



République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari Boumediene

Faculté d'Electronique et d'Informatique
Département Informatique

Mémoire de Master

Filière: Informatique

Spécialité:

Ingénierie du Logiciel

Thème

Conception et réalisation d'un outil pour la définition
Des besoins d'un système d'information coopératif
Orienté point de vue

Sujet Proposé par :

M (e) Z. ALIMAZIGHI

M (e) K. KESSI

Soutenu le : 22 / 06 / 2017

Présenté par : DELLALI Salim

Devant le jury composé de:

GHERMIT Sarah

Mr KHEMISSA (Président)

Mr ATIF (Membre)

Binôme N° : 016/2017

Remerciement

Il met particulièrement agréable avant de présenter mon travail, d'exprimer toute ma gratitude envers les personnes qui de près ou de loin m'ont apporté leur sollicitude.

Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

Mes premiers remerciements sont adressés à mes enseignants et enseignantes à L'USTHB pour leur contribution à notre formation de master.

Ensuite, bien sûr, je tiens à remercier notre encadreur Madame ALIMAZIGHI Professeur, doyenne de la faculté FEI à l'Université des sciences et de la technologie Houari Boumedienne USTHB Alger, et sa doctorante madame KESSI pour son encadrement, ses recommandations précieuses, sa disponibilité et ses remarques constructives.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions.

A nos familles et nos amis qui par leurs prières et leurs encouragements, on a pu surmonter tous les obstacles et pour leurs bonne humeur et leurs convivialité.

Nous tenons a remercier toute personne qui a participé de près ou de loin à l'exécution de ce modeste travail.

Enfin je tiens à remercier les membres du jury pour avoir assisté à cette soutenance.

Dédicaces

Je dédie ce travail à mes chers parents jamais je ne saurais m'exprimer quant aux sacrifices et aux dévouements que vous consacrés à mon éducation et mes études .les mots expressifs soient-ils restent faibles pour énoncer ma gratitude hautement profonde, à celle qui m'a transmis la vie, l'amour, le courage, à toi chère maman toutes mes joies, mon amour et ma reconnaissance.

A madame IGHILAZA qui a toujours était à mes coté

*A ma famille et mes chères amies : Anfel, Asma, Meriem, Sílía et Karima
toute mon amitié et tous mes collègues de la promotion,*

A mon binôme, Salim chez qui m'a supporté durant ces derniers temps.

..... Sarah

Je dédie ce travail à mes parents, avec leurs aides et compréhension qu'ils m'ont apportés durant tout mon parcours scolaire et universitaire.

Je dédie ce travail à mes amis et proches qui ont cru en moi.

Je dédie ce travail à la doctorante Mme K . KESSI notre promotrice qui nous a beaucoup aidé dans la réalisation de ce projet de fin d'études, et a Mme Z . ALIMAZIGHI qui encadre Mme K . KESSI, avec ces conseils et critiques constructive en notre faveur.

Je dédie ce travail également à l'organisation AIESEC (Spécialement AIESEC Local Committee Benak) dont je fais partie, et qui m'a beaucoup aidé à acquérir des compétences qui m'ont aidé à réaliser ce projet et arriver là où j'en suis maintenant.

..... Salim

Table des matières

Liste des figures	7
Introduction générale :	8
Chapitre I : Etat de l'art	
I.1. Introduction :	10
I.2. Les SI coopératifs :	10
I.2.1. Définition :	10
I.2.2. Complexité des SIC :	11
I.2.3. La coopération :	11
I.2.4. Concepts de base d'un SIC:	11
I.2.5. Le travail coopératif:.....	12
I.2.6. L'apport du travail coopératif :	12
I.2.7. Les avantages du travail coopératif :	12
I.3. Processus de développement d'un SIC :	12
I.4. Ingénierie des exigences (IE) :	13
I.4.1. Définition :	13
I.4.2. Les éléments de base de l'IE :	14
I.4.3. Motivations pour l'ingénierie des exigences :	14
I.4.4. Pourquoi l'IE est complexe et encore mal maîtrisé ? :	15
I.4.5. Le processus d'ingénierie des exigences :	15
I.4.6. Méthodes de représentation de la phase d'IE :	17
I.4.6.1. Les méthodes dirigées par les buts :	17
I.4.6.2. Les méthodes à base de scénario :	18
I.4.6.3. Les méthodes orientées points de vue :	18
I.4.6.4. L'intérêt d'utiliser des points de vue :	21
I.5. Les niveaux de modélisation et la méta-modélisation :	21
I.6. Problématique :	22
I.7. Conclusion :	22
Chapitre I : Analyse et Conception	
II.1. Introduction :	24
II.2. Approche VpCIS :	24
II.2.1. Niveau de modélisation M2 :	24
II.2.2. Niveau de modélisation M1 :	27
II.2.3. Niveau de modélisation M0 :	29
II.3. Conception de l'outil :	31
II.3.1. Définition des classes :	31

II.3.2. Identification des attributs :	32
II.4. Présentation d'UML :	34
II.4.1. Diagramme de cas d'utilisation globale	34
II.4.1.1 Description du diagramme de Use Case :	37
II.4.2. Diagramme de séquence :	37
II.4.2.1. Diagramme de séquence d'authentification :	37
II.4.2.2. Diagramme de séquence de création :	38
II.4.3. Diagramme d'activité :	38
II.5. Etude de l'existant :	40
II.5.1. Description de l'étude de cas :	40
II.5.2. Description des points de vue :	41
II.5.2.1. Point de vue 1 : E-santé basée identité :	41
II.5.2.2. Description du Point de vue E-santé basée identité :	41
II.5.2.3. Point de vue 2 : Sécurité :	42
II.5.2.4. Description du Point de vue Sécurité :	42
II.5.2.5. Point de vue 3: Big data/ Cloud :	44
II.5.2.6. Description du Point de vue Big data/ Cloud :	44
II.6 Conclusion :	47
Chapitre III : Implémentation	
III.1. Introduction :	49
III.2. Les outils de travaux :	49
III.2.1. Environnement matériel :	49
III.2.2. Environnement logiciel :	49
III.2.2.1. Microsoft Access (MS Access) :	49
III.2.2.2. Java :	50
III.2.2.3. Eclipse :	50
III.3. Architecture générale de l'application :	50
III.3.1. Description des packages :	50
III.3.2. Le Diagramme de composants :	51
III.4. Réalisation :	52
III.5. Conclusion :	72
Conclusion générale :	73
Bibliographie :	74

Liste des figures

Figure I. 1 Processus de développement de SI.....	13
Figure I. 2 Les phases de l'ingénierie des exigences [14,20]	16
Figure I. 3 Les 4 niveaux de modélisation produit et processus	21
Figure II. 4 Méta-modèle des exigences d'un SIC orienté points de vue.....	25
Figure II. 5 Modèle de point de vue VpCIS au niveau M1.....	28
Figure II. 6 Instanciation de point de vue VpCIS au niveau M0.....	30
Figure II. 7 Diagramme de cas d'utilisation "utilisation de l'outil VpCIS"	36
Figure II. 8 Diagramme de séquence de l'Authentification	37
Figure II. 9 Diagramme de séquence de création	38
Figure II. 10 Diagramme d'activité de l'outil VpCIS	39
Figure III.11 Architecture générale du Projet java	50
Figure III.12 Diagramme des composants de l'outil VpCIS.....	52
Figure III. 13 Fenêtre d'authentification pour accéder à VpCIS WorkSpace.....	52
Figure III. 14 Fenêtre principale VpCIS WorkSpace	53
Figure III. 15 Ajout d'un nouveau SIC.....	54
Figure III. 16 Panneau des Points de vue	54
Figure III. 17 Ajout d'un nouveau Point de vue	55
Figure III. 18 Lier deux Points de vue	55
Figure III. 19 Panneau des Niveaux d'Abstraction	56
Figure III. 20 Ajout d'un nouveau Niveau d'Abstraction	56
Figure III. 21 Panneau des Concepts de base du Niveau d'Abstraction.....	57
Figure III. 22 Panneau des Equipes	58
Figure III. 23 Ajout d'une nouvelle Equipe.....	58
Figure III. 24 Panneau des Activités des Equipes	59
Figure III. 25 Ajout d'une nouvelle Activité Equipe	59
Figure III. 26 Panneau des Informations Equipes	60
Figure III. 27 Ajout d'une nouvelle Information Equipe	60
Figure III. 28 Panneau des Connaissances Equipes d'une Information Equipe	61
Figure III. 29 Ajout d'une nouvelle Donnée Equipe.....	61
Figure III. 30 Panneau des Connaissances Equipes d'une Information Equipe	62
Figure III. 31 Ajout d'une nouvelle Connaissance Equipe.....	62
Figure III. 32 Panneau des Besoins Non Fonctionnel d'une Information Equipe	63
Figure III. 33 Ajout d'un nouvel Besoin Non Fonctionnel Equipe dans l'information Equipe	63
Figure III. 34 Panneau des Acteurs.....	64
Figure III. 35 Ajouter un nouvel Acteur	64
Figure III. 36 Ajout d'une nouvel Qualité à un Acteur	65
Figure III. 37 Panneau des Activité des Acteurs	65
Figure III. 38 Ajout d'une nouvelle Activité Acteur.....	66
Figure III. 39 Panneau des Informations des Acteurs.....	66
Figure III. 40 Ajout d'une nouvelle Information Acteur.....	67
Figure III. 41 Donnée, Connaissance et Besoin Non Fonctionnel Acteur d'une Information Acteur	67
Figure III. 42 Panneau des Données Acteur d'une Information Acteur.....	68
Figure III. 43 Ajout d'une nouvelle Donnée Acteur dans l'Information Acteur.....	68
Figure III. 44 Panneau des Connaissances Acteurs d'une Information Acteur	69
Figure III. 45 Ajout d'une nouvelle Connaissance dans Information Acteur.....	69
Figure III. 46 Panneau des Besoins Non Fonctionnel d'une information Acteur.....	70
Figure III. 47 Ajout d'un nouvel Besoin Non Fonctionnel d'une Information Acteur.....	70
Figure III. 48 Ajout d'un Lien.....	71
Figure III. 49 Panneau des Recherches	71
Figure III. 50 Exemple de Recherche	72

Introduction générale

L'avalanche technologique des dernières années a engendré des difficultés d'élaboration de SI fiables et modernes au service de la stratégie d'entreprise. Les évolutions technologiques ont un impact certain sur les processus d'entreprise, et ont entraîné celle-ci à modifier son organisation et à reconfigurer ces processus pour être plus rentable. De cela, est née une nouvelle génération de SI, les SI coopératifs, qui intègre dans la définition et l'élaboration du processus de développement la composante « coopération ». La coopération peut exister à deux niveaux : intra-entreprise et inter-organisations lorsque celles-ci acceptent de collaborer en vue d'objectifs communs.

Les systèmes d'information coopératifs, soit les systèmes basés sur le travail en équipe, sont aujourd'hui le cœur des entreprises. Ces systèmes font intervenir plusieurs équipes de domaines différents qui coopèrent afin de réaliser un même but.

En matière de méthodes d'analyse de SI, supportant les nouvelles technologies, nous nous proposons d'explorer le concept de Génie Logiciel : les points de Vue (Viewpoints) afin d'étudier et analyser les avantages qu'ils apportent dans la gestion du travail de groupe et les intégrer dans un processus de développement de SI coopératifs.

Dans un environnement complexe de coopération, le travail est morcelé suivant des objectifs et/ou rôles bien définis et réparti dans différentes équipes. En Génie Logiciel, la construction d'un système complexe fait appel au concept de Point de Vue. Un point de vue est une encapsulation de l'information partielle concernant un aspect du système, à laquelle on rajoute toutes les relations qui peuvent exister entre les différents points de Vue.

L'équipe de recherche a déjà défini une approche orientée points de vue pour la description de la phase d'analyse des besoins d'un SIC.

L'objectif de ce mémoire est de mettre en œuvre cette approche par la proposition d'un outil qui permet de définir et réaliser la phase d'analyse des besoins d'un SIC. Il est structuré selon les 3 chapitres suivants:

Le chapitre 1: Etat de l'art: nous présentons dans ce chapitre le contexte de travail: SI coopératifs, l'ingénierie des exigences (IE) ainsi que quelques méthodes de l'IE.

Le chapitre 2: Analyse et Conception: nous donnons un aperçu global sur la conception de notre outil et sur les spécifications des besoins.

Le chapitre 3: Implémentation: définit l'implémentation de notre travail effectué où nous mettons en œuvre notre outil.

Ce mémoire se termine par une conclusion et quelques perspectives futures.

Chapitre I : Etat de l'art

I.1. Introduction :

Les besoins en matière d'information ont beaucoup évolué ces dernières années et ont rendu nécessaire le partage des données entre systèmes d'information. Le partage des données n'est certes pas une idée nouvelle mais s'est démultiplié et complexifié avec l'avènement de l'Internet.

Les mutations technologiques de ces dernières décennies ont profondément bouleversé la vie des organisations. Progressivement, l'idée de la coopération dans les systèmes d'information est devenue une nécessité afin de répondre à de nouvelles exigences organisationnelles.

Cette nouvelle structuration d'un Système d'information « SI » nous amène à considérer que la connaissance nécessaire à la prise de décision dans une entreprise est constituée non seulement de la connaissance de ses membres mais aussi de la connaissance émergente de leurs interactions les uns avec les autres. Les SIs sont ouverts, distribués, hétérogènes et de plus en plus complexes, car ils font intervenir plusieurs parties prenantes de domaines différents qui doivent coopérer afin de réaliser un même but.

C'est donc le rôle du Système d'Information Coopératif « SIC » qui tend à favoriser, supporter et gérer des situations d'interaction entre les différents sous-ensembles du SI. Dans le paradigme des SICs, un SI doit être défini beaucoup plus par ses connexions et ses relations avec son environnement que par sa technologie interne ou ses fonctionnalités selon [1].

La réussite d'un projet de développement d'un système tel que le SIC, dépend d'une identification réelle des besoins que le système est censé satisfaire. Dans les années 70 à 80, les systèmes étaient développés selon la vision des concepteurs et des ingénieurs, sans l'implication effective des utilisateurs et autres parties prenantes. Cela a abouti à l'abandon de nombreux systèmes qui étaient très bien construits d'un point de vue technologique, mais qui ont été considérés comme des échecs car ils ne correspondaient pas aux besoins des utilisateurs. Afin de minimiser le taux d'échec de l'utilisation des systèmes informatiques dû à ce genre de lacunes, une étape dite *d'ingénierie des exigences (IE)* en particulier *la phase d'analyse des besoins* doit être clairement identifiée dans le processus de développement des systèmes d'information, et l'application des méthodologies et des technologies d'*ingénierie des exigences* s'avère nécessaire.

Ce chapitre vise à faire une présentation succincte permettant de comprendre le cadre général de notre travail à savoir définition de la phase d'analyse des besoins d'un SIC. Il est organisé comme suit. La première section présente le SIC de manière générale, la deuxième section définit le processus de développement d'un SIC. La troisième section présente la phase d'IE en général et les méthodes de représentation de cette dernière. La quatrième section présente les niveaux de modélisation définis par l'OMG. Pour finir, nous exposons notre problématique à la cinquième section.

I.2. Les SI coopératifs :

I.2.1. Définition :

Un SIC est « un ensemble de composants plus ou moins autonomes, souvent préexistants qui travaillent de manière synergique en échangeant information, expertise et en coordonnant leurs activités » [1].

I.2.2. Complexité des SIC :

Les trois principaux facteurs d'augmentation de la complexité d'un système d'information sont [2] :

- **Sociaux :** comportement et actions des acteurs dans l'organisation: plusieurs acteurs issus d'équipes différentes de domaines différents sont amenés à coopérer dans le but d'accomplir une même action, la coordination entre leurs activités est alors requise.
- **Techniques :** liés à la complexité des technologies employées.
- **Management :** relatifs à la difficulté de la prise de décision.

I.2.3. La coopération :

Une coopération est une association autonome de personnes volontairement réunies pour satisfaire leurs aspirations et besoins économiques, sociaux et culturels communs dans une organisation dont la propriété est collective et où le pouvoir est exercé démocratiquement.

Dans la recherche actuelle sur le "Computer Supported Cooperative Work" (CSCW), on présuppose habituellement que l'ensemble des personnes qui collaborent est un "groupe" ou une "équipe", un ensemble relativement fermé de personnes qui sont engagées dans un processus constant de communication et d'échange. [3]

Les groupes permettent d'accomplir des tâches irréalisables par une seule personne. C'est la base de toute organisation. L'efficacité d'une organisation dépend de l'efficacité de ses groupes de travail. On peut les définir comme des regroupements de personnes partageant le même but et engagées dans une communication continue.

Les performances et l'efficacité du groupe résultent de la coopération mise en œuvre et des décisions prises par ses membres. L'objectif des logiciels de groupe est de les aider à travailler ensemble. [4]

On appelle « activités coopératives » tout ce qui entraîne des besoins de coordination et de communication entre personnes afin d'assurer conjointement une activité professionnelle ou sociale. [3]

Ces définitions nous permettent de déduire les concepts de base nécessaire à la définition des besoins d'un SIC.

I.2.4. Concepts de base d'un SIC:

Afin de définir la phase d'analyse des besoins d'un SIC, il est nécessaire de définir les concepts de base suivants [5]:

- **L'acteur:** peut-être une équipe ou un acteur simple (une équipe est composée d'acteurs simples)
- **L'activité:** correspond à une tâche ou une fonction du SIC que doit réaliser un acteur ou une équipe (une activité peut être une activité composée réalisée par une équipe ou une activité simple réalisée par un acteur simple).
- **Les informations:** les informations dont a besoin un acteur ou une équipe pour accomplir son activité.
- **Les interactions:** représentent les différents liens qui peuvent exister entre l'acteur, l'activité et l'information.

I.2.5. Le travail coopératif:

Le travail coopératif (travail en équipe) "Teamwork" et la mise en réseau de compétences des collectifs de travail et des individus sont devenus des éléments clés de la société. Le management des entreprises est basé aujourd'hui sur la valorisation de la ressource humaine qui par le passé souvent productiviste [3], évolue vers de nouveaux modèles moins hiérarchiques, valorisant la mise en réseau et la coopération.

Le travail coopératif, dans sa définition la plus large, fait l'objet d'un champ d'étude pluridisciplinaire appelé "Computer Supported Cooperative Work" (CSCW) littéralement « Travail Coopératif Assisté par Ordinateur » encore appelée "Collectique".

Cette discipline étudie les mécanismes collectifs et individuels du travail de groupe, et recherche comment les technologies de l'information et de la communication (TIC) peuvent faciliter ce travail. Ces technologies assurent un équilibre entre les composantes humaines (culture, valeurs, comportement), organisationnelles (processus, structures) et technologiques (système d'information et de communication) et ont conduit à repenser l'organisation de l'entreprise.

Le travail coopératif répond aux 3 grandes fonctions (les 3 C): de communication (dialogue), de collaboration (partage) et de coordination (Workflow). [6]

La communication s'effectue par une transmission d'informations entre plusieurs personnes.

La coopération consiste en une participation de plusieurs personnes pour la réalisation d'une tâche, d'un objet, ou d'un projet, tandis que la coordination consiste à organiser les tâches et les ressources entre des personnes qui coopèrent.

I.2.6. L'apport du travail coopératif :

Le travail coopératif a des retombées intéressantes sur les nouvelles organisations [6] :

- **Meilleure rendement** : Le travail collaboratif permettra une productivité meilleure et une nette amélioration des performances.
- **Meilleure qualité de services** : Le travail coopératif apportera une amélioration de la qualité du service, avec un meilleur temps de réponse. Une vision claire des tâches à effectuer, et des tâches effectuées par les autres.
- **Visualisation claire des tâches** : Le travail coopératif permettra de connaître en tout temps la situation exact des projets, les coûts, etc. et pour ainsi apporter une meilleure connaissance permettant d'améliorer considérablement l'efficacité.

I.2.7. Les avantages du travail coopératif :

Le travail coopératif présente plusieurs avantages dont les plus significatifs sont [6] :

- Le partage des documents, et l'échange d'informations avec l'ensemble de ses coopérateurs.
- L'élaboration en commun d'une étude ou d'un projet.
- La coordination des activités au sein d'un processus.

I.3. Processus de développement d'un SIC :

Le développement d'un système d'information coopératif commence par la modélisation des processus à automatiser. Pour chaque étape du travail à effectuer, il faudra déterminer qui fait quoi, sur quoi, à quel moment, après quoi et avant quoi. Mais aussi définir les détenteurs de l'information, les types de documents traités, les points éventuels de blocage.

Un processus de développement de SI coopératifs comprend alors deux grandes étapes comme le montre la figure ci-dessous :

- l'ingénierie des besoins qui comporte les deux phases suivantes : la réalité organisationnelle (le problème) et la solution conceptuelle.
- l'étape d'ingénierie des systèmes qui comporte la phase de la solution implémentable.

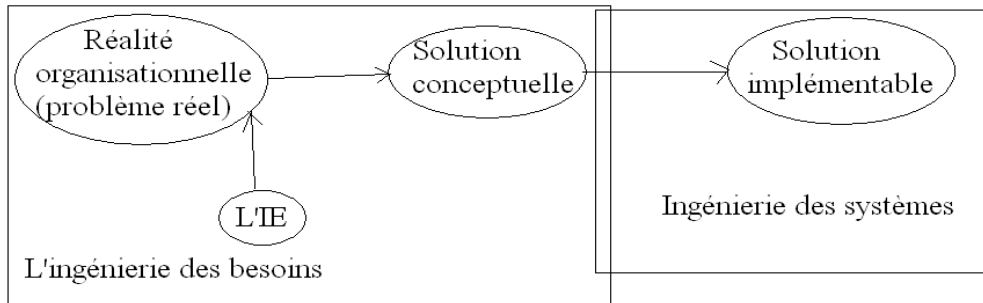


Figure I. 1 Processus de développement de SI

Dans le cadre de notre travail, nous nous intéressons au développement d'un système d'information coopératif, en particulier la phase de définition des besoins, nous nous situons alors à la phase d'ingénierie des exigences. Les SIC étant des systèmes complexes, nous proposons une approche qui permet de définir les besoins et d'exprimer le problème de complexité de ces derniers.

Nous détaillons dans ce qui suit la phase d'ingénierie des exigences.

I.4. Ingénierie des exigences (IE) :

I.4.1. Définition :

Selon le GTIE (Groupe de Travail sur l'Ingénierie des Exigences) de l'AFIS (Association Française de l'Ingénierie des Systèmes), l'IE désigne l'ensemble des activités destinées à découvrir, analyser, valider et faire évoluer un ensemble d'exigences relatives à un système. Elle permet de montrer la satisfaction des besoins et des engagements durant tout le cycle de vie d'un système [7].

L'Ingénierie des Exigences (IE) est une phase initiale du projet qui précède la conception du système. Elle consiste, au travers des méthodes, règles et processus, à établir et maintenir un référentiel unique qui s'affine et se complète au cours du développement et qui est maintenu tout au long de la vie du système.

L'ingénierie des exigences implique la capture, l'analyse et la résolution des nombreuses idées, des perspectives et des relations à divers niveaux de détail. Les méthodes basées sur des schémas globaux rigides ne tiennent pas suffisamment compte de la diversité des questions soulevées par les problèmes l'ingénierie des exigences. Les méthodes basées sur la notion de points de vue ont évolué pour résoudre ce problème [8].

Le but de l'ingénierie des exigences est donc d'obtenir un ensemble complet et cohérent de spécifications. Ce résultat se construit tout au long du processus ; au départ, les spécifications sont opaques, informelles et s'expriment uniquement par des vues personnelles. Ces vues

reflètent les compétences, objectifs et rôles de chaque participant. L'activité de spécification est donc une activité collective. Autoriser l'expression de vues multiples permet notamment une meilleure élicitation des besoins et ne pourra être que bénéfique au processus de spécification [9] [10].

I.4.2. Les éléments de base de l'IE :

- **Les parties prenantes :** c'est une entreprise, une organisation ou un individu ayant un intérêt ou une partie dans le résultat de l'ingénierie d'un système. Constitue une partie intéressée par l'utilisation et l'exploitation du système (voire par ses impacts sur son environnement), C'est aussi un agent qui participe à la conception du système, sa production, son déploiement, sa commercialisation, son maintien en condition opérationnelle et son retrait de service.
- **Les besoins :** c'est la perception que l'utilisateur a sur le système. Ce besoin s'exprime souvent sous forme de problèmes que rencontrent les utilisateurs auxquels le système est destiné.
- **Les exigences :** c'est une caractéristique visible, vue de l'extérieur du système. une condition ou une capacité qui doit être satisfaite par un système ou composant d'un système pour satisfaire un contrat, une norme, une spécification, ou autres documents formellement imposés. Comparativement au besoin, qui est lié à l'utilisateur, l'exigence est la vision que le concepteur ou le développeur a sur le système. Les exigences représentent la vision du système d'un point de vue concepteur, une exigence est un besoin dont la solution peut être implémentée.

On distingue 2 types d'exigences :

- **Les exigences fonctionnelles:** ce sont des exigences définissant une fonction du système à développer, décrivent le quoi, c'est à dire ce que le système doit faire.
- **Les exigences non fonctionnelles:** ce sont des exigences qui caractérise une propriété ou une qualité désirée du système, telle que sa performance, sa robustesse, sa convivialité, sa maintenabilité, etc ...

I.4.3. Motivations pour l'ingénierie des exigences :

En 1976, Bell & Tayer ont fait observer que l'inadéquation des fonctionnalités du système aux besoins des usagers, l'incohérence, l'incomplétude et l'ambiguïté des documents des exigences avaient une influence majeure sur la qualité du logiciel final [11].

Christel & kang (1992) ont classé les problèmes qui se posent lors de l'élicitation des exigences en dix points, appartenant à trois classes différentes [12]:

- Problèmes d'étendue du système étudié :
 - Les frontières du système sont mal définies.
 - Des informations non nécessaires sont fournies.
- Problèmes de compréhension :
 - Les utilisateurs ont une idée incomplète de leurs besoins.
 - Les utilisateurs connaissent mal les possibilités et contraintes des systèmes proposés.
 - Les analystes ont une faible connaissance du domaine.

- L'utilisateur et l'analyste parlent des langages différents.
- Il est facile d'omettre des informations.
- Il peut exister des conflits de points de vue entre différents utilisateurs.
- Les besoins sont souvent vagues et non mesurables.
- Problèmes de volatilité des exigences :
 - Les exigences évoluent au cours du temps.

I.4.4. Pourquoi l'IE est complexe et encore mal maîtrisé ? :

- Le champ est large puisqu'il s'étend du domaine social des organisations à celui des lois de la physique aux artefacts qui doivent y être intégrés; des objectifs stratégiques aux prescriptions logicielles fines; de l'informel au formel. Le système cible n'est pas seulement un logiciel mais comprend l'environnement organisationnel dans lequel il doit opérer. Celui-ci est complexe car il est constitué d'êtres humains, de machines et/ou de logiciels. Le système doit être considéré selon de multiples points de vue : socio-économiques, physiques, opérationnels, évolutifs, etc.
- Il y a d'autres aspects à prendre en compte en plus des aspects fonctionnels du système qui viennent à l'esprit, en premier lieu : la sécurité, la performance, la robustesse, la facilité d'utilisation, l'interopérabilité, etc. Tous ces aspects sont appelés non fonctionnels et ont la particularité d'être souvent en conflit les uns avec les autres.
- Il y a de nombreux acteurs. Citons, les ingénieurs des exigences, les clients, les experts du domaine, les utilisateurs, les commanditaires, les ingénieurs de maintenance et les ingénieurs d'applications. Chacun ayant son champ de compétences et de connaissances, sa propre culture, ses propres points de vue et intérêts. Ces différents acteurs ont des points de vue différents, voire conflictuels.
- Les spécifications des exigences peuvent être entachées de nombreuses erreurs. Certaines d'entre elles peuvent être terriblement pénalisantes et avoir des effets désastreux sur les étapes suivantes du développement et/ou sur la qualité du futur système. Ce sont par exemple les incomplétudes et incohérences, ambiguïtés, inadaptations aux exigences réelles; d'autres sont des faiblesses de la spécification qui peuvent aboutir à des pertes de temps ou la génération de nouvelles erreurs [13].

I.4.5. Le processus d'ingénierie des exigences :

Selon Nuseibeh & Easterbrook (2000) [14], un processus d'IE inclut les phases d'élicitation, de modélisation, d'analyse, de spécification, de validation et de gestion des exigences comme le montre la figure suivante:

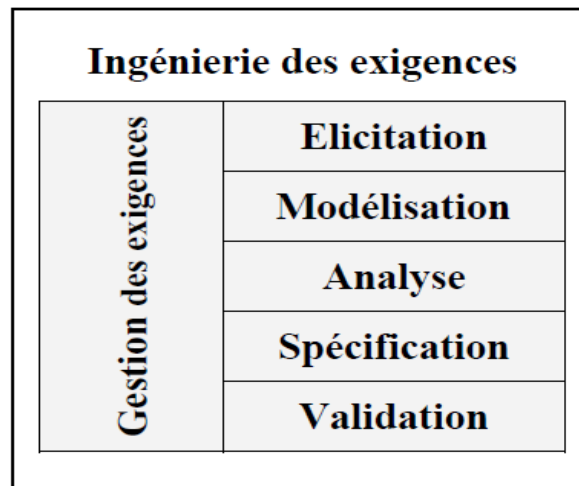


Figure I. 2 Les phases de l'ingénierie des exigences [14,20]

- **L'Elicitation des exigences** consiste en la collecte, la capture, la découverte et le développement des exigences à partir d'une variété de sources y compris les parties prenantes humaines.
- **La Modélisation** est une abstraction du problème par des représentations parfois graphiques. C'est la construction d'une description et d'une représentation abstraite d'un système pour les besoins de l'interprétation [14]. Divers aspects d'un système peuvent être modélisés pour avoir une compréhension approfondie du contexte ou de l'environnement.
- **L'Analyse** se focalise sur l'examen, la compréhension des exigences élicitées et leur vérification pour la qualité en termes d'exactitude, de complétude, de clarté et de consistance. C'est l'examen et l'interprétation des résultats issus de la phase de modélisation dans le but de clarifier les exigences, de supprimer les incohérences, et d'assurer la complétude et la non redondance. Les méthodes d'analyse sont entre autres la critique des connaissances et les techniques d'élicitation des exigences.
- **La Spécification** n'est autre que l'enregistrement et la documentation des exigences de sorte que celles-ci soient utilisables par les parties prenantes, et en particulier, par les développeurs qui doivent concevoir et construire le système. Elle consiste à établir la liste finale des exigences en les organisant suivant des catégories. Les exigences selon différentes perspectives doivent être intégrées dans le but d'avoir une vision commune du système.
- **La Validation** est la confirmation de la qualité des exigences et de leur conformité aux besoins et désirs des parties prenantes. Selon la norme EIA-632, la validation est focalisée sur la vérification de la version finale du document des exigences pour détecter les conflits, les omissions et les déviations par rapport aux normes [23]. La validation cherche à certifier que les exigences satisfont les attentes des parties prenantes et à assurer qu'elles définissent les fonctionnalités attendues du système. En cas de conflits, une négociation est envisagée pour amener les gens à un accord.
- **La Gestion des exigences** est exécutée tout au long du processus d'IE. Cette étape consiste à suivre l'évolution ou le changement des exigences, à faire la traçabilité et le contrôle de différentes versions de ces exigences durant tout leur cycle de vie de l'exigence, c'est-à-dire durant tout le processus d'IE.

I.4.6 Méthodes de représentation de la phase d'IE :

La modélisation d'un système prend en charge le processus d'analyse et de conception en introduisant un degré de formalité dans la façon dont les systèmes sont définis.

Elle permet non seulement de définir une syntaxe standard, mais aussi de fournir un support pour comprendre et communiquer les idées liées au développement du système.

Parmi les méthodes de représentation de l'IE nous citons : les méthodes orientées objets, les méthodes formelles, les méthodes orientées but, scénario et les méthodes orientées points de vue.

I.4.6.1. Les méthodes dirigées par les buts :

Les approches conventionnelles centrées sur le QUOI formalisent les contraintes imposées aux opérations et aux objets du système mais ne disent pas POUQUOI ces contraintes sont imposées et si elles sont suffisantes pour garantir que les objectifs que l'organisation place dans le système seront satisfaits. Yue a sans doute été le premier à argumenter en faveur de l'introduction de modèles de buts dans le document de spécification des exigences afin d'introduire un critère de complétude des exigences : une spécification des exigences est complète si elle permet de satisfaire le but qu'elle affine [15].

D'une façon générale, un but se définit comme un objectif que le système futur doit garantir par la coopération appropriée d'agents avec le système.

Les buts sont formulés à différents niveaux d'abstraction, allant de buts stratégiques de haut niveau tels que *Servir davantage de passagers* pour un service de transport de passagers par trains ou *fournir un service de retrait d'argent* pour un réseau de Guichets Automatiques Bancaires (GAB) à des buts techniques de bas niveau tels que activer la commande d'accélération à temps pour le système de transport ou absorber la carte de crédits après trois tentatives de saisie de code incorrectes pour le GAB.

Outre leur rôle essentiel pour la découverte des exigences fonctionnelles et non fonctionnelles, les buts ont montré leur utilité et leur importance dans d'autres activités de l'IE :

- Vérifier la complétude de la spécification des exigences : l'ensemble des buts fournit un critère de mesure de la complétude d'une spécification d'exigences; celle-ci est complète si l'on montre que l'ensemble des buts peuvent être satisfaits par l'ensemble des exigences de la spécification.
- Expliquer les exigences aux parties prenantes : les buts, notamment de haut niveau, représentent les justifications du développement du système. Une exigence n'existe que s'il y a un but qui justifie sa présence dans la spécification.
- Structurer la collection des exigences : le mécanisme de réduction des buts y contribue.
- Identifier et gérer les conflits : les différents acteurs du processus d'IE apportent des points de vue intéressants sur le système à développer; on sait que ces points de vue peuvent être divergents et générer des conflits [9]. Les buts peuvent aider à expliciter des conflits et à les résoudre

Cependant, l'expérimentation de ces approches sur le terrain a montré leurs limites : Parfois, le concept de but est flou et s'il est admis que l'on démarre le développement d'un système avec un certain nombre d'objectifs en tête, la question 'd'où viennent les buts ?' se pose.

I.4.6.2. Les méthodes à base de scénario :

Un scénario peut être défini comme "ordre des actions ou des événements pour un cas spécifique d'une certaine tâche générique qu'un système doit accomplir"[16].

Les scénarios permettent de capturer des exemples, des scènes, des descriptions narratives du contexte et du fonctionnement du système, des cas d'utilisation et des exemples de comportement d'utilisateur.

Une étude de Weidenhaupt (1998) montre que les scénarios sont des artefacts importants utilisés pour satisfaire de nombreux propos, avec différents contenus exprimés à différents niveaux d'abstraction et utilisant différentes notations [16].

Cependant, la majeure partie des approches actuelles se focalise sur la description des fonctionnalités du futur système et vise donc à la découverte des exigences fonctionnelles.

Les scénarios sont perçus comme étant plus faciles à utiliser que les buts réputés abstraits, mais leurs avantages s'accompagnent d'inconvénients : La réflexion qui conduit à spécifier un scénario tend à mettre l'accent sur des détails au détriment d'une vision générale.

I.4.6.3. Les méthodes orientées points de vue :

Dans la recherche, comme dans la pratique, de nombreuses notions sont utilisées dans un sens commun, sans qu'apparaisse le besoin réel de les formaliser davantage. C'est le cas pour la notion de "point de vue". L'émergence de ce concept dans le domaine de l'ingénierie des exigences nous fait suggérer qu'il est intéressant d'analyser ce concept et ses articulations théoriques.

La notion de *Point de vue*, appelée aussi *Vue*, revêt des significations diverses suivant les domaines et les travaux en Informatique. En général, on s'intéresse aux points de vue dès lors qu'on modélise des systèmes complexes faisant intervenir une grande masse de données, nécessitant la coopération de plusieurs experts de différents domaines de connaissances et points d'intérêts et s'adressant à une vaste panoplie d'utilisateurs.

Une approche basée sur les points de vue de l'ingénierie des exigences reconnaît que toutes les informations sur les exigences d'un système requises ne peuvent être découvertes en considérant le système à partir d'un point de vue unique. Au contraire, nous avons besoin de recueillir et d'organiser les exigences d'un certain nombre de points de vue différents. Un point de vue est une encapsulation de l'information partielle sur les exigences d'un système. Les informations des différents points de vue doivent être intégrées pour former la spécification du système final [17].

Parmi les méthodes orientées points de vue dans le domaine de l'IE, nous citons les méthodes suivantes:

I.4.6.3.1. VOSE :

ViewPoint Oriented System Engineering, VOSE [18] [19], a été développé à « Imperial College » à Londres, au début des années 90 par Finkelstein, Nuseibeh et al. (1992) comme un framework pour l'intégration des méthodes de développement dans les systèmes composés (composite systems). Ce sont des systèmes qui nécessitent la combinaison de différentes technologies et exigent le travail d'experts dans différents domaines d'application, chacun d'eux ayant son propre intérêt pour le système. Inévitablement ces intérêts se recouvrent, demandant un grand effort pour la gestion et la coordination des spécifications. Cependant, ces domaines de recouvrement ne sont pas facilement identifiables. Une fois que la connaissance dans chaque domaine technologique est représentée de différentes manières, plusieurs stratégies pour le

développement sont adoptées. Pour être comparés, les travaux des domaines distincts peuvent ne pas être dans la même étape de développement.

La méthode utilise les points de vue pour diviser les activités et la connaissance des participants concernant le système. Chaque point de vue est défini en tant qu'acteur (ou rôle dans le contexte de développement du système) et son point de vue sur le système, représenté par sa connaissance partielle du système (ses responsabilités). Cette information est organisée dans un point de vue et représentée par un arrangement (schéma) de cinq dimensions : style, domain, specification, work plan et work record.

- La dimension **Style** décrit le modèle de représentation utilisé par le point de vue.
- La dimension **Work plan** spécifie les actions de développement, le processus et la stratégie du point de vue.
- La dimension **Domain** décrit le sujet de préoccupation du point de vue dans le système global en cours de développement.
- La dimension **Spécification** précise le domaine du point de vue dans la notation décrite dans la dimension Style et développée en utilisant la stratégie décrite dans la dimension Work plan.
- La dimension **Work record** enregistre l'état de développement et l'historique des spécifications du point de vue. C'est le moyen d'assurer la traçabilité des spécifications.

La dimension Work Plan constitue le noyau du point de vue. On y spécifie les actions de contrôle contenant des actions disponibles au développeur pour vérifier la cohérence des spécifications. Les actions de contrôle sont de deux natures : intra ou inter points de vue (in/out-viewpoint checks). Les premiers consistent à vérifier la cohérence des spécifications à l'intérieur du point de vue tandis que les seconds contrôlent la cohérence des spécifications d'un point de vue avec celles des autres.

I.4.6.3.2. VORD :

VORD [20] a été développée à l'université Lancaster et proposée par Sommerville et Kotonya (1996). C'est une méthode pour la validation préalable des spécifications, principalement dédiée aux systèmes interactifs et à la résolution de points de vue. VORD introduit des points de vue focalisés sur les problèmes utilisateur et les préoccupations d'ordre organisationnel. Le modèle adopté pour les points de vue est orienté service ; les points de vue jouent un rôle analogue aux clients dans un système client-serveur. Les points de vue de VORD sont regroupés en deux classes :

- Les Points de vue directs correspondent directement aux clients du fait qu'ils reçoivent des services du système et lui envoient des informations de contrôle et des données. Ce sont des opérateurs/utilisateurs du système ou des sous-ensembles connectés au système à analyser.
- Les points de vue indirects ont un « intérêt » sur une partie ou sur tous les services qui sont fournis par le système mais qui n'interagissent pas directement avec lui. Les points de vue indirects produisent des spécifications qui contraignent les services fournis aux points de vue directs.

Les points de vue sont structurés dans une hiérarchie de classification pour satisfaire les variations des spécifications des utilisateurs. Un point de vue de VORD encapsule un ensemble d'attributs qui aident à définir et structurer les spécifications. Brièvement, un point de vue template comporte les composants suivants :

- **Identifier** attribue un numéro de référence unique à chaque point de vue.

- **Label** et description décrivent le nom et le rôle du point de vue.
- **Type** trace l'origine du point de vue à un point de vue indirect ou direct.
- **Attributes** caractérisent le point de vue dans le domaine de problème.
- **Requirements** dressent la liste des conditions du point de vue.
- **Event scenarios** décrivent l'interaction entre le point de vue et le système.
- **History** décrit l'évolution des spécifications du point de vue.

I.4.6.3.3. PREView :

PREview [21] (Process and Requirements Engineering Viewpoints) (Sommerville & Sawyer 1997) est une méthode qui a été développée dans le cadre du projet REAIMS (Requirements Engineering Adaptation and IMprovement for Safety and dependability). Son objectif était d'améliorer le processus d'élicitation au sein de quatre partenaires industriels.

L'ensemble des exigences résultant de l'application de cette méthode est :

- probablement complet;
- mutuellement consistant;
- compatible avec les préoccupations prédéfinies.

L'identification et l'élaboration des préoccupations “concerns” sont les premières activités à exécuter dans le processus PREview.

Brièvement, une préoccupation peut être définie comme une condition non négociable dont la satisfaction est essentielle pour l'entreprise.

Voici quelques caractéristiques complémentaires des préoccupations :

- Elles sont souvent non-fonctionnelles.
- Elles sont souvent formulées de façon vague, difficile à quantifier, et difficile à valider.
- Elles sont souvent une caractéristique intrinsèque au domaine du futur système. Par exemple la *sécurité* pour un système de transport.

Les préoccupations doivent être identifiées par un analyste au début du projet avec la participation des autres parties prenantes.

Une fois identifiées, les préoccupations doivent être élaborées en une forme qui est directement applicable.

Un point de vue dans PREview est représenté par une structure à six composants :

- **Le nom** de point de vue.
- **La portée** “focus” du point de vue : décrit la perspective ou le sujet du point de vue.
- **Les préoccupations** applicables au point de vue. On peut ne pas avoir de préoccupations dans un point de vue s'il peut être démontré qu'elles n'ont pas d'effet contraignant sur ce point de vue.
- **Les sources** du point de vue : elles définissent explicitement les sources des exigences liées à ce point de vue. Elles peuvent être des individus, des rôles, des groupes ou encore des documents.
- **Les exigences** du point de vue : c'est l'ensemble des exigences découvert par l'analyse du système à partir de la portée et des préoccupations applicables à ce point de vue.

- **L'historique** du point de vue. Ce composant contient les modifications apportées aux informations enregistrées dans le point de vue au fil du temps. Par exemple, on peut expliquer ici pourquoi une préoccupation particulière n'est plus applicable au point de vue.

I.4.6.4. L'intérêt d'utiliser des points de vue :

- Les parties prenantes d'un système sont hétérogènes - il n'y a pas un utilisateur typique.
- Les points de vue peuvent organiser les exigences d'un système à partir de classes d'utilisateurs différentes et d'autres parties prenantes.
- Différents types d'informations sont nécessaires pour spécifier les exigences d'un système, y compris des informations sur le domaine d'application, des informations sur l'environnement, des informations techniques, des informations sur le développement du système. Les points de vue peuvent être utilisés pour collecter et classer ces informations.
- Les points de vue peuvent être utilisés comme un moyen de structuration du processus d'élucidation des exigences.
- Les points de vue peuvent être utilisés pour encapsuler des modèles différents du système dont chacun fournit des informations de spécification.
- Les points de vue peuvent être utilisés pour structurer la description des exigences et exposer les conflits entre les différentes exigences [17].

I.5. Les niveaux de modélisation et la méta-modélisation :

La Figure ci-dessous présente les quatre niveaux de modélisation pour les produits et les processus tels que définis par l'OMG [22]. Les produits représentent le résultat à atteindre et les processus représentent le chemin à parcourir pour atteindre le résultat.

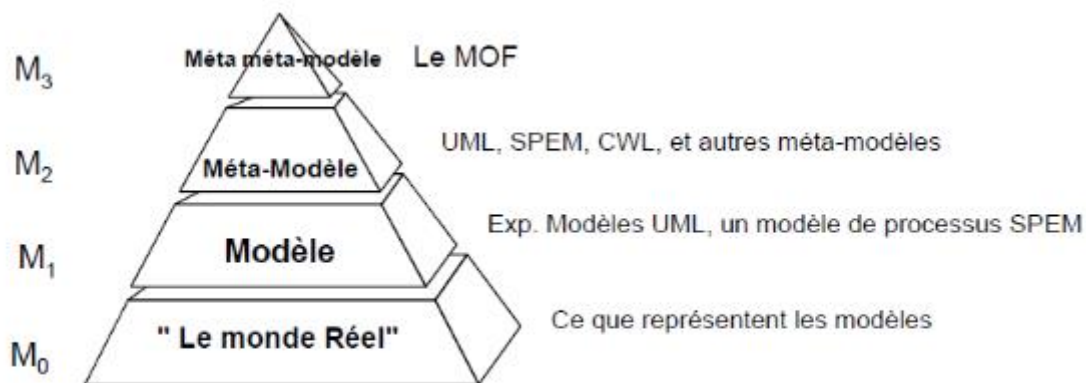


Figure I. 3 Les 4 niveaux de modélisation produit et processus

Dans cette figure on observe 4 niveaux de modélisation :

Le premier niveau M0 représente le niveau des instances, c'est-à-dire les objets du monde réel représentés dans le système dans notre cas, les processus d'ingénierie de SI.

Le niveau M1 est le niveau des modèles produit, ce sont des diagrammes de classes par exemple.

Le niveau M2 est le niveau du méta-modèle, le plus courant étant UML. UML définit les concepts qui pourront être instanciés pour créer des modèles composés de classes, d'associations et d'attributs, entre autres.

Le niveau M3 est le niveau de la méta-méta- modélisation (MOF par exemple).

Notre travail se base sur une approche nommée VpCIS (Viewpoint for Cooperative Information System) structuré selon les trois niveaux de modélisation M2, M1, M0.

I.6. Problématique :

Afin de spécifier et définir les besoins et exigences d'un SIC il est nécessaire de définir les concepts de base d'un SIC et de prendre en considération le problème de complexité de ce dernier.

Les méthodes qui existent dans l'élicitation des exigences d'un SIC tel que les méthodes orientées buts et scénarios représentent bien les concepts de base mais ne prennent pas en considération le problème de complexité. En revanche les méthodes orientées points de vue, elles ont été conçu pour répondre à la complexité des systèmes, mais nous avons constaté que ces méthodes tendent à définir chaque point de vue de manière isolée et la coopération n'est donc pas représentée.

Une approche orientée points de vue VpCIS, a été proposé par l'équipe de recherche, dans le but de représenter les concepts de base pour la définition de la phase d'analyse des besoins d'un SIC et de le décomposer selon ses différents points de vue et ainsi réduire le problème de complexité.

Le travail confié lors de ce stage, consiste à réaliser un outil qui permet de valider et de mettre en place cette approche. Cet outil permet alors de définir la phase d'analyse des besoins d'un SIC et servira de spécification pour la phase suivante qui est l'Ingénierie des systèmes du processus de développement d'un SI.

I.7. Conclusion :

Après avoir mis le projet dans son cadre et après avoir mis en place une démarche de développement qui nous aidera tout au long du projet, nous pouvons ainsi entamer la prochaine étape c'est-à-dire décortiquer le cahier de charges pour analyser les besoins qui s'y trouvent et ainsi les spécifier.

Chapitre II : Analyse et Conception

II.1. Introduction :

Une bonne conception mène généralement à la réussite du travail, l'étape de la réalisation n'est qu'un fruit d'une bonne analyse, celle-ci s'avère donc primordiale et fondamentale dans la totalité du travail de notre projet.

Dans le chapitre précédent, nous avons acquis les connaissances nécessaires pour concevoir notre solution, nous allons présenter en détail les différentes étapes de conception de notre système.

Dans ce chapitre, nous présentons dans un premier lieu une approche orientée point de vue sur laquelle nous nous sommes appuyés pour implémenter et réaliser notre outil, nous présentons ensuite la conception de l'outil et pour finir une étude de l'existant.

II.2. Approche VpCIS :

L'approche nommée VpCIS [5] (Viewpoints for Cooperative Information System) est définie selon les trois niveaux de modélisation de l'OMG comme suit:

- M2, ce niveau représente un méta-modèle orienté points de vue pour la définition des exigences d'un SIC.
- M1, ce niveau représente une instanciation du méta-modèle, et correspond au modèle de points de vue.
- M0, ce niveau représente une instanciation des points de vue du niveau M1 sur un exemple réel de SIC.

Nous détaillons dans ce qui suit chaque niveau.

II.2.1. Niveau de modélisation M2 :

La figure suivante représente le méta-modèle orienté points de vue pour la définition des besoins d'un SIC:

Dans ce méta-modèle, le SIC est décomposé en points de vue où chaque point de vue est décomposé en niveau d'abstraction.

Le niveau d'abstraction est un concept du génie logiciel, définit afin de réduire la complexité des systèmes, donnant la possibilité à l'architecte ou à l'analyste d'examiner les différents aspects du système à différents niveaux de détail.

Chaque niveau d'abstraction est composé des concepts de base : équipe, acteur, activité, information et interaction.

Nous détaillons, ci-dessous, les concepts définis par le méta-modèle:

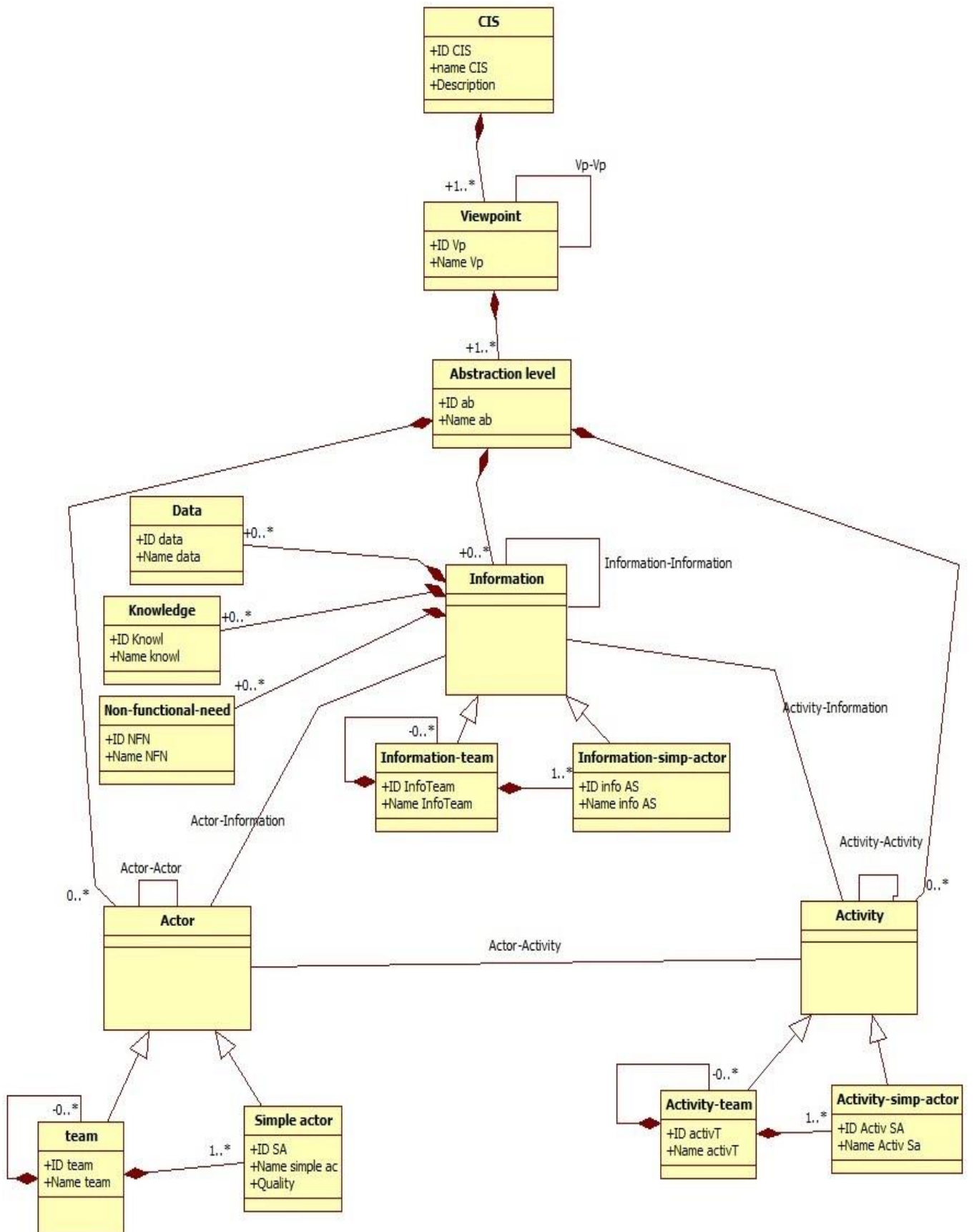


Figure II. 4 Méta-modèle des exigences d'un SIC orienté points de vue

- **SIC** : correspond au système d'information coopératif.
- **VpCIS viewpoint** : le SIC est décomposé en point de vue VpCIS, où un point de vue définit un aspect du système, l'ensemble de ces points de vue et les interactions entre ces derniers forment le SIC.
- **Niveau d'abstraction** : un point de vue est décomposé en niveau d'abstraction pour avoir un niveau de détail.
- **Actor** : appelé aussi "agent", "participant" ou "utilisateur", l'acteur est une entité (personne), membre d'une organisation ou équipe chargée de réaliser les activités qui lui sont confiées. Comme nous nous focalisons sur les SIC, le concept d'équipe est très important, peut être décomposé en sous équipe et composé de simple acteur (un acteur simple fait forcément partie d'une équipe).
- **Activity** : une activité est une étape d'un processus au cours de laquelle une action élémentaire (c'est-à-dire non décomposable) est exécutée. Elle définit les fonctionnalités du SIC. Une activité peut être une activité-simple réalisée par un acteur simple ou une activité-équipe réalisée par une équipe. Une activité-équipe est décomposée en activité simple (une activité simple fait forcément partie d'une activité-équipe). Une activité équipe peut être décomposée en sous activité équipe.
- **Information** : Une information peut être une information simple dont a besoin un acteur simple pour accomplir une activité ou une information-équipe dont a besoin une équipe pour accomplir une activité-équipe. L'information est composée de:
 - *Data* : L'objet manipulé durant la réalisation d'une activité.
 - *Knowledge* : Les connaissances nécessaires dont a besoin un acteur pour réaliser une activité.
 - *Non functional need* : caractérise une propriété ou une qualité désirée du système tel que la performance, la robustesse, la maintenance...
 - Une information-équipe peut être décomposée en information-simple (une information simple fait forcément partie d'une information-équipe). Une information équipe peut être décomposée en sous information équipe.
- **Interaction:**
 - Viewpoint - Viewpoint: représente un lien de coopération entre deux points de vue
 - Actor-Actor: représente la coopération entre deux acteurs. Comme nous distinguons deux types d'acteur (équipe et acteur simple), nous déduisons alors trois types de lien:
 - *Simple actor - Simple actor*: représente une interaction de coopération entre deux acteurs simples de la même équipe ou de deux équipes différentes.
 - *Team – Team*: représente une interaction entre deux équipes, nous déduisons deux type d'interaction: coopération ou sous équipe.
 - *Team - Simple actor*: représente une interaction de coopération entre équipes et acteur simple.
 - Actor- Activity: Un acteur réalise une activité, nous déduisons deux types de lien:
 - *Simple actor - Simple actor-activity*: un acteur simple réalise une activité simple
 - *Team - Team activity*: une équipe réalise une activité-équipe.

- Actor-Information: un acteur a besoin d'information pour réaliser son activité, nous distinguons trois types de lien:
 - *Simple actor - Simple actor information*: un acteur simple utilise une information simple.
 - *Team - Team information*: une équipe utilise une information-équipe
 - *Simple actor - Team information*: un acteur simple utilise une information d'une équipe.
- Information-Information: On peut avoir plusieurs interactions entre deux informations tel que l'association, la composition Nous déduisons trois types de lien entre:
 - *Simple actor Information - Simple actor Information*: lien de coopération
 - *Team Information - Team Information*: nous déduisons deux types de lien: coopération ou sous information équipe.
 - *Team Information - Simple actor Information*: lien de coopération.
- Activity-Information: Une activité a besoin d'information, nous déduisons deux types de lien:
 - *Simple Activity - Simple Information*: une activité simple a besoin d'information simple.
 - *Simple Activity - Team Information*: une activité simple a besoin d'information équipe.
 - *Team Activity - Team Information*: une activité-équipe a besoin d'information-équipe.
- Activity-Activity: On peut avoir plusieurs interactions entre deux activités tel que l'association, la composition Nous déduisons trois types de lien entre:
 - *Simple actor Activity - Simple actor Activity*: lien de coopération.
 - *Team Activity - Team Activity* : nous déduisons deux types de lien: coopération ou sous activité équipe.
 - *Team Activity - Simple actor Activity*: lien de coopération.

Nous passons maintenant au niveau de modélisation M1 où nous instancions le modèle de point de vue à partir du méta-modèle du niveau M2.

II.2.2. Niveau de modélisation M1 :

Le modèle de point de vue VpCIS est une instantiation du méta-modèle et a la forme suivante :

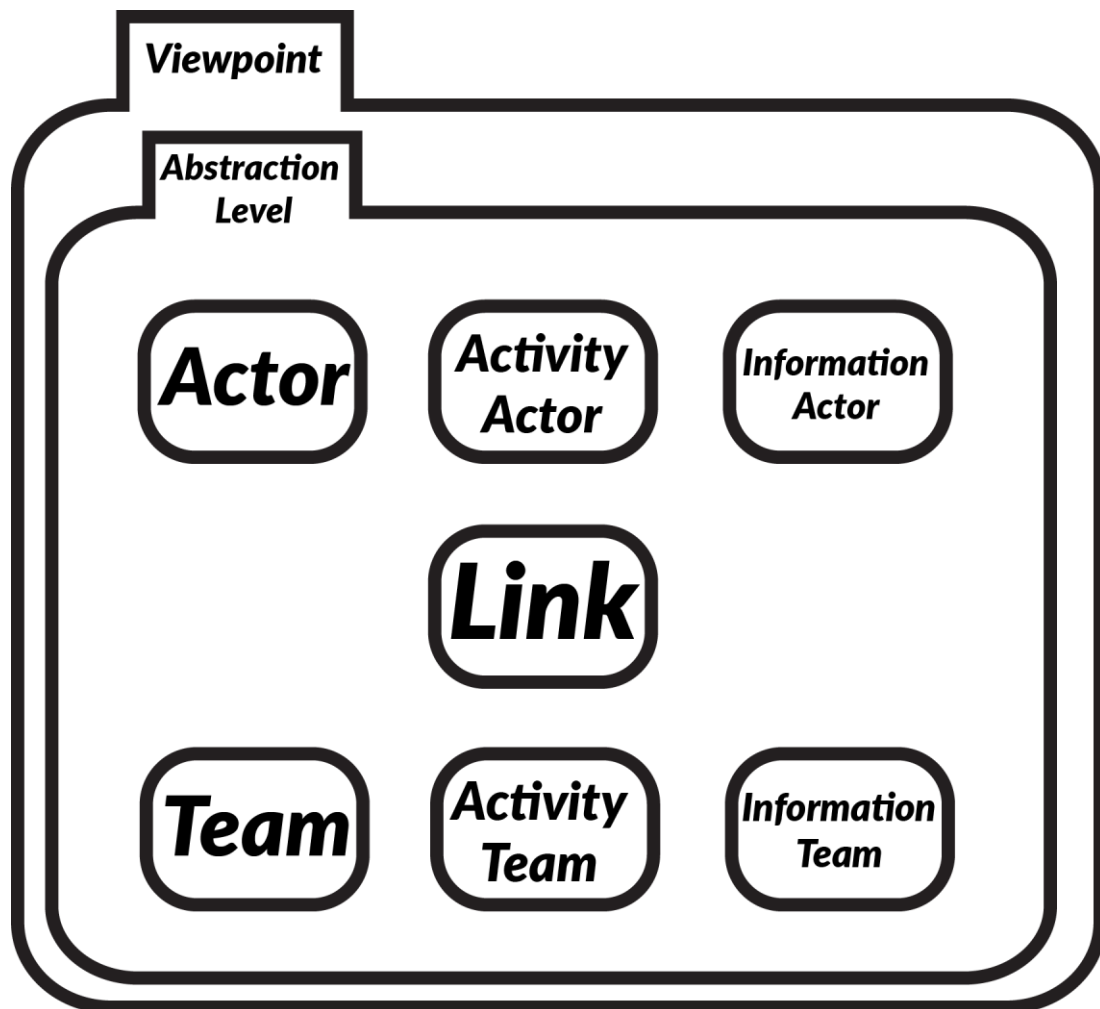


Figure II. 5 Modèle de point de vue VpCIS au niveau M1

- Niveau d'abstraction :
 - Equipe:
 - Activité Equipe:
 - Information Equipe:
 - Données Equipe:
 - Connaissances Equipe:
 - Besoins non fonctionnels Equipe:
 - Acteur:
 - Activité Acteur:
 - Information Acteur:
 - Données Acteur:
 - Connaissances Acteur:
 - Besoin non fonctionnels Acteur:

- Links:
 - Link équipe-équipe:
 - Link équipe-Activité équipe:
 - Link équipe-Information Equipe:
 - Link Activité Equipe - Information Equipe:
 - Link Activité Equipe- Activité Acteur:
 - Link Information Equipe - Information Equipe:
 - Link Acteur -Acteur:
 - Link Acteur -Activité Acteur:
 - Link Acteur - Information Acteur:
 - Link Acteur - Information équipe:
 - Link Activité Acteur - Information Acteur:
 - Link Activité Acteur - Information Equipe:
 - Link Activité Acteur - Activité Acteur :
 - Link Information Acteur - Information Acteur:
 - Link équipe - Acteur :
 - Link Activité équipe - Activité Acteur :
 - Link information équipe - information acteur:

Un point de vue est alors composé de niveau d'abstraction, où chaque niveau d'abstraction est composé des concepts de base: Equipe, Acteur, Activité, Information et Interaction.

II.2.3. Niveau de modélisation M0 :

Le niveau de modélisation M0 représente une instanciation des points de vue du niveau M1 sur une étude de cas réelle.

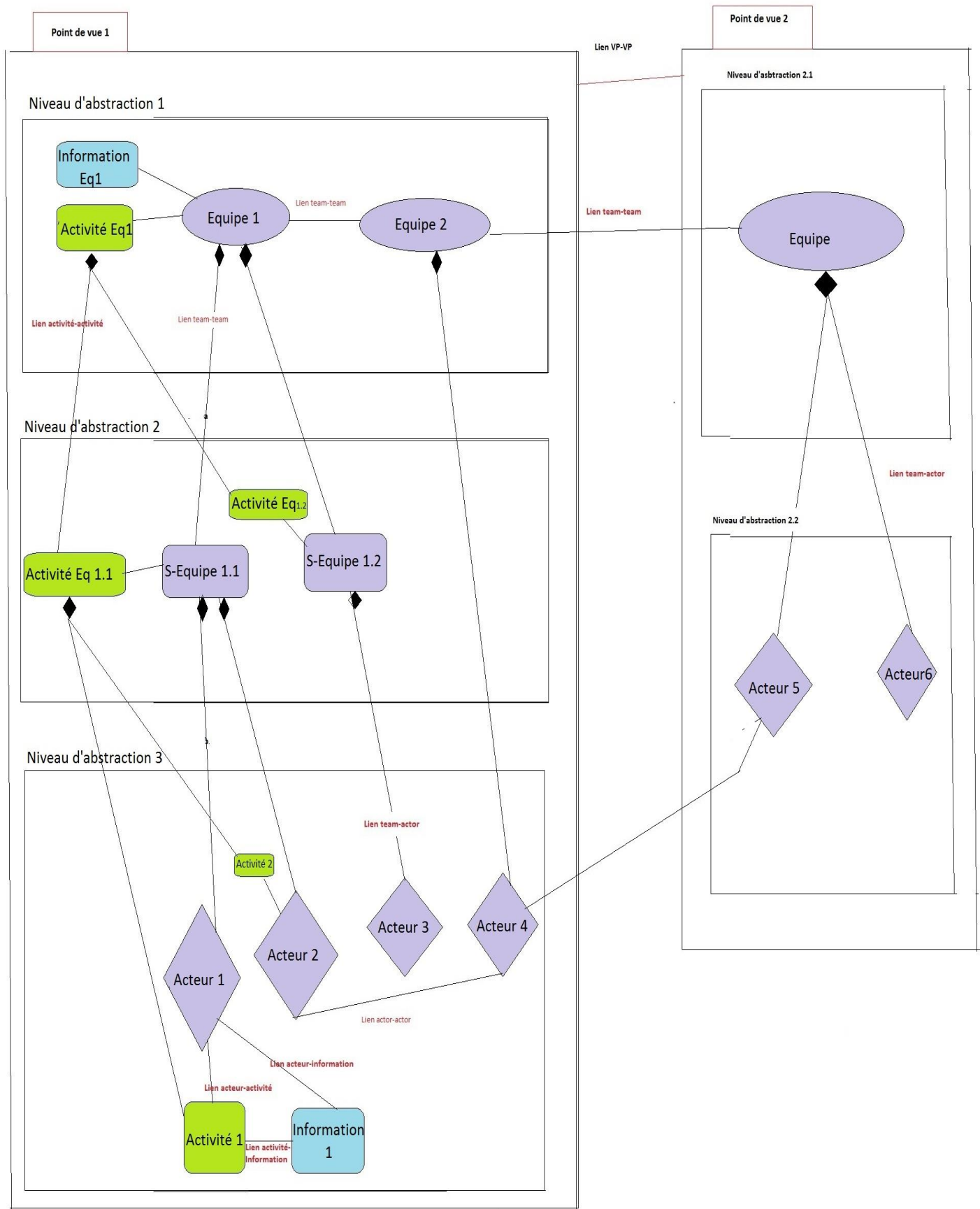


Figure II. 6 Instanciation de point de vue VpCIS au niveau M0

II.3. Conception de l'outil :

Afin d'implémenter cette approche, nous proposons un outil réalisé avec java et la base de données Access. Nous avons dans un premier temps créé la base de donnée en créant une table pour chaque classe du méta-modèle, par la suite, nous avons créé une interface java pour chaque classe pour permettre à l'utilisateur de remplir les informations concernant le SIC à travers ces interfaces.

II.3.1. Définition des classes :

Les classes équipe, acteur, activité équipe, activité acteur, information équipe, information acteur, sont définis de manière indépendantes et sont reliés par les classes Link.

- CIS (**idCIS**, nameCIS, description)
- Viewpoint (**idViewpoint**, nameViewpoint, idCIS*)
- AbstractionLevel (**idAbstractionLevel**, nameAbstractionLevel, idViewpoint*)
- Team (**idTeam**, nameTeam, idAbstractionLevel*)
- Actor (**idActor**, **quality**, nameActor, idTeam*, idAbstractionLevel*)
- ActivityTeam (**idActivityTeam**, nameActivityTeam, idAbstractionLevel*)
- ActivityActor(**idActivityActor**, nameActivityActor, idActivityTeam*, idAbstractionLevel*)
- InformationTeam(**idInformationTeam**, nameInformationTeam, idAbstractionLevel*)
- InformationActor(**idInformationActor**, nameInformationActor, idInformationTeam*, idAbstractionLevel*)
- DataTeam (**idDataTeam**, nameDataTeam, idInformationTeam*)
- DataActor (**idDataActor**, nameDataActor, idInformationActor*)
- NonFunctionalRequirementTeam(**idNonFunctionalRequirementTeam**, nameNonFunctionalRequirementTeam, idInformationTeam*)
- NonFunctionalRequirementActor(**idNonFunctionalRequirementActor**, nameNonFunctionalRequirementActor, idInformationActor*)
- KnowledgeTeam (**idKnowledgeTeam**, nameKnowledgeTeam, idInformationTeam*)
- KnowledgeActor (**idKnowledgeActor**, nameKnowledgeActor, idInformationActor*)
- LinkViewpointViewpoint (**id LinkViewpointViewpoint**, sourceIdViewpoint(idViewpoint*), destinationIdViewpoint (idViewpoint*))
- Link Team-Team(**idLinkTeamTeam**, typeLink, sourceIdTeam(idTeam*), destinationIdTeam (idTeam*)) (on a deux type de liens: subTeam ou cooperation)
- Link Actor-Actor (**idLinkActorActor**,sourceIdActor (idActor*), sourceQuality (quality*), destinationIdActor(idActor*), destinationQuality(quality*))
- LinkTeam-Actor(**idLinkTeamActor**,sourceIdTeam(idTeam*), destinationIdActor(idActor*),destinationQuality(quality*))
- Link Team-ActivityTeam (**idLinkTeamActivityTeam**, sourceIdTeam(idTeam*), destinationIdActivityTeam(idActivityTeam*))

- Link Team-InformationTeam (**idLinkTeamInformationTeam**, sourceIdTeam (idTeam*), destinationIdInformationTeam (idInformationTeam*))
- Link ActivityTeam-InformationTeam (**idLinkActivityTeamInformationTeam**, sourceIdActivityTeam (idActivityTeam*), destinationIdInformationTeam (idInformationTeam*))
- Link ActivityActor-InformationActor (**idLinkActivityActorInformationActor**, sourceIdActivityActor (idActivityActor*), destinationIdInformationActor (idInformationActor*))
- Link ActivityActor-InformationTeam (**idLinkActivityActorInformationTeam**, sourceIdActivityActor (idActivityActor*), destinationIdInformationTeam (idInformationTeam*))
- Link Actor-ActivityActor (**idLinkActorActivityActor**, sourceIdActor(idActor*), sourceQuality (quality*) destinationIdActivityActor (idActivityActor*))
- Link Actor-InformationActor (**idLinkActorInformationActor**, sourceIdActor (idActor*), sourceQuality(quality*), destinationIdInformationActor(idInformationActor*))
- Link Actor-InformationTeam (**idLinkActorInformationTeam**, sourceIdActor (idActor*), sourceQuality(quality*), destinationIdInformationTeam(idInformationTeam*))
- Link ActivityTeam-ActivityTeam (**idLinkActivityTeamActivityTeam**, typeLink, sourceIdActivityTeam (idActivityTeam*), destinationIdActivityTeam (idActivityTeam*))
- LinkActivityActor-ActivityActor(**idLinkActivityActorActivityActor**, sourceIdActivityActor(idActivityActor*),destinationIdActivityActor (idActivityActor*))
- Link ActivityTeam-ActivityActor (**idLinkActivityTeamActivityActor**, sourceIdActivityTeam(idActivityTeam*), destinationIdActivityActor (idActivityActor*))

II.3.2. Identification des attributs :

Les attributs sont détaillés dans le tableau suivant: (AN : Alphanumérique, N : Numérique)

Codification	Désignation	Type	Longueur
IdCIS	Identificateur D'un CIS	N	10
nameCIS	Nom D'un CIS	AN	225
Description	Description d'un CIS	AN	Texte long
idViewpoint	Identificateur d'un Viewpoint	N	10
nameViewpoint	Nom d'un Viewpoint	AN	225
idAbstractionLevel	Identificateur d'un niveau d'abstraction	N	10
nameAbstractionLevel	Nom d'un niveau d'abstraction	AN	225
idActor	Identificateur d'un Acteur	N	10

nameActor	Nom d'un Acteur	AN	225
idTeam	Identificateur d'une Equipe	N	10
nameTeam	Nom d'une Equipe	AN	225
IdActivityActor	Identificateur d'une Activité d'un Acteur	N	10
nameActivityActor	Nom d'une Activité d'un Acteur	AN	225
IdActivityActor	Identificateur d'une Activité d'un Acteur	N	10
nameActivityActor	Nom d'une Activité d'un Acteur	AN	225
IdActivityTeam	Identificateur d'une Activité d'une Equipe	N	10
nameActivityTeam	Nom d'une Activité d'une Equipe	AN	255
idInformationTeam	Identificateur d'une information d'une équipe	N	10
nameInformationTeam	Nom d'une information d'une équipe	AN	225
idInformationActor	Identificateur d'une information d'un Actor	N	10
nameInformationActor	Nom d'une information d'un Actor	AN	225
idDataTeam	Identificateur d'une Donnée d'équipe	N	10
nameDataTeam	Nom d'une Donnée d'équipe	AN	225
idDataActor	Identificateur d'une Donnée d'un acteur	N	10
nameDataActor	Nom d'une Donnée d'un acteur	AN	225
IdNonFunctionalRequirementTeam.	Identificateur d'un besoin non fonctionnel d'une équipe	N	10
nameNonFunctionalRequirementTeam	Nom d'un besoin non fonctionnel d'une équipe	AN	225
idNonFunctionalRequirementActor	Identificateur d'un besoin non fonctionnel d'un acteur	N	10
nameNonFunctionalRequirementActor	Nom d'un besoin non fonctionnel d'un acteur	AN	225

idKnowledgeTeam	Identificateur d'une connaissance d'une équipe	N	10
nameKnowledgeTeam	Nom d'une connaissance d'une équipe	AN	225
idKnowledgeActor	Identificateur d'une connaissance d'un Acteur	N	10
nameKnowledgeActor	Nom d'une connaissance d'un Acteur	AN	225

II.4. Présentation d'UML :

En regardant les objectifs fixés pour la réalisation du projet, nous remarquons que nous sommes face à une application modulaire et qui devra rester ouverte pour les améliorations futures. De ce fait, il est très important d'utiliser un langage universel pour la modélisation afin de clarifier la conception et de faciliter les échanges. Notre choix s'est porté sur le langage UML puisqu'il convient pour toutes les méthodes objet et se prête bien à la représentation de l'architecture du système.

UML (Unified Modeling Language), est un langage de modélisation des systèmes standard, qui utilise des diagrammes pour représenter chaque aspect d'un système ie: statique, dynamique, en s'appuyant sur la notion d'orienté objet qui est un véritable atout pour ce langage.

UML se positionne exclusivement comme un ensemble de formalismes. Il faut y associer une démarche et une organisation pour constituer une méthode.

UML permet de couvrir le cycle de vie d'un logiciel depuis la spécification des besoins jusqu'au codage en offrant plusieurs moyens de description et de modélisation des acteurs et de utilisation système, du comportement des objets, du flot de contrôle internes aux opérations, des composants d'implémentation et leurs relations, de la structure matérielle et de la distribution des objets et des composants indépendamment des techniques d'implémentation et peut être mis à jour selon les besoins.

Les avantages :

- Universel.
- Adopté par les grandes entreprises.
- Notation unifié
- Facile à comprendre.
- Adopté par plusieurs processus de développement
- Limite les risques d'erreur.
- N'est pas limité au domaine informatique.

Après cette définition d'UML, nous présentons les diagrammes UML que nous avons défini pour notre conception.

II.4.1. Diagramme de cas d'utilisation globale

Cas d'utilisation : un cas d'utilisation représente un ensemble de séquences d'actions à réaliser par le système et produisant un résultat observable intéressant pour un acteur particulier représenté par des ellipses et limité par un rectangle pour représenter le système.

Ci-dessous, nous présentons le diagramme de cas d'utilisation pour la compréhension du fonctionnement du système.

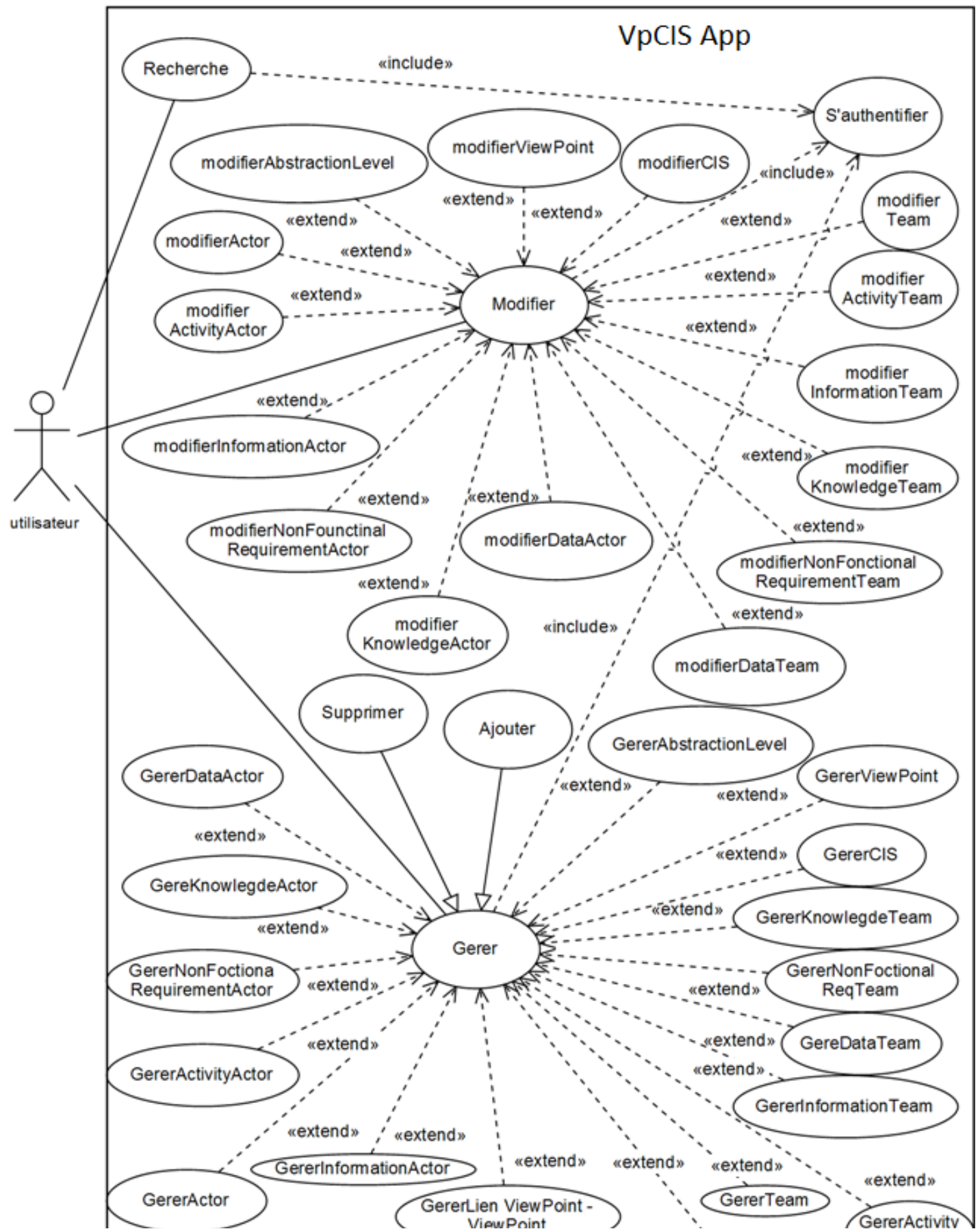


Figure II. 7 Diagramme de cas d'utilisation "utilisation de l'outil VpCIS"

II.4.1.1 Description du diagramme de Use Case :

L'utilisateur une fois authentifié, il peut ajouter ou supprimer un SIC, un point de vue, un niveau d'abstraction, les concepts de base « Acteur » et « Equipe », et les liens entre points de vue.

Il peut modifier un SIC, un point de vue, un niveau d'abstraction, les concepts de base « Acteur » et « Equipe » sauf le concept de base Link.

Il peut également effectuer une recherche.

II.4.2. Diagramme de séquence :

Un diagramme de séquence montre les interactions entre les objets, arrangées en séquence dans le temps.

II.4.2.1. Diagramme de séquence d'authentification :

Dans un souci de sécurité de notre système, et la sécurisation de l'information; avant d'effectuer une tâche quelconque une authentification doit être effectuée. Le diagramme de séquence de la figure suivante décrit les étapes détaillées de phase d'authentification d'un utilisateur.

L'utilisateur doit d'abord saisir son identifiant et son mot de passe. Le système vérifie par la suite ses informations et lui autorise l'accès en cas de réussite (correspondances trouvé).

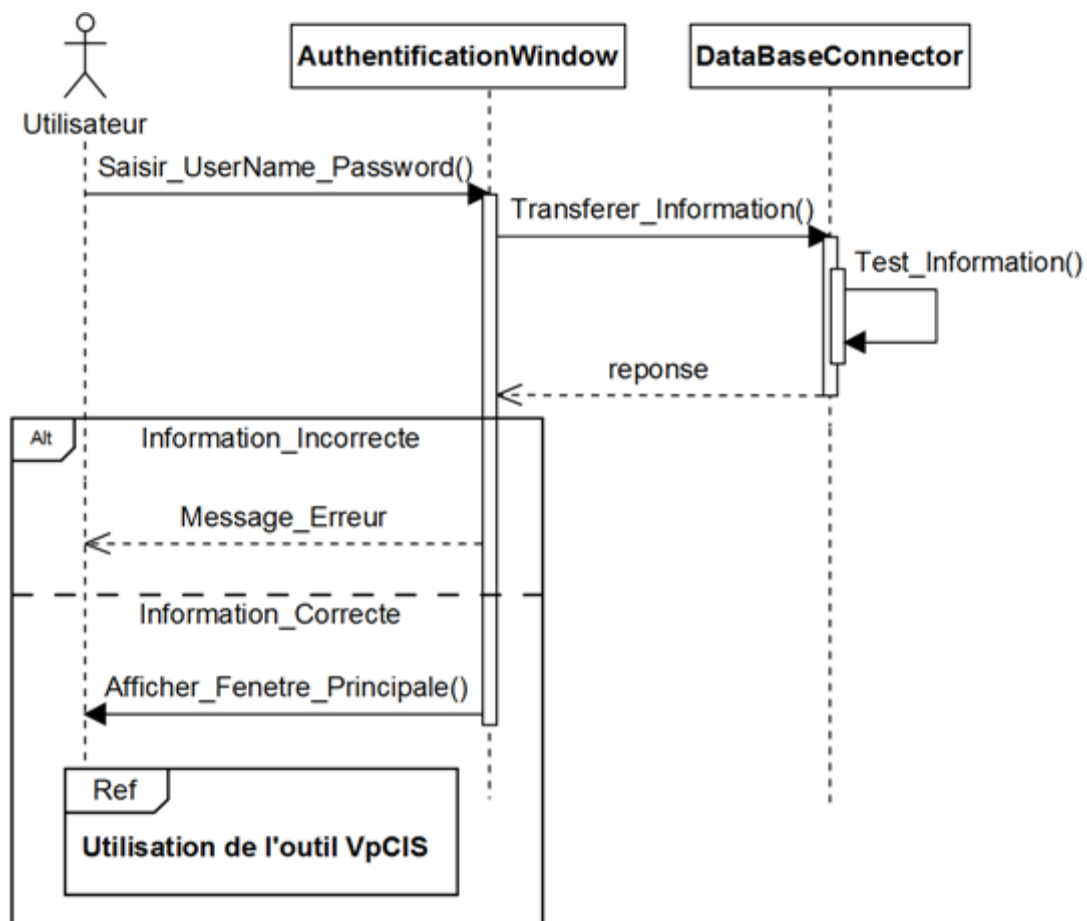


Figure II. 8 Diagramme de séquence de l'Authentification

II.4.2.2. Diagramme de séquence de création :

La figure suivante nous montre la succession des étapes pour créer les différents objets manipulables pendant l'utilisation de l'outil VpCIS.

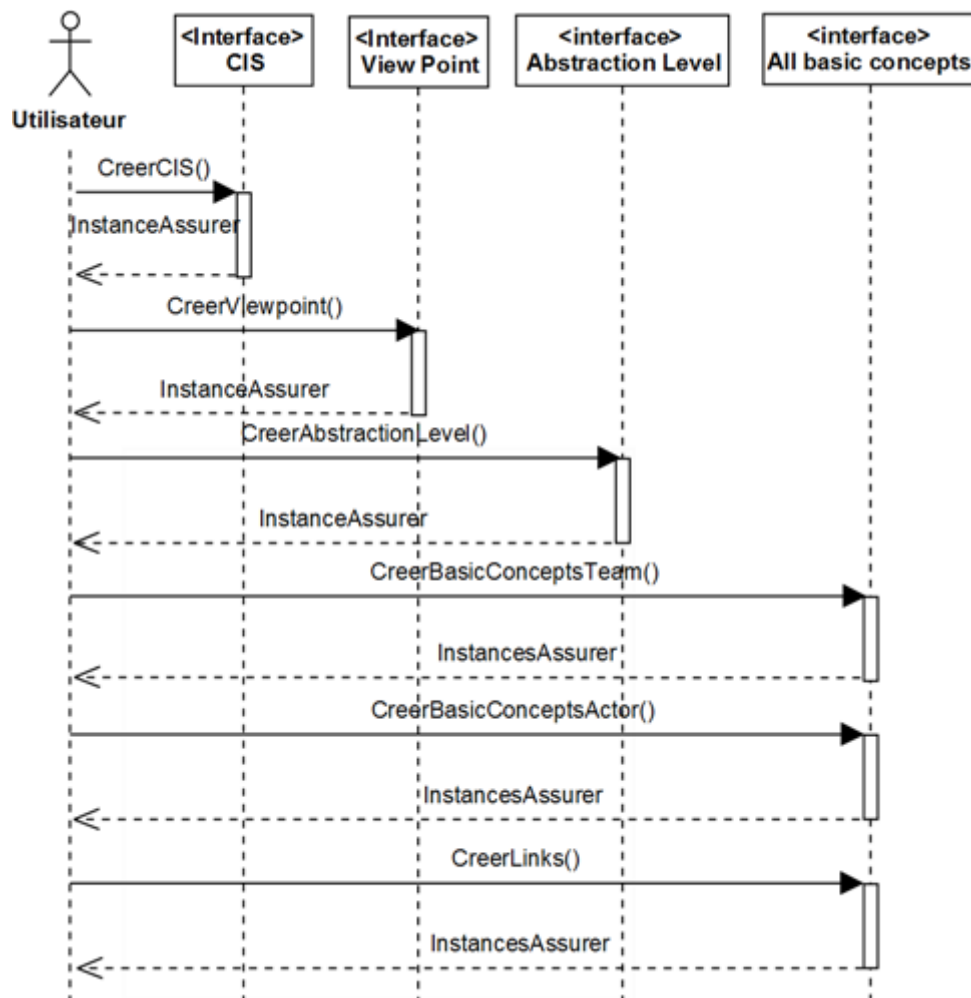


Figure II. 9 Diagramme de séquence de création

II.4.3. Diagramme d'activité :

Le diagramme d'activité permet de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables. Il donne une vision des activités propres à une opération ou à un cas d'utilisation.

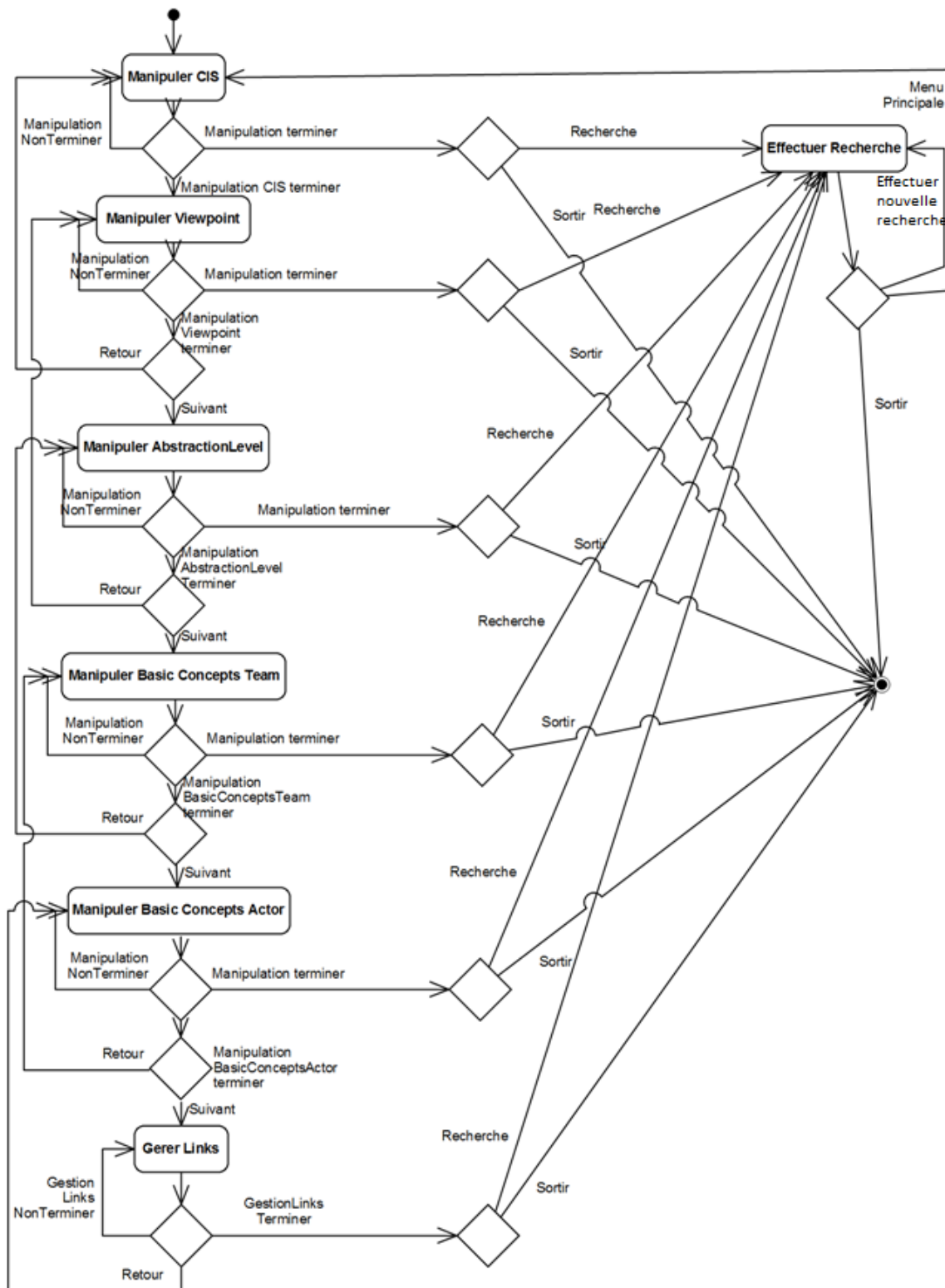


Figure II. 10 Diagramme d'activité de l'outil VpCIS

Il faut noter que Manipuler inclus Ajouter, Mettre à jour et Supprimer tant dis que Gérer inclus qu' Ajouter et Supprimer seulement.

II.5. Etude de l'existant :

Le travail coopératif que nous illustrons, consiste en un système intégré de gestion de l'identité numérique d'un citoyen, ce travail va être réalisé par les équipes du laboratoire LSI.

Notre objectif est de définir la phase d'analyse des besoins de ce SIC en utilisant l'outil que nous proposons.

II.5.1. Description de l'étude de cas :

La carte d'identité électronique (eID) contient une puce électronique qui protège les données personnelles, de manière optimale, et permet de répondre à de nombreuses questions administratives, depuis un terminal. L'objectif est le déploiement d'une solution d'administration électronique basée sur l'identification par une carte à puce. L'objectif principal réside en la conception et mise en œuvre d'une plateforme de services sécurisé basé identité par carte à puce.

Les équipes SI, Sécurité, GL, Cloud, mobilité et veille sociale du laboratoire LSI, sont amenés à coopérer afin de réaliser une plateforme e-gouvernement qui supporte divers services basés identité entre autre l'E-santé basée identité.

Les aspects (points de vue) à considérer sont les suivants:

Identité numérique: il s'agit de définir une plateforme e-gouvernement qui supporte divers services basés identité.

Point de vue 1: E-santé basée identité: c'est la projection de la plateforme sur le domaine de santé. Il y aura la prise en charge des objets communicants pour l'e-santé. (Réalisé par l'équipe SI)

Point de vue 2: Sécurité de service basés identité: il s'agit de sécuriser les services de la plateforme ainsi que les données relatives. (Réalisé par l'équipe Sécurité)

Point de vue 3: Développement de logiciels: la plateforme engage toutes les structures étatiques et privés pour la fourniture de services au citoyen. C'est un environnement Cloud par excellence. Il s'agit de développer un modèle logiciel agile et adaptable à différents environnement. (Réalisé par l'équipe GL)

Point de vue 4: Big Data/ Cloud: les données générées par la plateforme e-gouvernement sont de tailles importantes. Il s'agit de mettre en place une solution permettant le stockage et l'accès à l'information de manière rationnelle. Les solutions Cloud permettent d'offrir des services performants sur ces données dans une plateforme de haute performance payable à l'utilisation. (Réalisé par l'équipe Big data/ Cloud)

Point de vue 5: Validation de la plateforme: la validation de la plateforme est un préalable pour sa mise en service. La modélisation et la vérification de la plateforme constitue un gage de garantie de son déploiement.

Point de vue 6: Veille sociale et veille sociétale: l'objectif de la veille sociale est de détecter et ou prévoir les conflits afin de réajuster des décisions pour l'amélioration de la vie du groupe donc de la plateforme et l'amélioration de la rentabilité.

N'ayant pas toutes informations concernant tous les points de vue, nous allons décrire uniquement les points de vue: **E-santé basée identité, Sécurité de service basés identité et Big data/ Cloud.**

II.5.2. Description des points de vue :

II.5.2.1. Point de vue 1 : E-santé basée identité :

L'équipe SI est chargée du dossier médical partagé d'un citoyen (DMP). Le DMP est un dossier numérique traçant toute l'histoire de la vie d'un patient, depuis son identité, à ses diverses pathologies, ou toute autre information intéressante tant le patient lui-même, que le médecin que l'administratif de la santé. Il permet une transmission de données de manière sécurisée.

La constitution d'un DMP nécessite la mise en place en amont des SIS. Les SIS ou les Systèmes d'Information de la Santé représente un concept incontournable pour le développement de ce domaine.

L'objectif est de mettre en place un environnement de développement aboutissant à l'élaboration d'un ensemble de contenus pour la réalisation d'un DMP numérique.

II.5.2.2. Description du Point de vue E-santé basée identité :

Le point de vue E-santé est composé de deux niveaux d'abstraction: Niveau 1 pour décrire l'équipe, et le niveau 2 où nous avons plus de détails sur l'équipe, c'est à dire les acteurs de l'équipe SI.

II.5.2.2.1. Niveau d'abstraction 1 :

- **Equipe:** SI (composée de Alimazighi et Azzouz).
- **Activité Equipe:** Réaliser un SIS pour la constitution d'un DMP numérique (cette *activité-équipe* est décomposée en *activité-acteur*: conception d'un SIS et *activité-acteur*: réalisation de la base de données).
- **Information Equipe:** Info DMP E-santé (cette information-équipe est décomposée en *information-acteur* : DMP patient et *information-acteur* : Info Conception du SIS).
 - **Données Equipe:** format du dossier médical du patient.
 - **Connaissances Equipe:** Système d'information, base de données.
 - **Besoins non fonctionnels Equipe:** Optimisation du traitement du patient, baisse des coûts, meilleure qualité de service.
- **Links:**
 - **Link équipe-Activité équipe:** L'équipe SI est chargée de Réaliser un SIS pour la constitution d'un DMP numérique.
 - **Link équipe-Information Equipe:** L'équipe SI a besoin de Info DMP E-santé.
 - **Link Activité Equipe - Information Equipe:** Pour réaliser un SIS pour la constitution d'un DMP numérique, on a besoin de Info DMP E-santé.

II.5.2.2.2. Niveau d'abstraction 2 :

- **Acteur1:** Alimazighi (chef d'équipe).
- **Activité Acteur:** Conception du SIS.
- **Information Acteur:** DMP patient.
 - **Donnée Acteur :** DMP patient.

- **Connaissance Acteur:** Système d'Information, base de données.
- **Besoin non fonctionnels Acteur:** réalisation de la conception dans les délais.
- **Links:**
 - **Link Acteur -Activité Acteur:** Alimazighi réalise la conception du SIS.
 - **Link Acteur - Information Acteur:** Alimazighi a besoin du DMP du patient pour réaliser la conception du SIS.
 - **Link Activité Acteur - Information Acteur:** Pour réaliser la conception du SIS, on a besoin de DMP du patient.
- **Acteur 2:** Azzouz (membre simple).
- **Activité Acteur:** Réalisation de la base de données.
- **Information Acteur:** Info Conception du SIS.
 - **Donnée Acteur:** diagramme de classe du SIS.
 - **Connaissances Acteur:** Base de données, SQL.
 - **Besoins non fonctionnels Acteur:** Intégrité de la base donnée, respecter les délais.
- **Links:**
 - **Link Acteur - Acteur:** Azzouz et Alimazighi (nous avons mis une contrainte si on veut ajouter le lien Alimazighi -Azzouz, ça ne marche pas vu que le lien existe déjà (Azzouz et Alimazighi) pour éviter la répétition).
 - **Link Acteur - Activité Acteur:** Azzouz réalise la base donnée.
 - **Link Acteur - Information Acteur:** Azzouz a besoin du diagramme de classes du SIS pour réaliser la base de données.
 - **Link Activité Acteur - Information Acteur:** Pour réaliser la base de données on a besoin Info Conception SIS.

II.5.2.3. Point de vue 2 : Sécurité :

Nous avons un deuxième point de vue Sécurité, où l'équipe Sécurité est chargée d'assurer la sécurité de la base de donnée E-santé de tel manière à afficher les données nécessaires tout en protégeant la vie privée du citoyen, elle est donc en coopération avec l'équipe SI, nous avons alors une coopération entre le point de vue E-santé et le point de vue sécurité.

II.5.2.4. Description du Point de vue Sécurité :

Le point de vue Sécurité est composé de deux niveaux d'abstraction: Niveau 1 pour décrire l'équipe, et le niveau 2 où nous avons plus de détails sur l'équipe, c'est à dire les acteurs de l'équipe Sécurité.

II.5.2.4.1. Niveau d'abstraction 1 :

- **Equipe:** Sécurité (composée de Belkhir et Bouyakoub).
- **Activité Equipe:** Sécuriser la base de donnée (*l'activité équipe* est décomposée en *activité acteur*: proposer une solution qui affiche que les données nécessaires d'un patient et *activité acteur*: protéger la base de donnée d'éventuelles attaques).
- **Information Equipe:** Info donnée des patients.

- **Données Equipe:** Donnée des patients.
- **Information Equipe:** Info équipe sécurité (*Information-équipe* est décomposée en *information acteur*: Info sécurité vie privée et *information-acteur*: Info protection de donnée)
 - **Donnée Equipe:** donnée de sécurité.
 - **Connaissances Equipe:** Réseau et sécurité.
 - **Besoins non fonctionnels Equipe:** Assurer la sécurité de la base de données en protégeant la vie privée du citoyen.
- **Links:**
 - **Link équipe-équipe:** Sécurité et SI (lien de coopération).
 - **Link équipe-Activité équipe:** L'équipe sécurité est chargée d'assurer la sécurité de la base de données E-santé.
 - **Link équipe-Information Equipe:** L'équipe Sécurité a besoin de Info donnée des patients.
 - **Link équipe-Information Equipe:** L'équipe Sécurité a besoin de Info équipe sécurité.
 - **Link Activité Equipe - Information Equipe:** Pour assurer la sécurité de la base de données, on a besoin de Info donnée des patients.

II.5.2.4.2. Niveau d'abstraction 2 :

- **Acteur1:** Belkhir (chef d'équipe et responsable du SIC).
- **Activité Acteur:** Proposer une solution qui affiche que les données nécessaires d'un patient.
- **Information Acteur:** Info sécurité vie privée.
 - **Donnée:** a besoin aussi de donnée patients qui sont définis dans Information Equipe: Info donnée des patients, nous n'allons donc pas définir une deuxième fois cette information, nous ajoutons juste un lien entre acteur - information équipe.
 - **Connaissance Acteur:** Sécurité et réseau.
 - **Besoin non fonctionnels Acteur:** assurer la sécurité des données dans les délais et protéger la vie privée des patients.
- **Links:**
 - **Link Acteur -Acteur:** Belkhir et Alimazighi.
 - **Link Acteur -Activité Acteur:** Belkhir est chargé de proposer une solution qui affiche que les données nécessaires d'un patient.
 - **Link Acteur - Information Acteur:** Belkhir a besoin de Info sécurité vie privée.
 - **Link Acteur - Information Equipe:** Belkhir a besoin de l'information équipe: Info donnée des patients.
 - **Link Activité Acteur - Information Acteur:** Pour proposer une solution qui affiche que les données nécessaires d'un patient, on a besoin de Info sécurité vie privée.
 - **Link Activité Acteur - Information Equipe:** Pour proposer une solution qui affiche que les données nécessaires d'un patient, on a besoin de Info donnée des patients.

- **Acteur2:** Bouyakoub (membre simple).
- **Activité Acteur:** Protéger la base de données d'éventuelles attaques.
- **Information Acteur:** Info protection de données.
 - **Donnée Acteur:** a besoin aussi de données patients qui sont définies dans Information Equipe: Info donnée des patients, nous n'allons donc pas définir une deuxième fois cette information, nous ajoutons juste un lien entre acteur - information équipe.
 - **Connaissance Acteur:** Sécurité et réseau.
 - **Besoin non fonctionnels Acteur:** sécurité des données.
- **Links:**
 - **Link Acteur -Acteur:** Bouyakoub et Belkhir
 - **Link Acteur -Activité Acteur:** Bouyakoub est chargé de protéger la base de données d'éventuelles attaques.
 - **Link Acteur - Information Acteur:** Bouyakoub a besoin de Info protection de donnée.
 - **Link Acteur - Information Equipe:** Bouyakoub a besoin de l'information équipe: Info donnée des patients.
 - **Link Activité Acteur - Information Acteur:** Pour protéger la base de données d'éventuelles attaques, on a besoin de Info protection de donnée.
 - **Link Activité Acteur - Information Equipe:** Pour protéger la base de données d'éventuelles attaques, on a besoin de Info donnée des patients.

II.5.2.5. Point de vue 3: Big data/ Cloud :

Exploiter et interpréter une masse de données gigantesque devient un enjeu majeur de création de valeur dans de nombreux domaines économiques. Analyser le potentiel de données issues de différentes sources, sous différentes formes, afin de produire des résultats exploitables pour améliorer le quotidien, reste un enjeu pour les années à venir. Des solutions Cloud permettent aux organismes de réduire leurs coûts d'investissement en termes de TIC. En effet, dans une solution Cloud, les acteurs ne font aucun investissement matériel ni logiciel (toute l'infrastructure matérielle et logicielle est chez le fournisseur de Cloud) et payent ces services à l'utilisation.

L'équipe Big data/ Cloud est alors chargée de réaliser des solutions Cloud, elle est en coopération avec l'équipe SI et l'équipe sécurité, nous avons alors une coopération entre le point de vue E-santé et le point de vue sécurité.

II.5.2.6. Description du Point de vue Big data/ Cloud :

Le point de vue Big data/ Cloud est composé de deux niveaux d'abstraction: Niveau 1 pour décrire l'équipe, et le niveau 2 où nous avons plus de détails sur l'équipe, c'est à dire les acteurs de l'équipe Big data/ Cloud.

II.5.2.6.1. Niveau d'abstraction 1 :

- **Equipe:** Big data/ cloud (composée de Boukhalifa, Derbal et Djiroune).
- **Activité Equipe:** Réaliser des solutions Cloud (l'activité équipe est décomposée en *activité acteur*: réalisation d'une nouvelle approche de stockage et de transformation de données

hétérogènes et *activité acteur*: Analyse de grands volumes de données hétérogènes et *activité acteur*: Optimisation des accès au Cloud).

- **Information Equipe:** Info cloud (Information Equipe est décomposée en *Information acteur*: Info stockage de donnée et *information acteur*: Info analyse de donnée et *information acteur*: Info optimisation de donnée).
 - **Données Equipe:** a besoin de donnée patients qui sont définies dans Information Equipe: Info donnée des patients que nous avons créée dans le point de vue sécurité, nous n'allons donc pas définir une deuxième fois cette information, nous ajoutons juste un lien entre équipe - information équipe).
 - **Connaissances Equipe:** Système d'Information, base de données.
 - **Besoins non fonctionnels Equipe:** Réduction du coût, sécurité des données.
- **Links:**
 - **Link équipe-équipe:** Big data/ cloud et SI (lien de coopération).
 - **Link équipe-équipe:** Big data/ cloud et Sécurité (lien de coopération).
 - **Link équipe-Activité équipe:** L'équipe Big data/ cloud est chargée de réaliser des solutions Cloud.
 - **Link équipe-Information Equipe:** L'équipe Big data/ cloud a besoin de Info Cloud.
 - **Link équipe - Information Equipe:** L'équipe Big data/ cloud a besoin de Info donnée des patients.
 - **Link Activité Equipe - Information Equipe:** Pour réaliser des solutions Cloud, on a besoin de Info cloud.

II.5.2.6.2. Niveau d'abstraction 2 :

- **Acteur1:** Boukhalifa (chef d'équipe).
- **Activité Acteur:** Réaliser une nouvelle approche de stockage et de transformation de données hétérogènes.
- **Information Acteur:** Info stockage de données.
 - **Donnée Acteur:** a besoin de donnée patients qui sont définies dans Information Equipe: Info donnée des patients que nous avons créée dans le point de vue sécurité, nous n'allons donc pas définir une deuxième fois cette information, nous ajoutons juste un lien entre acteur - information équipe
 - **Connaissance Acteur:** SI, base de données.
 - **Besoin non fonctionnels Acteur:** réaliser l'approche en un temps raisonnable.
- **Links:**
 - **Link Acteur -Acteur:** Boukhalifa et Alimazighi.
 - **Link Acteur - Acteur:** Boukhalifa et Azzouz.
 - **Link Acteur - Acteur:** Boukhalifa et Belkhir.
 - **Link Acteur -Activité Acteur:** Boukhalifa est chargée de réaliser une nouvelle approche de stockage et de transformation de données hétérogènes.
 - **Link Acteur - Information Acteur:** Boukhalifa a besoin de Info stockage de donnée.

- **Link Acteur - Information Equipe:** Boukhalfa a besoin de Info donnée des patients.
- **Link Activité Acteur - Information Acteur:** Pour réaliser une nouvelle approche de stockage et de transformation de données hétérogènes, on a besoin de Info stockage de donnée.
- **Link Activité Acteur - Information Equipe:** Pour réaliser une nouvelle approche de stockage et de transformation de données hétérogènes, on a besoin de Info donnée des patients.
- **Acteur 2:** Derbal (membre simple).
- **Activité Acteur:** Analyse de grands volumes de données hétérogènes.
- **Information Acteur:** Info analyse de données.
 - **Donnée Acteur:** a besoin de donnée patients qui sont définies dans Information Equipe: Info donnée des patients que nous avons créée dans le point de vue sécurité, nous n'allons donc pas définir une deuxième fois cette information, nous ajoutons juste un lien entre acteur - information équipe.
 - **Connaissances Acteur:** SI, Base de données.
 - **Besoins non fonctionnels Acteur :** Analyser les données en un temps raisonnable.
- **Links:**
 - **Link Acteur - Acteur:** Derbal et Boukhalfa.
 - **Link Acteur - Activité Acteur:** Derbal est chargée d'analyser de grands volumes de données hétérogènes.
 - **Link Acteur - Information Acteur:** Derbal a besoin de Info analyse de donnée.
 - **Link Acteur - Information Equipe:** Derbal a besoin de Info donnée des patients.
 - **Link Activité Acteur - Information Acteur:** pour analyser de grands volumes de données hétérogènes, on a besoin de Info analyse de donnée.
 - **Link Activité Acteur - Information Equipe:** pour analyser de grands volumes de données hétérogènes, on a besoin de Info donnée de patients.
- **Acteur3:** Djiroune (membre simple).
- **Activité Acteur:** Optimisation des accès au Cloud.
- **Information Acteur:** Info optimisation de donnée.
 - **Donnée Acteur:** a besoin de donnée patients qui sont définies dans Information Equipe: Info donnée des patients que nous avons créée dans le point de vue sécurité, nous n'allons donc pas définir une deuxième fois cette information, nous ajoutons juste un lien entre acteur - information équipe.
 - **Connaissance Acteur:** SI, base de données.
 - **Besoin non fonctionnels Acteur:** réduction des coûts.
- **Links:**
 - **Link Acteur - Acteur:** Djiroune et Boukhalfa.
 - **Link Acteur - Acteur:** Djiroune et Derbal.

- **Link Acteur -Activité Acteur:** Djiroune est chargée de l'optimisation des accès au Cloud.
- **Link Acteur - Information Acteur:** Djiroune a besoin de Info optimisation de donnée.
- **Link Acteur - Information Equipe:** Djiroune a besoin de Info donnée des patients.
- **Link Activité Acteur - Information Acteur:** Pour optimiser les accès au Cloud, on a besoin de Info optimisation de donnée.
- **Link Activité Acteur - Information Equipe:** Pour optimiser les accès au Cloud, on a besoin de Info donnée des patients.

II.6 Conclusion :

Dans ce chapitre, nous avons conçu et documenté le code que nous devons produire. Dans cette phase, toutes les questions concernant la manière de réaliser le système à développer ont été élucidées. Le produit obtenu est un modèle graphique (ensemble de diagrammes) prêt à être codé. Dans le chapitre suivant nous allons étudier en détails les outils et les langages utilisés durant la phase de construction.

Chapitre III :

Implémentation

III.1. Introduction :

Après avoir achevé l'étape de conception de l'application, on va entamer dans ce chapitre la partie réalisation et implémentation dans laquelle on s'assure que le système est prêt pour être exploité par les utilisateurs finaux. Nous allons présenter maintenant dans le cadre de la réalisation du projet l'environnement de travail de logiciel et matériel et les choix technologiques en premier lieu, puis nous exposerons quelques scénarios d'exécution à travers des captures d'écran.

III.2. Les outils de travaux :

III.2.1. Environnement matériel :

Au cours de ce projet nous avons utilisé un ordinateur portable personnel qui présente les caractéristiques suivantes :

- **Marque** : Dell.
- **Processeur** : Intel core (i7-6500U).
- **Mémoire** : 8 Go de RAM.
- **Disque dur** : 1T.
- **Système d'exploitation** : Windows 8.1.

III.2.2. Environnement logiciel :

III.2.2.1. Microsoft Access (MS Access) :

Access est un SGBD combiné à un langage de programmation qui permet de créer des applications de façon rapide. Il permet aussi le développement et l'exportation de rapports et données en format Word, Excel ou PDF et qui permet de :

- Créer des schémas relationnels et donc créer des **tables**, des contraintes sur les champs de ces tables et des contraintes référentielles entre ces tables.
- Saisir des données dans les tables, avec l'instruction standard LMD INSERT ou à travers une interface graphique composée de **formulaire**.
- Écrire des **requêtes** et des **vues** en utilisant le langage SQL.
- Réaliser des **formulaires** permettant d'alimenter ou interroger la BD.
- Réaliser des **états** permettant de mettre en forme des résultats de requête de type SELECT

Avantages :

- Rapidité de mise en œuvre.
- Facilité de maintenance ou reprise.
- Rapidité de création d'IHM.
- Langage graphique permettant un apprentissage rapide.
- Schéma de données robustes (intégrité référentielle, contraintes, type de données, etc.).
- La portabilité.

III.2.2.2. Java :

Nous avons choisi le langage Java SE1.8 pour l'implémentation de notre système. C'est un langage de programmation à usage général, évolué et orienté objet. Ses caractéristiques ainsi que la richesse de son écosystème et de sa communauté lui ont permis d'être très largement utilisé pour le développement d'applications de types très disparates. Java est notamment largement utilisée pour le développement d'applications d'entreprises et mobiles.

Avantages :

- Java est interprétée.
- Java est portable : il est indépendant de toute plate-forme il n'est pas lié à un système d'exploitation en particulier. Son code peut être interprété très facilement sur n'importe quel environnement disposant d'une machine virtuelle.
- Java est orienté objet et simple à maîtriser.
- Java est fortement typé.
- Java assure la gestion de la mémoire.
- Java est multitâche.

III.2.2.3. Eclipse :

Pour l'environnement de développement, nous avons choisi Eclipse Neon Entreprise Edition, qui est un environnement de développement intégré(IDE)

Eclipse est un environnement de développement intégré libre extensible, universel et polyvalent, permettant de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions.

III.3. Architecture générale de l'application :

III.3.1. Description des packages :

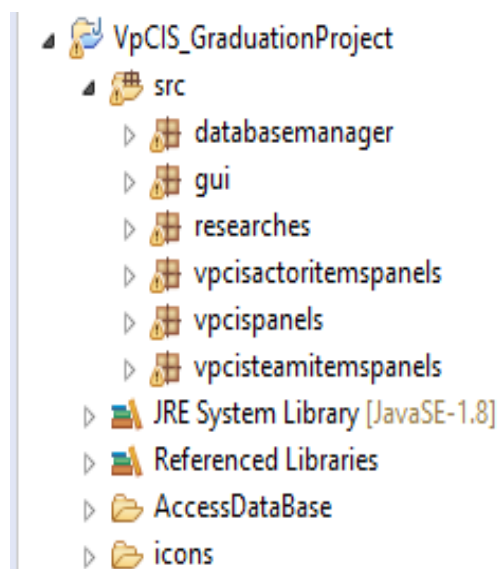


Figure III.11 Architecture générale du Projet java

Nous avons décomposé notre projet en 6 packages :

- `databasemanager` (ou Database Manager): ce package est chargé de fournir la connexion à la base de données MS Access.
- `gui` (ou Graphique User Interface) : contient l'implémentation de la fenêtre d'authentification ainsi que la fenêtre principale.
- `vpcispanels` (ou VpCIS Panels) : ce package contient les fenêtres de manipulation de CIS, Viewpoint et Abstraction Level, ce package assure l'ajout, la mise à jour et la suppression de ces derniers.
- `vpcisactoritempanels` (ou VpCIS Actor Items Panels) : ce package contient les fenêtres de manipulation des basic concepts (items) relatif à « Acteur Simple », ce package assure l'ajout, la mise à jour et la suppression de Actor, Activity Actor, Information Actor, Data Actor, Knowledge Actor, Non Functional Requirement Actor et Link (le basic concept Link exceptionnellement assure l'ajout et la suppression des liens entre les basic concepts relatif à « Acteur Simple » et les basic concepts relatif à « Equipe » qui seront défini dans le point suivant.
- `vpcisteamitempanel` (ou VpCIS Team Items Panels) : ce package contient les fenêtres de manipulation des basic concepts (items) relatif à « Equipe », ce package assure l'ajout, la mise à jour et la suppression de Team, Activity Team, Information Team, Data Team, Knowledge Team, Non Functional Requirement Team et Link (le basic concept Link exceptionnellement assure l'ajout et la suppression des liens entre les basic concepts relatif à « Equipe » et les basic concepts relatif à « Acteur Simple »
- `researches` (ou Researches) : ce package contient les recherches possibles à effectuer sur la base de données.

III.3.2. Le Diagramme de composants :

Nous décrivons dans le diagramme suivant les composants et leurs dépendances dans l'environnement de réalisation. C'est une vue statique de l'implémentation du système illustrant les choix de réalisation.

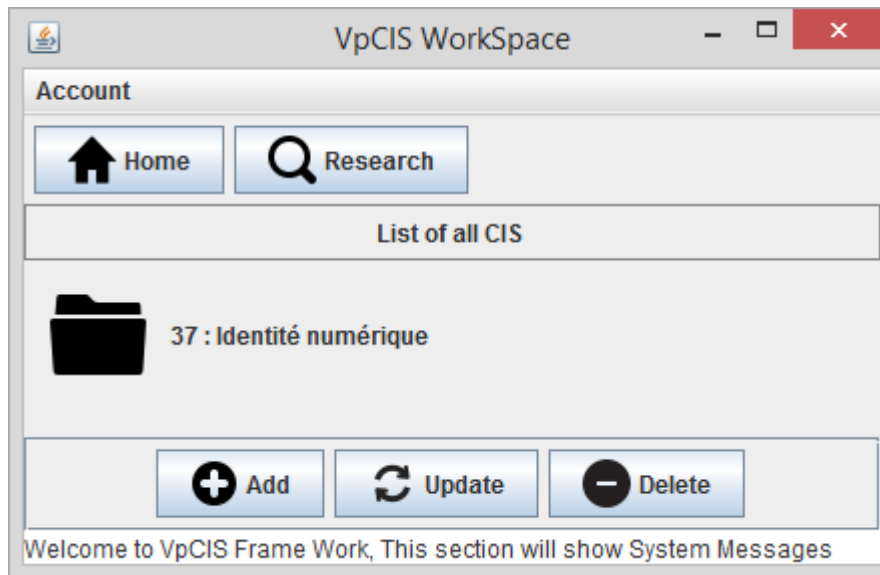


Figure III. 14 Fenêtre principale VpCIS WorkSpace

La fenêtre principale se décompose en :

- Un menu : qui nous permet de se déconnecter.
- Un panneau principal : ce dernier et lui-même décomposé en trois autres panneaux:
 - Panneau raccourcis : contient les raccourcis « Home » et « Research » (« Home » permet de revenir à n'importe quel moment à la fenêtre principale, « Research » permet d'accéder à n'importe quel moment au panneau de recherche qui sera décrit ultérieurement).
 - Panneau VpCIS : on peut l'appeler aussi le panneau principale car c'est la partie dynamique de l'application.
 - Panneau messages systèmes : contient un espace réservé pour l'affichage des messages systèmes. Il sert à notifier l'utilisateur et l'aider dans la manipulation de l'application.

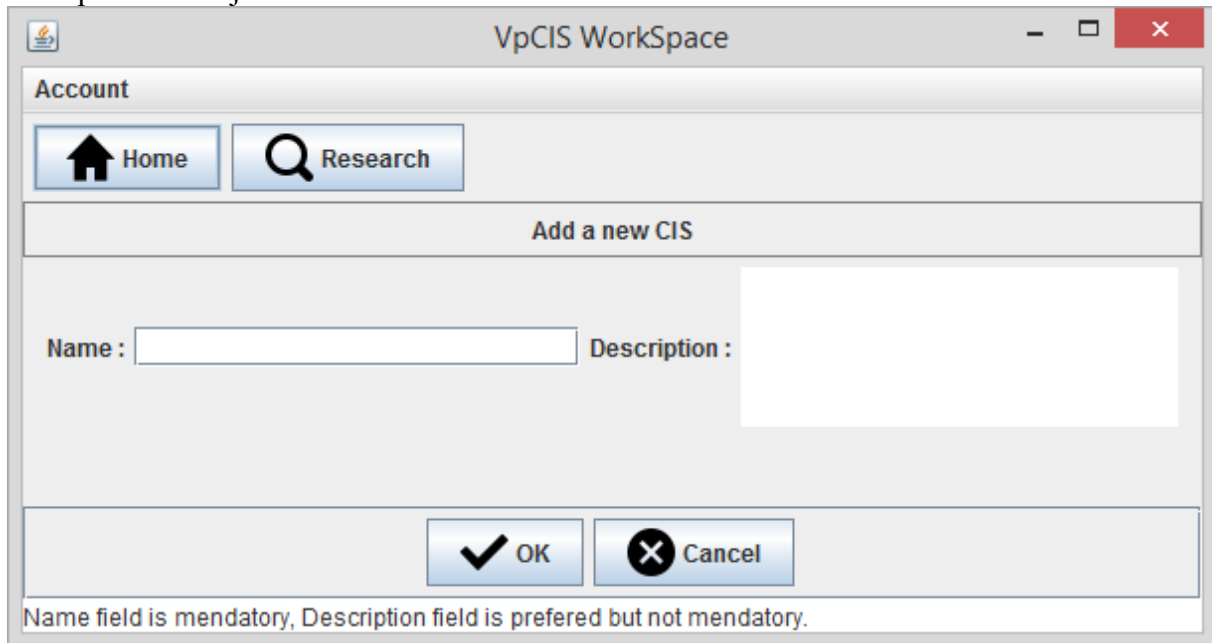
Dans ce qui suit nous allons nous concentrer sur ce qui sera afficher dans le panneau VpCIS, car tout le reste est statique et ne change pas pendant l'utilisation de l'application.

Le panneau VpCIS est décomposé en 3 autres panneaux :

- Panneau nord : affiche le titre du panneau principal.
- Panneau centre : affiche les éléments nécessaires pour travailler avec l'application.
- Panneau sud : contient les boutons nécessaires pour travailler avec l'application.

Dans la Figure III. ci-dessus, on montre la fenêtre principale, qui affiche les SIC créés. On remarque que le SIC créé est accompagné d'une icône et de [son identifiant : son nom].

Il est possible d'ajouter un nouveau SIC :

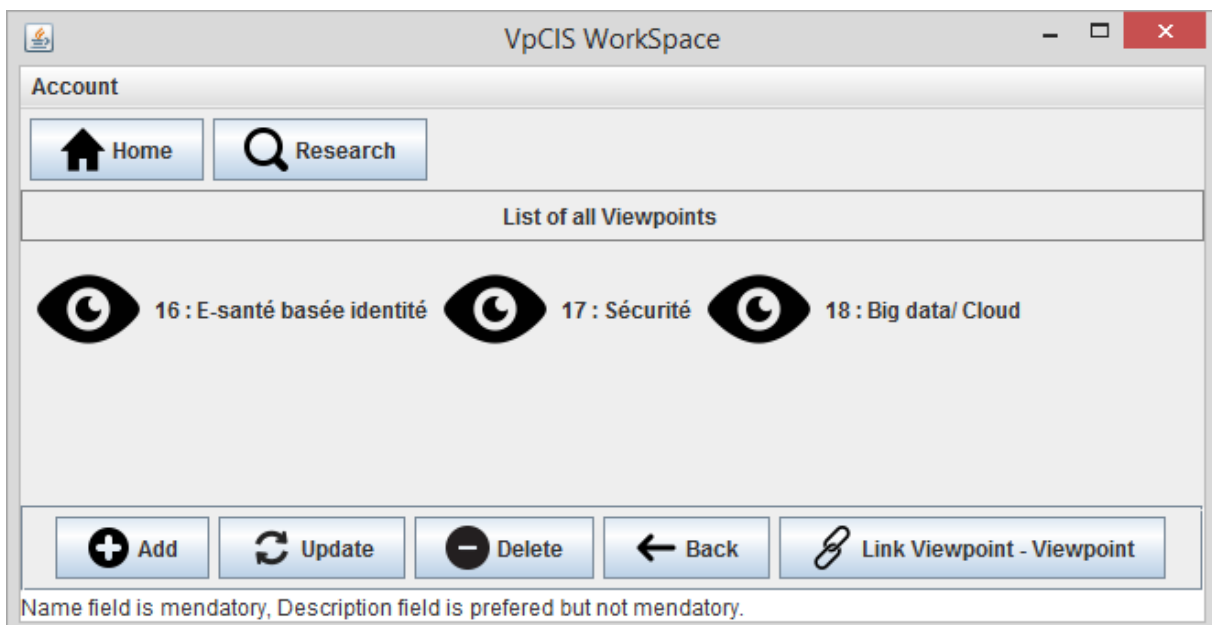


The screenshot shows a window titled "VpCIS Workspace" with a standard Windows-style title bar. Inside, there's a section labeled "Account" with two buttons: "Home" (with a house icon) and "Research" (with a magnifying glass icon). Below this is a header for "Add a new CIS". The main area contains two input fields: "Name :" followed by a text box, and "Description :" followed by a larger text area. At the bottom right are "OK" and "Cancel" buttons. A footer message states: "Name field is mandatory, Description field is preferred but not mandatory."

Figure III. 15 Ajout d'un nouveau SIC

On peut ainsi modifier ou supprimer un SIC déjà créé.

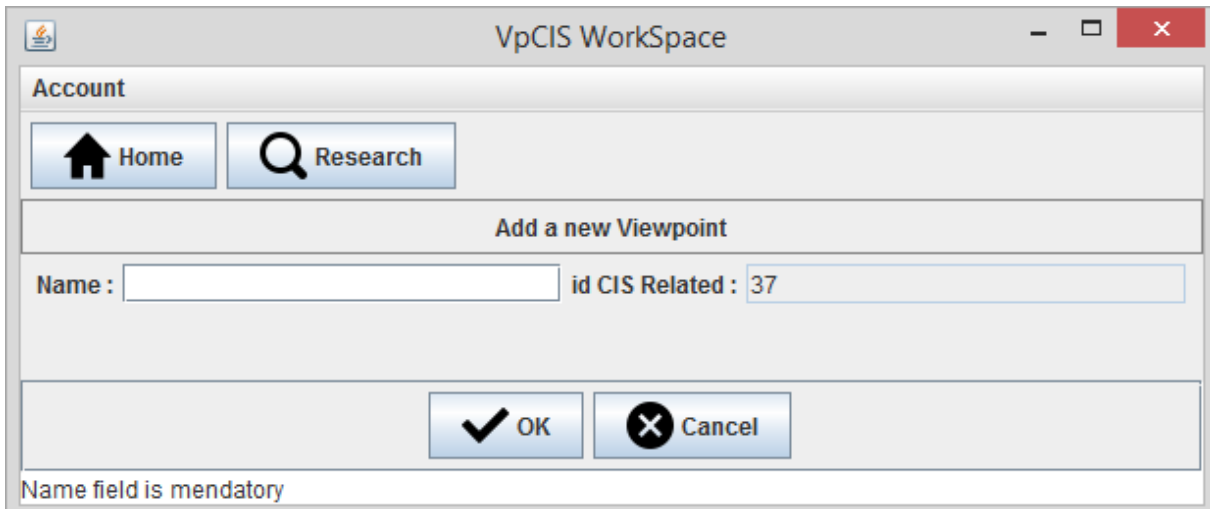
Après la création d'un SIC, on passe au panneau concernant les points de vue:



The screenshot shows the same "VpCIS Workspace" window, but now displaying the "List of all Viewpoints" section. It features the same "Account" header with "Home" and "Research" buttons. Below the header, three viewpoint icons (represented by eyes) are shown with their respective labels: "16 : E-santé basée identité", "17 : Sécurité", and "18 : Big data/ Cloud". At the bottom, there is a row of five buttons: "Add" (plus icon), "Update" (refresh icon), "Delete" (minus icon), "Back" (left arrow icon), and "Link Viewpoint - Viewpoint" (link icon). The same footer message is present: "Name field is mandatory, Description field is preferred but not mandatory."

Figure III. 16 Panneau des Points de vue

On peut ajouter un nouveau point de vue :

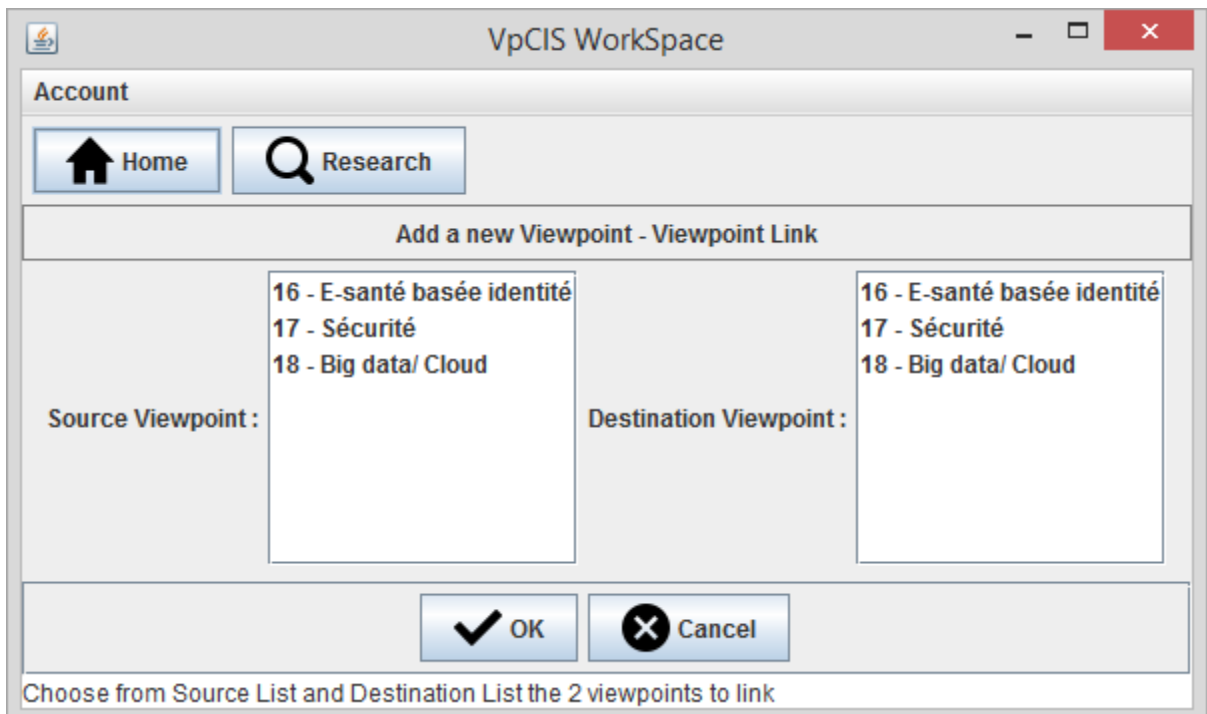


The screenshot shows a window titled "VpCIS Workspace" with a standard Windows-style title bar. Inside, there's a section labeled "Account" with two buttons: "Home" (with a house icon) and "Research" (with a magnifying glass icon). Below this is a section titled "Add a new Viewpoint". It contains two input fields: "Name :" followed by an empty text box, and "id CIS Related :" followed by a text box containing the number "37". At the bottom of this section are two buttons: "OK" (with a checkmark icon) and "Cancel" (with a close icon). A status bar at the very bottom of the window displays the text "Name field is mandatory" in red.

Figure III. 17 Ajout d'un nouveau Point de vue

On peut de même modifier ou supprimer un point de vue déjà créé.

Si le besoin exige de relier des points de vue entre eux, il est possible d'ajouter un lien entre les points de vue en cliquant sur le bouton « Link Viewpoint – Viewpoint » :



The screenshot shows a window titled "VpCIS Workspace" with the same "Account" section as the previous figure. Below it is a section titled "Add a new Viewpoint - Viewpoint Link". This section contains two large text boxes. The left box is labeled "Source Viewpoint :" and the right box is labeled "Destination Viewpoint :". Both boxes contain the same list of text: "16 - E-santé basée identité", "17 - Sécurité", and "18 - Big data/ Cloud". At the bottom of this section are "OK" and "Cancel" buttons. A status bar at the bottom of the window displays the text "Choose from Source List and Destination List the 2 viewpoints to link" in red.

Figure III. 18 Lier deux Points de vue

On donne la possibilité de supprimer un lien entre deux points de vue déjà créé.

En cliquant sur un point de vue créé, on accède au panneau concernant les niveaux d'abstraction :

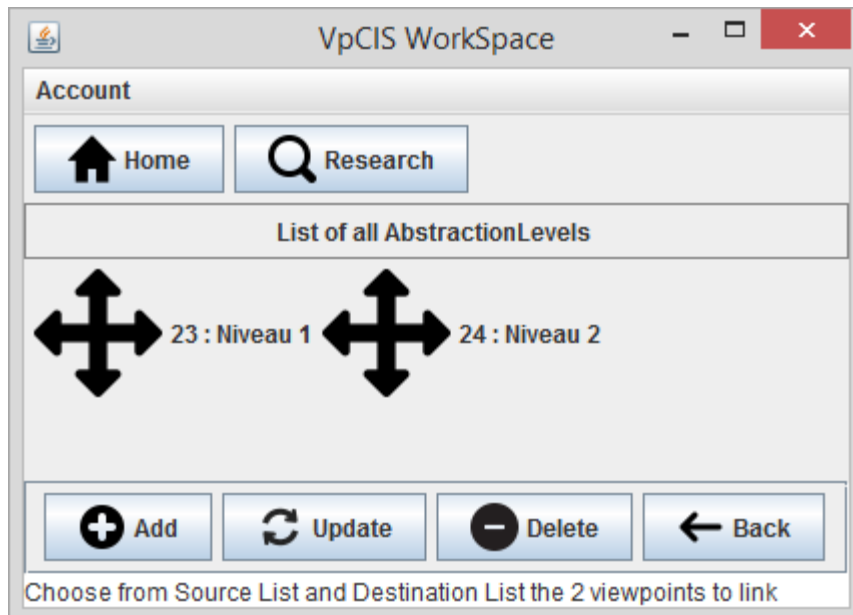


Figure III. 19 Panneau des Niveaux d'Abstraction

Pareil pour les niveaux d'abstraction. On peut ajouter un nouveau niveau d'abstraction :

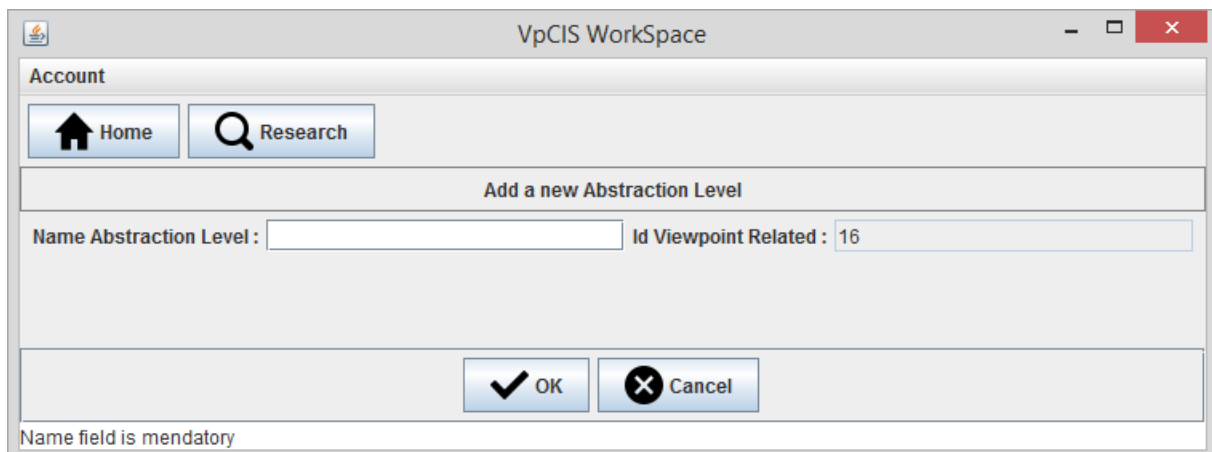


Figure III. 20 Ajout d'un nouveau Niveau d'Abstraction

Modification et suppression d'un niveau d'abstraction déjà crée est possible.

A la fin de création d'un niveau d'abstraction, on passe maintenant à la création des Concepts de base :

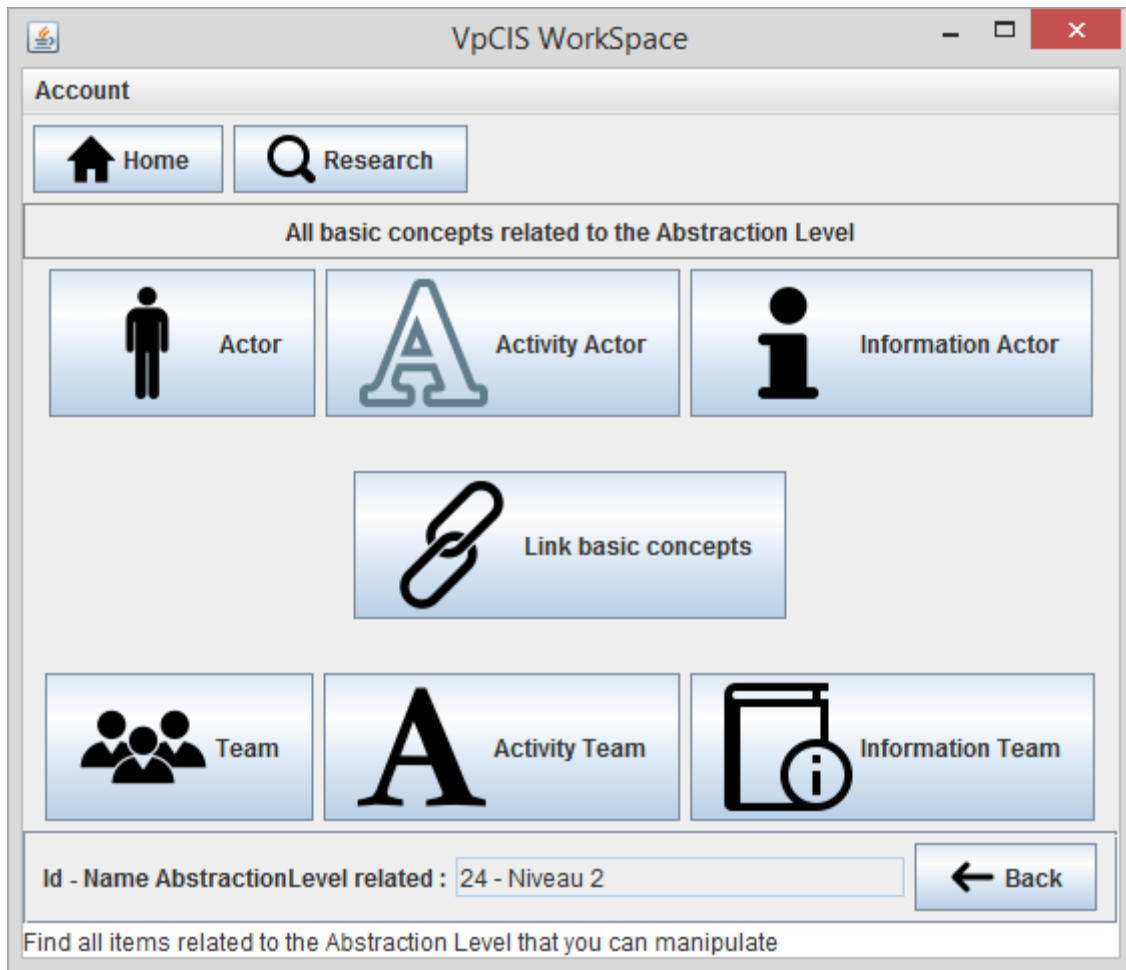


Figure III. 21 Panneau des Concepts de base du Niveau d'Abstraction

Nous allons passer par chaque groupe de Concepts de base dans l'ordre suivant :

- Concepts de base concernant « Equipe ».
- Concepts de base concernant « Acteur Simple ».
- Concepts de base « Link ».

- Concepts de base « Equipe » :
 - Equipe :

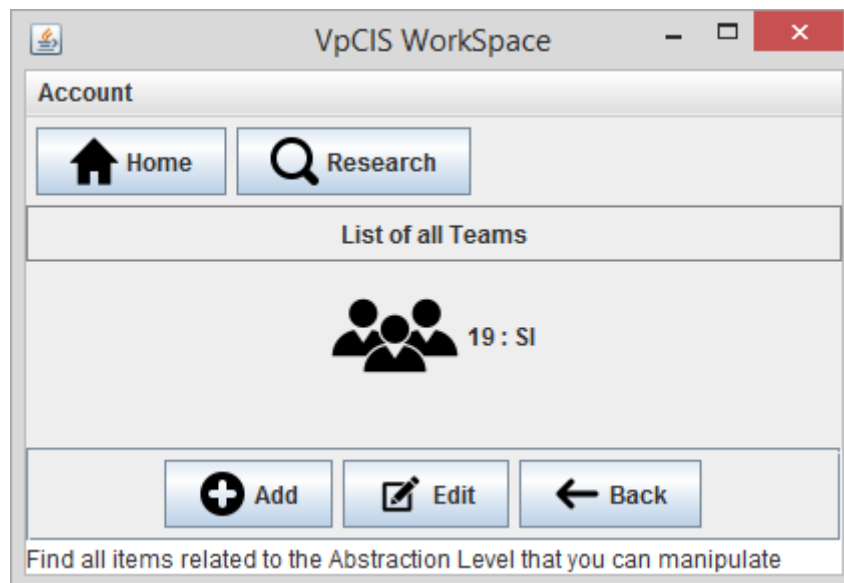


Figure III. 22 Panneau des Equipes

L'ajout d'une Equipe :

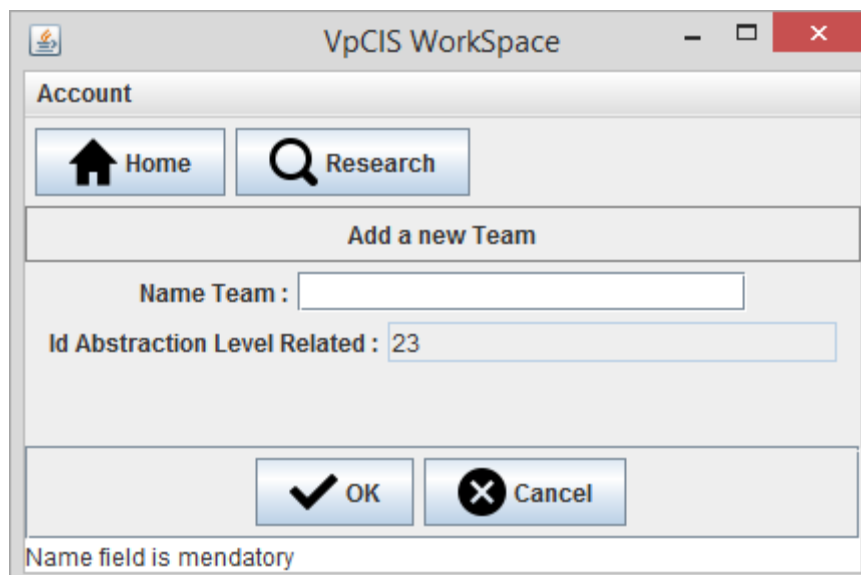


Figure III. 23 Ajout d'une nouvelle Equipe

Il est possible de modifier ou de supprimer une Equipe déjà créée.

- Activité Equipe:

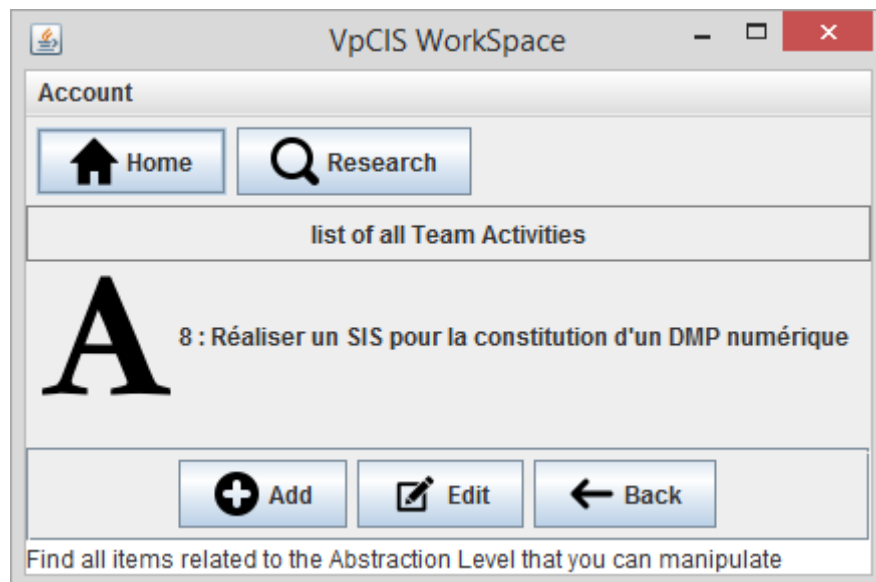


Figure III. 24 Panneau des Activités des Equipes

L'ajout d'une Activité Equipe :

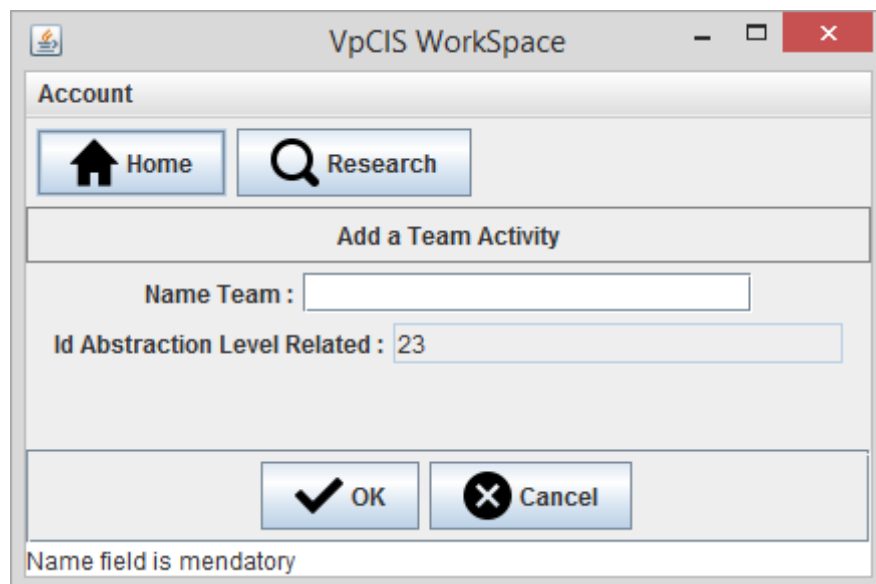


Figure III. 25 Ajout d'une nouvelle Activité Equipe

On peut ainsi modifier ou supprimer une Activité Equipe déjà créé.

- Information Equipe :

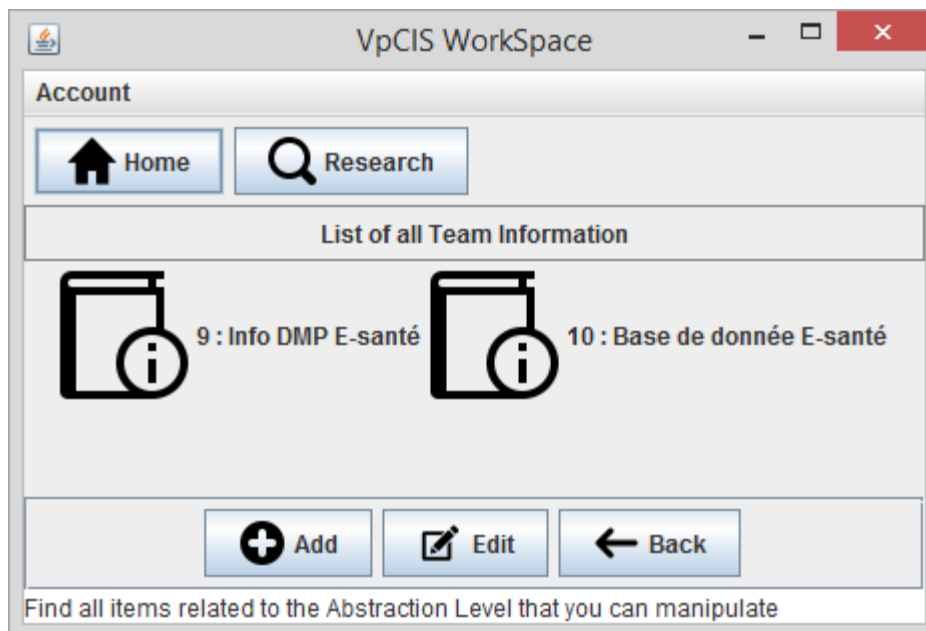


Figure III. 26 Panneau des Informations Equipes

Il est possible d'ajouter une Information Equipe :

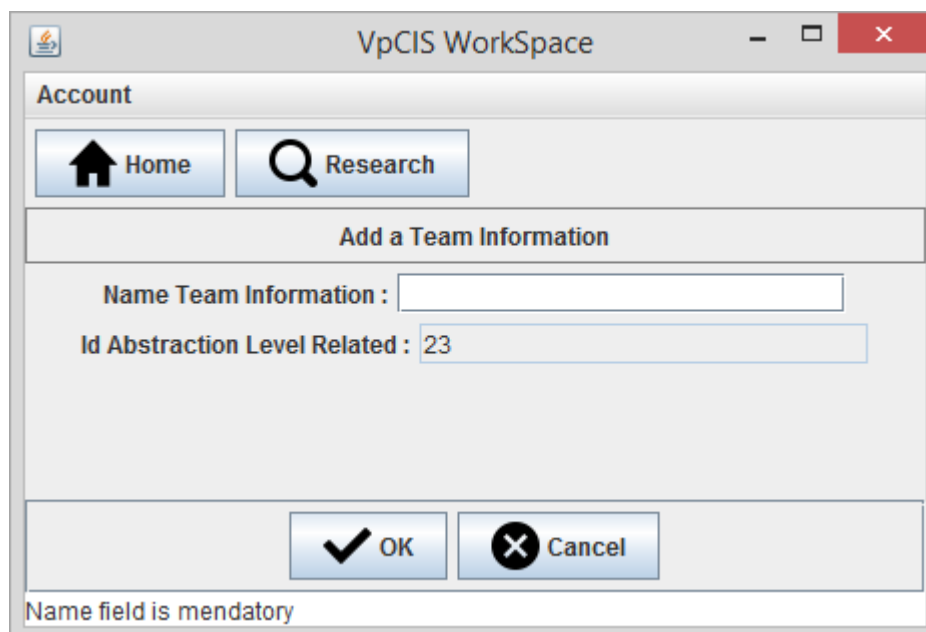


Figure III. 27 Ajout d'une nouvelle Information Equipe

On peut de même modifier et supprimer une Information Equipe déjà créée.

Quand on clique sur une information créée, on a la possibilité de lui associer des Données Equipes, des Connaissances Equipes et des Besoin Non Fonctionnel Equipes, nous allons parler de ces derniers dans ce qui suit :

- Donnée Equipe :

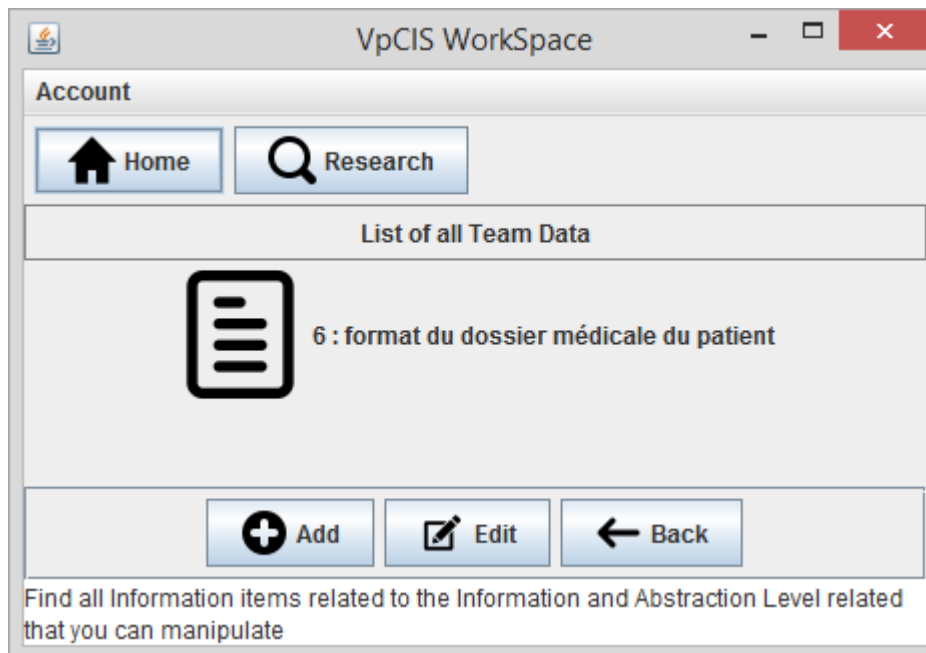


Figure III. 28 Panneau des Connaissances Equipes d'une Information Equipe

On peut ajouter une Donnée Equipe :

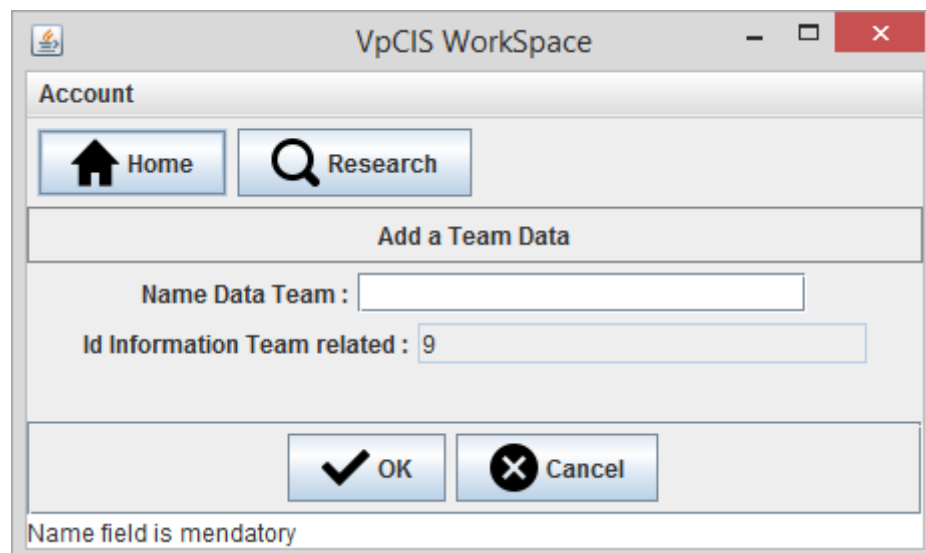


Figure III. 29 Ajout d'une nouvelle Donnée Equipe

Possibilité de modifier ou de supprimer des Connaissances Equipes crée déjà.

- Connaissances Equipe :

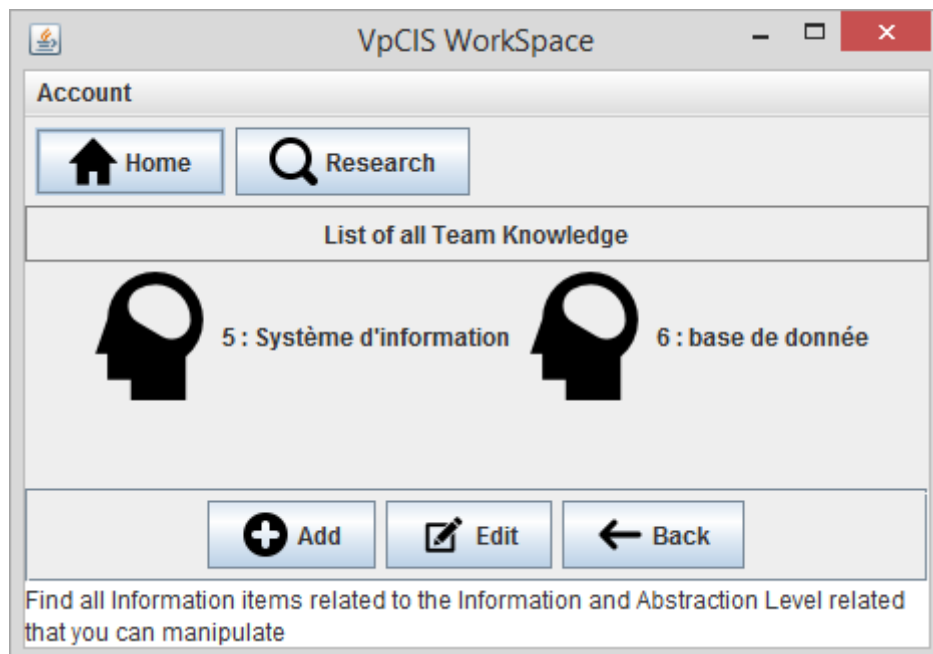


Figure III. 30 Panneau des Connaissances Equipes d'une Information Equipe

On peut ajouter une Connaissance Equipe :

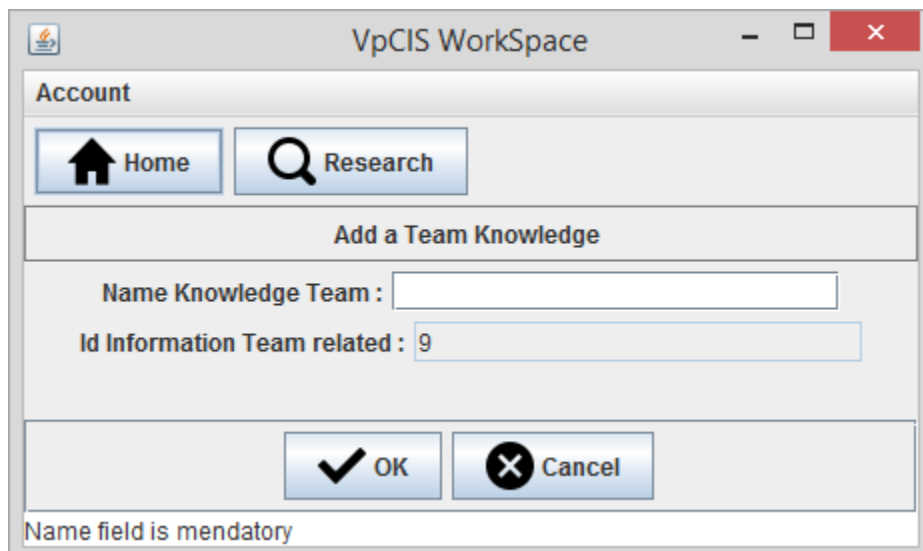


Figure III. 31 Ajout d'une nouvelle Connaissance Equipe

- Besoin Non Fonctionnel Equipe :

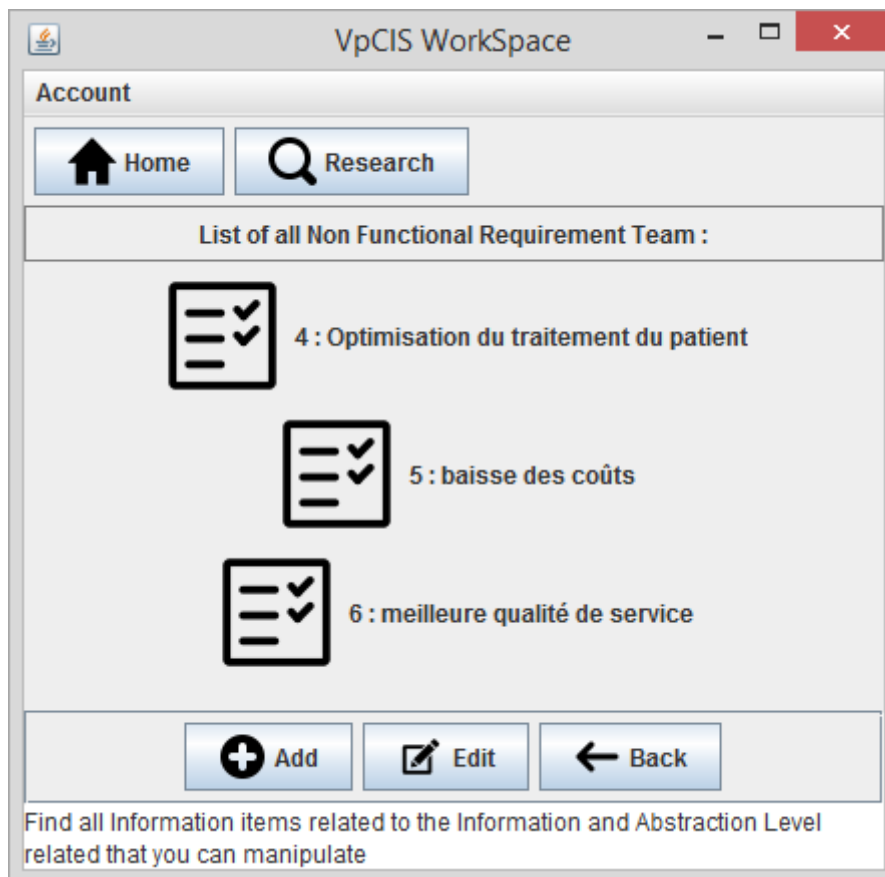


Figure III. 32 Panneau des Besoins Non Fonctionnel d'une Information Equipe

Possibilité d'ajout d'un Besoin Non Fonctionnel Equipe :

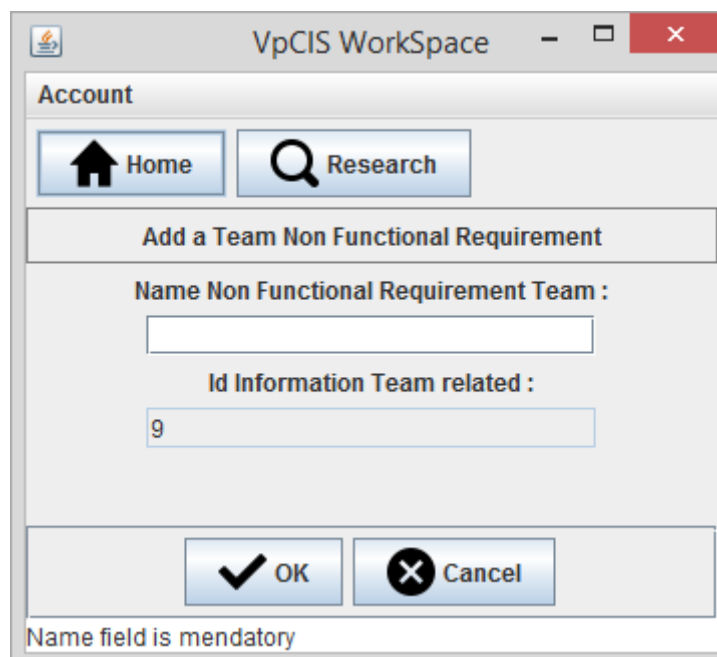


Figure III. 33 Ajout d'un nouvel Besoin Non Fonctionnel Equipe dans l'information Equipe

Il est également possible de modifier ou de supprimer un Besoin Non Fonctionnel Equipe déjà créé.

- Concept de base « Simple Acteur » :
 - Acteur :

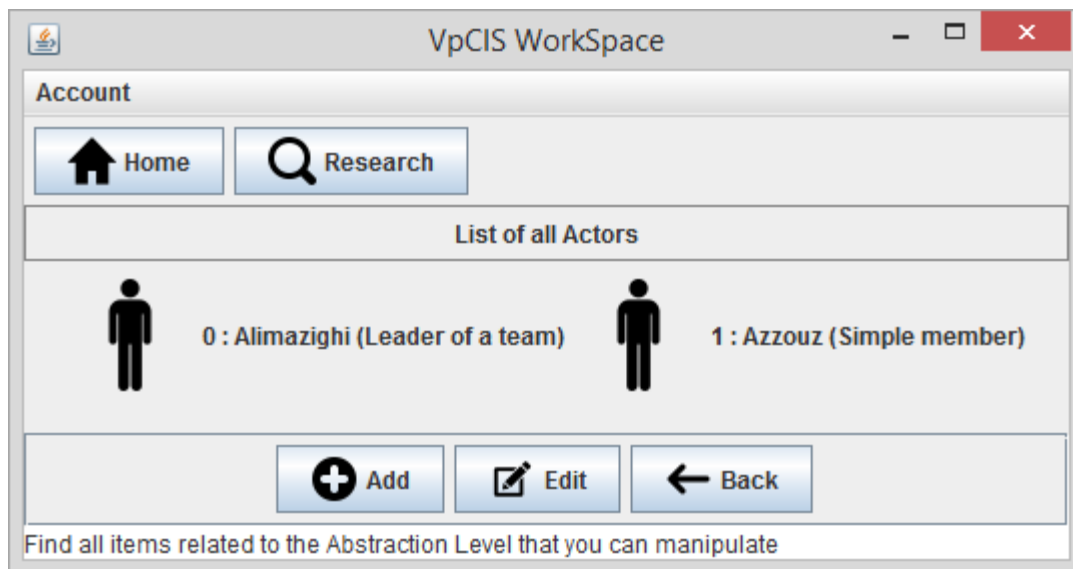


Figure III. 34 Panneau des Acteurs

Il est possible d'ajouter un nouvel Acteur :

Figure III. 35 Ajouter un nouvel Acteur

On peut modifier ou supprimer un Acteur déjà créé, il est ainsi possible d'ajouter une nouvelle qualité à un Acteur existant :

VpCIS WorkSpace

Account

Home Research

Add another quality to an existing Actor

id Actor :

From Team :

Id Abstraction Level :

Quality to add:

OK Cancel

Quality : can be a simple member, or leader of a team but only one in the team, or Responsible of the CIS but only one in the whole CIS Project, Name field is mandatory,

Figure III. 36 Ajout d'une nouvel Qualité à un Acteur

On peut modifier ou supprimer une qualité d'un acteur.

- Activité Acteur :

VpCIS WorkSpace

Account

Home Research

List of All Actor Activities

A 6 : Conception du SIS **A** 7 : Réalisation de la base de donnée

+ Add Edit ← Back

Find all items related to the Abstraction Level that you can manipulate

Figure III. 37 Panneau des Activité des Acteurs

Ajouter une Activité Acteur :

The screenshot shows a window titled 'VpCIS Workspace' with a standard Windows-style title bar. Inside, there's a header bar with 'Account' and two buttons: 'Home' (with a house icon) and 'Research' (with a magnifying glass icon). Below this is a section titled 'Add an Actor Activity'. It contains a form with the following elements: a label 'Name Actor Activity :' followed by a text input field; a label 'From Team Activity :' followed by a text input field; and a label 'Id Abstraction Level Related :' followed by a text input field containing the value '24'. To the right of the form is a text area containing the text '8 - Réaliser un SIS pour la constitution d'un DMP numérique'. At the bottom of the form are two buttons: 'OK' (with a checkmark icon) and 'Cancel' (with an 'X' icon). A small note at the bottom left of the window states 'Name field is mandatory'.

Figure III. 38 Ajout d'une nouvelle Activité Acteur

On peut de même modifier ou supprimer une Activité déjà créé.

- Information Acteur :

The screenshot shows a window titled 'VpCIS Workspace' with a standard Windows-style title bar. Inside, there's a header bar with 'Account' and two buttons: 'Home' (with a house icon) and 'Research' (with a magnifying glass icon). Below this is a section titled 'List of all Information Actor'. It displays two items, each with an information icon (a lowercase 'i' in a circle) and a label: '4 : DMP patient' and '5 : Info Conception du SIS'. At the bottom of the list are three buttons: 'Add' (with a plus icon), 'Edit' (with a pencil icon), and 'Back' (with a left arrow icon). A footer bar at the bottom of the window contains the text 'Find all items related to the Abstraction Level that you can manipulate'.

Figure III. 39 Panneau des Informations des Acteurs

On peut ajouter une Information Acteur :

The screenshot shows a window titled 'VpCIS Workspace' with a standard Windows title bar. Inside, there's a header 'Account' with 'Home' and 'Research' buttons. The main section is titled 'Add an Actor Information'. It contains three input fields: 'Name Actor Information' (empty), 'From Team Information' (with a dropdown menu showing '9 - Info DMP E-santé' and '10 - Base de donnée E-santé'), and 'Id Abstraction Level Related' (containing '24'). At the bottom are 'OK' and 'Cancel' buttons. A red error message 'Name field is mandatory' is visible at the very bottom.

Figure III. 40 Ajout d'une nouvelle Information Acteur

La modification et suppression d'une information Acteur est possible.

Quand on clique sur une information créé, on a la possibilité de lui associer des Données Acteur, des Connaissances Acteur et des Besoins Non Fonctionnel Acteur, nous allons parler de ces derniers dans ce qui suit :

The screenshot shows a window titled 'VpCIS Workspace'. It has a header 'Account' with 'Home' and 'Research' buttons. The main section is titled 'All Information items related to the Actor information and the Abstraction Level'. Below this title are three buttons: 'Data' (with a list icon), 'Knowledge' (with a head icon), and 'Non Functional Requirement' (with a checklist icon). At the bottom, there are two input fields: 'Id - Name Information Actor related' (containing '4 - DMP patient') and 'Id - Name AbstractionLevel related' (containing '24 - Niveau 2'). A 'Back' button is to the right of the second field. A footer text reads 'Find all Information items related to the Information and Abstraction Level related that you can manipulate'.

Figure III. 41 Donnée, Connaissance et Besoin Non Fonctionnel Acteur d'une Information Acteur

- Donnée Acteur :

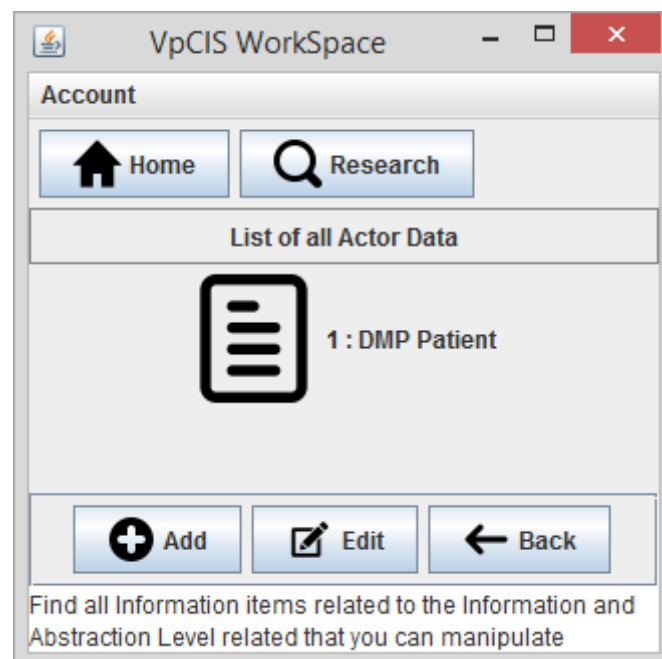


Figure III. 42 Panneau des Données Acteur d'une Information Acteur

L'ajout d'une Donnée Acteur :

Figure III. 43 Ajout d'une nouvelle Donnée Acteur dans l'Information Acteur

Il est également possible de modifier ou de supprimer une Activité Acteur déjà créé.

- Connaissances Acteur :

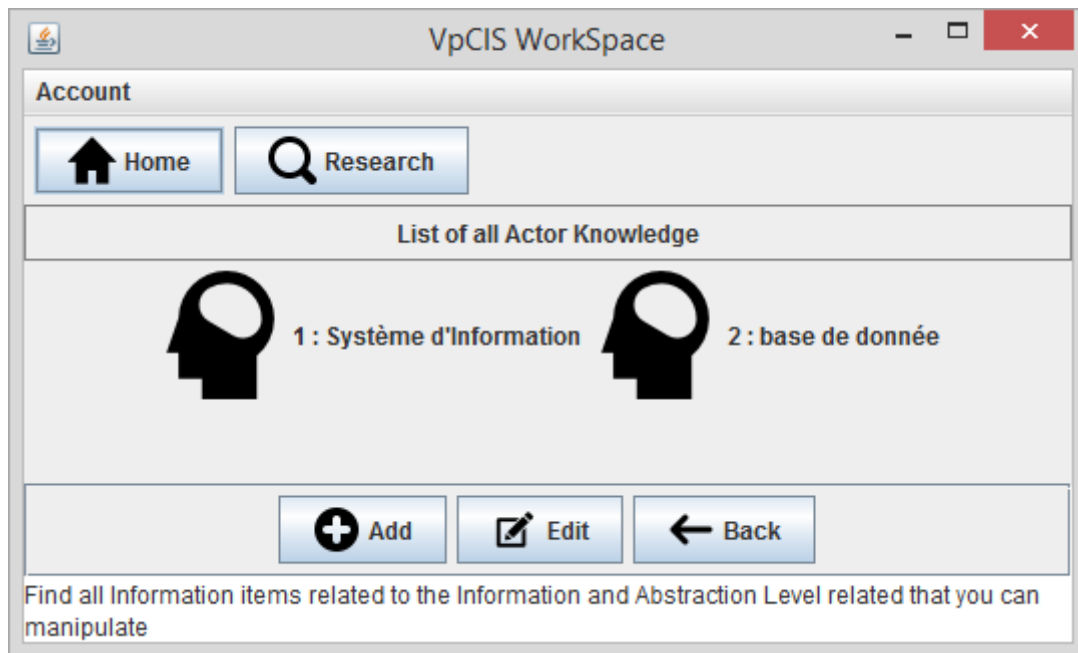


Figure III. 44 Panneau des Connaissances Acteurs d'une Information Acteur

On peut ajouter une Connaissance Acteur :

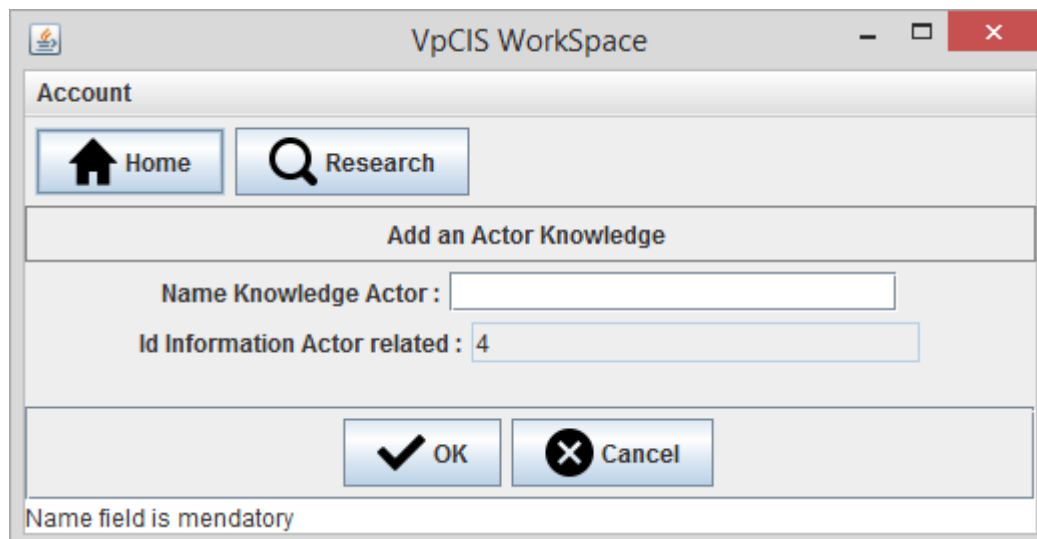


Figure III. 45 Ajout d'une nouvelle Connaissance dans Information Acteur

Il est possible d'ajouter ou de supprimer des Connaissances Acteur déjà créé.

- Besoin Non Fonctionnel Acteur :

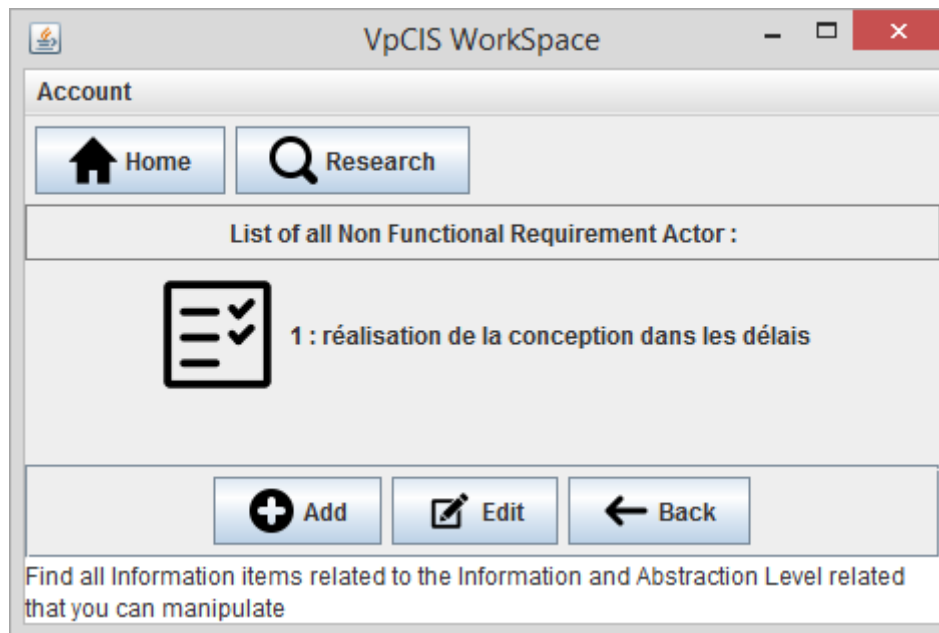


Figure III. 46 Panneau des Besoins Non Fonctionnel d'une information Acteur

Il est possible d'ajouter un Besoin Non Fonctionnel :

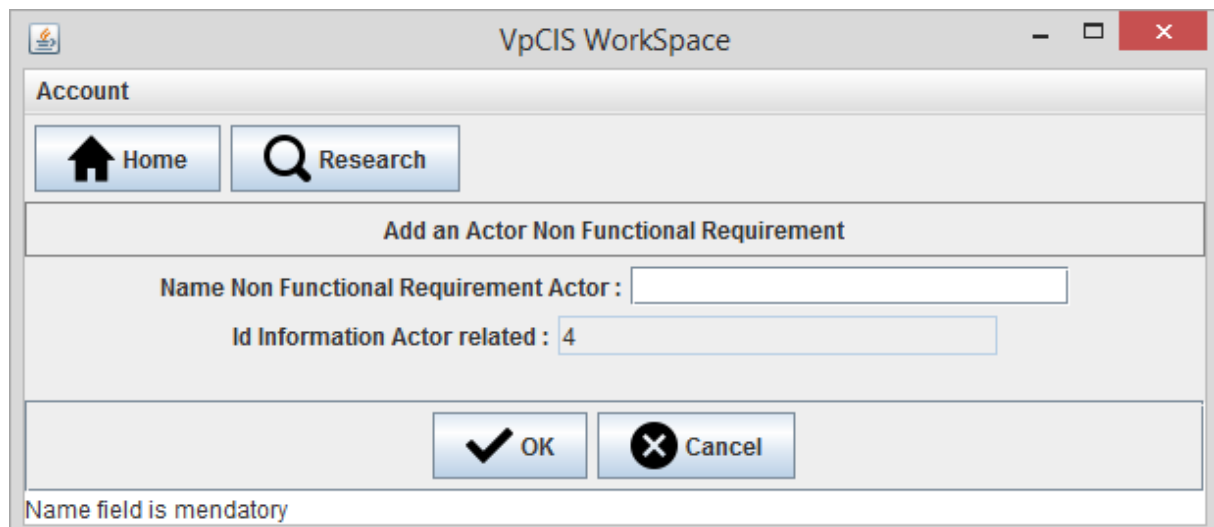


Figure III. 47 Ajout d'un nouvel Besoin Non Fonctionnel d'une Information Acteur

Il est également possible de modifier ou de supprimer un Besoin Non Fonctionnel Acteur.

- Concept de base « Lien » :

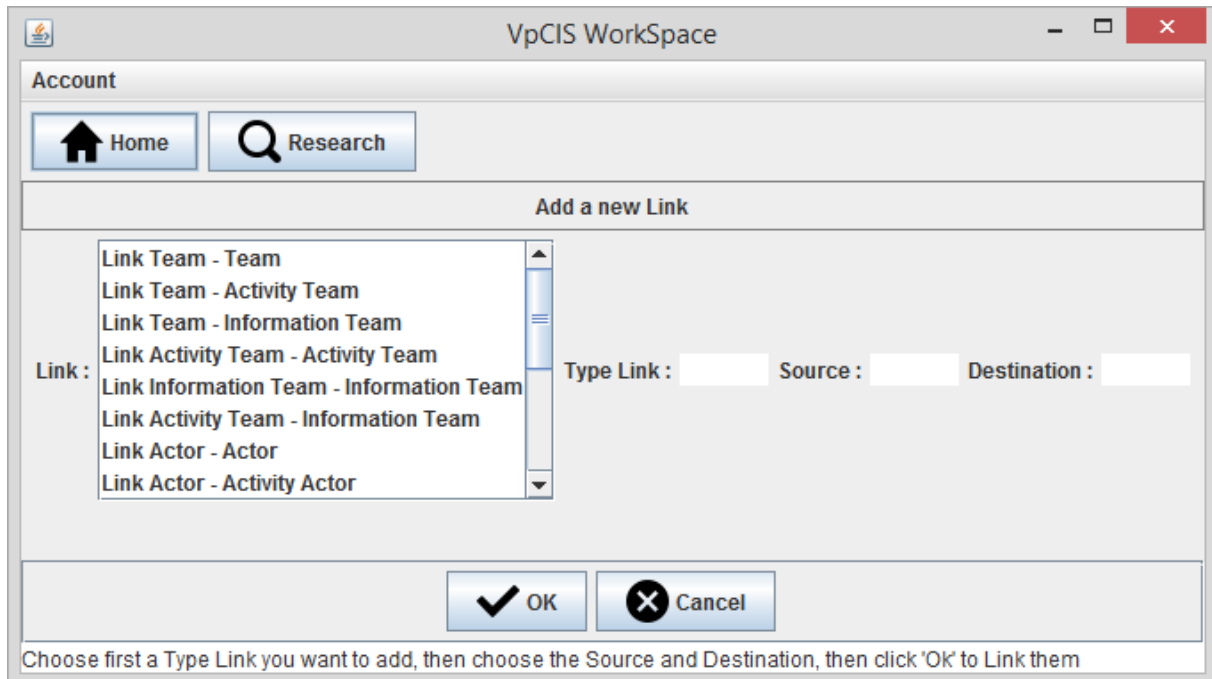


Figure III. 48 Ajout d'un Lien

Il est également possible de supprimer un lien déjà créé.

Enfin, en cliquant sur le Bouton « Researches » dans la barre de raccourcis, on a la possibilité d'effectuer des recherches prédéfinies :



Figure III. 49 Panneau des Recherches

Un exemple de la 1^{er} Recherche :

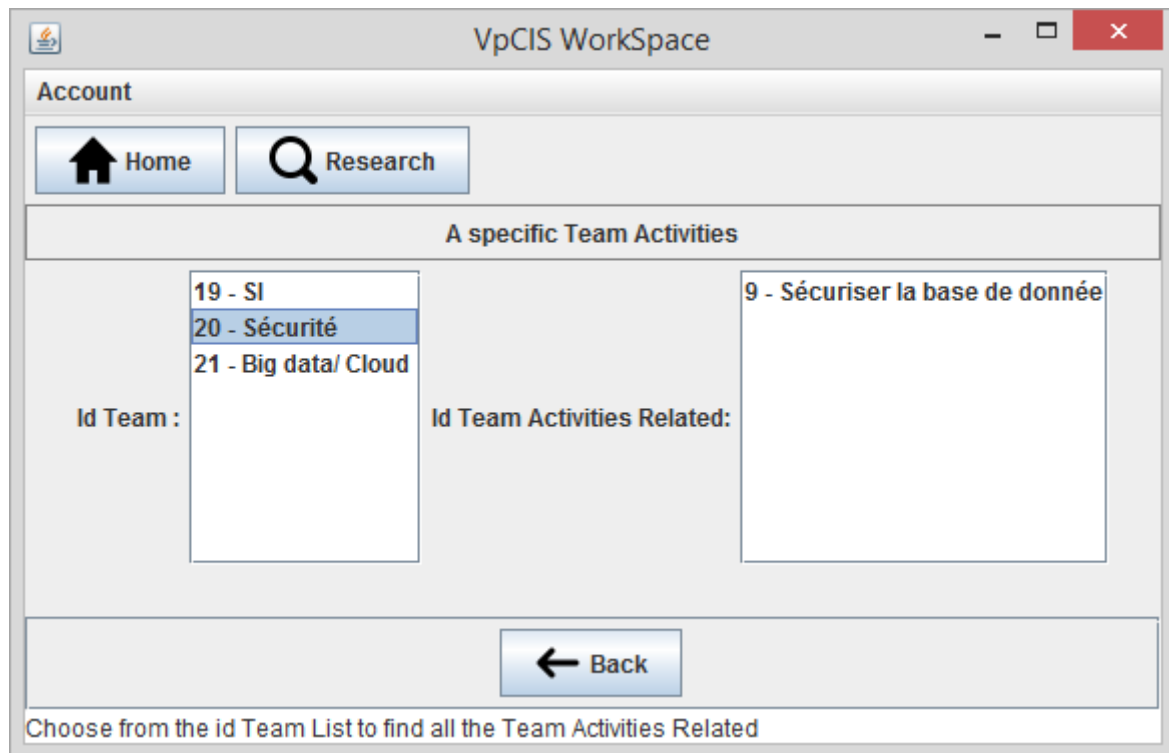


Figure III. 50 Exemple de Recherche

III.5. Conclusion :

A travers ce chapitre, nous avons présenté la réalisation de l'application en justifiant nos choix technologiques, en représentant quelques interfaces graphiques que nous avons jugé les plus importantes et en décrivant brièvement comment nous avons planifié notre projet.

Conclusion générale

L'objectif de notre projet de fin d'étude était de concevoir et implémenter un outil de gestion des systèmes d'information avec la notion des points de Vue sur le laboratoire LSI. Le point de départ de la réalisation de ce projet était une récolte des informations nécessaires pour dresser un état de l'existant, présenter un aperçu sur la problématique ainsi que l'architecture utilisée.

Par la suite, nous nous sommes intéressés à l'analyse et la spécification des besoins qui nous a permis de distinguer les différents acteurs interagissant avec l'application visée. L'objectif de la partie suivante était la conception détaillée, dans laquelle nous avons fixé la structure globale de l'application.

Le dernier volet de notre projet était la partie réalisation qui a été consacrée à la présentation des outils du travail et les interfaces les plus significatives de notre application.

L'apport de ce travail a été d'une importance très considérable, en effet, il nous a permis : de suivre une méthodologie de travail bien étudié, d'approfondir nos connaissances dans le monde de développement des applications et qui nous a permis de nous adapter avec l'environnement du développement ainsi perfectionné nos connaissances acquises en programmation et en conception.

La réalisation d'un tel projet, nous a permis d'apprendre et de toucher du doigt une partie de divers aspects du métier de développeur et de celui du concepteur.

Comme perspective, nous espérons voir notre application évoluer par une étape d'approfondissement du volet transactionnel avec des services distribués, un serveur web et des bases données afin de rendre notre application plus intéressante. Ce qui permettrait l'échange, des diverses données en utilisant des réseaux et des formats normalisés, permettre un affichage sous forme de graphe (des nœuds et des arcs représentant les liens), aussi d'avoir la possibilité de faire plus de recherches sur l'application, ajouter des chemins d'accès d'une part et d'autre part nous espérons améliorer la disposition des composants (JLabel, JTextField, JList).

Une autre perspective serait d'utiliser les spécifications obtenues à la fin de notre outil pour l'élaboration du SIC, soit de s'attaquer à la deuxième grande étape du cycle de développement d'un SIC, à savoir l'ingénierie des systèmes.

Nous espérons enfin que le travail que nous avons effectué a été à la hauteur de la confiance qui nous a été donnée.

Tout au long de l'élaboration du projet, nous avons rencontré plusieurs difficultés tant au niveau conceptuel qu'au niveau de la réalisation. Tout de même, nous avons réussi à les surpasser pour présenter en fin de compte une application opérationnelle.

Bibliographie :

1. D. Boulanger, G. Dubois, Objets et coopération de systèmes d'information, ingénierie objet : concepts, techniques et méthodes. Inter Editions, 1997.
2. A. Krabbel, I. Wetzel, H. Züllighoven, On the Inevitable Intertwining of Analysis and Design: Developing Systems for Complex Cooperations. DIS, 1997.
3. S.K. Levan, "Le projet Workflow". Paris : Hermès, 2000.
4. S. Khoshafian, M. Buckiewics, «Groupware & Workflow», MASSON editions, 1998.
5. K. Kessi, Z. Alimazighi, M. Oussalah, Requirement Meta Model Of A Cooperative Information System Oriented Viewpoints, CENTERIS, Potugal, October 2015.
6. M. Saadoun, Le projet groupware: des techniques de management au choix du logiciel groupware. Paris: Edition Eyrolles, 1996.
7. GTIE : Groupe de Travail Ingénierie des Exigences. Ingénierie des Exigences. Technical report, Association Française d'Ingénierie Système, [http ://www.afis.fr/nav/gt/ie/ie.html](http://www.afis.fr/nav/gt/ie/ie.html).
8. E. Hull, K. Jackson, J. Dick, Requirements Engineering. third edition. Springer, 2010.
9. B. Nuseibeh, J. Kramer, A. Finkelstein, A framework for expressing the relationships between multiple views in requirements specification. IEEE CS Press, 1994.
10. I. Sommerville, Software Engineering, 6th edn, Addison-Wesley, USA, 2001.
11. T.E. Bell, T.A. Thayer, Software Requirements: Are They Really a Problem?. Proc. ICSE-2, San Francisco, pp. 61-68, 1976.
12. M. G. Christel, K. C. Kang, Issues in Requirements Elicitation. Technical Report CMU/SEI-92-TR-12, Software Engineering Institute, Carnegie Mellon University, 1992.
13. C. Rolland, C. Ben Achour, Guiding the construction of textual use case specifications, North Holland Elsevier Science, 1998.
14. B. Nuseibeh, S. Easterbrook, Requirements Engineering: A Roadmap. ACM Press, Limerick, Ireland, 2000.
15. K. Yue, What does it mean to say that a specification is complete?. Proc. IWSSD-4. Four International Workshop on Software Specification and Design, Monterrey, 1987.
16. C. Rolland, C. Souveyet, C. Ben Achour, Guiding goal modelling using scenarios. IEEE Transactions on Software Engineering, Special Issue on Scenario Management, 1998.
17. I. Sommerville, P. Sawyer, Viewpoints: Principles, Problems and a Practical Approach to Requirements Engineering. Annals of Software Engineering 3, 1997.
18. A. Finkelsetin, J. Kramer, B. Nuseibeh, L. Finkelstein, M. Goedicke, Viewpoints: A Framework for Integrating Multiple Perspectives in System Development. International Journal of Software Engineering and Knowledge Engineering 2(1), 1992.
19. M. Sabetzade, A. Finkelstein, M. Goedicke, Viewpoints, in Encyclopedia of Software Engineering. Ed.Taylor and Francis, New York, 2010.
20. G. Kotonya, I. Sommerville, Requirements Engineering: Process and Techniques. John Wiley and Son, Great Britain, 1998.
21. I. Sommerville, P. Sawyer, S. Viller, Viewpoints for requirements elicitation: a practical approach. IEEE International Conference Requirements Engineering, USA, 1998.
22. OMG, Unified Modeling Language.
23. EIA-632, Electronic Industries Alliance, Government Electronics And Information Technology Association Engineering Department, EIA STANDARD. Processes for Engineering a System, January 1999.

Résumé

Dans ce projet, nous nous intéressons aux systèmes d'informations coopératifs, Un Système d'Information Coopératif (SIC) fait intervenir plusieurs parties prenantes qui collaborent entre elles, chacune avec son propre point de vue sur le système et œuvrant dans un même but.

La taille et la complexité des systèmes d'information coopératifs ne cessent de croître, leur complexité rend plus difficile leur conception. Il est donc très important de comprendre les besoins et les exigences du système, ce qui nous amène à nous intéresser à la première phase de développement d'un SIC, à savoir la phase d'analyse des besoins.

Afin de concevoir et de réaliser un outil qui permet de définir les besoins d'un SIC, en vue de réduire le problème de complexité, nous nous sommes basés sur une approche qui utilise une notion du génie logiciel qui est « les points de vue ». Cela va nous permettre de décomposer le SIC selon les points de vue de chaque partie prenante. Un point de vue, dans une première approximation est une vision partielle que peuvent avoir des individus ou groupes d'individus sur un système.

L'approche est organisée selon les trois niveaux de modélisation de l'OMG: M2, M1, M0:

Pour cela, sur la base d'un méta-modèle orienté points de vue pour la définition des besoins d'un SIC au niveau de modélisation M2, nous avons dérivé un modèle de point de vue au niveau de modélisation M1.

Afin d'implémenter cette approche nous proposons un outil conforme au méta-modèle, que nous avons appliqué pour validation à une étude de cas réel (niveau M0) : « système intégré de gestion de l'identité numérique d'un citoyen », réalisé par les équipes du laboratoire LSI du département Informatique de l'USTHB.

Mot Clés : Système d'information coopératif, point de vue, analyse des besoins, méta-modèle, base de données, interface utilisateur.