



INSTITUT  
FRANCOPHONE  
INTERNATIONAL



**VNU**  
ĐẠI HỌC QUỐC GIA HÀ NỘI  
Vietnam National University, Hanoi

**INSTITUT FRANCOPHONE INTERNATIONAL**

**MASTER I INFORMATIQUE**

OPTION : SYSTÈMES INTELLIGENTS ET MULTIMÉDIA

**TRAVAIL PRATIQUE**

MODULE : GÉNIE LOGICIEL AVANCE

**CONCEPTION ET REALISATION  
D'UNE APPLICATION  
DE GESTION DES  
TACHES**

Par :

CIBAMBO Masugentwali Steven

**et**

KASANDA Malamba Gustave

Enseignant :

Dr. HO TUONG VINH

*Hanoi, Novembre 2018*

## LISTE DES FIGURES

Figure 2.1. Diagramme de cas d'utilisation :.....	12
Figure 2.2. Diagramme de classes :.....	13
Figure 2.3. Diagramme de séquence :.....	15
Figure 2.4. Diagramme d'activité :.....	16
Figure 3.1. Fenêtre de connexion :.....	18
Figure 3.2. Fenêtre principale de l'application :.....	19
Figure 3.3. Ajout d'un membre :.....	20
Figure 3.4. Modifier un membre :.....	21
Figure 3.5. Supprimer un membre :.....	22
Figure 3.6. Lister tous les membre :.....	23
Figure 3.7. Créer une tâche :.....	24
Figure 3.8. Modifier une tâche :.....	25
Figure 3.9. Supprimer une tâche :.....	26
Figure 3.10. Lister toutes les tâches :.....	27
Figure 3.11. Assigner une tâche:.....	28
Figure 3.12. Lister toutes les assignations :.....	29

## TABLE DE MATIÈRES

0. INTRODUCTION :	5
0.1. Contexte :	5
0.2. Problématique :	5
0.3. Objectif du travail :	6
0.4. Délimitation du travail :	6
0.5. Subdivision du travail :	6
Chapitre 1. Spécifications de l'application :	7
1.1. Analyse du système proposé :	7
1.2. Spécification des besoins :	8
1.2.1. Besoins non fonctionnels :	8
1.2.2. Besoins fonctionnels :	8
Chapitre 2. La conception du nouveau Système :	11
2.1. Diagramme de cas d'utilisation :	11
2.2. Diagramme de classes :	12
2.3. Diagramme de séquence :	13
2.4. Diagramme d'activité :	15
Chapitre 3. Implémentation et Test :	17
3.1. Outils matériel :	17
3.2. Outils logiciel :	17
3.2.1. MySQL :	17
3.2.2. Eclipse :	17
3.2.3. Langage de programmation utilisé :	17
3.3. Technologie utilisée :	18
3.3.1. MVC :	18
3.4. Interfaces de l'application et fonctionnalités:	18
3.4.1. La fenêtre de connexion :	18
3.4.2. La fenêtre principal :	19
3.4.3. Ajout d'un membre :	20
3.4.4. Modifier un membre :	20
3.4.5. Supprimer un membre :	21
3.4.6. Lister tous les membre :	22

3.3.7. Créer une tâche :.....	22
3.3.8. Modifier une tâche :.....	24
3.3.9. Supprimer une tâche :.....	25
3.3.10. Lister toutes les tâches :.....	26
3.3.11. Assigner une tâche :.....	27
3.3.12. Lister toutes les assignations :.....	28
Conclusion :.....	30
Annexe :.....	32

## **0. INTRODUCTION**

### **0.1. Contexte**

La communication est un facteur de développement dans toute organisation pour l'atteinte des informations depuis un certain temps réel, de façon précise et à moindre coût. Cependant beaucoup d'entreprises et d'organisations ne parviennent plus à atteindre leurs objectifs par suite des contraintes liées à l'accès aux informations nécessaires en temps utile. Il se pose alors des sérieux problèmes comme celui d'*incertitude* de la source, du coût *élevé* d'acheminement du message (données) et de *manque de contrôle* sur les échanges des informations entre les membres d'une communauté ou d'une organisation.

Par ailleurs comme au sein d'une organisation existe une large hiérarchie et dont chacune est appelée à remplir une tâche bien définie , d'où la conception d'un Système de Gestion des Tâches facilitant la suivie s'avère indispensable.

### **0.2. Problématique**

Actuellement, la non numérisation du processus de suivie de projet constitue un désavantage tant pour le gestionnaire que pour tous les membres de l'organisation.

En effet, des irrégularités sont observées lors de la distribution de tâches au sein d'une organisation voir même pendant l'exécution. Entre autres ;

Pour le gestionnaire (Chef d'équipe pour notre cas) :

- Vue les multiples tâches qu'on peut retrouver dans une organisation ; et suivant leurs évolutions ; le Chef d'équipe peut facilement se tromper sur l'état d'évolution d'une ou une autre tâche.
- Étant donné que, une tâche est destinée être exécuté par un membre de l'équipe (comme dans le cas présent) mais ça peut arrivé que plusieurs tâches soient assignées à un seul membre, c'est probable qu'à un moment donné qu'on oublie à qui est assigné une tâche quelconque.
- Etc

Pour un membre de l'équipe :

- N'est pas être au courant qu'une tâche lui est assignée (ou avoir une confusion);
- Etc.

Face aux problèmes épinglés en quelques volets ci-haut que peut connaître une organisation pour la gestion de tâches à son sein, il nous a semblé injuste de rester indifférents sans apporter notre

contribution afin de trouver une solution qui peut l'aider à atteindre ses objectifs et maximiser ses rentabilités.

### 0.3. Objectif du travail

Étant étudiant en Première année de Master option Systèmes Intelligents et Multimédia, le but majeur du présent projet est de nous rappeler les concepts d'analyse et modélisation de Système d'Information avec le fameux langage UML mais aussi ceux de la programmation orientée objet précisément avec le Java.

### 0.4. Délimitation du travail

Délimiter un sujet revient à le situer dans le temps, dans l'espace et aussi dans le domaine.

- **Dans l'espace** : notre travail est circonscrit au niveau d'un groupe de personnes qui ont les multiples tâches à réaliser (soit une équipe donnée).
- **Dans le domaine** : Comme l'indique son intitulé, il s'agit d'une Conception et Réalisation d'une application informatique de Gestion des tâches au sein d'une équipe.
- **Dans le temps** : Ce travail tient compte de la constitution ordinaire d'une équipe, c'est-à-dire un groupe de deux ou plusieurs personnes avec au sein duquel on trouve un chef d'équipe et les membres simples.

### 0.5. Subdivision du Travail

Pour permettre une meilleure compréhension de notre travail, il nous a été demandé de le subdiviser en 3 grandes parties, lesquels sont précédés d'une Introduction Générale et bouclés par une Conclusion Générale.

Au cours du premier chapitre nous allons découvrir les spécifications techniques et fonctionnelles du nouveau Système.

Le deuxième chapitre sera réservé à la Conception du Système.

Le troisième chapitre englobe l'Implémentation et Test du Système proposé.

En fin, nous passerons à la conclusion générale où nous évaluerons notre prototype et proposerons les éventuelles extensions.

## **Chapitre 1. Spécification de l'application**

En réalité, la réalisation d'un logiciel nécessite une phase de conceptualisation, de développement et finalement une phase pendant laquelle on s'assure que le produit développé correspond à ce qui est attendu.

Cette dernière phase permet de se prononcer sur les différentes qualités du logiciel qui sont : la correction, la fiabilité, la robustesse ou encore l'efficacité du logiciel et porte communément le nom de phase de test.

### **1.1. Analyse du Système proposé**

Le but de l'activité d'analyse des besoins est d'éviter de développer un logiciel qui ne correspond pas aux attentes et aux besoins des utilisateurs. Elle vise à étudier le domaine d'application ainsi que l'environnement du futur système afin d'en déterminer les frontières, le rôle, les ressources disponibles et requises et les contraintes d'utilisation et de performance.

En ce qui concerne notre travail, l'analyse des besoins a pour objectif la détermination des spécifications du futur logiciel, c'est-à-dire les services qu'il doit rendre. L'analyse des besoins comprend deux étapes à savoir l'étude de faisabilité et l'expression des besoins.

#### **➤ Étude de faisabilité**

L'étude de faisabilité a pour but de déterminer si le projet répond à un besoin effectif, et s'il peut se traduire par une application informatique. Elle permet d'améliorer la définition des services à offrir. L'étude de faisabilité est une phase cruciale puisque c'est elle qui permet d'établir les premières définitions du logiciel, du contexte d'utilisation, des fonctionnalités et des services attendus.

#### **➤ L'Expression des besoins**

L'expression des besoins consiste à s'accorder sur les fonctionnalités et les contraintes à respecter afin d'obtenir la définition détaillée de ses futures fonctionnalités.

Plusieurs éléments définissent l'expression des besoins :

- Les exigences des utilisateurs qui définissent par la suite les services attendus ;
- Les spécifications globales détaillant les services et contraintes du futur logiciel ;
- Les spécifications de conception détaillée décrivant de manière très précise les fonctions du futur logiciel.

La qualité du produit final dépend en grande partie de la qualité des spécifications globales.

## 1.2. Spécification des besoins

### 1.2.1. Besoins non fonctionnels

Les besoins non fonctionnels permettent de décrire les contraintes que doivent surmonter les services et les fonctionnalités offertes par le futur logiciel (temps et réponse, plate-forme d'exécution, niveau de fiabilité, etc.).

Les besoins non fonctionnels sont souvent plus simples à exprimer, mais, parfois, elles sont aussi plus difficiles à satisfaire que les besoins fonctionnels. Les besoins non fonctionnels ou besoins système ou encore besoins techniques sont les besoins liés directement au système mais non à l'exécution des fonctionnalités.

Pour le cas de notre Système, certains critères sont à prendre en considération ; à savoir :

- **La rapidité** : vu le nombre important des tâches qu'on peut avoir au sein d'une organisation et l'effet d'assigner chaque tâche un membre peut prendre du temps ; le nouveau Système effectue cette opération par quelques clics.
- **La confidentialité et la sécurité d'information** : vue la spécification de droit d'utilisateur du Système ; cela limite l'accès aux données et certaines options à certains utilisateurs.
- **La performance** : ce logiciel doit répondre à toutes les exigences des usagers d'une manière optimale vu la façon dont il était conçu.
- **Convivialité** : du côté interface utilisateur, le futur logiciel doit répondre aux critères de facilité, de simplicité et d'adaptabilité envers les utilisateurs.

### 1.2.2. Besoins fonctionnels

Les spécifications fonctionnelles ont pour objectif la direction des services et des fonctionnalités que le futur logiciel doit obtenir. Elles doivent préciser comment le logiciel doit réagir lorsque des événements ou des situations particulières arrivent.

Pour ce faire notre application doit permettre entre autres :

- **La mobilité** : avec le présent Système ; le chef d'équipe n'a plus besoin d'organiser une réunion pour la distribution de tâche ou savoir leur évolutions ; vu que c'est un Système qui peut fonctionner dans un réseau la quasi totalité des opérations peuvent s'effectuer sur le poste de travail sans pour autant se déplacer.



- **L'archivage** : Étant donné que ; toutes les opérations effectuées sont directement stockées dans la Base des données ; cela permet une archivage numérique fiable et automatique.

Les besoins fonctionnels ou *functional requirements* en anglais sont des besoins qui décrivent le fonctionnement du système. Ils se rapportent à l'identification des acteurs et l'identification des cas d'utilisation.

- **Identification des acteurs** : Les acteurs principaux que nous avons identifié au sein de notre Système sont :

- Utilisateur principal : Chef d'équipe
- Utilisateur secondaire : Membre simple

Leur accès aux fonctionnalités du Système est décrit dans le dernier chapitre qui s'articule sur l'Implémentation et Test

- **Identification des cas d'utilisation** : les différents cas d'utilisation identifié au sein de notre système sont :

- **Authentification** : au lancement de l'application l'utilisateur est appelé à renseigner son login et son mot de passe. Cet étape permet de déterminer les options auxquelles l'utilisateur peut avoir accès suivant qu'il est Chef d'équipe ou membre simple mais aussi sécurisé les informations aux nos autorisés.
- **Ajout d'un membre** : permet au Chef d'équipe d'ajouter un membre de l'équipe dans le Système en renseignant toutes informations et son type.
- **Modifier un membre** : permet au Chef d'équipe de modifier les informations d'un membre qui existe déjà dans le Système.
- **Supprimer un membre** : permet de supprimer un membre du Système en lui ôtant tous les droits d'accès à ce dernier.
- **Lister tous les membres** : cet option permet d'afficher les informations de tous les membres présent dans le Système.
- **Quitter** : pour se déconnecter du Système et revenir à la fenêtre d'authentification.
- **Créer une tâche** : ceci permet d'ajouter une nouvelle tâche dans le Système.
- **Modifier une tâche** : pour modifier les informations d'une tâche dans le Système.
- **Supprimer une tâche** : permet de supprimer une tâche qui existe déjà du Système.
- **Lister toutes les tâches** : pour afficher les informations de toutes les tâches se trouvant dans le Système.

- **Rechercher un membre** : Une fois sur liste des membres on a la possibilité d'en rechercher a partir de la barre de recherche situé au quai supérieur droit.
- **Assigner une tâche** : permet d'assigner une tâche à un membre.
- **Assignment** : permet de voir la liste de toutes les assignments déjà effectuer.

## **Chapitre 2. La Conception du Nouveau Système**

Afin de rendre efficace l'activité d'analyse et d'expression des besoins, il s'avère nécessaire d'utiliser une méthode structurée et de fournir des documents hiérarchisés définissant les spécifications globales et les spécifications de conception détaillée qui sont intimement liées à la phase de programmation.

Le langage de conception que nous avons choisi est le langage UML.

Pour la modélisation des besoins, il nous a été demandé d'utiliser les diagrammes UML suivants :

- ✓ Diagramme de cas d'utilisation ;
- ✓ Diagramme de classe ;
- ✓ Diagramme de séquence ;
- ✓ Diagramme d'activité.

### **2.1. Diagramme de cas d'utilisation**

Le diagramme de cas d'utilisation décrit de manière générale toutes les actions possible du Système liées à un acteur. Sur le présent diagramme chaque utilisateur est associé à un ou plusieurs cas d'utilisation comme le montre la figure suivant ;

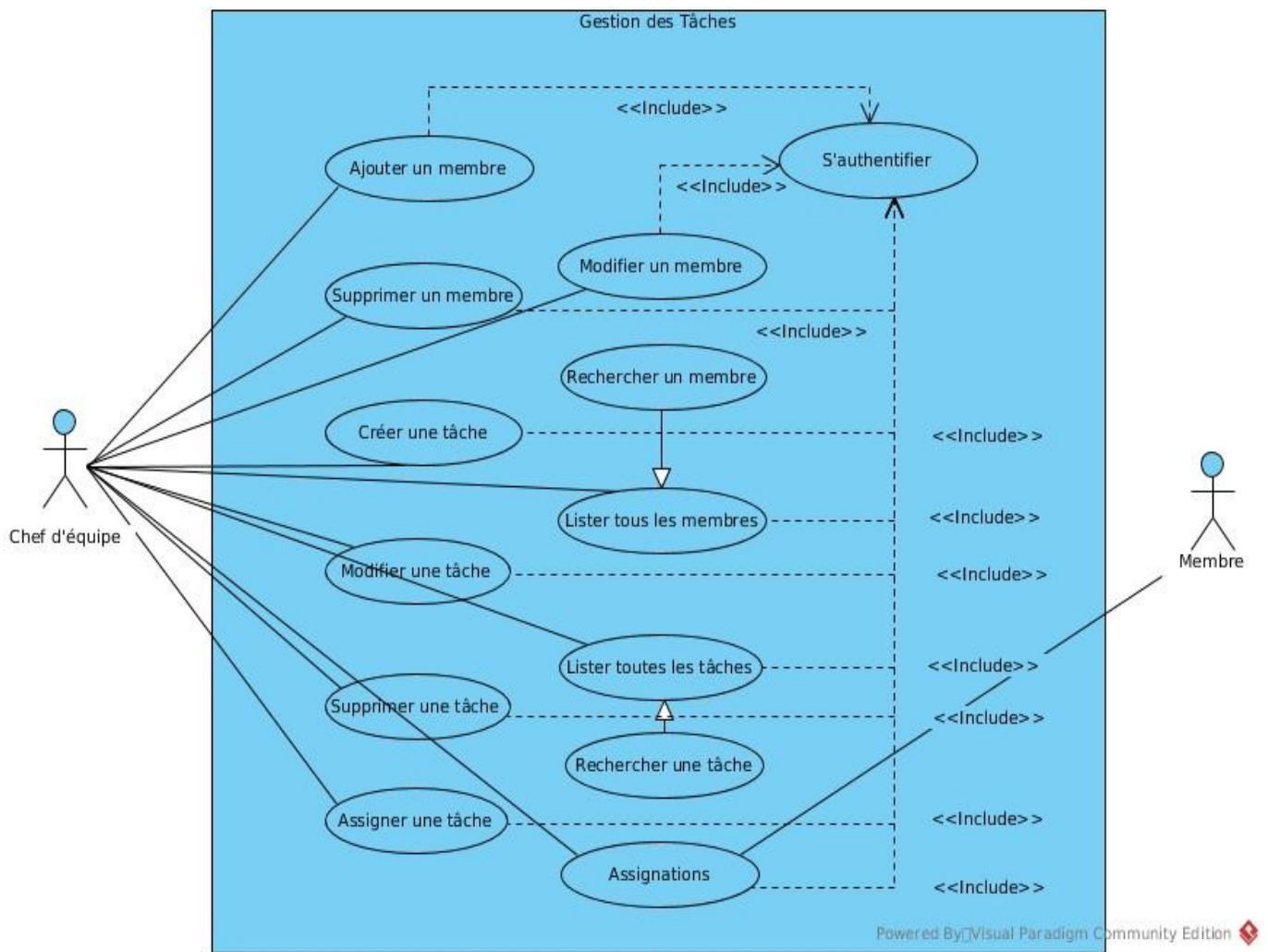


Fig.2.1. Diagramme de cas d'utilisation

## 2.2. Diagramme de classes

Ce présent diagramme illustre les relations qui unissent nos 4 classes, le résultat à la figure suivante ;

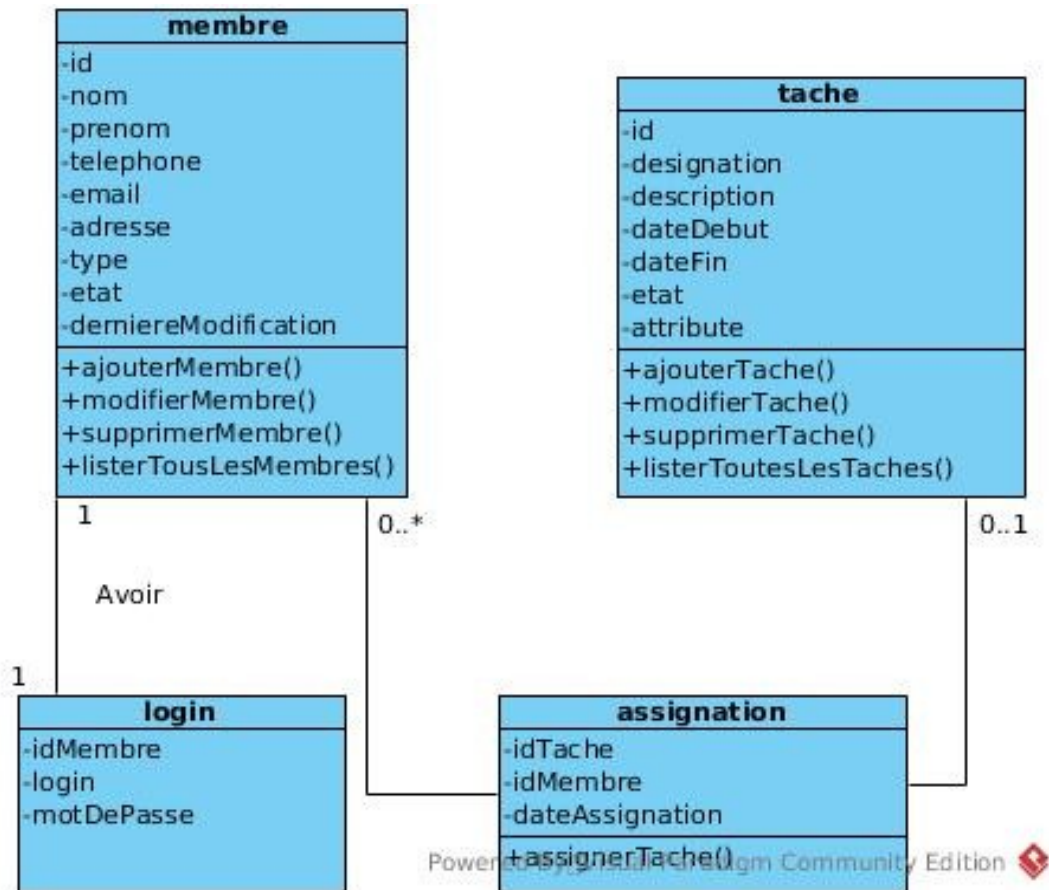


Fig.2.2. Diagramme de classes

### 2.3. Diagramme de Séquence : Assignment d'une tâche

Pour effectuer une assignation ; il faut bien sûr se connecter avec le statut d'un chef d'équipe. C'est ainsi que le présent diagramme de séquence illustre comment y arriver.

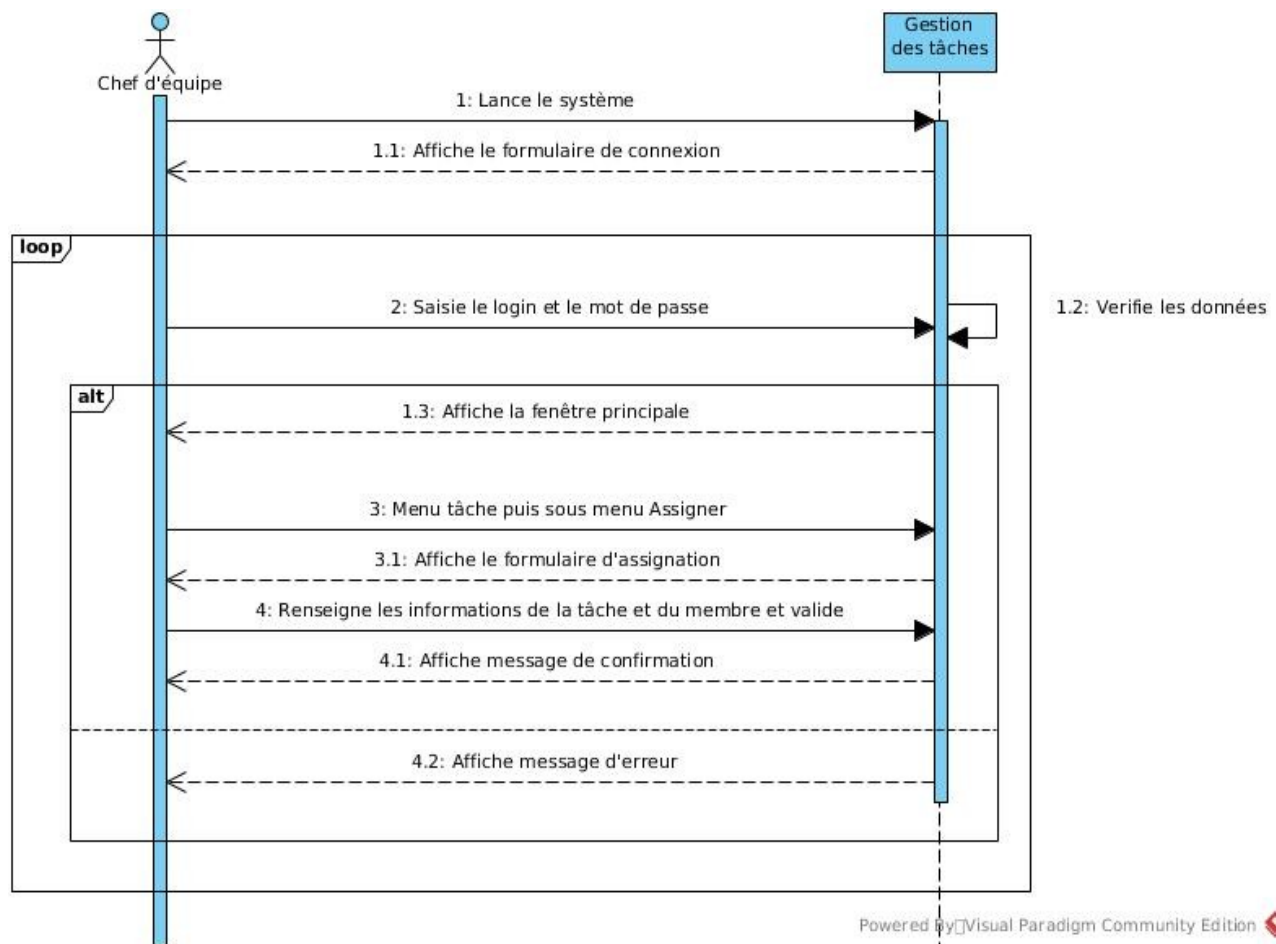


Fig.2.3. Diagramme de Séquence : Assignment d'une tâche

## 2.4. Diagramme d'Activité : Assigner une tâche

Même cas que sur le diagramme précédent ; sauf que ici c'est avec le Diagramme d'activité :

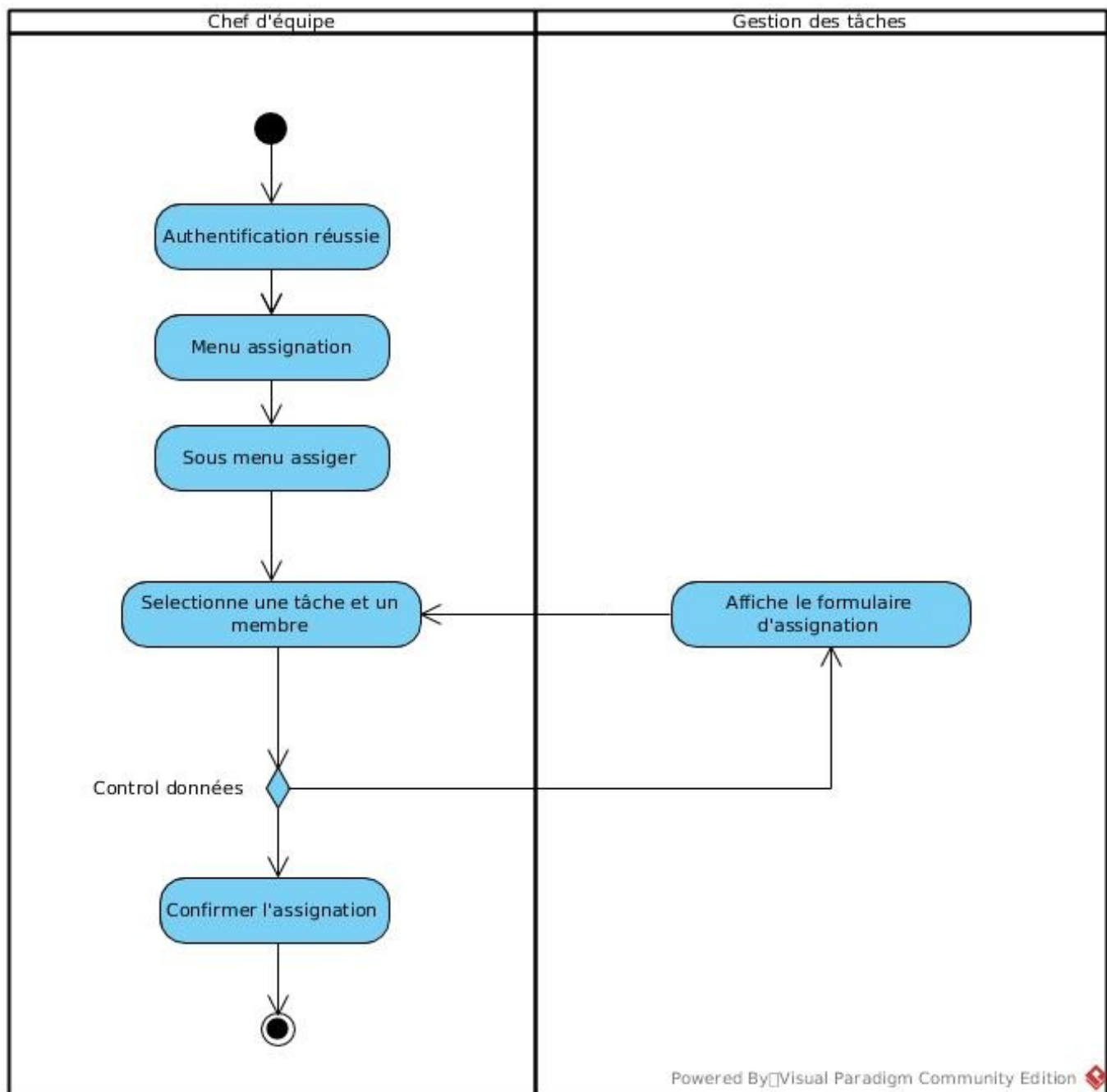


Fig.2.4. Diagramme d'activité

D'après la réalisation de ce chapitre, nous avons constaté que UML est une technique plus générique : elle ne « force » pas l'utilisateur de certains concepts qui peuvent s'avérer très importants dans la modélisation de processus, entraînant ainsi une perte d'information ou la création de modèles incomplets, mais elle est par contre flexible et extensible, ce qui permet à ses utilisateurs de l'adapter précisément à leurs besoins. UML permet également la génération automatique de code applicatif.

## **Chapitre 3. Implémentation et Test**

Cette partie présente le dernier volet de notre travail. Elle a pour objet d'exposer le travail réalisé.

D'abord nous commencerons par la présentation de l'environnement matériel puis l'environnement logiciel et en fin la technologie utilisées. Ensuite, nous illustrerons quelques aperçus d'écrans montrant les différentes fonctionnalités mise en place.

### **3.1. Outils matériel**

Pour réaliser ce travail nous avons utilisé :

- ✓ Un PC pour le développement, ayant les caractéristiques suivantes :
  - Processeur Intel i5 (2.60GHZ),
  - 4.00GB de Mémoire vive
  - Disque dure 320GB
  - Système d'exploitation Linux distribution Ubuntu version 18.04 LTS

### **3.2. Outils logiciel**

#### **3.2.1. MySQL**

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 multi-thread et multi-utilisateur. Pour notre cas il nous a été utile pour la sauvegarde de données.

#### **3.2.2. Eclipse**

Eclipse est un environnement de développement (IDE) historiquement destiné au langage Java, même si grâce à un système de plugins il peut également être utilisé avec d'autres langages de programmation, dont le C/C++ et le PHP. Pour le présent projet il nous a permis de saisir nos codes Java et de les compiler.

#### **3.2.3. Langage de programmation utilisé**



Java est un langage de programmation orienté objet et un environnement d'exécution, développé par Sun Microsystems. Il fut présenté officiellement en 1995. Le Java était à la base un langage pour Internet, pour pouvoir rendre plus dynamiques les pages (tout comme le JavaScript aujourd'hui). Mais le Java a beaucoup évolué et est devenu un langage de programmation très puissant permettant de presque tout faire. Java est aujourd'hui officiellement supporté par Sun, mais certaines entreprises comme IBM font beaucoup pour Java. C'est avec ce dernier qu'il nous a été demandé de réaliser ce projet.

### **3.3. Technologie utilisée**

#### **3.3.1. MVC**

La technologie MVC qui est l'acronyme de Modèle Vue Contrôleur, permet de créer des applications Web ou Desk composées :

- ✓ **D'un Modèle** : ce dernier est constitué d'un ensemble de classes permettant de créer les objets métiers manipulés dans l'application, et d'exécuter les traitements métiers ;
- **De Vues** : constituant des éléments graphiques tels que des contrôles d'utilisateurs, des pages Web ou encore des Master Pages. Ces éléments graphiques sont implémentés de manière radicale par rapport à leurs homologues en ASP.NET Web Forms ;
- **De Contrôleurs** : permettant de piloter l'application, d'exécuter des actions et fournir une vue en réponse aux requêtes reçues. L'une des fonctionnalités fondamentales des contrôleurs est d'assurer la communication entre le modèle et la vue.

### **3.4. Interfaces de l'application et fonctionnalités**

Dans cette section, nous présentons quelques fenêtres de notre application, ainsi qu'une petite description de leurs fonctionnalités.

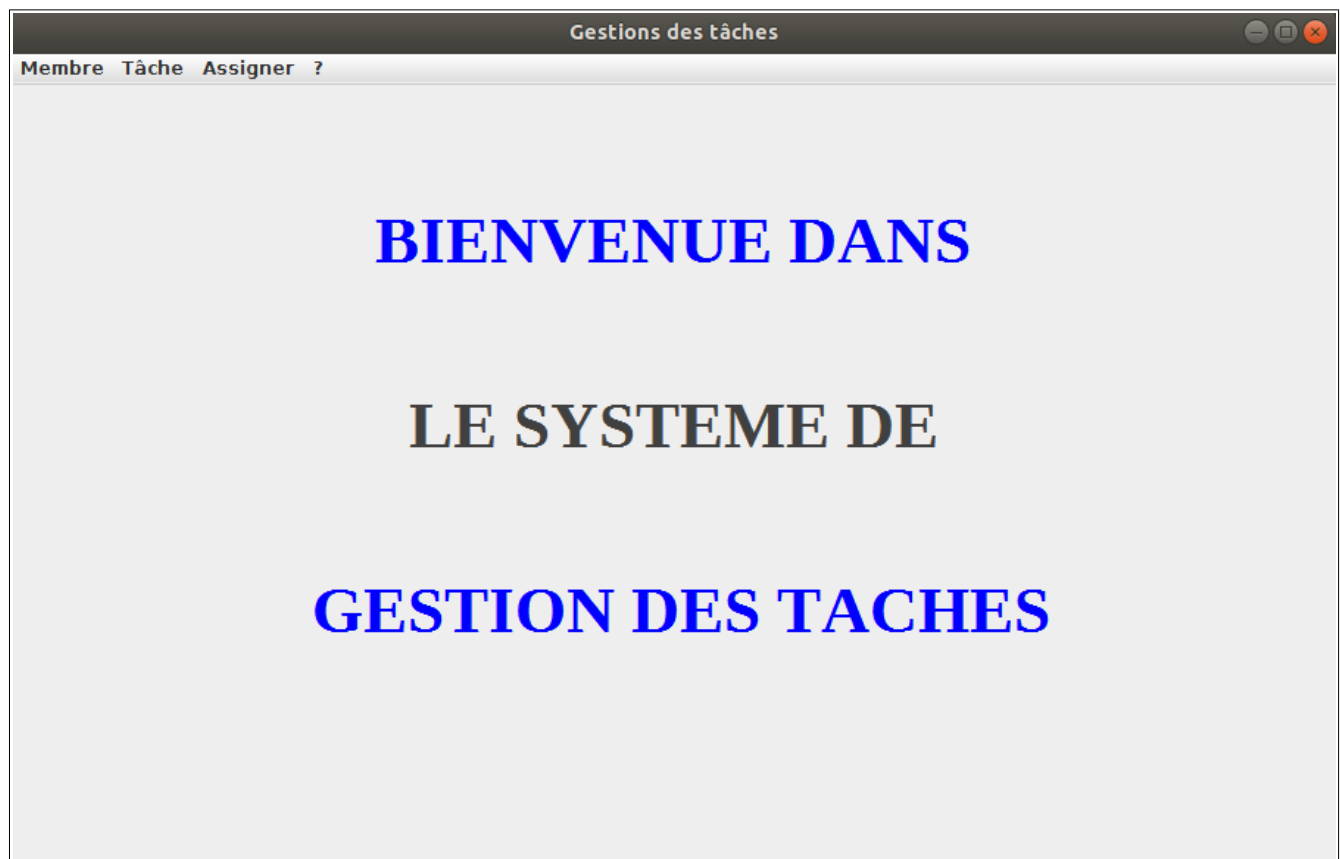
#### **3.4.1. La fenêtre de connexion**

Au lancement de l'application, l'utilisateur est appelé de saisir son login et son mot de passe pour que le système puisse savoir les informations à afficher pour lui ;



Fig.3.1. Fenêtre de connexion

### 3.4.2. La fenêtre principal



Une fois les informations sont correctes ; le système affiche directement la fenêtre principal afin de donner à l'utilisateur la possibilité d'accéder aux différentes options.

Fig.3.2. Fenêtre principale de l'application

### 3.4.3. Ajout d'un membre

Supposons que l'utilisateur connecté est un chef d'équipe et il veut ajouter un membre dans le Système le menu Membre lui ouvre une liste de sous-menus ou on trouve Ajouter, qu'il faut cliquer pour accéder à un formulaire pour saisir ses informations ; le tout à la figure suivante .

**Gestions des tâches**

Membre Tâche Assigner ?

**Informations personnel**

Nom :  E-mail :

Prénom :  Adresse :

Téléphone :  Type :

**Liste de Membres**

N°	Nom	Prénom	Téléphone	E-mail	Adresse	Type
1	Admin	Admin	+84356554309	admin@hotmail	My Dinh II, KTX DN...	Chef d'Equipe
2	Espoir	Espoir	+84125785241	espoir@hotmail.fr	KTX METRE	Membre
3	Fiston	Fiston	+8452416385	fiston@hotmail.fr	Hanoi	Membre
4	Gaston	Gaston	+857451263	gaston@hotmail.fr	Ha noi	Membre
5	Christian	Christian	+8451275411	christian@hotmail...	Hanoi	Membre
6	BAFUNYEMBAKA	CRISPIN CRISPIN	+25478411447785	bafunyembaka@...	KTX MY DINH II	Membre
9	BATUMIKE	BATUMIKE	+8452758965332	batumike@gmail....	Hoch Minh City	Administrateur
10	GASPARD	GERARD	+8475126395	gaspard@hotmail...	KTX MY DINH	Membre
11	Edgar	Benoit	+8475126395	edg@hotmail.fr	KTX MY DINH	Chef d'Equipe
12	FABIEN	EMMANUEL	+854715263214	ema@hotmail.fr	KTX MY DINH	Chef d'Equipe
13	ESSAI	BELLEVUE	+84517485	essai@hotmail.fr	KTX METRI	Membre
14	GUSTAVE	GUSTAVE	+84254784512	gustava@gmail.c...	KTX MY DINH	Chef d'Equipe
15	Emmanuel	Saulu	+841257485412	emmanuel@hotmail...	KTX METRI	Membre
16	Kayembe	Gustave	+845217	gustave@hotmail.fr	KTX METRI	Chef d'Equipe

Fig.3.3. Ajout d'un membre

### 3.4.4. Modifier un membre

Pour modifier les informations d'un membre, on passe par le menu Membre et puis le sous-menu Modifier. Une fois sur la liste de tous les membres ; on sélectionne un membre ou on le recherche avant de cliquer sur le bouton Modifier au quai supérieur droite.

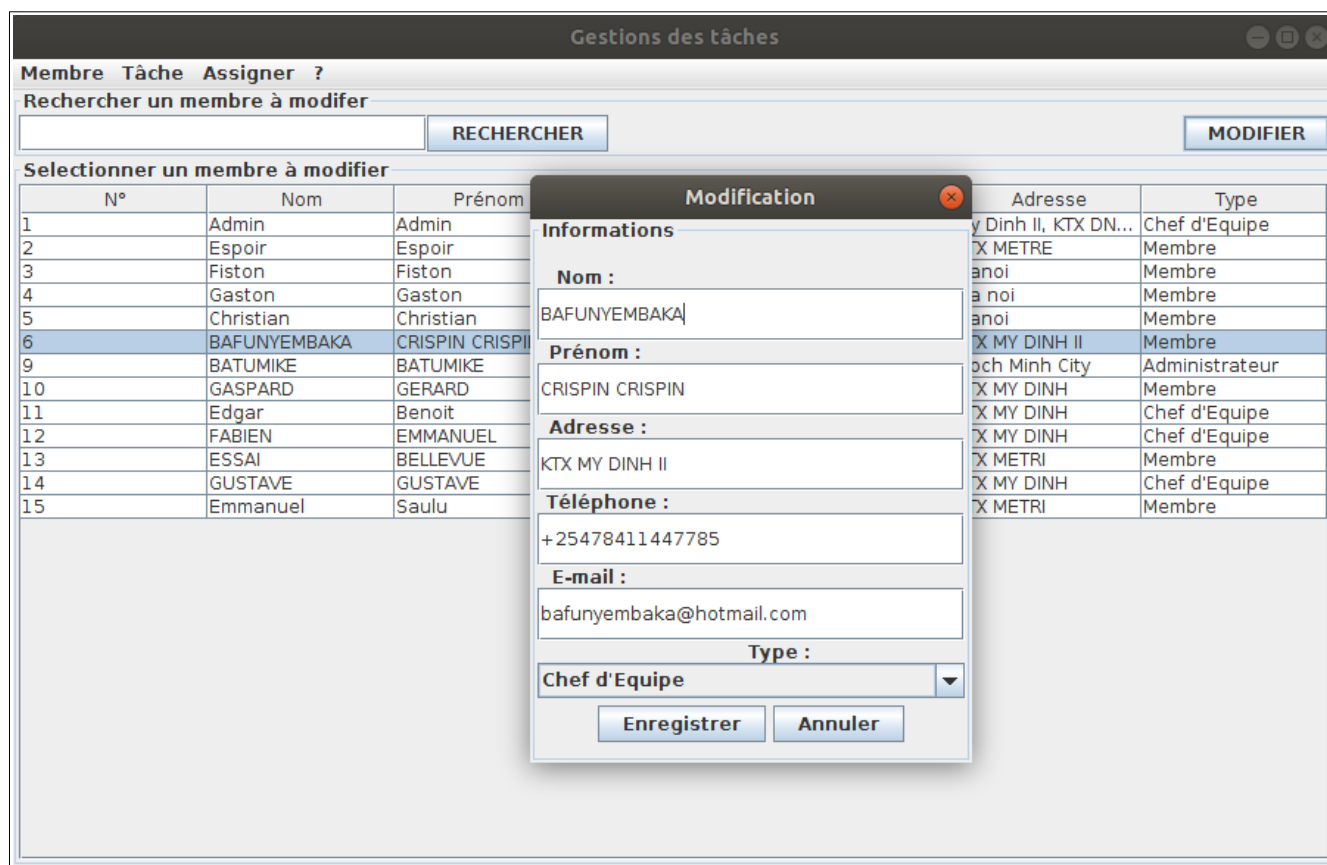


Fig.3.4. Modifier un membre

### 3.3.5. Suppression d'un membre

Idem modification

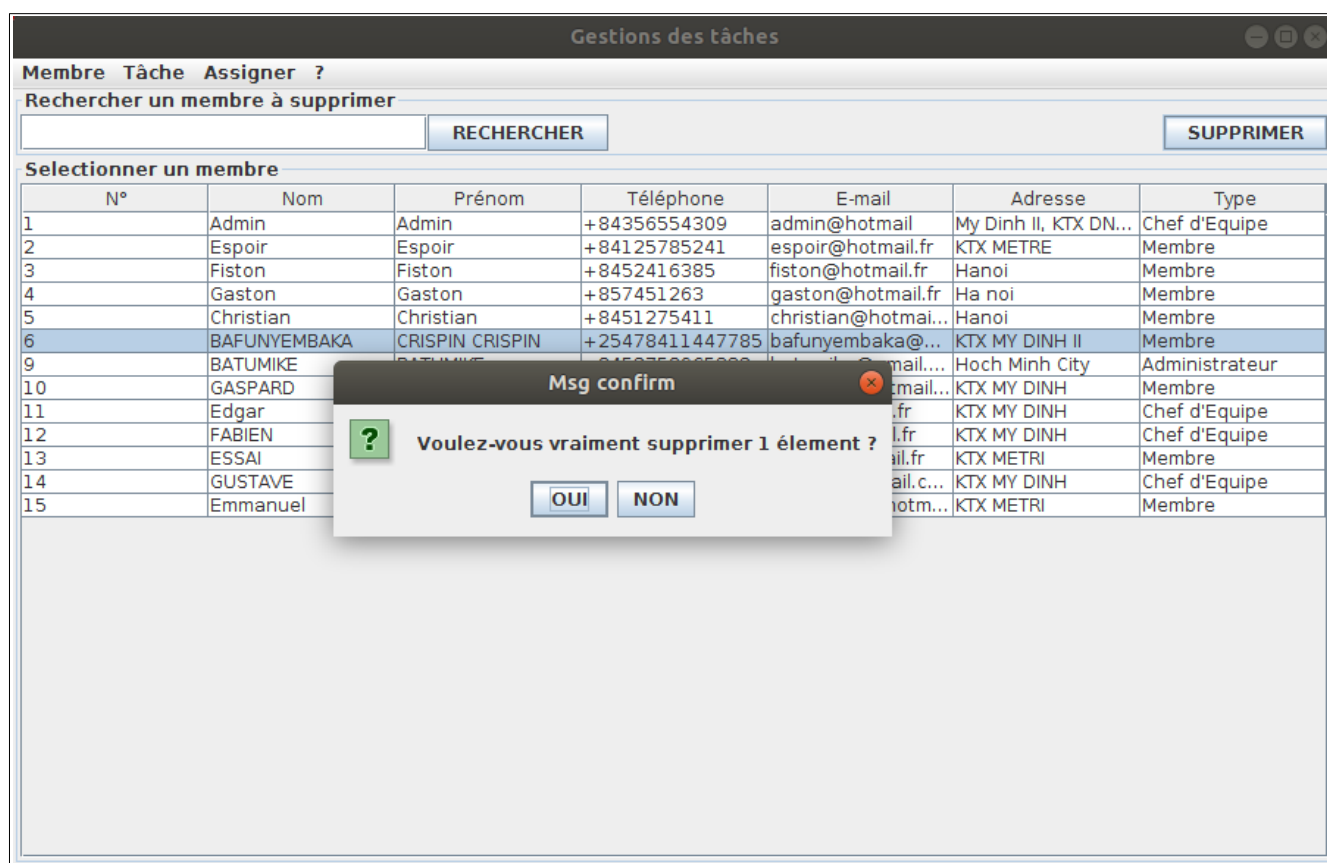


Fig.3.5. Supprimer un membre

### 3.3.6. Lister tous les membres

Pour voir tous les membres qui sont dans le Système ; il faut bien sûr passer par le menu Membre et dans la liste de sous-menu ; on choisie Lister.

Une fois sur la fenêtre qui liste tous les membres ; on a la possibilité de rechercher un membre grâce à la barre de recherche située au quais supérieur droit.

Gestions des tâches						
Membre Tâche Assigner ?						
Liste de Membres						
				<input type="text"/>	RECHERCHER	
N°	Nom	Prénom	Téléphone	E-mail	Adresse	Type
1	Admin	Admin	+84356554309	admin@hotmail	My Dinh II, KTX DN ...	Chef d'Equipe
2	Espoir	Espoir	+84125785241	espoir@hotmail.fr	KTX METRE	Membre
3	Fiston	Fiston	+8452416385	fiston@hotmail.fr	Hanoi	Membre
4	Gaston	Gaston	+857451263	gaston@hotmail.fr	Ha noi	Membre
5	Christian	Christian	+8451275411	christian@hotmail...	Hanoi	Membre
6	BAFUNYEMBAKA	CRISPIN CRISPIN	+25478411447785	bafunyembaka@h...	KTX MY DINH II	Membre
9	BATUMIKE	BATUMIKE	+8452758965332	batumike@gmail....	Hoch Minh City	Administrateur
10	GASPARD	GERARD	+8475126395	gaspard@hotmail...	KTX MY DINH	Membre
11	Edgar	Benoit	+8475126395	edg@hotmail.fr	KTX MY DINH	Chef d'Equipe
12	FABIEN	EMMANUEL	+854715263214	ema@hotmail.fr	KTX MY DINH	Chef d'Equipe
13	ESSAI	BELLEVUE	+84517485	essai@hotmail.fr	KTX METRI	Membre
14	GUSTAVE	GUSTAVE	+84254784512	gustava@gmail.c...	KTX MY DINH	Chef d'Equipe
15	Emmanuel	Saulu	+841257485412	emmanuel@hotm...	KTX METRI	Membre

Fig.3.6. Lister les membres

### 3.3.7. Créer une tâche

Pour créer une tâche, on passe par le menu Tâche ; une fois les sous-menus de celui-ci visible on choisie créer.

Gestions des tâches

Membre Tâche Assigner ?

AJOUT D'UNE TACHE

Désignation :

Description :

Date début :

Date Fin :

Etat :

Initial

Enregistrer

Tâches

N°	Désignation	Description	Date début	Date fin	Etat
1	Mise en oeuvre d'un ...	La mise en oeuvre d'...	2018-11-18	2018-11-18	Initial
2	Humaniste	Intervention sur le te...	2018-11-18	2018-11-18	Initial
3	Cadastre	la cadastre	2018-11-18	2018-11-18	Initial
4	Analyste Programmeur	Recherche des donn...	2018-11-18	2018-11-18	Initial
5	Dispenser le cours	Mise en place du pro...	2018-11-18	2018-11-18	Initial
7	Architectes	Mise en place de plac	2018-11-18	2018-11-23	Initial

Fig.3.7. Créer une tâche

On remarque que ; une fois l'enregistrement effectué ; le tableau est mis à jour automatiquement aussi.

### 3.3.8. Modifier une tâche

Avant de modifier une tâche ; il est demandé d'abord de sélectionner une tâche que l'on veut modifier et en suite cliquer sur le bouton Modifier situé au quais supérieur droit.

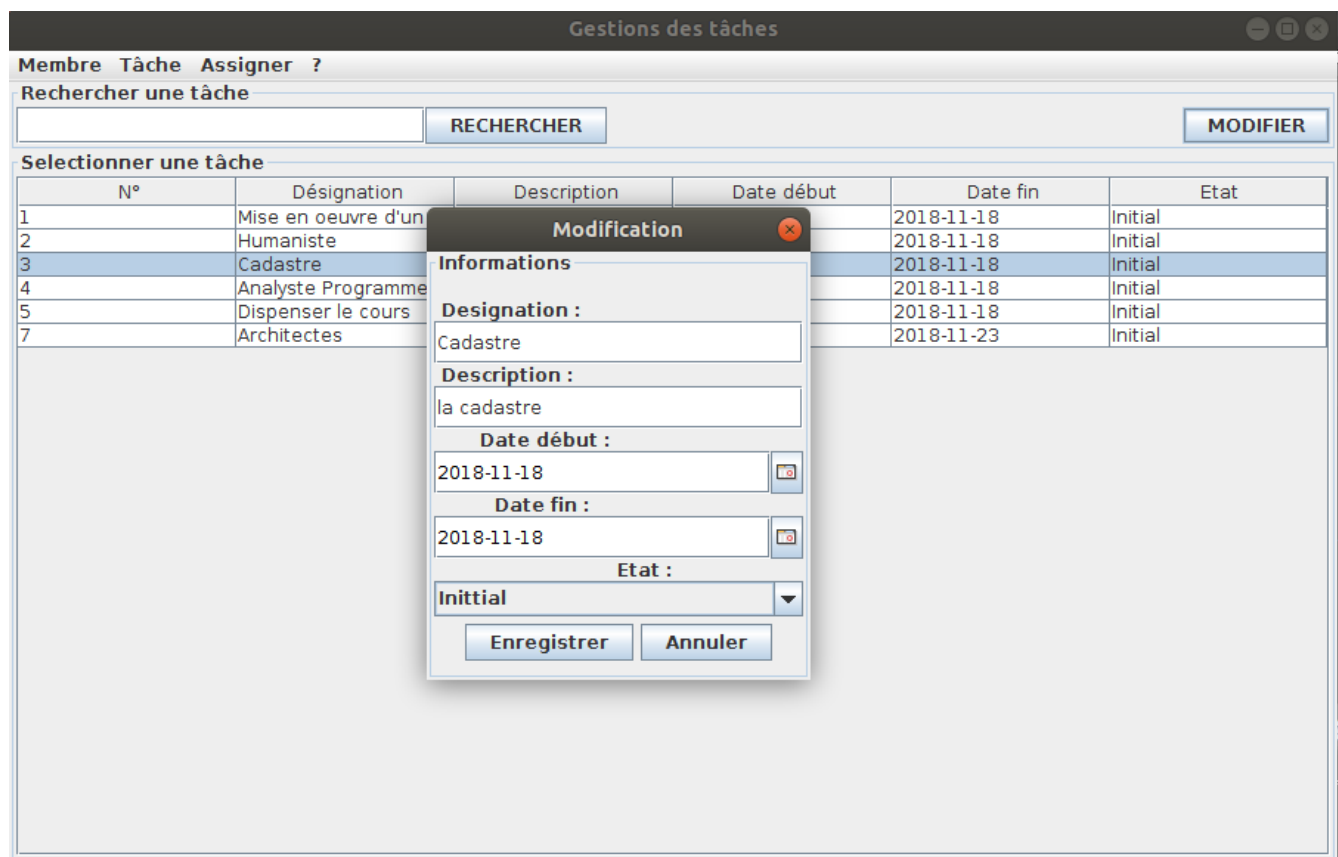


Fig.3.8. Modifier une tâche

### 3.3.9. Supprimer une tâche

Idem que modification



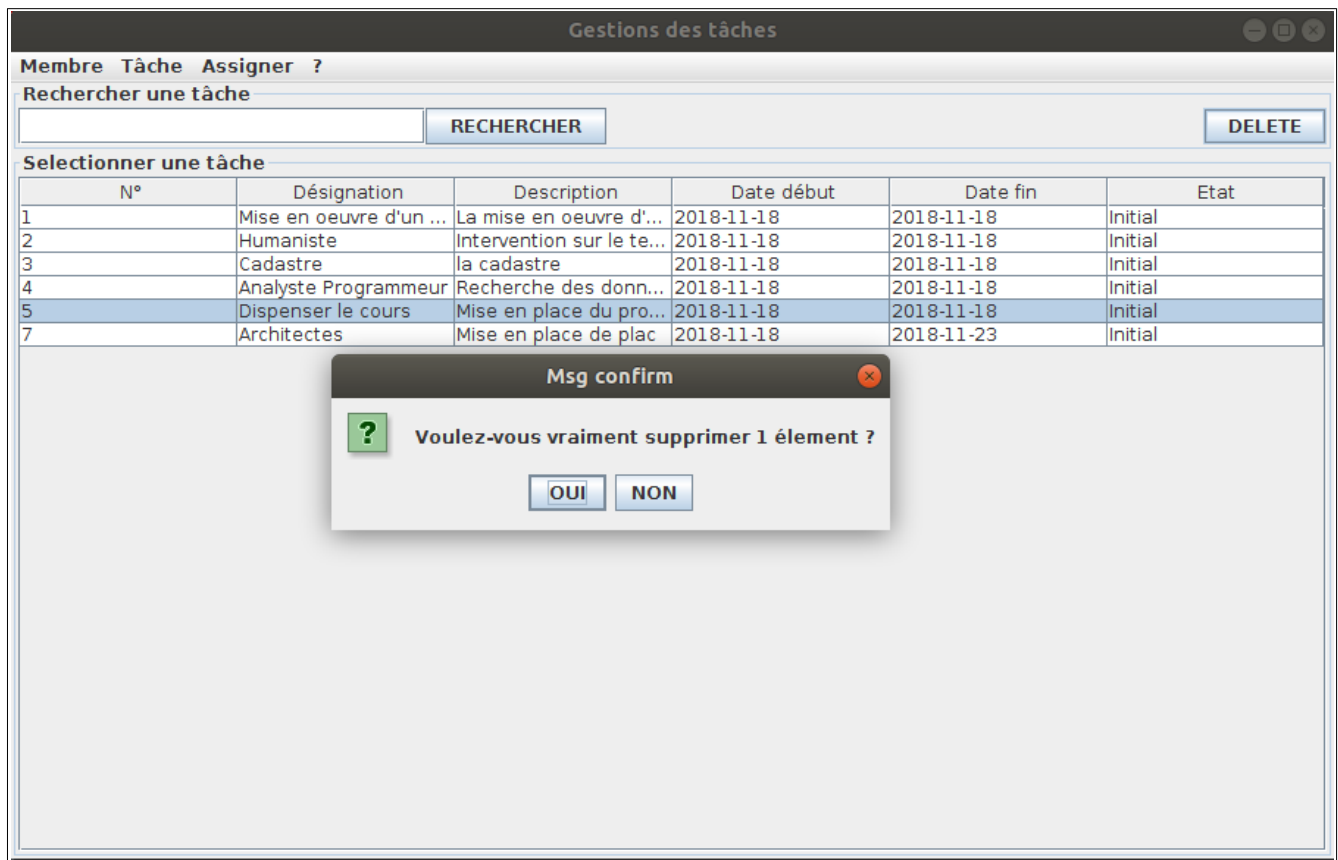


Fig.3.9. Supprimer une tâche

### 3.3.10. Lister toutes les tâches

Ceci nous permet de voir toutes les taches du Système en passant par les sous-menus du menu Tâche. Le résultat à la figure suivante ;

Gestions des tâches					
Membre Tâche Assigner ?					
Liste de Tâches					
				<input type="text"/>	RECHERCHER
N°	Désignation	Description	Date début	Date fin	Etat
1	Mise en oeuvre d'un ...	La mise en oeuvre d'...	2018-11-18	2018-11-18	Initial
2	Humaniste	Intervention sur le te...	2018-11-18	2018-11-18	Initial
3	Cadastre	la cadastre	2018-11-18	2018-11-18	Initial
4	Analyste Programmeur	Recherche des donn...	2018-11-18	2018-11-18	Initial
5	Dispenser le cours	Mise en place du pro...	2018-11-18	2018-11-18	Initial
7	Architectes	Mise en place de plac	2018-11-18	2018-11-23	Initial

Fig.3.10. Lister toutes les tâches

### 3.3.11. Assigner une tâche

Pour assigner une tâche à un membre on passe le menu Assigner, puis le sous-menu Assigner une tâche. Sur la fenêtre qui apparaît on sélectionne la tâche et le membre à qui on veut assigner la tâche.

Gestions des tâches

Membre Tâche Assigner ?

Tâche

ID Tâche :

Désignation :

Choisir la tâche

Description :

Date début :

Date Fin :

Etat :

Choisir état

Assigner

Membre

ID Membre :

Membre :

Choisir Membre

Téléphone :

E-mail :

Adresse :

Type :

Tâches

ID TACHE	TACHE	MEMBRE	DATE ASSIGNATION	DATE DEBUT	DATE FIN	ETAT
1	Mise en oeuvre d'...	Espoir Espoir	2018-11-19	2018-11-18	2018-11-18	Initial
2	Humaniste	Espoir Espoir	2018-11-20	2018-11-18	2018-11-18	Initial
3	Cadastre	Fiston Fiston	2018-11-20	2018-11-18	2018-11-18	Initial
4	Analyste Program...	Gaston Gaston	2018-11-20	2018-11-18	2018-11-18	Initial
7	Architectes	MAGNIFIQUE ANTO...	2018-11-20	2018-11-18	2018-11-23	Initial

Fig.3.11. Assigner une tâche

### 3.3.12. Lister toutes les assignments

La dernier option permet de lister toutes les assignments du Système. A partir du menu Assigner

Gestions des tâches						
Membre Tâche Assigner ?						
Tâches						
ID TACHE	TACHE	MEMBRE	DATE ASSIGNATION	DATE DEBUT	DATE FIN	ETAT
1	Mise en oeuvre d'...	Espoir Espoir	2018-11-19	2018-11-18	2018-11-18	Initial
2	Humaniste	Espoir Espoir	2018-11-20	2018-11-18	2018-11-18	Initial
3	Cadastre	Fiston Fiston	2018-11-20	2018-11-18	2018-11-18	Initial
4	Analyste Program...	Gaston Gaston	2018-11-20	2018-11-18	2018-11-18	Initial
7	Architectes	MAGNIFIQUE ANTO...	2018-11-20	2018-11-18	2018-11-23	Initial

Fig.3.12. Lister toutes les assignments

## Conclusion

Au terme de notre travail, nous estimons avoir atteint notre but à savoir la conception et la réalisation d'un Système de Gestion des Tâche au sein d'une équipe ou une organisation.

Tout au début ; nous avons commencé par une brève introduction de notre travail . Dans le premier chapitre nous avons fait le survol sur les spécifications de l'application à réaliser cela nous a permis d'avoir une vue global sur le projet à mettre en place.

Dans le deuxième chapitre nous avons parlé de la Conception et la modélisation du Système informatique à proposé, nous avons utilisé UML pour concevoir la nouvelle solution.

Ainsi dans le troisième et dernier chapitre nous avons développé la nouvelle solution en utilisant le langage Java avec MySQL comme système de gestion de base des données.

Une fois utilisé ; l'application que nous avons réalisée permettra :

- ◆ Pour l'utilisateur principal (Chef d'équipe)
  - ◆ D'ajouter un membre du système
  - ◆ De modifier les informations d'un membre
  - ◆ De supprimer un membre du système
  - ◆ De rechercher un membre et afficher toutes ses informations
  - ◆ De lister tous les membres ;
  - ◆ De créer une nouvelle tâche
  - ◆ De modifier les informations d'une tâche
  - ◆ De supprimer une tâche
  - ◆ De lister toutes les tâche qui sont dans le système
  - ◆ D'assigner une tâche à un membre
  - ◆ De voir toutes les assignations.
- ◆ Pour l'utilisateur simple (Membre de l'équipe)
  - ◆ De voir les tâches qui lui sont assignées
  - ◆ De changer son mot de passeron

Ayant constaté ce bon succès dans notre démarche, nous nous réjouissons du résultat convergeant dans le maximum avec l'objectif que nous nous sommes fixé au départ de nos recherches, bien que certaines gens disent encore que la communication en temps réel et le transport de personnes sont deux rivaux engagés dans la course et que la victoire de l'un signifiera l'obsolescence de l'autre. Le partage de ressources à distance et la navigation du monde en temps utile devient à l'ordre du jour de notre époque.

La réalisation de ce travail nous a été utile dans la mesure où nous avons pu confronter nos connaissances théoriques avec la pratique. En effet, les notions apprises au niveau de la licence ont été une des bases solides dans lesquelles nous avons puisé pour concevoir notre application.

Pour clore notre travail ; nous aimerions émettre quelques recommandations :

- Subdivision des membres en sous-équipes
- Savoir la formation d'un membre mais aussi le domaine de la tâche pour savoir qui est bien placé pour exécuter une ou une autre tâche.

## **ANNEXE**

Cette partie contient le code source de notre travail. Comme la technologie MVC le prévoit, nous avons trois grand répertoire qui contiennent l'ensemble de nos fichiers source suivant leur type ; saut la page Main.java qui est dans package par défaut.

[Code source du projet](#)