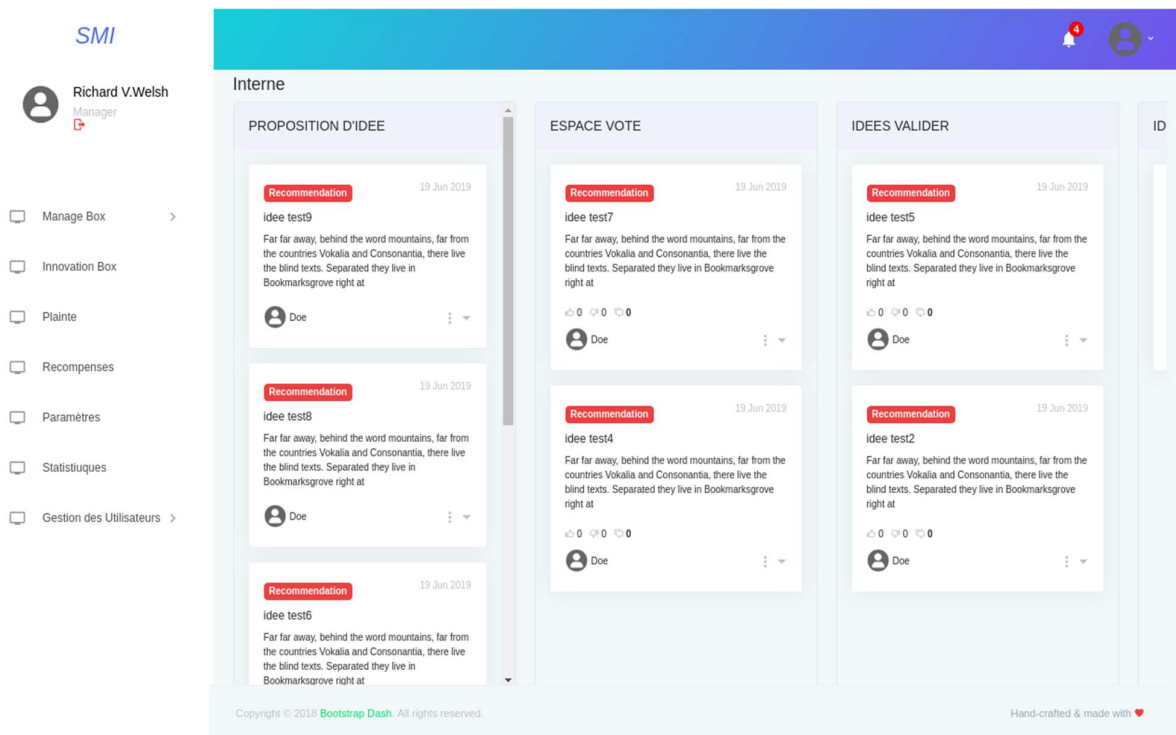


Figure 17 : Interface de gestion des idées pour les rôles **admin** et **responsable**

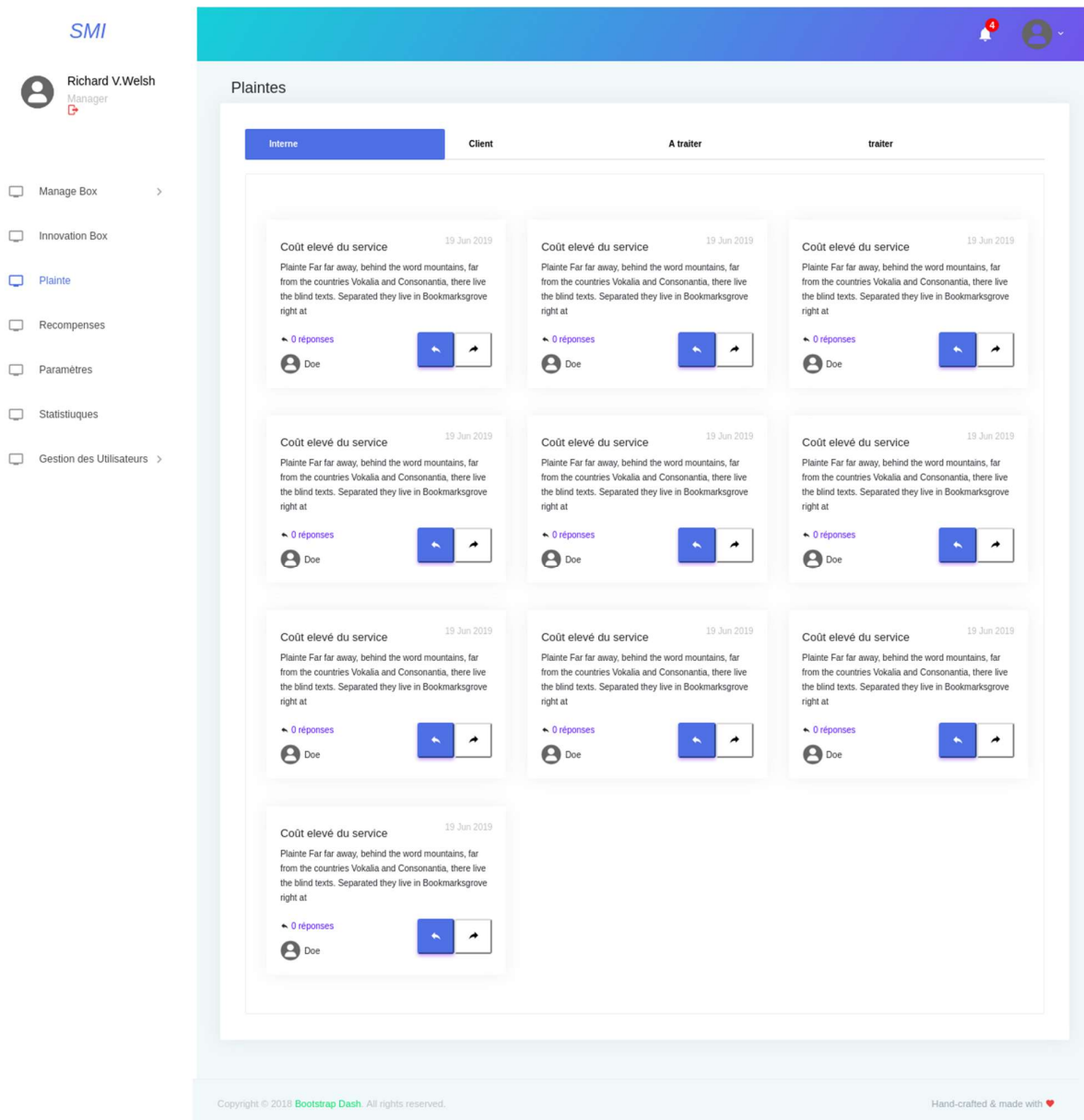


Figure 18 : Interface de gestion des plaintes pour les rôles admin et responsable

2. Codes Source

- Fichier de routing pour les api : api.php

```
<?php
```

```
use Illuminate\Http\Request;
```

```
/*
```

```
|-----
```

```
-----
```

```
| API Routes
|-----
|
| Here is where you can register API routes for your
application. These
| routes are loaded by the RouteServiceProvider within a group
which
| is assigned the "api" middleware group. Enjoy building your
API!
|
*/

Route::middleware('auth:api')->get('/user', function (Request
$request) {
    return $request->user();
});
Route::post('user/change/avatar/{user}',
'UserController@changeAvatar')->name('user.change.avatar');

Route::post('invitation/{secret}/confirm','InvitationController@
confirm');
Route::get('invitation/{id}', 'InvitationController@get')->
name('invitation.get');

Route::post('register', 'AuthController@register');
Route::post('/register/collaborator',
'UserController@register')->name('user.register');
Route::post('login', 'AuthController@login');
Route::post('recover', 'AuthController@recover');
Route::group(['middleware' => ['jwt.auth']], function() {
    Route::get('logout', 'AuthController@logout');
    Route::get('/stat/global', 'HomeController@GlobalStat');
    Route::get('/stat/user', 'HomeController@UsersStat');

    Route::get('test', function(){
        return response()->json(['foo'=>'bar']);
    });
});

Route::prefix('plainte')->middleware('jwt.auth')->
group(function() {
    Route::get('/', 'PlainteController@getAll')->
name('plainte.index');

    Route::get('/{plainte}', 'PlainteController@get')->
name('plainte.get');

    Route::get('/{plainte}/a_traiter',
'PlainteController@a_traiter')->name('plainte.a_traiter');
```

```

        Route::get('/{plainte}/traiter',
        'PlainteController@traiter')->name('plainte.traiter');
        Route::get('/{plainte}/default',
        'PlainteController@default')->name('plainte.default');

        // Route::get('/getWith', 'PlainteController@getWith')-
        >name('plainte.get.all');

        Route::post('/{plainte}/repondre',
        'PlainteController@repondre')->name('plainte.response');

        Route::post('/', 'PlainteController@create')-
        >name('plainte.create');

        Route::put('/{plainte}', 'PlainteController@update')-
        >name('plainte.update');

        Route::delete('/{plainte}', 'PlainteController@delete')-
        >name('plainte.delete');

        Route::get('/{plainte}/disable',
        'PlainteController@disable')->name('plainte.disable');

    });

```

○ Ficher controller pour les plaintes : `PlainteController.php`

<?php

```

namespace App\Http\Controllers;

use App\Models\Plainte;
use App\Models\Motif;
use App\Models\Reponse;
use Illuminate\Http\Request;

class PlainteController extends Controller
{
    public function getAll(){
        $plaintes=Plainte::where('entreprise_id',auth()->user()-
        >entreprise_id)->with(['user','motif'])->get();
        // dd($plaintes);
        $data=array();
        $i=0;
        $data['client']=[];

        foreach ($plaintes as $plainte) {
            if($plainte->user->role=="client" && $plainte->statut==0){
                $data['client'][$i]['plainte']=$plainte;

                $data['client'][$i]['responses']=Reponse::with('user')-
                >where('plainte_id',$plainte->id)->get();
            }
            $i++;
        }
    }
}

```



```

        $i++;
    }
}

$a=0;
$data['a_traiter']=[];

foreach ($plaintes as $plainte) {
    if($plainte->statut==1){
        $data['a_traiter'][$a]['plainte']=$plainte;

        $data['a_traiter'][$a]['responses']=Reponse::with('user')->where('plainte_id',$plainte->id)->get();
        $a++;
    }
}

$b=0;

$data['traiter']=[];
foreach ($plaintes as $plainte) {
    if($plainte->statut==2){
        $data['traiter'][$b]['plainte']=$plainte;

        $data['traiter'][$b]['responses']=Reponse::with('user')->where('plainte_id',$plainte->id)->get();
        $b++;
    }
}

$j=0;
$data['employers']=[];
foreach ($plaintes as $plainte) {
    if($plainte->user->role!="client" && $plainte->statut==0){
        $data['employers'][$j]['plainte']=$plainte;

        $data['employers'][$j]['responses']=Reponse::with('user')->where('plainte_id',$plainte->id)->get();
        $j++;
    }
}

$k=0;
$data['my']=[];
foreach ($plaintes as $plainte) {
    if($plainte->user->id==auth()->id()){
        $data['my'][$k]['plainte']=$plainte;
        $data['my'][$k]['responses']=Reponse::with('user')->where('plainte_id',$plainte->id)->get();
        $k++;
    }
}

```

```

        }
    }

    $l=0;
    $data['all']=[];
    foreach ($plaintes as $plainte) {

        $data['all'][$l]['plainte']=$plainte;
        $data['all'][$l]['responses']=Reponse::with('user')-
>where('plainte_id',$plainte->id)->get();
        $l++;

    }
    return response()->json(['success'=>true,'data'=>$data],200);
}

public function get($plainte){
    $plainte=Plainte::find($plainte);
    $reponses=array();
    $responses=Reponse::with('user')->where('plainte_id',$plainte-
>id)->get();
    return response()-
>json(['success'=>true,'plainte'=>$plainte,'responses'=>$responses],20
0);
}

public function repondre(Request $request,Plainte $plainte){
    $reponse=Reponse::create([
        'plainte_id'=>$plainte->id,
        'user_id'=>auth()->id(),
        'contenu'=>$request->reponse
    ]);
    $plainte=Plainte::with(['user','motif']->find($plainte->id);
    $reponses= Reponse::with('user')->where('plainte_id',$plainte-
>id)->get();
    return response()-
>json(['success'=>true,'plainte'=>$plainte,'reponses'=>$reponses],200)
;
}

```