

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab Blida



Faculté des Sciences
Département d'Informatique

Mémoire de fin d'étude

En vue d'obtenir le diplôme de master
en informatique

Option : Ingénierie des logiciels

Sujet :

***Déploiement d'une infrastructure cyber-physique
RFID***

Réaliser par :

KACHOUANE Hadjer
HAMDANIA Amel

Soutenu le :

M.	Président
M.	Examineur
M.	Examineur
Mlle. N. BOUSTIA	Promotrice
Mr. A. HENTOUT	Encadreur

**Promotion
2014 / 2015**

Remerciement

Tout d'abord, nous tenons rendre grâce à Dieu tout puissant pour nous avoir donné le courage et la détermination nécessaire pour finaliser ce travail et le mener à terme.

Mon premier remerciement s'adresse au directeur de ce mémoire, Mr Hentout, pour la qualité de son encadrement. Sa disponibilité et son attention du début à la fin de ce mémoire, ont été pour moi une aide précieuse et une source d'équilibre pour la réalisation de ce travail.

Je voudrai également remercier Mme Boustia qui nous a fait l'honneur d'être notre promotrice ainsi que pour son aide, ses conseils et son soutien durant ce projet.

Je tiens aussi à remercier Mr.Messous pour son aide et son encouragement.

Je remercie les membres de jury pour nous avoir fait l'honneur de juger notre travail.

Enfin ,A

Mes parents qui ont tant souffert pour m'éduquer et m'éclaircir le chemin de la réussite.

Et mes frères, mes sœurs, mon neveu, mes nièces et à tous mes amis, symbole de complémentarité.

Kachouane Hadjer

Remerciement

Tout d'abord, je tiens à rendre grâce à Dieu tout puissant pour nous avoir donné le courage et la détermination nécessaires pour finaliser ce travail et le mener à terme.

En premier lieu, Je remercie mon encadrant Mr. Hentout pour avoir accepté de diriger ce travail. Sa disponibilité, sa patience, ses compétences et ses conseils avisés m'ont été d'une aide inestimable.

Je voudrais également remercier Mme. Boustia qui nous as fait l'honneur d'être notre promotrice ainsi que pour son aide, ses conseils et son soutien durant ce projet.

Je tiens aussi à remercier Mr. Messous pour son aide et son encouragement.

Je remercie les membres de jury pour nous avoir fait l'honneur de juger notre travail.

Un grand Merci à mes amis, à tous ceux qui m'ont soutenu et à tous ceux qui ont participé de près ou de loin à l'aboutissement de ce travail.

Je termine par dédicacer ce mémoire à mon père, ma mère et mes frères qui sans eux, tout cela n'aurait pas été possible ...

ملخص

الأنظمة السيرانية الفيزيائية تعتبر مجال أساسي للأبحاث، هي تأخذ دورا أساسيا في عدة مجالات خصوصا مجال الروبوتات. هذه الأنظمة قادرة على اكتشاف العالم الفيزيائي و تحويله إلى معلومات الكترونية. الاكتشاف و التعريف في النظام الروبوتي السيراني الفيزيائي يعتبر أمرا مهما حيث يتطلب مجموعة من النهج لتحقيق المطلوب.

الهدف من عملنا هذا هو نشر البنية التحتية باستعمال تكنولوجيا تحديد الترددات الراديوية RFID بطريقة حيث إن مجموع اجهزة استشعار تحديد الترددات الراديوية الموضوعة على الروبوتات تمكننا من تجميع و تعريف بطاقات تحديد الترددات الراديوية الموصولة مع مجموعة من الاشياء في الوسط و تنظيم المعلومات المستخلصة بواسطة قاعدة البيانات مع استخراج المسارات المختلفة (و المسار الأقصر) التي مر بها الروبوت. بمقارنة النتائج التي توصلنا اليها مع التي توصل اليها مختلف الباحثين يمكننا التأكيد على أن عملنا يقود الى نتائج مقبولة و مرضية.

الكلمات المفتاحية: نشر البنية التحتية بتحديد الترددات الراديوية، قاعدة بيانات، الأنظمة الروبوتية السيرانية

الفيزيائية، تحديد الترددات الراديوية RFID.

Abstract

Cyber-physical systems (CPS) are considered as a priority area of research. It occupies an important role in different fields especially in *robotics*. These systems are capable to detect the physical world and reconfigure it in computer data.

The detection and identification in robotic cyber-physical system (RCPS) is a very important question which needs a set of approaches in order to perform this task.

The goal of our work is the deployment of an infrastructure using the RFID technology; in such a way that all RFID readers placed on robots will be able to assimilate and identify RFID tags attached to the environment objects, to organize the resultant information in a database, in order to extract the different paths (and shortest path) moving the robots from an initial to a final location.

When comparing the obtained results with those found in the literature by other researchers, we can confirm that our work leads to a satisfactory result.

Keywords: Deployment of RFID infrastructure, database, robotic cyber-physical system, RFID.

Résumé

Les systèmes cyber-physiques (SCP) sont considérés comme un domaine de recherche prioritaire. Ils occupent un rôle important dans de nombreux domaines, notamment la robotique. Ces systèmes sont capables de détecter le monde physique et de le reconfigurer en données informatiques.

La détection et l'identification dans un système robotique cyber-physique (SRCP) est une question importante, ce qui nécessite un ensemble d'approches pour effectuer cette tâche. Le but de notre travail est le déploiement d'une infrastructure cyber-physique utilisant la technologie RFID, de telle sorte que l'ensemble des lecteurs RFID placés sur les robots soient capables d'assimiler et d'identifier les tags RFID attachés aux objets de l'environnement et, par la suite, organiser les informations résultantes dans une base de données centralisée afin d'en extraire les différents chemins empruntés (ainsi que le plus court chemin) amenant d'une position initiale à une position finale.

En comparant les résultats que nous avons obtenus avec ceux trouvés dans la littérature, nous pouvons affirmer que notre travail a permis d'obtenir un résultat satisfaisant.

Mot clés : Déploiement d'une infrastructure RFID, Base de données, système robotique cyber-physique, RFID.

Liste des Figures

Figure 1.1 : Trois principaux composants dans un SCP.....	6
Figure 1.2 : Exemple de structure d'un SCP.....	7
Figure 1.3 : Détection par Ultrason.....	9
Figure 1.4 : Détection par Infrarouge.....	10
Figure 1.5 : Détection par Wifi.....	11
Figure 1.6 : Principe de la RFID.....	12
Figure 1.7 : Structure et vue externe de tags RFID.....	14
Figure 1.8 : Un patient équipé avec un bracelet RFID.....	15
Figure 1.9 : Chaine d'approvisionnement avec la RFID.....	16
Figure 1.10 : Approchement du robot PR2 au objet.....	20
Figure 1.11 : Le robot kompai dans une classe de l'université Umea.....	21
Figure 2.1 : Schéma descriptif du notre système.....	25
Figure 2.2 : Diagramme de cas d'utilisation du système.....	31
Figure 2.3 : Diagramme d'activité pour la détection d'objets et l'affichage dans fichier .txt.....	32
Figure 2.4 : Diagramme de séquence pour l'authentification.....	33
Figure 2.5 : Diagramme de séquence pour traiter le fichier.txt.....	34
Figure 2.6 : Diagramme de séquence pour l'extraction d'un chemin.....	35
Figure 2.7 : Diagramme de séquence pour ajouter un tag.....	35
Figure 2.8 : Diagramme de séquence pour modifier un tag.....	36
Figure 2.9 :Diagramme de séquence pour supprimer un tag.....	37
Figure 2.10 : Diagramme de séquence pour ajouter un robot	37
Figure 2.11 :Diagramme de séquence pour modifier un robot	38
Figure 2.12 : Diagramme de séquence pour supprimer un robot	39
Figure 2.13 :Diagramme de séquence pour ajouter un lecteur	39
Figure 2.14 :Diagramme de séquence pour modifier un lecteur	40
Figure 2.15 :Diagramme de séquence pour supprimer un lecteur	41

Figure 2.16: Diagramme de classe.....	42
Figure 3.1 : (a) Le simulateur en 2D, (b) Le simulateur en 3D.....	49
Figure 3.2 : Modèle Client/serveur du Player.....	50
Figure 3.3 : Stage et serveur Player.....	51
Figure 3.4 : Définition de l'environnement dans simple1.world.....	52
Figure 3.5 : Un modèle d'un robot dans simple1.world.....	52
Figure 3.6 : Un modèle d'un objet (table) dans map.inc.....	52
Figure 3.7 : Simple1.cfg.....	53
Figure 3.8 : Environnement simulé.....	54
Figure 3.9 : Format du modèle Fiducial finder.....	55
Figure 3.10: Format d'un modèle « objet-fiducial ».....	55
Figure 3.11 : Suivre une phéromone.....	56
Figure 3.12 : Format fichier.txt	57
Figure 3.13 : Interface d'accueil	59
Figure 3.14 : Interface menu principal	59
Figure 3.15 : Interface « Traitement Fichier ».....	60
Figure 3.16: Interface « Tag ».....	60
Figure 3.17: Interface «Extraction Chemin ».....	61
Figure 4.1 : Premier scénario.....	64
Figure 4.2: Deuxième scénario	64
Figure 4.3 : Troisième scénario.....	65
Figure 4.5: Comparaison entre les scénarios.....	67
Figure 4.6 : Extraction de l'ensemble de chemins.....	68
Figure 4.7 : Extraction chemin.....	69
Figure 4.8: Extraction chemin optimal.....	69

Table des acronymes

SCP :Système cyber-physique .

SRCP :Système Robotique cyber-physique.

RFID : Radio Fréquence Identification.

IR :Infrarouge.

US :Ultrasons.

UWB : bande ultra-large.

Liste des tableaux

Tableau 2.1 : description textuelle des cas d'utilisations.....	29
Tableau 2.2 : Description des classes.....	42
Tableau 2.3 : Passage du modèle Objet vers le modèle Relationnel.....	43
Tableau 3.1 : Résultats obtenus pour le premier scénario	65
Tableau 3.2 : Résultats obtenus pour le premier scénario	65
Tableau 3.3 : Résultats obtenus pour le premier scénario	65

Sommaire

Introduction générale.....	1
Chapitre -I- : Systèmes cyber-physiques	
1.Introduction.....	5
2. Système cyber-physique	5
2.1.Définition.	5
2.2.Exemple de structure de SCP	6
2.3.Caractéristiques des SCP	7
3.Les SCP et l'internet des objets.....	8
3.1.L'identification par Ultrason.....	8
3.2.L'identification par infrarouge.....	9
3.3.L'identification par bande ultra-large (UWB)	10
3.4.L'identification par Wifi	11
3.5.L'identification par RFID.....	12
3.6.Comparaison entre les différentes technologies d'identification.....	13
4. SCP et l RFID	13
5. Les applications des SCP.....	14
5.1.Dans le domaine médical.....	14
5.2.Dans le domaine militaire.....	15
5.3.Dans le domaine logistique	16
5.4.Domaine de l'automobile.....	17
5.5.Bâtiment intelligent.....	17
5.6.Les avantages du système RFID	18
5.7.Les inconvénients du système RFID.....	18
6.Applications robotique et SCP.....	19
6.1.La RFID et les robots ménagers.....	19
6.2.Stratégie de la localisation d'objet en utilisant la technologie RFID.....	20
6.3.La chirurgie médicale	21
6.4.Coopération entre caméras d'ambiance et capteur embarqués	21
6.5.Les avantages du système RFID dans la robotique.....	22
6.6.Les inconvénients du système RFID dans la robotique.....	22

7.Conclusion	22
Chapitre -II- : Description de la solution proposée	
1.Introduction	25
2.Description de la solution proposée.....	25
2.1.Première Phase: Déploiement d'une infrastructure RFID	26
2.1.1.Première Partie : Détection et identification	26
2.1.2.Deuxième Partie : Extraction chemin	26
2.2.Deuxième Phase :Traitement de base de donnée.....	27
3.Conception	27
3.1.Présentation du langage UML.....	27
3.2.Modélisation	28
3.2.1.Diagrammes des cas d'utilisation.....	28
3.2.1.1.Description textuelle des cas d'utilisations	28
3.2.1.2.Diagramme de cas d'utilisation du système.....	30
3.2.2.Diagrammes d'activité.....	31
3.2.3.Diagrammes de séquence.....	32
3.2.3.1.Diagramme de séquence pour l'authentification.....	32
3.2.3.2.Diagramme de séquence pour traiter le fichier.txt	33
3.2.3.3.Diagramme de séquence pour l'extraction d'un chemin.....	34
3.2.3.4.Diagramme de séquence pour ajouter un tag	35
3.2.3.5.Diagramme de séquence pour modifier un tag	36
3.2.3.6.Diagramme de séquence pour supprimer un tag	36
3.2.3.7.Diagramme de séquence pour ajouter un robot	37
3.2.3.8.Diagramme de séquence pour modifier un robot	38
3.2.3.9.Diagramme de séquence pour supprimer un robot	38
3.2.3.10.Diagramme de séquence pour ajouter un lecteur	39
3.2.3.11.Diagramme de séquence pour modifier un lecteur	40
3.2.3.12.Diagramme de séquence pour supprimer un lecteur	40
3.2.4.Diagrammes de classes	41
3.2.4.1.Description des classes.....	42
3.2.4.2.Passage du modèle objet vers le modèle relationnel	44
3.2.4.3.Traduction du modèle objet en base de données relationnelles.....	44

4.Codification du système	45
4.1.Codification des robots	45
4.2.Codification des opérateurs humains.....	46
4.3.Codification des objets.....	46
5.Conclusion	46
Chapitre -III- : Implémentation de la solution proposée	
1.Introduction	48
2.Simulateur Player/Stage.....	48
2.1.Player	49
2.2.Stage.....	50
3.Déploiement d'une infrastructure RFID.....	51
3.1.Fichier.world.....	51
3.2.Fichier.inc	52
3.3.Fichier.cfg	53
3.4.Fiducial et Player/Stage	54
3.5.Le langage de programmation choisi (C++)	55
4.Extraction chemin.....	56
4.1.Suivre un chemin.....	56
4.2.Fichier «txt ».....	56
5.Implémentation de la base de données RFID.....	57
5.1.Le Système de Gestion de Base de données(SGBD).....	57
5.2.MySQL	58
5.3.Le langage de programmation choisi (java).....	58
5.4.L'environnement de développement intégré EDI NetBeans	58
6. Application développée.....	58
6.1.Interface d'accueil	59
6.2.Interface menu principal	59
6.3.Interface « Traitement Fichier ».....	59
6.4.Interface « Tag »	60
6.5.Interface « Extraction Chemin»	61
7.Conclusion.....	61
Chapitre-IV- :Test et validation	
1.Introduction	63

2.Scénario de validation de la détection des tag.....	63
2.1.Premier Scénario.....	63
2.2.Deuxième Scénario.....	64
2.3.Troisième Scénario.....	65
2.4.Discussion des résultats obtenus de détection des tags	65
2.5.Comparaison entre les résultats obtenus.....	67
3.Scénarios de validation de l'extraction de chemins.....	68
3.1.Premier scénario.....	68
3.2.Deuxième scénario.....	68
3.3.Troisième scénario.....	69
4.Conclusion.....	69
Conclusion générale et perspective.....	72
Références bibliographiques	75

Introduction Générale

La robotique autonome devient de plus en plus importante dans le domaine de la recherche, et fait face à une croissance rapide dans la complexité des besoins et des exigences pour des robots chargés de tâches multiples, capables de se coordonner, et développés de telle manière que des garanties de sûreté et de sécurité puissent être vérifiées et certifiées pour remplacer l'homme [1]. En parallèle, une évolution similaire dans le domaine des systèmes temps-réel embarqués répartis a justifié l'émergence du domaine des « Systèmes cyber-physiques (SCP) » reflétant une montée similaire en complexité. À la différence des systèmes embarqués traditionnels, un SCP à part entière est généralement conçu comme un réseau d'éléments informatiques en interaction avec des entrées et des sorties physiques au lieu de dispositifs autonomes en interaction. Les SCP peuvent être trouvés dans divers domaines tel que l'aéronautique, la santé, l'automobile, les processus chimiques, etc.

Ces systèmes sont capables de collecter des informations sur le monde physique et de les reconfigurer en données informatiques. « *En sciences informatiques, les données sont relativement limitées. En physique, à l'inverse, ce qui peut être observé est très divers* ». En plus du rôle qu'y joue la logique, un SPC doit donc être capable de prendre en compte le monde réel. Ces notions sont étroitement liées aux concepts de la robotique et des réseaux de capteurs [2]. Les systèmes robotiques sont donc une classe importante de SCP ; l'habileté des robots d'interagir intelligemment avec le monde repose sur le calcul embarqué et de la communication, le contrôle en temps réel, et la perception du monde qui les entoure. Ceci est dans le but d'accomplir automatiquement des tâches, le plus souvent, reconnues comme pénibles, répétitives, impossibles pour les humains ou dangereux comme l'analyse de substances à l'intérieur d'un volcan, des centrales nucléaires, dans l'espace ou dans les profondeurs de l'océan ; ou dans un but d'efficacité supérieure notamment dans la vie quotidienne, y compris le ménage domestique en

assimilant les différents objets, soins des malades, des personnes âgées et la chirurgie médicale avec des robots capables de réaliser des opérations, détecter et identifier les instruments médicaux, détecter des maladies, etc.

L'identification et la localisation des objets est un processus pertinent. Elle s'applique généralement au traitement de nombreux objets qui exige un suivi rapide au-delà des capacités humaines, et obtenir des informations sur les personnes, les animaux, les biens, les produits en transit ou en stockage, etc.

En robotique, la capacité de détecter et d'identifier à tout moment des éléments fixes et mobiles dans un environnement est une question importante. Il existe un ensemble de méthodes, de technologies, d'approches et de dispositifs qui permettent de répondre à cette question. Parmi ces approches, on trouve le Zigbee, les Ultrasons, le Wifi, le NFC, la RFID et d'autre encore. Par la suite, la collecte, le traitement, l'interprétation correcte et le stockage des différentes informations obtenues nécessite des outils plus efficaces pour gérer et utiliser ces données.

Dans ce projet, nous nous intéressons à l'approche RFID (identification par radiofréquence) qui est un élément relativement récent de l'identification automatique. Cette technologie utilise la radio fréquence pour identifier les objets étiquetés lorsqu'ils passent à proximité d'un lecteur RFID. Dans ce contexte, notre travail vise à :

1. Simuler une infrastructure *RFID* cyber-physique dans un environnement dynamique et inconnu pouvant contenir des opérateurs humains, des robots, des obstacles statiques et mobiles, des objets connus et inconnus, etc. en utilisant le simulateur Player/Stage.
2. Simuler les lecteurs RFID par des capteurs fiducial finder, et les tags RFID par des fiducials attachés à tous les éléments étiquetés de l'environnement de telle sorte qu'ils puissent être lus par ces capteurs.
3. Enregistrer l'ensemble de toutes ces données dans un fichier texte (.txt) accessible par tous les agents (robots) du SCP.
4. Organiser le fichier généré précédemment dans une base de données centralisée pour pouvoir effectuer les différents traitements sur le système.
5. Extraire l'ensemble des chemins empruntés par les robots durant la détection, dont chaque chemin possède une position initiale, une position finale et un identifiant phéromone qui le caractérise.
6. Enfin, il s'agit de sélectionner le chemin le plus court amenant le robot d'une position

initiale à une position finale.

Outre cette introduction générale, le présent document est organisé en quatre chapitres donnés comme suit :

- Dans le premier chapitre, nous allons présenter les systèmes cyber-physiques sur lesquels est basé le déploiement de l'infrastructure RFID. Ensuite, nous allons aborder une étude comparative entre les différentes technologies et systèmes de détection et d'identification. Enfin, nous allons donner des exemples d'applications des systèmes cyber-physiques dans divers domaines notamment la dans le domaine de la robotique en utilisant la technologie d'identification RFID.
- Dans le second chapitre, nous allons décrire, d'une manière détaillée notre solution proposée ainsi que les aspects conceptuels et les différents détails de développement de cette solution.
- Dans le troisième chapitre, nous présentons l'implémentation de la solution que nous avons proposée.
- Dans le dernier chapitre, nous présentons et discutons les principaux résultats obtenus lors des tests de validations afin de prouver l'efficacité de la solution que nous avons proposée.

Enfin, la conclusion générale de ce mémoire synthétisera principales contributions et donnera quelques perspectives à notre travail.

1. Introduction

Les systèmes informatisés ont envahi la plupart des activités humaines dans la société, à toutes les échelles. La plupart des systèmes de production, tous les moyens de transports allant des automobiles aux navires et aux avions, utilisent des systèmes informatisés qui évaluent de façon continue leurs propres situations et proposent ou prennent de manière autonome des décisions d'action selon l'état actuel de leur environnement. On est donc dans un monde où des appareils électroniques très informatisés permettent de communiquer pour réaliser des actions, pour donner des conseils, pour prendre les initiatives satisfaisantes que l'individu a oublié de prendre. Cela constitue le domaine des *Systèmes Cyber-Physiques* (SCP), un domaine qui a pris une importance considérable dans l'économie et dans la recherche, avec des applications dans tous les secteurs, utilisant différentes approches de détection (RFID, Zigbee, etc.) [3].

Afin de montrer l'importance des SCP, la présentation de ces systèmes dans un premier lieu est nécessaire. Ensuite, nous allons présenter une étude comparative entre les différentes approches de détection. Enfin, nous allons détailler les applications des SCP dans différents domaines utilisant l'approche RFID.

2. Système cyber-physique

Le terme *Systèmes cyber-physiques* est apparu vers 2006, quand il a été inventé par *Helen Gill* à la *National Science Foundation* aux États-Unis [4].

2.1. Définition

Dans la littérature, un grand nombre de définitions des SCP peuvent être trouvées [5] [6] [7] [8] [9] [10] [11] [12]. Les différentes définitions sont comparées, et combinées en une seule définition en se basant sur les caractéristiques communes des SCP [13]. Pour résumer, « Les systèmes cyber-physiques (SCP) sont des systèmes embarqués complexes qui interagissent continuellement et de manière dynamique avec leur environnement grâce à l'association d'éléments du monde physique et d'éléments du monde de l'information distribuée ». Autrement dit, « les SCP utilisent des calculs et de la communication profondément ancrées dans l'interaction avec les processus physiques afin d'ajouter de nouvelles fonctionnalités au système physique, pour former des systèmes complexes fiables, de manière efficace et en temps réel qui offrent de nouvelles capacités ». Les SCP sont donc l'union du calcul, de communication et de contrôle.

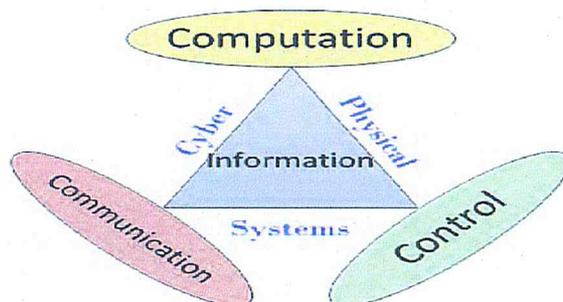


Figure 1.1 : Trois principaux composants dans un SCP [14].

Les SCP peuvent être trouvés dans des domaines aussi variés que l'aéronautique, l'automobile, les processus chimiques, l'infrastructure civile, l'énergie et l'automatisation industrielle, la santé, l'agriculture, la fabrication, le transport, la robotique, le divertissement et les appareils électroménagers. Parmi les applications réalisées de ces systèmes dans l'environnement qui nous entoure on trouve les réseaux électriques, bâtiments intelligents, les maisons intelligentes, les centres de données, les soins de santé, les technologies médicales intelligentes, etc. [15].

2.2. Exemple de structure de SCP

Tous les SCP ont des processus informatiques qui interagissent avec les composants physiques. Ceux-ci peuvent être relativement simple (appareil de chauffage, machine de découpe, etc.) ou comporter plusieurs composants dans les assemblages complexes (véhicules, avions, raffineries, etc.). Les processus informatiques et physiques de ces systèmes sont étroitement liés et coordonnés pour travailler ensemble efficacement, et souvent avec l'homme dans la boucle de contrôle [16].

On peut distinguer trois parties principales dans un SCP tel que montré par la figure 1.2 suivante :

- Tout d'abord, l'installation physique est la partie *physique* d'un SCP. C'est simplement la partie du système qui n'est pas réalisée avec des ordinateurs ou des réseaux numériques. Elle peut inclure des pièces mécaniques, des procédés biologiques ou chimiques, ou des opérateurs humains.
- Deuxièmement, il existe une ou plusieurs plateformes de calcul, qui consistent en des capteurs, actionneurs, un ou plusieurs ordinateurs et éventuellement, un ou plusieurs systèmes d'exploitation.

- Troisièmement, il existe une structure de réseau, qui fournit des mécanismes de communication aux ordinateurs. Toutes ces parties ensemble, les plateformes et la structure réseau forment la partie « cyber » du SCP [4].

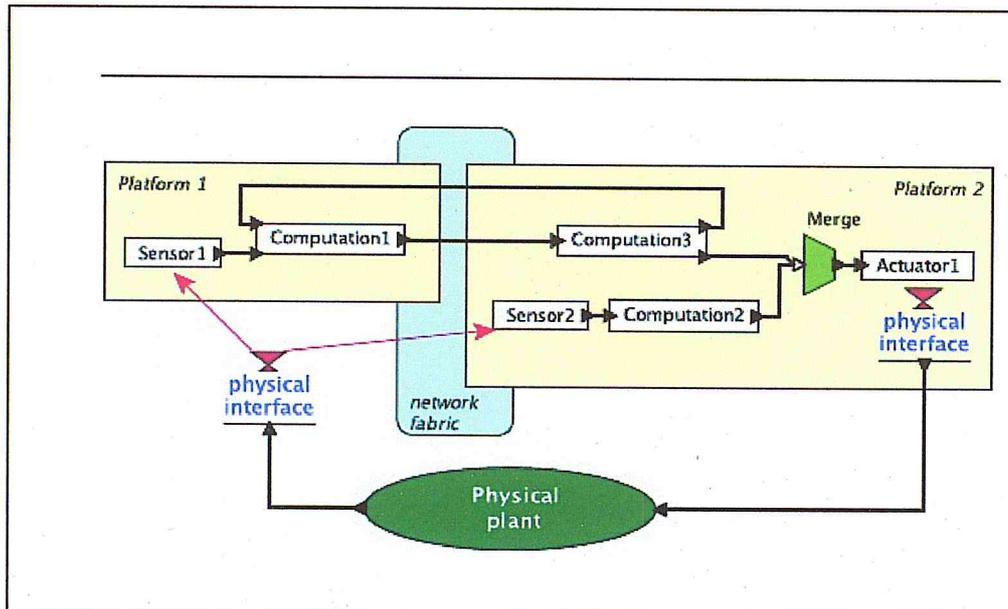


Figure1.2 : Exemple de structure d'un SCP [4].

2.3. Caractéristiques des SCP

Un SCP est un système logiciel intégré dans des dispositifs, bâtiments, moyens et voies de transport, systèmes de production, procédés médicaux, processus logistiques, processus de coordination, processus de gestion, etc. Un SCP permet de [7] [17] :

- Enregistrer directement les données physiques à l'aide de capteurs et influencer sur les processus physiques à l'aide des actionneurs.
- Évaluer et enregistrer les données, et interagir de façon active ou réactive avec le monde physique et numérique.
- Relier avec d'autres SCP et dans les réseaux mondiaux via les moyens de communication numériques (sans fil et/ou câblés, locaux et/ou mondiaux).
- Utiliser les données et les services disponibles à l'échelle mondiale.
- Comporter une série d'interfaces homme/machine dédiées.
- Entrée et feedback possible issues de l'environnement physique.
- Contrôle distribué.
- Performances temps-réel.

Les SCP apportent de nombreux avantages [18] :

- Construire des systèmes plus sûrs et plus efficaces.
- Réduire le coût des systèmes de construction et d'exploitation.
- Construire des systèmes complexes qui fournissent de nouvelles capacités.
- Baisser les coûts de calcul, de réseautage et de détection.
- Les ordinateurs et la communication sont omniprésents, ce qui permet de développer des SCP à une échelle nationale ou mondiale.

3. SCP et Internet des objets

Un SCP quelconque présente une capacité particulière de partage d'informations et de services avec d'autres SCP disséminés dans un cadre basé sur un environnement d'« Internet des Objets (IoT) ». Cependant, la frontière entre les SCP et l'IoT n'est pas clairement identifiée, puisque les deux concepts ont été chronologiquement conduits en parallèle, découlant de deux communautés autonomes, mais ils sont étroitement liés [19]. L'IoT et les SCP visent à accroître le lien entre le cyber espace et le monde physique, en utilisant les informations de détection et la technologie de l'interactif [20]. Il existe plusieurs technologies qui permettent de détecter et d'identifier directement et sans ambiguïté des entités numériques et des objets physiques dans un environnement d'intérieur ou d'extérieur, et ainsi de pouvoir récupérer, stocker, transférer et traiter, sans discontinuité entre les mondes physiques et virtuels, les données s'y rattachant.

Dans ce qui suit, quelques technologies d'identification et de détection dans un environnement d'intérieur seront analysées et comparées.

3.1. Identification par Ultrason

Le système à ultrasons (US) est constitué d'un ou plusieurs détecteurs de signaux US et une ou plusieurs balises à ultrasons. La communication est généralement point à point, entre chaque étiquette et chaque détecteur. Un tag US est composé d'un appareil sans fil avec sa propre alimentation électrique, qui est associé à l'objet. Cette balise émet périodiquement son identifiant au système en utilisant des ondes ultrasonores. Le plus proche détecteur US reçoit le signal d'identification de la balise, il l'analyse et la communique à un contrôle central qui maintient les informations de tous les détecteurs du système. Cette technologie utilise la mesure TOA (Time Of Arrival) ou TOF (Time Of Flight) qui mesure le temps de propagation de l'onde acoustique entre le capteur et la cible.

Les systèmes à US présentent de nombreux avantages :

- Permettent une mesure de l'espace libre.
- Les capteurs US sans contact travaillent indépendamment des caractéristiques du produit.

D'autre part :

- Leur très faible portée (ne dépasse pas quelques centimètres) limite les informations accessibles.
- Cette technologie nécessite une infrastructure de haute densité de détecteurs qui augmenterait le coût [21].

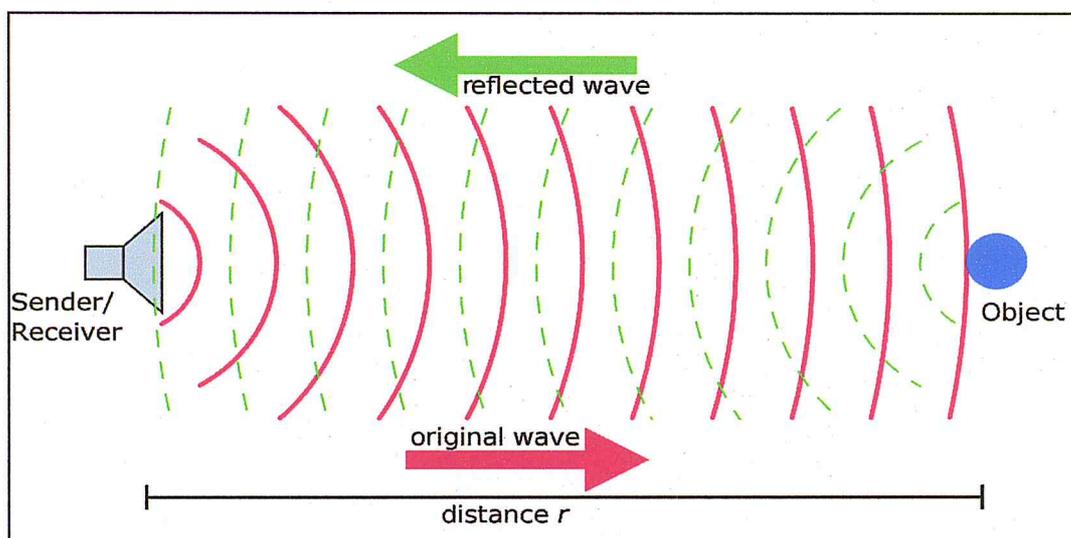


Figure 1.3 : Détection par Ultrason [22].

3.2. Identification par infrarouge

L'infrarouge (IR) est un faisceau de lumière. Les transmissions en IR doivent être très intenses afin qu'il n'y ait pas de confusion avec les nombreuses sources de lumière qui existent dans une pièce (fenêtres, néons, télévision, ampoules, etc.). La lumière IR possède une large bande passante, les débits sont relativement importants, mais la portée est faible (excellente à 10 cm, elle régresse de plus en plus jusqu'au 80 cm). Un réseau IR est commode et rapide. Le faisceau ne doit jamais être coupé sinon la transmission est interrompue.

Les systèmes IR présentent de nombreux avantages :

- La transmission IR offre l'avantage d'être très économique et la mieux adaptée.
- La technique IR permet également la confidentialité des transmissions.

D'autre part :

- La modulation directe en amplitude de la luminosité des diodes IR ne permet pas d'obtenir une couverture suffisante [23].

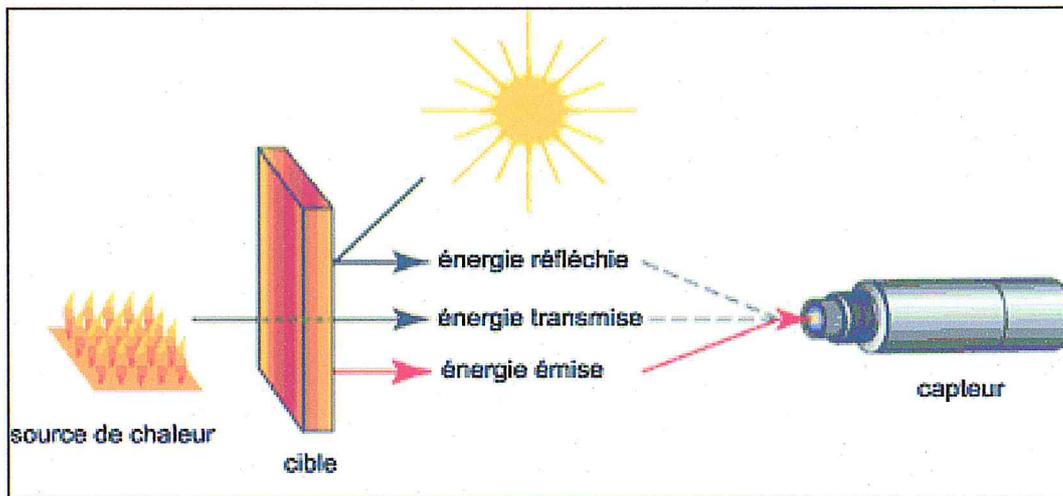


Figure 1.4 : Détection par Infrarouge [24].

3.3. Identification par bande ultra-large(UWB)

La technologie à bande ultra-large (*UWB, Ultra Wide Band*) est une technologie sans fil qui peut être exploitée à très faible densité de puissance pour la communication à des débits de données élevés sur de courtes distances. Les dispositifs UWB utilisent généralement des impulsions de très courte durée, de sorte qu'ils occupent de très grandes largeurs de bande d'émission. UWB peut être comparée à une autre technologie sans fil à courte distance, Bluetooth, qui est un standard pour connecter des périphériques portatifs sans fil avec d'autres appareils similaires.

La technologie à bande ultra-large (UWB) possède certaines caractéristiques intéressantes [25] :

- Permet la détection et l'identification de cibles malgré certains obstacles et certaines nuisances.
- Une caractéristique importante des systèmes UWB est leur gain de traitement élevé, lequel donne une indication de la résistance d'un système au brouillage.
- Dans les systèmes de communication sans fil qui utilisent la technologie UWB, l'information de la bande de base peut moduler directement des impulsions courtes ce qui produit des systèmes très simples.

D'autre part [21] :

- Les interférences avec d'autres technologies sans fil dépendent de la fréquence de travail.

3.4. L'identification par Wifi

La technologie Wifi est largement utilisée par les systèmes de détection et de localisation. Elle utilise les données de différents points d'accès (AP) [21] et une onde porteuse sur laquelle est modulé le signal à transmettre. En fait, il y a deux modes de fonctionnement [26] :

- **Mode de fonctionnement point à point (ou Peer-to-Peer) :** chaque ordinateur dispose d'un adaptateur réseau Wifi et se connecte directement à l'ordinateur cible.
- **Mode de fonctionnement en étoile :** un routeur est installé comme point central de communication, chacune des communications entre les ordinateurs équipés d'adaptateurs Wifi passent par le routeur.

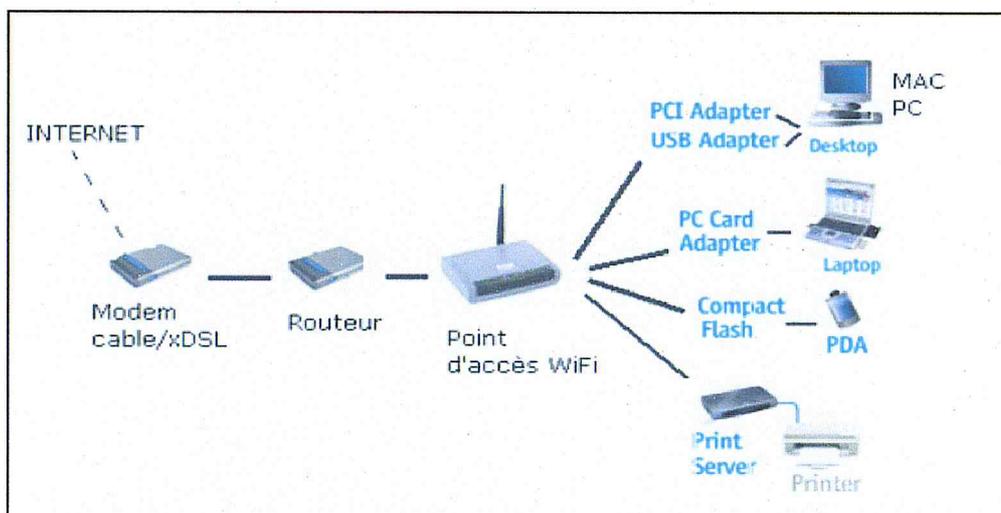


Figure 1.5 : Détection par Wifi [26].

Les avantages que présente cette technologie sont [21] :

- Peu infrastructures nécessaires.
- Localisation peut être atteinte par le dispositif lui-même et les données peuvent être facilement envoyées.
- Bande passante de travail ouverte.

D'autre part, elle présente certains inconvénients [21] :

- Les périphériques Wifi sont plus chers que ceux utilisés dans Bluetooth, ZIGBEE ou des technologies RFID.

- La consommation électrique est plus élevée que dans autres technologies.

3.5. Identification par RFID

L'abréviation RFID signifie « Radio Frequency IDentification » ou « Identification par Radio Fréquence ». Cette technologie permet d'identifier un objet, d'en suivre le cheminement et d'en connaître les caractéristiques et d'avoir très rapidement des informations précises, fiables, plus conséquentes, à l'instance grâce à une étiquette émettant des ondes radio, attachée ou incorporée à l'objet [27].

Un système RFID repose sur les éléments suivants [28] :

- Tag RFID (étiquette ou transpondeur) : c'est un dispositif d'identification électronique qui est constitué d'une puce et d'une antenne.
- Lecteur RFID (station de base) : il a pour but d'identifier le transpondeur.
- Le système de traitement des données : c'est une infrastructure informatique qui sert à collecter et à exploiter les données.

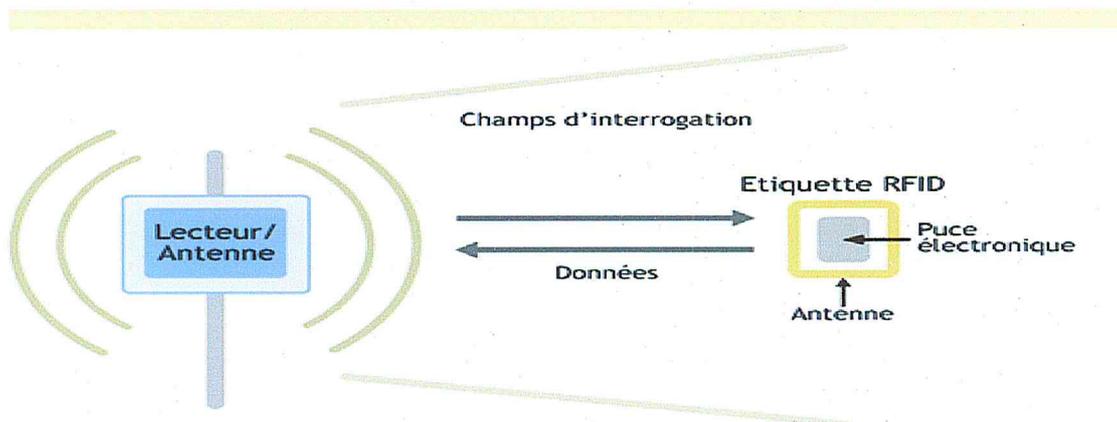


Schéma de principe de la RFID

Figure 1.6 : Principe de la RFID [29].

La RFID présente de nombreux avantages [30] :

- La possibilité de mise à jour du contenu par les intervenants.
- Une grande (et grandissante) capacité de contenu.
- Une vitesse de lecture et écriture rapide.
- Une sécurisation possible d'accès au contenu.
- Une très grande durée de vie.
- Une grande souplesse de positionnement.

- Une faible sensibilité à l'environnement d'utilisation.

D'autre part, elle présente certains inconvénients [30] :

- Un cout toujours trop élevé même si il est à la baisse.
- Les interrogations (et normes) concernant l'impact des ondes électromagnétiques sur la santé.

3.6. Comparaison entre les différentes technologies d'identification

Les technologies UWB et US offrent la meilleure précision. Toutefois, les deux technologies sont ceux avec le niveau de déploiement le plus bas. La technologie US est en développement. Les signaux UWB sont potentiellement plus difficiles à détecter que les signaux de radio communication classiques, d'où UWB est donc difficile à réglementer. En plus, certains des systèmes commerciaux basés sur UWB sont disponibles, mais ils sont trop chères pour les déploiements réels.

D'autre part, la technologie Wifi est largement utilisée par rapport aux autres. Néanmoins, le niveau de précision atteint par ces systèmes peut ne pas être approprié pour certaines applications.

Le système IR offre une bonne précision et assure un grand débit d'information. Mais, il nécessite une visibilité directe.

L'identification par RFID semble être un choix attractif ayant un équilibre approprié entre la précision et le niveau de déploiement.

4. SCP et RFID

Un système RFID est un SCP typique essentiellement en raison de ses composants fonctionnels et physiques [14] :

- **Élément de calcul** : même si une étiquette RFID passive n'a normalement qu'à la fonction de stockage, l'ensemble du système RFID et le système de post-traitement (principalement le lecteur de tags RFID) contiennent les fonctions de calcul et de traitement des données.
- **Élément de contrôle** : généralement un système RFID est sous le contrôle d'une micro-unité interne de contrôle (MCU).
- **Élément de communication** : dans un système RFID, la quasi-totalité des informations sont échangées en utilisant les ondes radiofréquence, les données et les flux de contrôle sont établis via une communication RF à deux voies. Pendant ce

processus, le système RFID traditionnel utilise des tags électroniques placés sur les différents objets afin de suivre leurs emplacements ou descriptions. La figure 1.7 représente une étiquette RFID simple [31].



Figure 1.7 : *Structure et vue externe de tags RFID [14].*

5. Applications des SCP

Les SCP deviennent un spectacle de plus en plus courant dans de nombreuses parties de la société moderne, que ce soit dans l'industrie, dans l'armée, ou dans le contexte civil. La recherche sur ces systèmes est d'une importance fondamentale pour les systèmes d'ingénierie dans de nombreux domaines d'applications importants tels que le transport, la logistique, et les systèmes médicaux [8]. Quelques applications des SCP utilisant la technologie RFID seront présentées dans ce qui suit :

5.1. Domaine médical

Dans le domaine médical, la RFID servira à l'identification des patients, des objets et des médicaments afin de réduire les erreurs médicales et d'améliorer les conditions de vie des patients.

En 2007, dans l'hôpital de Nice en France, ils ont testé cette nouvelle méthode de travail. L'objectif principal était d'éviter les goulets d'étranglement et d'améliorer la prise en charge du patient. Dans cet hôpital, chaque patient est équipé à son arrivée d'un bracelet RFID comportant son identité et le degré de gravité de ses symptômes, évaluée sur une échelle chiffrée. Ils ont réalisé une application informatique qui permet, par la suite, d'établir un parcours médical prenant en compte le degré d'urgence relatif, la disponibilité des médecins et les différents équipements étiquetés par des puces RFID [32].



Figure 1.8 : Un patient équipé avec un bracelet RFID [33].

- **Aide aux personnes dépendantes :** les bracelets RFID peuvent aider les personnes fragiles et dépendantes de manière que lorsqu'un patient tombe, le bracelet avertit automatiquement le personnel en précisant le lieu où il se trouve. Les bracelets RFID serviront aussi à contrôler les personnes démentes à travers les lecteurs placés sur la porte de sortie qui permet de verrouiller la porte dès qu'un patient s'en approche [32].
- **Gestion des prélèvements sanguins :** les étiquettes et les lecteurs RFID peuvent être utilisés pour connaître les différentes informations relatives au sang (groupe sanguin, rhésus, etc.) et contrôler à tout moment la qualité de l'échantillon pendant sa durée de vie. Ceci permet d'éliminer les pertes de temps, augmenter la disponibilité et la fiabilité des données liées à chaque échantillon et donc faciliter le processus de gestion des prélèvements sanguins [32].
- **Suivi des médicaments :** la RFID permet d'assurer l'identité d'un médicament; où les produits sans étiquettes sont considérées comme dangereux [32].

5.2. Domaine militaire

L'armée américaine exige que tous les cartons et les palettes soient livrés avec une étiquette d'expédition militaire qui affiche des données d'expédition. Le ministère de la défense américaine a proposé la solution RFID de conformité militaire comme un moyen d'aider les fournisseurs et les fabricants à satisfaire aux nouvelles normes de l'armée pour la RFID.

L'armée américaine utilise des étiquettes RFID actives et passives pour les systèmes de communication et de transport en Irak et en Afghanistan, elle est en mesure de diagnostiquer et résoudre les problèmes beaucoup plus rapidement qu'auparavant. La mise en œuvre de la RFID dans tout ce domaine permettra d'économiser la clôture militaire. L'armée

a réussi à créer une meilleure visibilité tout au long de la chaîne d'approvisionnement en augmentant la productivité et la stabilité [34].

5.3. Domaine logistique

La gestion de la chaîne d'approvisionnement en matière de logistique est l'une des applications de la RFID en entreprises. Trois niveaux peuvent être distingués [35] :

1. **Expédition** : l'étiquette peut faciliter le picking des produits, la constitution des palettes, leurs tris par destination et le contrôle du chargement. L'étiquette contient l'identification du produit, ou du contenu d'une palette, celle du numéro de lot de fabrication, l'identification du destinataire, le numéro de commande, des particularités de manutention, etc. Ces informations sont stockées dans l'étiquette RFID afin de faciliter les contrôles en cours de transit.
2. **Réception** : les données pourront automatiquement être collectées pour effectuer les contrôles et mettre à jour les stocks et effectuer les rapprochements avec les documents commerciaux.
3. **En transit** : l'étiquette permet de tracer les produits à chaque point de chargement et de déchargement ou simplement de passage. Ainsi, l'expéditeur peut être à tout moment informé du déroulement du transport local [35].

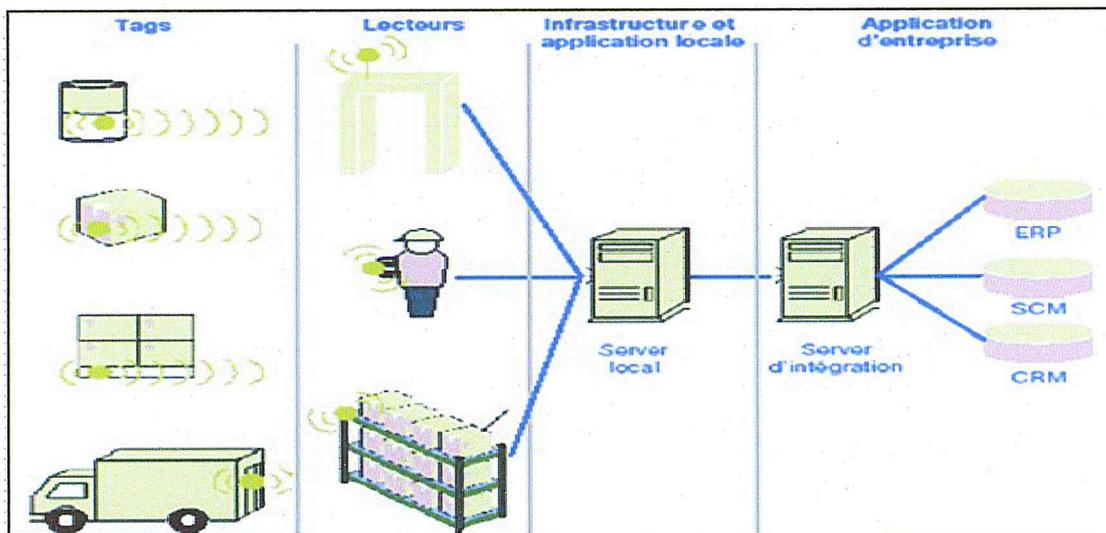


Figure 1.9 : Chaîne d'approvisionnement avec la RFID [35].

5.4. Domaine de l'automobile

Le véhicule est développé à partir d'un système purement physique sur la base des lois de la mécanique et de la chimie, à un SCP plus sophistiqué qui intègre les composants électroniques et les systèmes de contrôle pour améliorer la performance et la sécurité.

Les véhicules sont bien adaptés à la collecte d'informations sur le monde physique local, mais ils n'ont pas la capacité de recueillir des informations globales sur l'environnement dans lequel ils évoluent. La plupart des capteurs automobiles sont spécifiquement conçus à surveiller l'infrastructure dans le véhicule lui-même, telle que la température du moteur ou les pannes de certaines composantes. La détection localisée est efficace pour la gestion des questions telles que la fiabilité de véhicule, mais elle peut seulement fournir des solutions limitées pour des aspects de SCP à grande échelle telle que la sécurité, planification d'itinéraire, etc. En plus, elle ne prévoit pas de surveillance suffisante pour prévenir les accidents.

Dans le cadre d'amélioration du SCP automobile actuel, et pour éliminer les limitations citées auparavant. La meilleure solution était d'ajouter des capteurs RFID puissants à l'extérieur du véhicule qui permettent de déterminer la localisation des véhicules voisins et les échanges des différentes informations environnantes telles que des conditions routières, des conditions de vent, etc. D'autre part, un panneau niveau de ces capteurs nécessite une reconfiguration de tout le système. En plus, la majorité des véhicules ne sont pas équipés avec ces capteurs vus leur cherté [36].

5.5. Bâtiment intelligent

En 2008 à l'université de Washington, ils ont réalisé un écosystème RFID afin de tester le déploiement de l'infrastructure RFID dans les étages d'un bâtiment. Ce déploiement comprend des participants qui portent des étiquettes RFID sous forme des badges et étiquettes attachées à des objets ainsi qu'un ensemble de lecteur s'installés dans l'immeuble. Ceci permet d'acheminer ou de trouver la trace de ces participants dans n'importe quel endroit de cet immeuble.

Tous les lecteurs exécutent un logiciel personnalisé qui traite les nouvelles données RFID. Ce logiciel interroge le matériel de lecture des étiquettes RFID nouvellement détectées et génère un événement TRE (Tag Read Event) par mot clé, par antenne et par seconde avec le triplet (tag ID, ID antenne, temps). Toutes ces données sont stockées dans une base de données répartie.

Dans ce projet, un ensemble d'outils Web a été développé permettant aux utilisateurs de faciliter leur compréhension, la gestion et le contrôle des données avec toute confidentialité. Une étude de quatre semaines sur les utilisateurs a été menée pour mesurer les tendances dans l'adoption et l'utilisation des outils et des applications ainsi que les réactions qualitatives des utilisateurs [37].

5.6. Avantages du système RFID

D'après la présentation des applications des SCP utilisant la technologie RFID, quelques avantages du système RFID sont cités dans ce qui suit :

- **Une plus grande capacité de contenu** : dans une étiquette RFID une capacité de 1000 caractères est aisément stockable.
- **Une sécurité d'accès au contenu** : l'étiquette RFID peut être protégée par mot de passe en écriture ou en lecture. Dans une même étiquette, une partie de l'information peut être en accès libre, et l'autre protégée. Cette faculté fait de l'étiquette, un outil adaptée à la lutte contre le vol et la contrefaçon.
- **Une plus grande souplesse de positionnement** : avec l'étiquette RFID, il est possible de s'abstraire des contraintes liées à la lecture optique, elle n'a pas besoin d'être vue. Il lui suffit d'entrer dans le champ du lecteur pour que sa présence soit détectée.
- **Une moindre sensibilité aux conditions environnementales** : les étiquettes RFID n'ont pas besoin d'être positionnées à l'extérieur de l'objet à identifier. Elles peuvent donc être mieux protégées des agressions liées aux stockages, aux manutentions ou au transport [35].

5.7. Inconvénients d'un système RFID

D'autre part, le système RFID provoque quelques inconvénients :

- **Interrogations sur l'impact de la radio fréquence sur la santé** : les étiquettes RFID ne présentent aucun risque quel que soit leur nombre puisqu'elles ne sont actives que lorsqu'elles se trouvent dans le champ d'un lecteur. Les études portent donc essentiellement sur les lecteurs et visent à définir les critères de régulation de leur puissance d'émission afin d'éviter qu'ils ne créent des perturbations sur les équipements de santé [35].
- **L'inacceptation de la société envers cette nouvelle technologie** : les gens sont inquiets que leurs informations soient stockées.

- **Le coût d'implantation d'un système RFID est relativement élevé.**

6. Applications robotique et SCP

L'utilisation de SCP dans le domaine de la robotique est une tendance qui prend de l'ampleur, les applications les plus spectaculaires demeurent dans la robotique chirurgie, les usines automatisées, la robotique militaire, etc. Quelques applications des SCP dans le domaine robotique existantes seront présentées dans ce qui suit :

6.1. RFID et robots ménagers

En 2014, l'équipe de Georgia Tech a appliqué des étiquettes RFID sur plusieurs types d'objets (flacons de médicaments, télécommandes de téléviseurs, téléphones, etc.). Ils ont également équipé un robot PR2 de Willow Garage avec deux antennes RFID placées sur ses épaules. Ces antennes peuvent détecter jusqu'à 150 tags par seconde et à une distance de 3m dans des conditions pratiques ou dépassant 6m sous la ligne idéale de conditions de visibilité. L'objectif est de permettre aux robots de trouver et accéder à des objets quotidiens dans l'environnement. En tant que tel, la position d'un objet étiqueté dans son environnement peut changer de façon inattendue.

Ils ont formulé la recherche et la navigation vers un objet étiqueté comme un problème d'optimisation où le robot doit trouver la position d'une antenne directionnelle qui maximise le RSSI (The received signal strength indicator) d'une étiquette avec un ID correspondant à l'objet cible.

Enfin, ils ont démontré que cette approche peut permettre à un PR2 de trouver et accéder à différents objets marqués dans une vraie maison. Grâce à des protocoles d'anti collisions entre le lecteur et les étiquettes, certaines balises peuvent coexister dans l'environnement sans interférence [38].



Figure 1.10 : Approchement du robot PR2 aux objets [38].

6.2. Stratégie de la localisation d'objet en utilisant la technologie RFID

En 2012, à l'université Umeå en suède, ils ont réalisé un projet sur la détection des objets par un robot mobile en utilisant la technologie RFID. L'environnement de travail contient des objets étiquetés et des lecteurs RFID avec double antennes montés sur les deux cotés et devant le robot Kompai. Le projet a été réalisé suivant trois méthodes :

La première méthode consiste à mettre en œuvre le robot comme un véhicule Braitenberg¹. Ce véhicule est capable de détecter le signal reçu de l'étiquette RFID et peut se déplacer. Il est composé de deux parties principales : les capteurs et les roues.

La deuxième et troisième méthodes permettent d'estimer la localisation de l'objet et de déplacer le robot Kompai vers l'objet en utilisant la mesure AOA (Angle of Arrival) qui permet de déterminer l'angle entre la ligne de récepteur émetteur et la direction de référence.

Afin de rendre le contrôle plus facile, l'interface utilisateur graphique (GUI) a été construite avec les paramètres nécessaires. Le programme a des fonctions, qui effectue ces méthodes et mis en œuvre avec une base de données simple qui contient les informations suivantes : l'identifiant du tag, la description de l'objet marqué, l'image de l'objet, ainsi qu'un ensemble d'opérations pour gérer cette base de donnée telle que l'ajout, la modification et suppression [39].

¹Valentino Von Braitenberg est connu pour son livre *Véhicules : Expériences en psychologie synthétique* dans lequel il décrit comment de petits « véhicules » peuvent évoluer par l'ajout de capteurs, d'actionneurs et de connexions simulant des réseaux de neurones basiques.



Figure 1.11 : Le robot kompai dans laboratoire de l'université Umea[39].

6.3. Chirurgie médicale

Dans le monde, il existe plusieurs types de robots dans différents domaines, la médecine est un des plus concernés par les évolutions technologiques. On distingue les robots en tant qu'outils chirurgicaux, et ceux d'aide au diagnostic. Ces types de robot sont, en effet, été développé, offrant aux médecins de nouvelles perspectives, en terme de perception des pathologies, de précision dans le geste médical ou encore d'entraînement.

DaVinci est un robot chirurgical, plus précisément une machine dirigée par un chirurgien pour réaliser des opérations. C'est un système embarqué qui combine la robotique, l'informatique et technologies IA, pour fournir, non seulement, un environnement connecté des dispositifs d'interaction, mais, aussi, en mettant l'accent sur l'exécution des tâches physiques. Il est composé de deux parties suivantes [40] :

Une console permettant au chirurgien de guider les bras mécaniques avec des sortes de joysticks.

La partie active du robot, c'est-à-dire l'ensemble des bras opérateurs situés à côté de la table d'opération.

DaVinci est composé d'une caméra stéréoscopique et de deux tiges dont l'extrémité est flexible et auxquelles sont fixés les instruments chirurgicaux nécessaires à l'intervention [41].

6.4. Coopération entre caméras d'ambiance et capteur embarqués

En 2012 à l'université de Toulouse (France), une stratégie de coopération entre des caméras d'ambiance et des capteurs embarqués sur un robot mobile a été réalisée. Le but était

de suivre une personne donnée et identifiable par son badge RFID. Le système est composé d'un robot mobile et deux caméras fixées sur les murs. Ce robot embarque divers capteurs tel qu'un laser SICK, une caméra numérique et un lecteur de badges RFID. Il embarque également deux PCs et une connexion Ethernet sans fil, la communication entre le robot mobile et le PC dévolu aux caméras est assurée par Wifi. Les fonctionnalités perceptuelles du système se basent sur divers modules de détection, dévolues aux capteurs précités, sont mises en œuvre ; ceux-ci sont définis comme suit [42] :

- **Laser SICK** : La démarche vise à segmenter les jambes humaines grâce à des contraintes géométriques spécifiques.
- **Lecteur RFID** : Ce lecteur embarqué sur le robot détecte tous les tags sur 360° et qui se trouvent à une distance entre 0.5 et 4.5 m relativement au robot. Il permet d'identifier les badges RF ensuite les localiser grossièrement en distance et azimut.
- **Caméras d'ambiance** : Les deux caméras fixées aux murs procurent des champs de vue larges et en recouvrement.
- **Vision embarquée** : Cette caméra est dévolue à la perception approximative de la personne taguée.

6.5. Avantages d'un système RFID dans la robotique

- Le robot peut reconnaître l'objet inconnu et non visibles en utilisant le signal RF transmis par l'étiquette RFID attachée à l'objet.
- Le système RFID peut être installé facilement sur les robots.

6.6. Inconvénients d'un système RFID dans la robotique

- La difficulté de détecter les transpondeurs multiples par les robots en même temps.
- La portée de transmission de transpondeur est limitée, ce qui interdit un robot d'interagir avec un transpondeur distant.

7. Conclusion

Les SCP présentent un intérêt considérable et apportent de nombreux avantages dans les principaux secteurs (santé, militaire, logistique, l'automobile, etc.). Ces systèmes combinent le calcul, la communication, et de la dynamique physique. Ils sont difficiles à modéliser, à concevoir, et à analyser. Dans ce chapitre, nous avons présenté les SCP, leurs applications dans différents domaines notamment dans la robotique utilisant la technologie

RFID. Cependant, nous avons remarqué que le déploiement d'une infrastructure cyber-physique utilisant l'approche RFID reste toujours une question importante dans ce domaine.

Dans le prochain chapitre nous allons présenter notre solution proposée pour répondre à cette thématique en détaillant les différentes étapes de développement.

1. Introduction

Dans un système robotique cyber-physique (SRCP), la capacité de détecter et d'identifier des éléments fixes et mobile dans un environnement et le stockage des différentes données obtenues via un système pertinent et fiable est une tâche importante.

Les travaux évoqués et traités dans le chapitre précédent nous ont permis de proposer notre propre solution pour atteindre nos objectifs. Dans ce chapitre, nous allons présenter et décrire notre solution ainsi que les étapes de conception et de développement utilisant le langage de modélisation unifié (UML).

2. Description de la solution proposée

Afin de répondre à notre problématique principale, nous allons utiliser la technologie RFID dans un système robotique cyber-physique. À cet effet, nous avons décidé de simuler un système multi-robots au niveau de rez-de-chaussée de la Division Productique et Robotique (DPR) du Centre de Développement des Technologies Avancées (CDTA) de manière que toutes les entités de cet environnement tel que les opérateurs humains, les robots, les obstacles statiques et dynamiques et les différents objets soient étiquetées par des tags RFID.

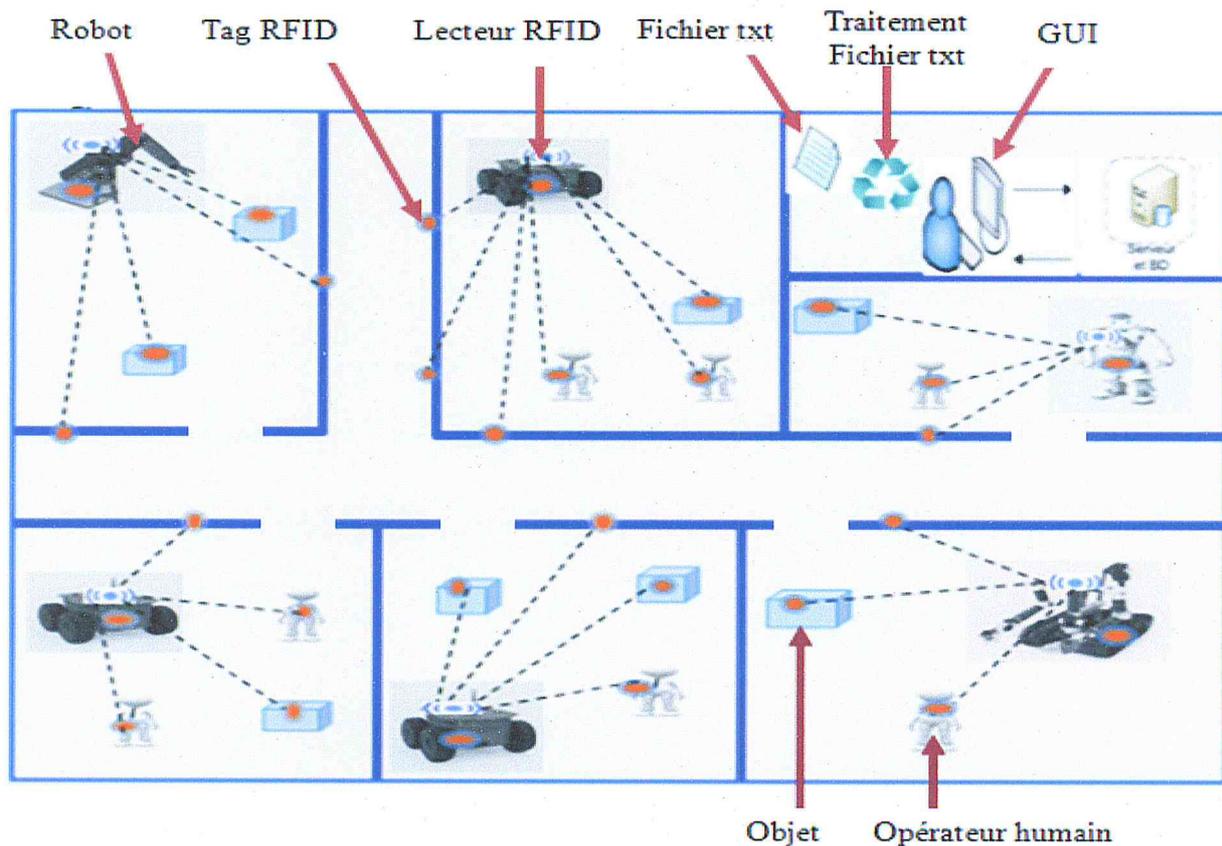


Figure 2.1 : Schéma descriptif de notre système.

Notre solution proposée est basée sur les deux phases suivantes :

2.1. Première phase : Déploiement d'une infrastructure RFID

Cette phase est réalisée en deux parties.

2.1.1. Première partie : Détection et identification

Consiste à programmer les lecteurs RFID installés sur les robots de telle sorte qu'ils soient capables, de détecter et d'identifier tous les tags situés dans leurs voisinages. Nous considérons que les étiquettes (les tags) peuvent être inscriptibles, attachées à n'importe quel objet quelle que soit sa taille, sa nature actif ou passif. Les objets actifs sont ceux qui peuvent changer leurs positions fréquemment (par exemple : des tables, des chaises, des robots, des opérateurs humains, etc.). En revanche, les objets passifs sont des éléments avec des positions fixes (par exemple : des murs, des plateformes, etc.). Ces étiquettes comprennent un identifiant unique et des informations supplémentaires notamment leurs emplacement et encore d'autres pour décrire et particulariser ces différents objets.

2.1.2. Deuxième partie : Extraction chemin

Pour conclure cette phase, nous allons sauvegarder la trace des différentes positions par lesquelles sont passés les robots pendant la détection, en s'inspirant de l'algorithme de Colonie de fourmis [43].

Algorithme : Extraction chemin

Entrées : les tags.

Sorites : chemin avec son propre phéromone O. // Phéromone O est un identifiant qui caractérise un chemin précis.

Début

Tant que *le robot se navigue* **faire**

Détecter les tags ;

Si $L(t) \neq L(t-1)$; // $L(t)$ représente l'ensemble des tags détectés à l'instant t.

Déposer une phéromone O et récupérer position courante du robot ;

Sinon

Arrêter ;

Fin

Fin

Fin

Après avoir gardé la trace des différents chemins des robots, de telle sorte que chaque chemin possède un ensemble de positions, une position initiale (x_i, y_i) et finale (x_f, y_f) et une distance euclidienne qui le caractérise, nous allons extraire le plus court chemin avec la distance minimale.

$$Distance_{chemin\ i} = \sum_i^n \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2}$$

2.2. Deuxième phase : Traitement de base de données

Dans cette phase, les informations perçues par les détecteurs telle que les identifiants des objets, la position courante du robot, le temps de détection, etc. sont envoyées pour la création d'un fichier texte « .txt » accessible par tous les robots du SRCP. Par la suite, nous allons traiter ce fichier et exploiter une base de données centralisée pour stocker ces données plus les autres informations supplémentaires de chaque balise existante dans l'environnement. La mise à jour de cette base de données et l'insertion des nouveaux objets détectés s'effectuent après chaque étape de détection. En plus, nous allons réaliser des interfaces utilisateur graphique pour gérer et traiter les entités de notre système.

3. Conception

3.1. Présentation du langage UML

UML « Unified Modeling Language », ou « Langage de Modélisation Unifié » est un langage graphique dont les tâches et les données sont représentées par assemblage des éléments graphiques, qui est une formalisation très aboutie dans la modélisation objet.

UML est fondé sur le concept *orienté objets* ; on ne raisonne plus en terme d'actions mais plutôt en objets du monde physique. Un objet intègre à la fois des données et des opérations.

Ce langage de modélisation est conçu pour élaborer des logiciels, il permet une description efficace de n'importe quel système. UML est standard, intuitif, homogène et cohérent. Les symboles embrouillés, redondants ou superflus ont été éliminés en faveur d'un meilleur rendu visuel.

Les éléments de base du langage UML se composent souvent de rectangles, flèches, etc. Ces éléments s'assemblent pour former des diagrammes, qui sont dépendants et se complètent de façon à permettre la modélisation d'un projet tout au long de son cycle de vie.

La force et la vitesse du langage UML se trouvent dans le nombre de diagrammes utilisés [44].

3.2. Modélisation

Dans la plupart des méthodes d'analyse et de conception orientées objets, l'étude d'un problème est réalisée suivant trois aspects :

1. **Aspect fonctionnel** : l'un des avantages majeurs de ce langage réside dans la prise en considération de l'ingénierie des besoins. L'objectif de la capture des besoins ou exigences consiste à déterminer ce que le système informatique doit faire, c'est-à-dire le quoi afin de favoriser une meilleure compréhension des fonctionnalités du système lors du développement (*savoir-faire de l'objet*). Généralement, cet aspect est représenté par un diagramme de cas d'utilisation [44].
2. **Aspect dynamique** : où on précise le comportement des objets, les différents états par lesquels ils passent et les événements qui déclenchent ces changements d'états. On parle souvent de cycle de vie d'un objet (*États et messages de l'objet*). Cet aspect est représenté par un diagramme de séquences ou un diagramme d'activités [44].
3. **Aspect statique** : où on identifie les propriétés des objets et leurs liaisons avec les autres objets (*structure de l'objet*). Cet aspect est souvent représenté par un diagramme de classes [44].

3.2.1. Diagramme des cas d'utilisation

Avant de développer un système, il faut savoir précisément à quoi il devra servir. C'est-à-dire à quel besoin il devra répondre.

Un diagramme de cas d'utilisation (*use case*) répond à la question « Qu'est-ce que les utilisateurs attendent ? ». Il peut facilement exprimer les besoins et décrire ce que le futur système devra offrir comme fonctionnalités.

Un diagramme Use Case ou Cas d'utilisation doit aussi permettre de répondre à la question « Qui fait quoi ? ». Il doit décrire sous forme d'actions et de réactions le comportement d'un système du point de vue d'un utilisateur externe [44] [45].

3.2.1.1. Description textuelle des cas d'utilisations

Le tableau ci-dessous (tableau 2. 1) représente une description détaillée des différents cas d'utilisation du notre système ainsi que les acteurs qui interagissent avec ce système.

Cas d'utilisation	Description de cas	Acteur	Sous cas
Détecter les objets.	Ce cas assure les tâches de détection des différents objets de l'environnement y compris la récupération de l'identifiant de l'objet et la vérification de la portée.	Robot (Acteur principal)	Récupérer l'identifiant de l'objet. Vérifier la portée.
Récupérer l'identifiant de l'objet.	Cette tâche permet de récupérer l'identifiant du tag de chaque objet détecté dans l'environnement.	Robot (Acteur principal)	
Vérifier la portée.	Cette fonction consiste à vérifier si l'objet est dans la portée du lecteur monté sur le robot, c.-à-d. il faut que la distance entre le robot et cet objet soit ≤ 9 m pour qu'il puisse être détecté.	Robot (Acteur principal)	
Enregistrer dans un fichier « .txt ».	Après la récupération de l'identifiant du tag, cette information, la position courante du robot et de l'objet ainsi que le temps de détection sont enregistrés dans un fichier « .txt ».	Robot (Acteur principal)	
Traiter le fichier « .txt ».	Ce cas assure les tâches de traitement de fichier « .txt » y compris la visualisation et l'intégration du fichier dans la base de données.	Administrateur	Visualiser le fichier « .txt ». Intégrer le fichier dans la BDD.
Visualiser le fichier « .txt ».	Quand le fichier est chargé, on pourra par la suite visualiser les différents détails de détection.	Administrateur	

Intégrer le fichier dans la BDD.	Après la visualisation des informations, il y a lieu de procéder à l'intégration des données dans la BDD.	Administrateur	
Extraire un chemin.	Permet d'extraire un chemin de robot.	Administrateur	Rechercher le plus court chemin.
Rechercher le plus court chemin	Quand il s'agit de plusieurs chemins avec des positions initiales et finales identiques, nous recherchons le plus court chemin.		
Gérer la BDD.	Ce cas assure les tâches de gestion des tags y compris l'ajout, la modification et la suppression.	Administrateur	Ajouter. Modifier. Supprimer. Rechercher.
Ajouter	Ajouter un nouveau tag d'un objet, d'un robot ou d'un lecteur dans l'environnement.	Administrateur	
Modifier	Modifier les informations relatives à un tag, un robot ou un lecteur déjà ajouté.	Administrateur	
Supprimer	Permet de supprimer un tag ou robot ou un lecteur par son ID.	Administrateur	
Rechercher	Permet de rechercher un tag ou un robot ou un lecteur par son ID.	Administrateur	

Tableau 2.1 : Description textuelle des cas d'utilisations.

3.2.1.2. Diagramme de cas d'utilisation du système

Ce diagramme permet de donner une vision globale du comportement fonctionnel du notre système ainsi que les acteurs (robot qui est l'acteur principal et administrateur) et leurs interaction avec ce système. Le rôle de notre acteur principal robot est de détecter les

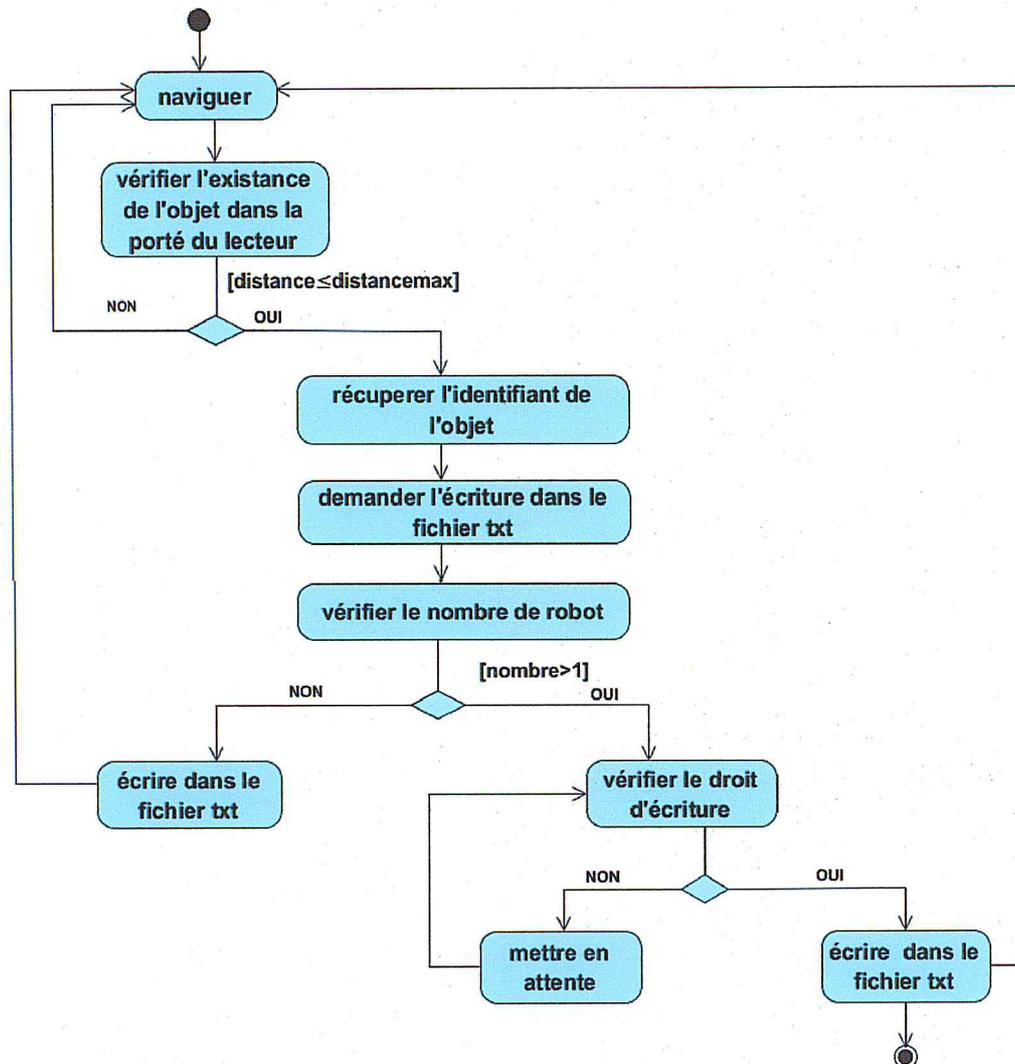


Figure 2.3 : Diagramme d'activité pour la détection d'objets et l'écriture dans un fichier « .txt »

3.2.3. Diagramme de séquence

Le diagramme de séquence permet de décrire comment les éléments du système interagissent entre eux et avec les acteurs pour accomplir une tâche du système. Ce diagramme montre les interactions selon un point de vue temporel.

Un diagramme de séquence représente l'interaction entre les objets en insistant sur la chronologie des envois de messages [44] [45].

3.2.3.1. Diagramme de séquence pour l'authentification

Le diagramme de la figure 2.4 représente les collaborations entre l'objet administrateur, système et base de données selon un point de vue temporel pour effectuer l'authentification.

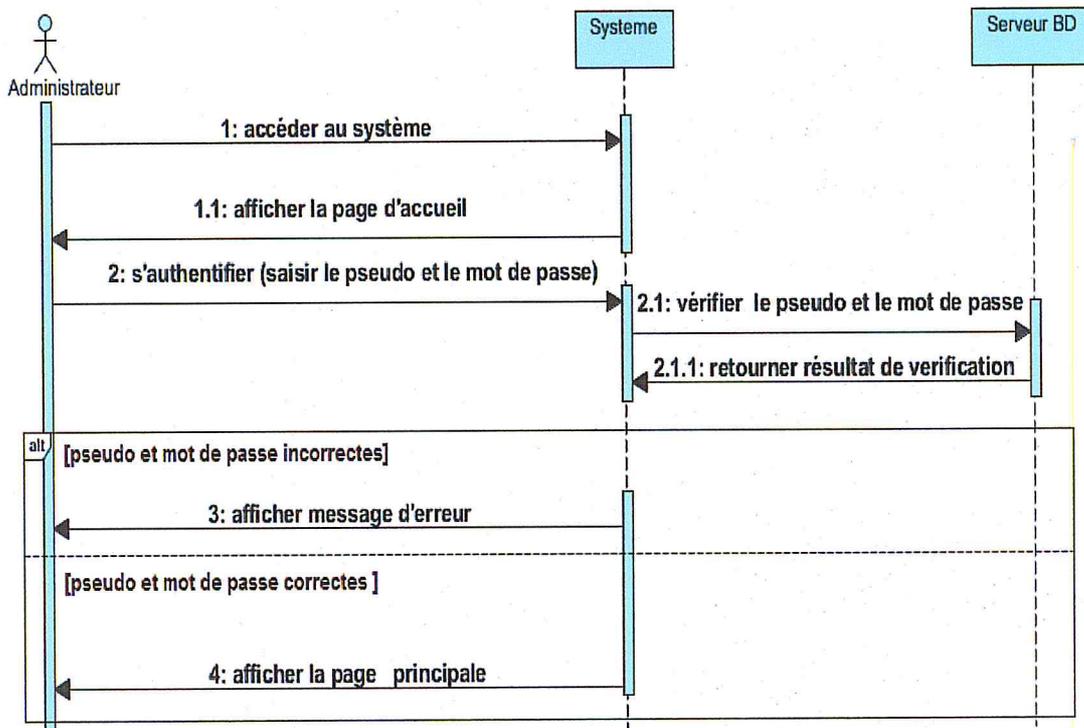


Figure 2. 4 : Diagramme de séquence pour l'authentification.

3.2.3.2. Diagramme de séquence pour traiter le fichier « .txt »

Le diagramme de la figure 2.5 décrit l'interaction entre l'acteur administrateur avec le système et la base de données selon un ordre chronologique. Il décrit également les différents cas alternatifs possibles afin de traiter le fichier « .txt » qui contient les informations de détection et d'identification pour mettre à jour et intégrer ces informations dans la BDD.

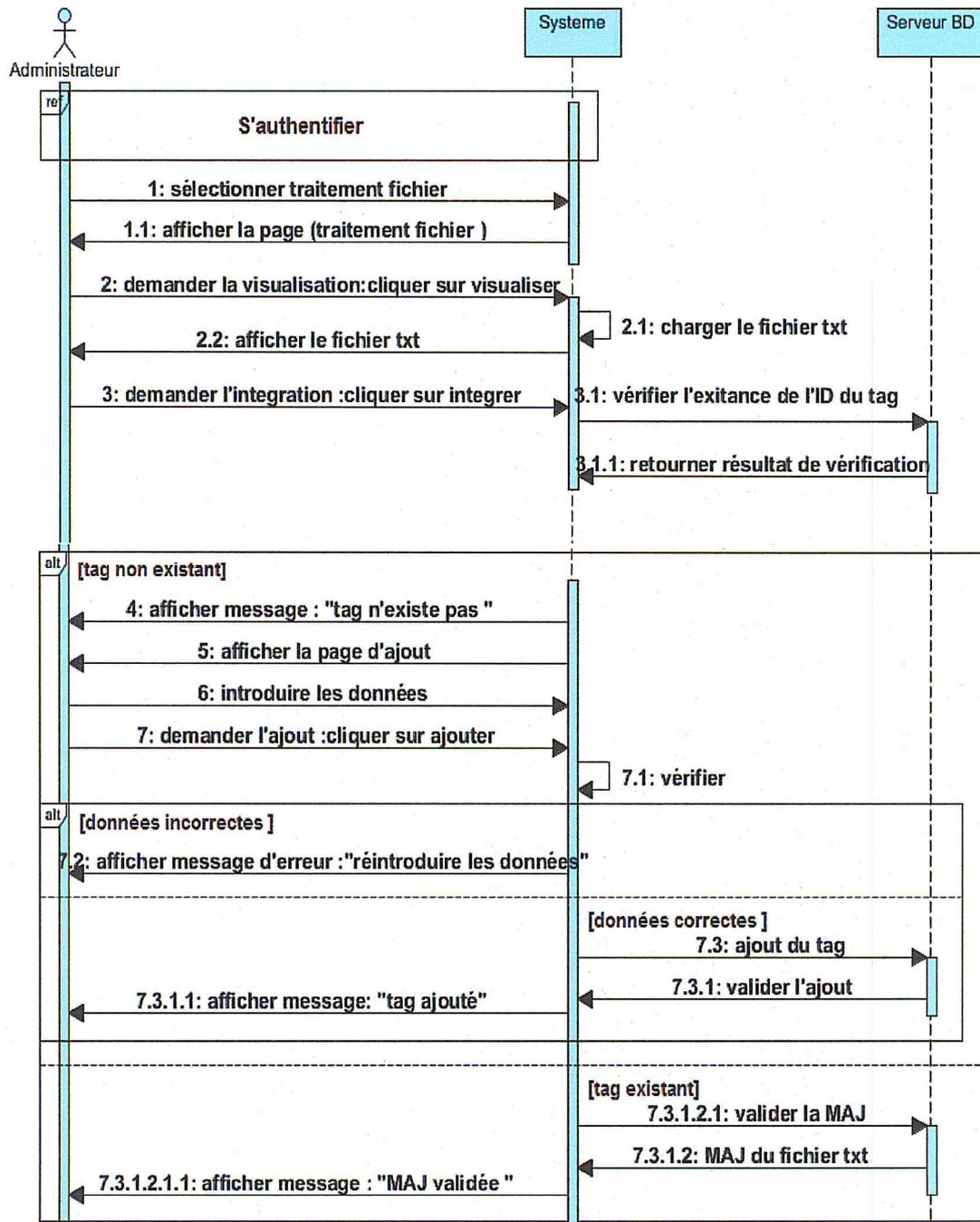


Figure 2.5 : Diagramme de séquence pour traiter le fichier « .txt »

3.2.3.3. Diagramme de séquence pour l'extraction d'un chemin

Le diagramme de la figure 2.6 montre comment l'administrateur interagit avec le système pour extraire les différents chemins des robots et rechercher le plus court.

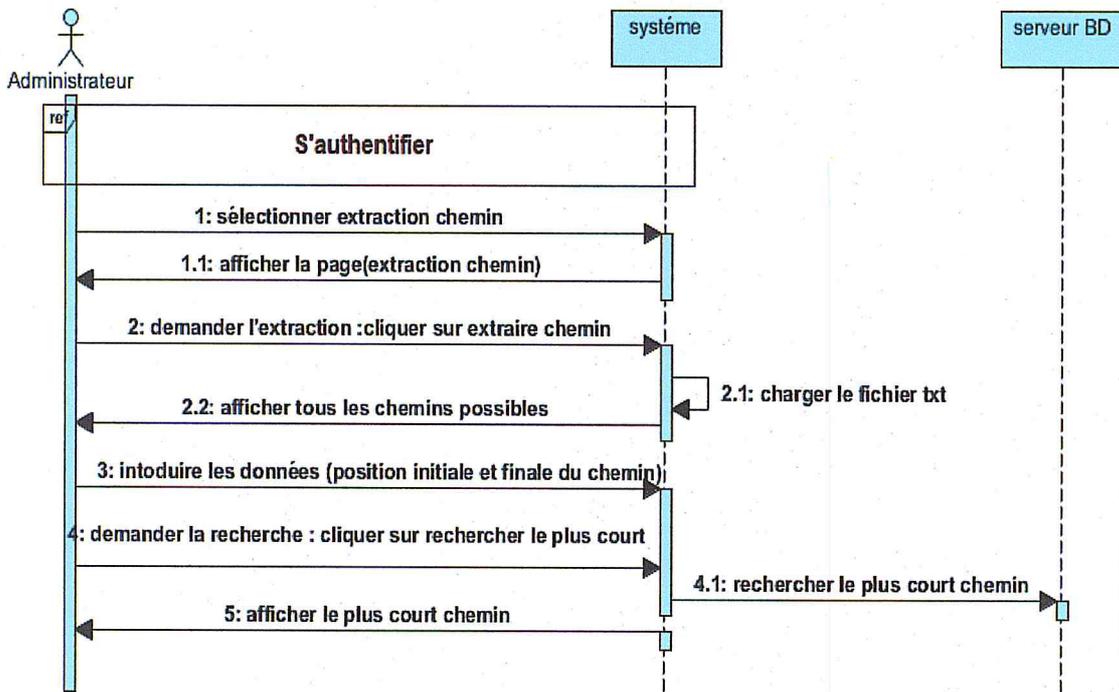


Figure 2.6 : Diagramme de séquence pour l'extraction d'un chemin.

3.2.3.4. Diagramme de séquence pour ajouter un tag

Le diagramme de la figure 2.7 modélise la chronologie des messages échangés entre l'administrateur et le système pour ajouter un nouveau tag à la base de données.

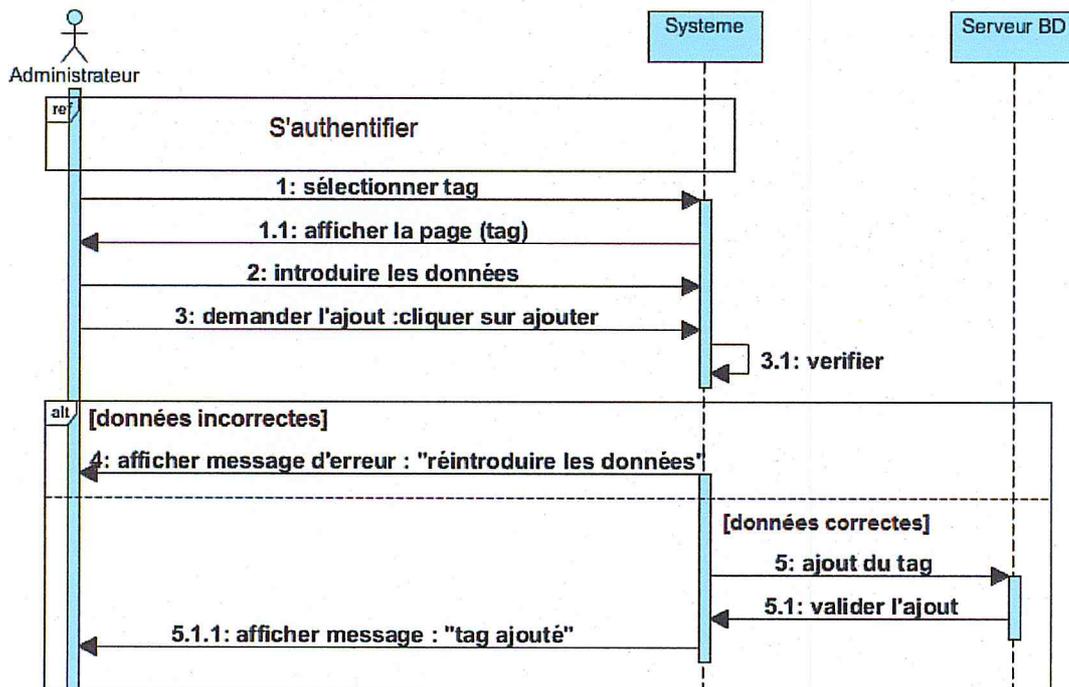


Figure 2.7 : Diagramme de séquence pour ajouter un tag.

3.2.3.5. Diagramme de séquence pour modifier un tag

Le diagramme suivant (figure 2.8) représente l'interaction entre l'acteur administrateur, le système et la base de données pour ajouter un nouveau tag.

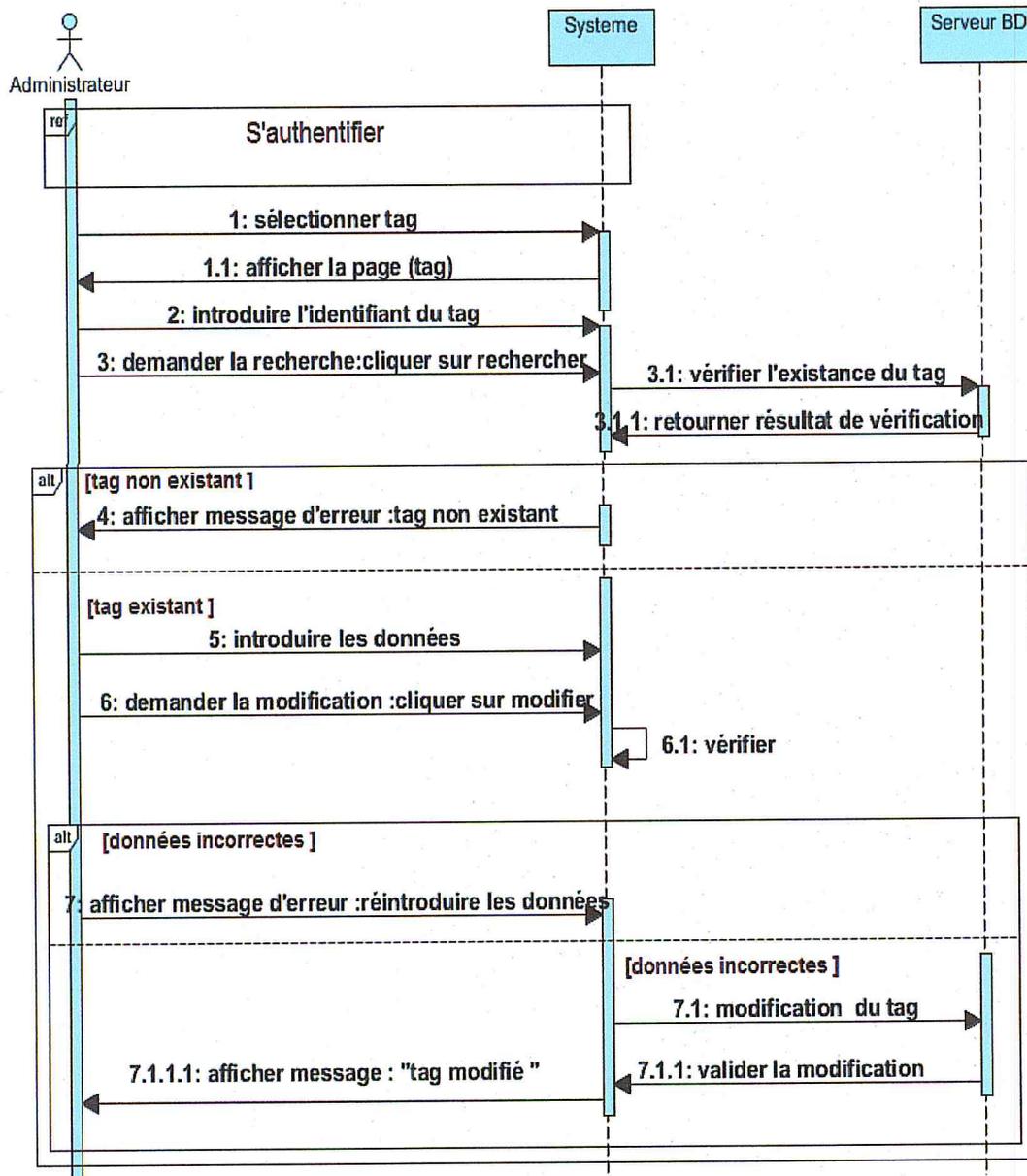


Figure 2.8 : Diagramme de séquence pour modifier un tag.

3.2.3.6. Diagramme de séquence pour supprimer un tag

Le diagramme suivant (figure 2.9) montre comment l'administrateur interagit avec le système pour supprimer un tag existant dans la base de données en traitant les différents cas alternatifs possibles.

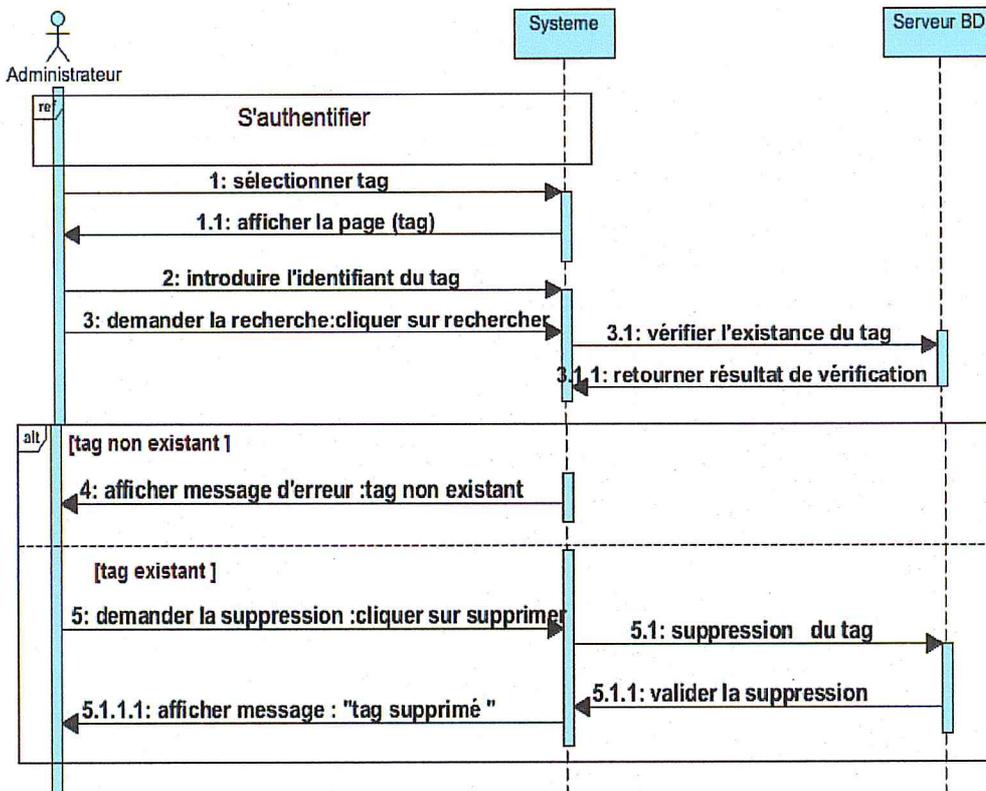


Figure 2.9 : Diagramme de séquence pour supprimer un tag.

3.2.3.7. Diagramme de séquence pour ajouter un robot

Le diagramme de la figure 2.10 modélise la chronologie des messages échangés entre l'administrateur et le système et la base de données pour ajouter un nouveau robot.

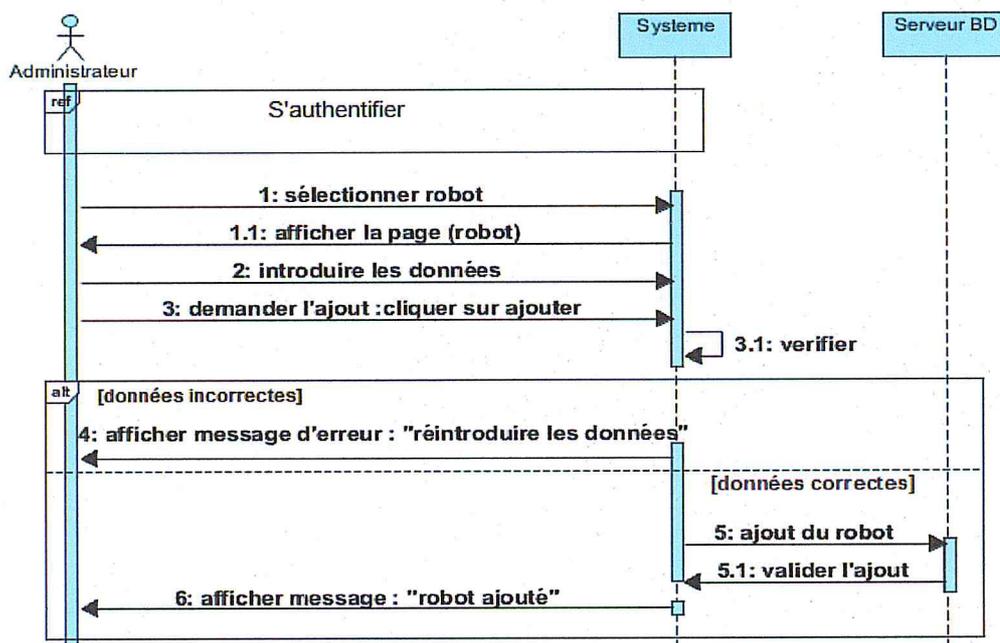


Figure 2.10 : Diagramme de séquence pour ajouter un robot.

3.2.3.8. Diagramme de séquence pour modifier un robot

Le diagramme suivant (figure 2.11) montre comment l'administrateur interagit avec le système pour modifier un robot existant dans la base de données en traitant les différents cas alternatifs possibles.

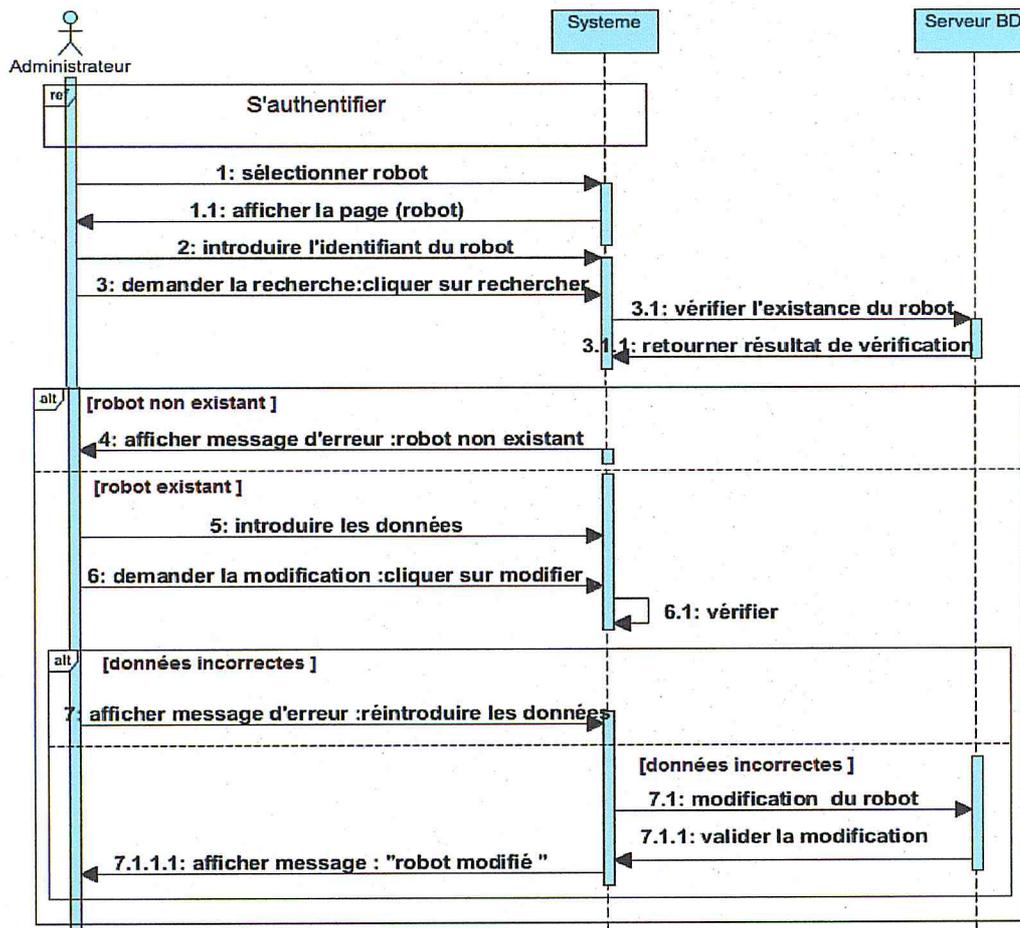


Figure 2.11 : Diagramme de séquence pour modifier un robot.

3.2.3.9. Diagramme de séquence pour supprimer un robot

Le diagramme suivant (figure 2.12) représente l'interaction entre l'acteur administrateur, le système et la base de données pour supprimer un robot existant.

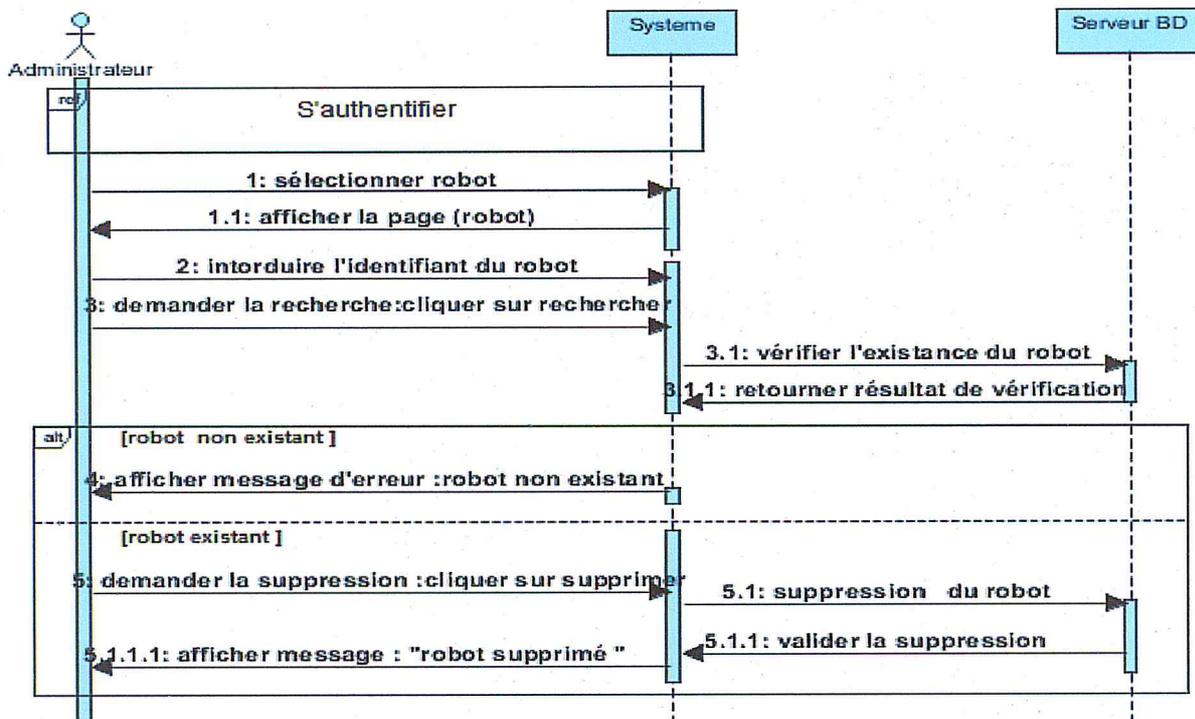


Figure 2.12 : Diagramme de séquence pour supprimer un robot.

3.2.3.10. Diagramme de séquence pour ajouter un lecteur

Le diagramme de la figure 2.13 modélise la chronologie des messages échangés entre l'administrateur et le système pour ajouter un nouveau lecteur à la base de données.

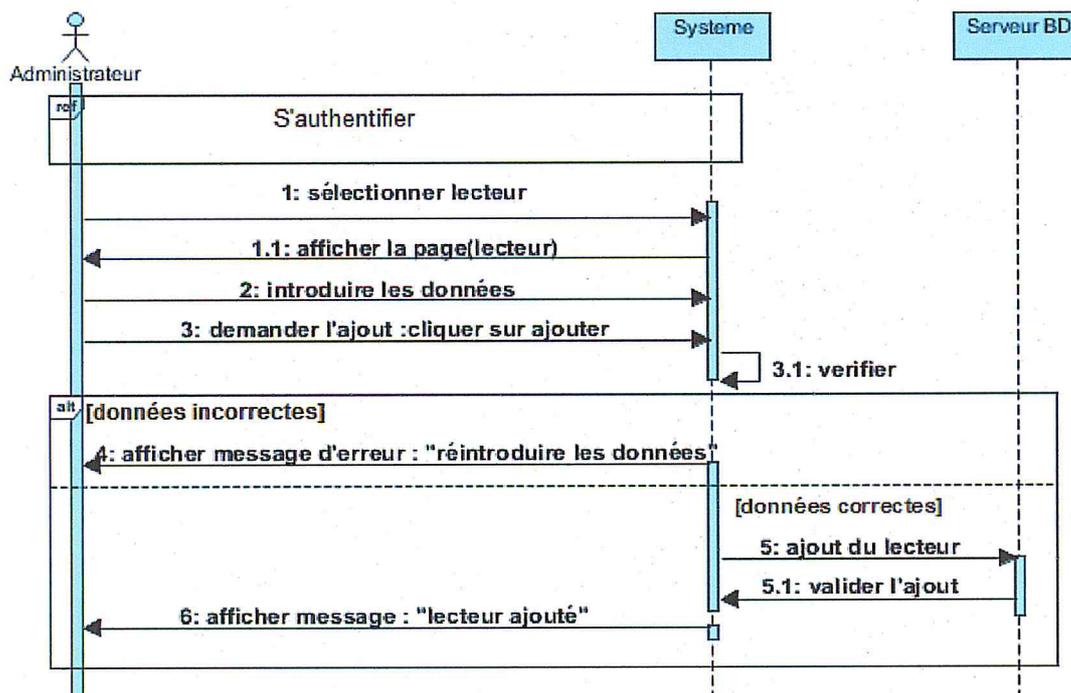


Figure 2.13 : Diagramme de séquence pour ajouter un lecteur.

3.2.3.11. Diagramme de séquence pour modifier un lecteur

Le diagramme de la figure 2.14 modélise la chronologie des messages échangés entre l'administrateur et le système et la base de données pour modifier un lecteur.

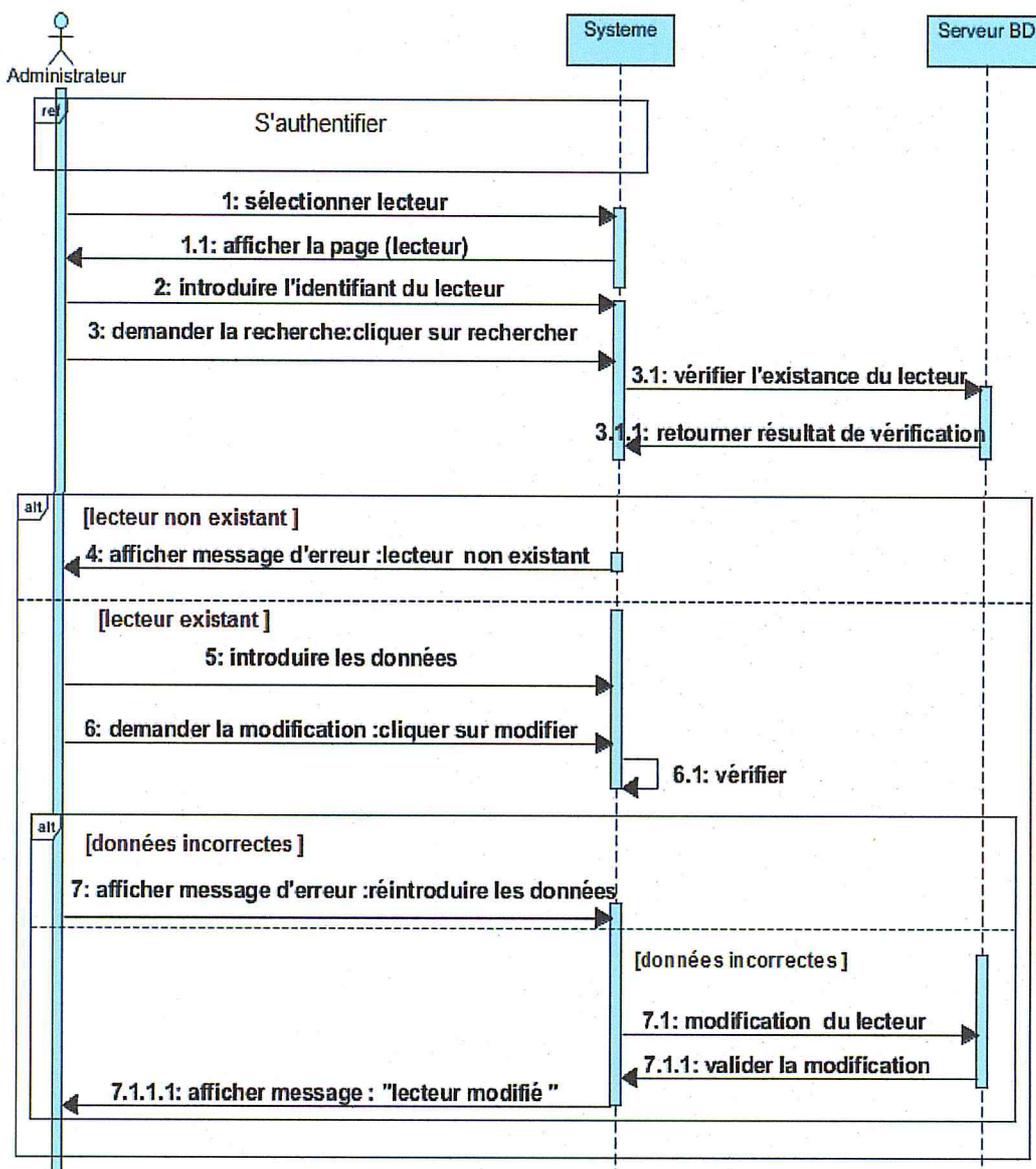


Figure 2.14 : Diagramme de séquence pour modifier un lecteur.

3.2.3.12. Diagramme de séquence pour supprimer un lecteur

Le diagramme suivant (figure 2.15) représente l'interaction entre l'acteur administrateur, le système et la base de données pour supprimer un lecteur existant.

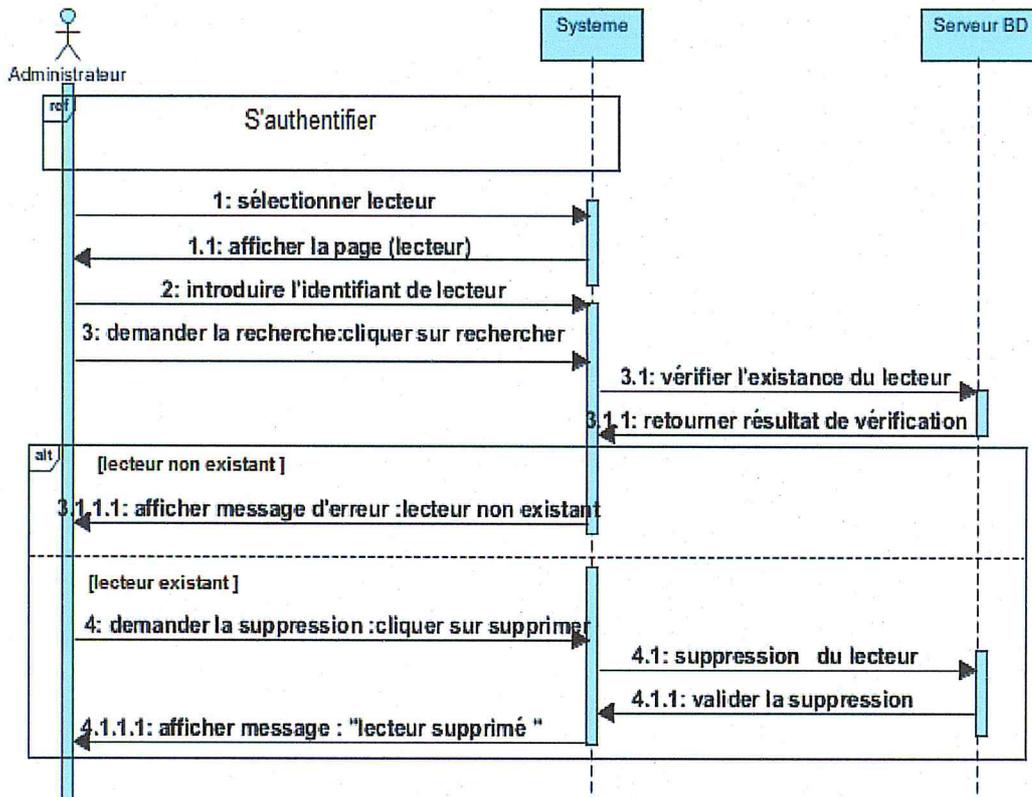


Figure 2.15 : Diagramme de séquence pour supprimer un lecteur.

3.2.4. Diagrammes de classe

Le diagramme de classe est considéré comme l'élément le plus important dans la modélisation orienté objet, donc ce diagramme est considéré comme le point central en UML.

Contrairement aux diagrammes des cas d'utilisations qui décrivent le système d'un point de vue externe, le diagramme de classe décrit la structure interne du système et de ces composants. Il décrit aussi les tâches définissant le comportement de ces derniers [44] [45].

Le diagramme ci-dessous (figure 2.16) décrit l'ensemble des classes de notre système et les différentes relations entre elles ainsi que les attributs et les opérations propres à chaque classe.

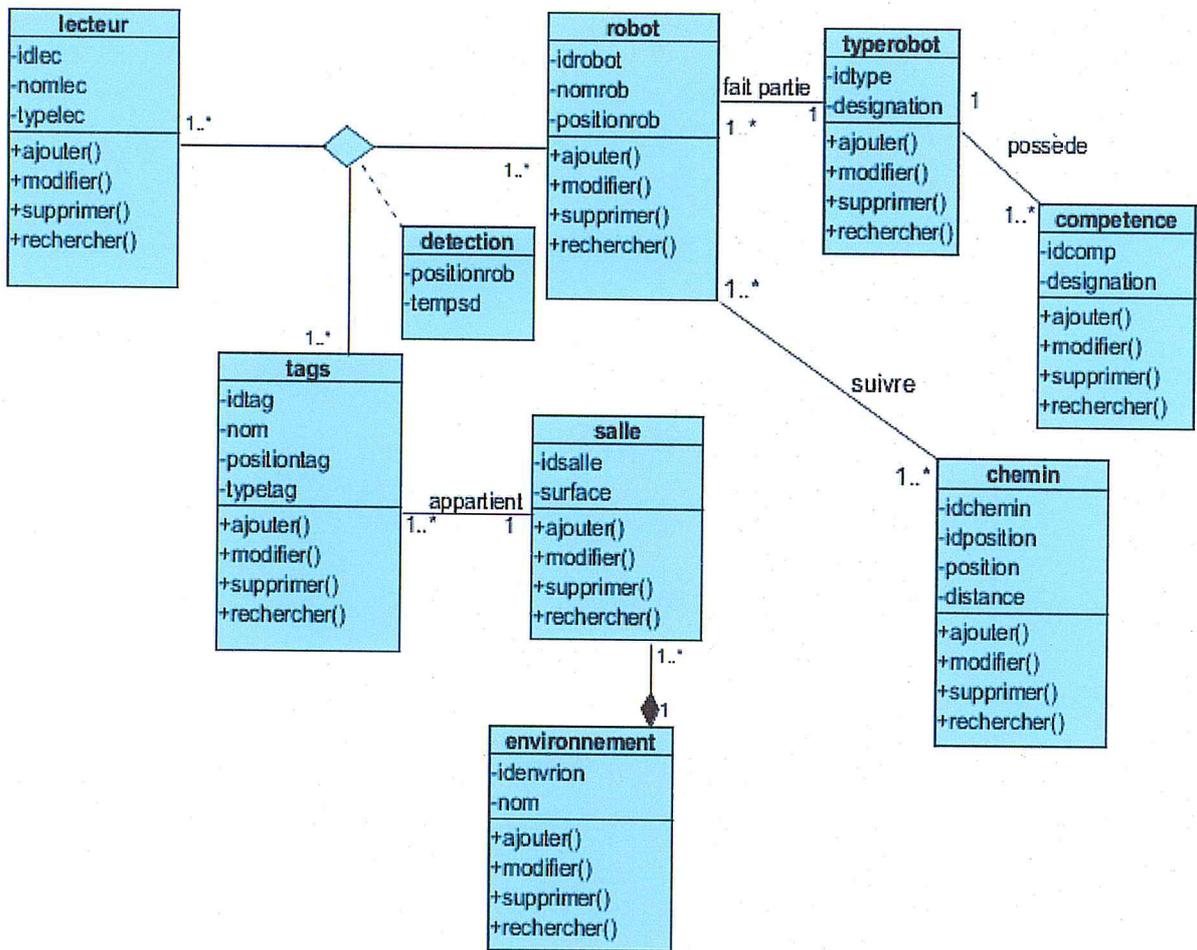


Figure 2.16 : Diagramme de classe.

3.2.4.1. Description des classes

Le tableau suivant (tableau 2.2) décrit les différents attributs avec leurs types, ainsi que les méthodes de chaque classe de diagramme.

Classe	Attributs	Désignation	Type	Méthodes
robot	idrobot nomrob positionrob	Identifiant du robot. Nom du robot. Position initial du robot(x, y, θ).	Varchar Varchar Varchar	Ajouter () Modifier () Supprimer () Rechercher ()
typerobot	idtype designation	Identifiant du type de robot. Désignation du type robot (manipulateur, mobile, manipulateur mobile, etc.).	Varchar Varchar	Ajouter () Modifier () Supprimer () Rechercher ()

tags	idtag	Identifiant du tag de l'objet.	Varchar	Ajouter ()
	nom	Nom de l'objet.	Varchar	Modifier ()
	positiontag	Position initial de l'objet (x, y, z).	Varchar	Supprimer () Rechercher ()
	typetag	Type de tag (opérateur humain, objet actif, objet passif, etc.).	Varchar	
lecteur	idlec	Identifiant du lecteur.	Varchar	Ajouter ()
	nomlec	Nom du lecteur.	Varchar	Modifier ()
	typelec	Type du lecteur.	Varchar	Supprimer () Rechercher ()
competence	idcomp	Identifiant de la compétence du robot.	Varchar	Ajouter () Modifier ()
	designation	Désignation de la compétence.	Varchar	Supprimer () Rechercher ()
salle	idsalle	Identifiant de la salle.	Varchar	Ajouter ()
	surface	Surface de la salle.	Varchar	Modifier () Supprimer () Rechercher ()
environnement	idenviron	Identifiant de l'environnement.	Varchar	Ajouter ()
	nom	Nom de l'environnement (étage, division, etc.)	Varchar	Modifier () Supprimer () Rechercher ()
chemin	idchemin	Identifiant de chemin (phéromone).	Varchar	Ajouter () Modifier ()
	idposition	Identifiant de la position.	Varchar	Supprimer ()
	position	Position de robot.	Varchar	Rechercher ()
	distance	Distance de chemin	Varchar	
detection (classe associative)	position rob	Position du robot lors de la détection.	Varchar	
	tempsd	Temps de détection	Varchar	

Tableau 2.2 : Description des classes.

3.2.4.2. Passage du modèle objet vers le modèle relationnel

Le passage du modèle objet au modèle relationnel doit respecter un ensemble de règles qui sont données comme suit [46] :

- **Règle 1** : Une classe devient une table; ses attributs deviennent ceux de la table et son identifiant est la clé primaire de cette dernière.
- **Règle 2** : Pour chaque association « plusieurs à plusieurs », on crée une nouvelle table ; sa clé primaire est la concaténation des clés des relations traduisant les classes intervenantes dans l'association.
- **Règle 3** : Une association « un à plusieurs » est traduite en incluant dans la relation de multiplicité un, la clé primaire de l'autre comme clé étrangère.
- **Règle 4** : Une association « un à un » est traduite en incluant la clé primaire de l'une des relations comme clé étrangère dans l'autre.
- **Règle 5** : La traduction d'une « association N-aire » est une relation ayant comme clé primaire, un sous ensemble des clés des relations traduisant les classes qui interviennent dans l'association.
- **Règle 6** : L'héritage se traduit par une table pour chacune des sous classes, en reportant les attributs de la super classe dans les tables des sous-classes.

Le tableau suivant résume les équivalences entre les deux structures :

Modèle Objet	Modèle Relationnel
Classe	Table
Attribut	Attribut
Identifiant	Clé primaire
Association	Clé étrangère
Héritage	Clé primaire identique sur plusieurs tables

Tableau 2.3 : Passage *du modèle objet vers le modèle relationnel*.

3.2.4.3. Traduction du modèle objet en base de données relationnelles

En appliquant les règles de passage du modèle d'objet au model relationnel sur notre diagramme de classe, nous obtenons les résultats suivants :

- robot(idrob, nomrob, positionrob, idtype*).

- typerobot(idtype, designation).
- tags(idtag, nom, , positiontag, typetag, idsalle*)
- lecteur(idlec, nomlec, typelec).
- detection(idrob, idtag, idlec, positionrob, tempsd).
- competence(idcomp, designation, idtype*).
- salle(idsalle, surface, idenviron*).
- environnement(idenviron, nom).
- chemin(idchemin, idposition).
- chemin-robot(idchemin, idposition, idrobot, position, distance)

4. Codification du système

Les différents tags attachés aux objets situés dans notre environnement sont codés comme suit :

4.1. Codification des robots

- **Robots mobiles**



→ Codification Normalisée.

Exemple : 10001 robot mobile n°1.

- **Robot manipulateur**



→ Codification Normalisée.

Exemple : 20001 robot manipulateur n°1.

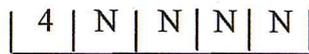
- **Robots manipulateurs mobiles**



→ Codification Normalisée.

Exemple : 30001 robot manipulateur mobile n°1.

4.2. Codification des opérateurs humains



Codification Normalisée.

Exemple : 40001 opérateur n°1.

4.3. Codification des objets

- **Objets actifs :**



Codification Normalisée.

Exemple : 50001 table n°1.

- **Objets passifs**



Codification Normalisée.

Exemple : 60001 mur n°1.

5. Conclusion

Dans ce chapitre, nous avons présenté notre solution qui inclue deux parties principales :

- Dans la première partie, nous avons traité l'identification et la détection des différents objets situés dans notre environnement et le stockage des données obtenus dans une base de données centralisée ainsi que l'extraction des différents chemins des robots.
- La seconde partie consiste en la mise à jour ainsi que la gestion de cette base de données.

Nous avons réalisé notre solution suivant l'étude conceptuelle que nous avons abordée tout au long de ce chapitre. Cette étude nous a permis aussi de mettre en évidence les différentes classes du système.

Dans le chapitre suivant, nous allons implémenter et mettre en œuvre la solution que nous avons proposée dans l'étude conceptuelle détaillée dans ce chapitre.

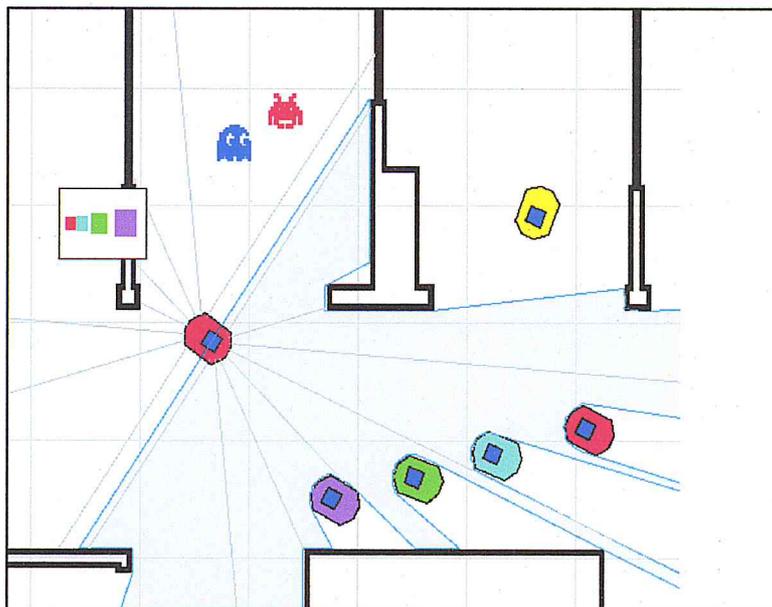
1. Introduction

Après avoir présenté dans le chapitre précédent notre solution proposée et sa modélisation UML, l'objectif de ce chapitre est de présenter l'implémentation de notre système.

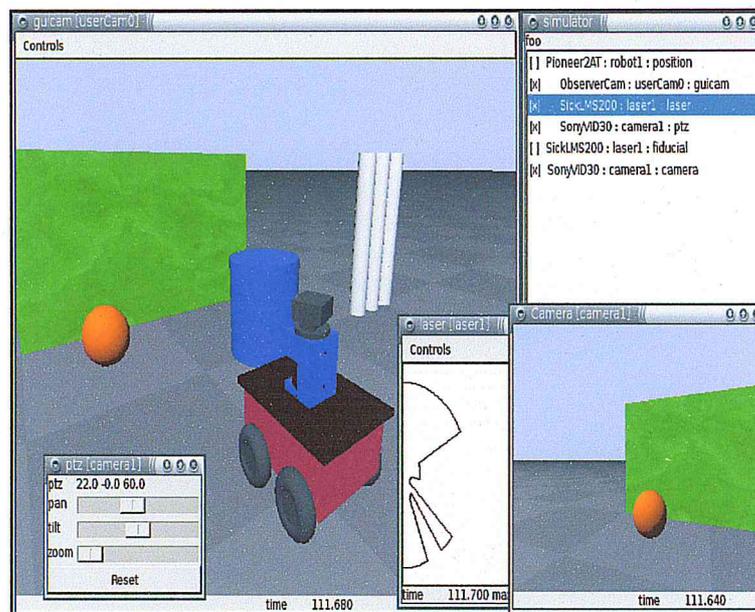
Dans ce troisième chapitre, nous allons simuler la partie de détection et d'identification des objets de l'environnement utilisant le simulateur Player/Stage. Ainsi, nous allons mettre en œuvre la proposition pour en extraire les chemins des robots. Nous allons aussi décrire les différentes interfaces réalisées de notre application pour gérer notre système. Nous allons aussi définir les outils de développement utilisés pour l'implémentation de chaque phase.

2. Simulateur Player/Stage

Player/Stage/Gazebo est un projet qui permet la simulation de robots sur un ordinateur avec le trio Player, Stage et Gazebo. Ce simulateur a été créé par les chercheurs de l'Université South California [47]. Il permet de simuler un très grand nombre de matériels y compris les caméras, laser, gripper, fiducial, blobfinder, etc. de manière que chacun de ces capteurs possède sa propre interface. Ainsi, des robots peuvent être aussi simulés. On dispose de deux interfaces de visualisation : 2D et 3D (voir figure 3.1).



(a) Player/Stage



(b)Gazebo

Figure 3.1 : (a) Le simulateur en 2D, (b) Le simulateur en 3D [47].

Leur interface donne, à la fois, des outils pour simuler, ainsi que pour communiquer avec un système robotique réel [47].

Remarque

Dans notre projet, nous avons uniquement utilisé Player/Stage, version Player-3.0.2et Stage-3.2.2.

2.1. Player

Player est en réalité une interface réseau de type client/serveur qui permet de réaliser la simulation de robots dans leur environnement. Il donne une possibilité d'appliquer un algorithme d'intelligence avec les trois langages proposés (Java, C et C++). Il permet de recueillir et de traiter les informations portées par les différents capteurs que possède chacun des robots de la simulation. Il utilise un protocole TCP/IP pour la connexion entre le client et le serveur. Il permet aussi de contrôler les robots via les ports (attribution d'un port supérieur à 1024 pour chaque robot). Le port par défaut étant le port 6665, ce qui, en terme pratique, signifie que si la simulation ne comporte qu'un seul robot, il n'est pas nécessaire de lui assigner un port, le 6665 lui sera attribué par défaut. Le modèle client/serveur utilisé permet en outre de partager les données de la simulation en cours avec n'importe quel ordinateur mis

en réseau avec celui qui est le siège de la simulation, ce qui, pour une équipe de recherche, peut se révéler particulièrement efficace et utile[47].

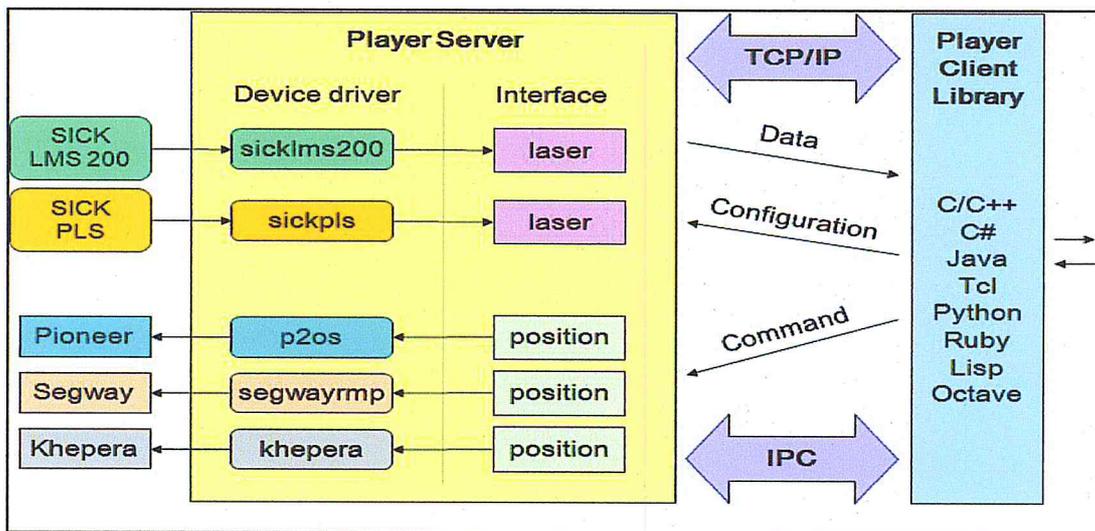


Figure 3.2 : Modèle Client/serveur du Player [47].

Player fonctionne sous le modèle client/serveur qui présente beaucoup d'avantages [47] :

- **Distribution** : Le modèle permet à un client d'accéder à des capteurs et à des actionneurs n'importe où dans le réseau. Un client peut accéder à plusieurs serveurs et un serveur peut gérer plusieurs clients. Cela permet à un programme de contrôler les différents aspects d'un robot.
- **Indépendance** : Les programmes clients peuvent être écrits dans n'importe quel langage et sur n'importe quelle plateforme matérielle proposant une programmation des sockets.
- **Commodité** : Le serveur fournit un abstract interface à des périphériques (device). Le programme client s'abonne à un ensemble de périphériques et à une fréquence par périphérique. La fréquence permet au client de retirer les données d'un périphérique donné à un instant donné.

2.2. Stage

Stage est comme un driver de Player. Il fournit une interface graphique à la simulation en 2D, modélisant ainsi les déplacements des différents robots. Selon les programmeurs, on peut utiliser exactement le même mécanisme pour communiquer entre Player et les robots réels. La différence est uniquement dans le fichier de configuration. Avec Stage, pour simuler

les robots, on utilise le driver Stage avec plugin `_libstageplugin_`. Quand on travaille avec Stage, on s'intéresse au fichier de configuration qui décrit le modèle des robots et l'environnement du simulateur [47].

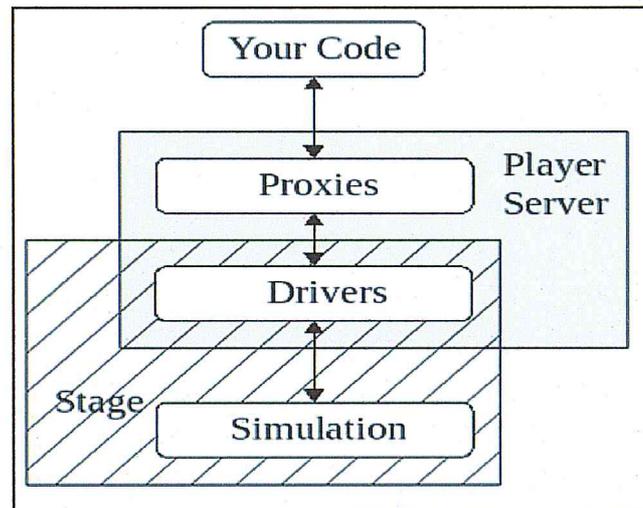


Figure 3.3 : Stage et serveur Player [48].

3. Déploiement d'une infrastructure RFID

Dans Player/Stage il existe trois types de fichiers importants qui sont nécessaires pour réaliser un environnement simulé [48] :

- Le fichier `a.world`.
- Le fichier de configuration `a.cfg`.
- Le fichier include `a.inc`.

3.1. Fichier.world

Il définit la liste des modèles nécessaires pour la simulation. Cela comprend la configuration de l'environnement de base (taille, repère du plan composé d'un axe des X et axe des Y, etc.), des robots et d'autres objets.

Les figures ci-dessous (figure 3.4 et figure 3.5) décrivent la configuration de notre environnement de base ainsi que le modèle de robot que nous avons créé.

```

# load an environment bitmap
floorplan
(
  name "Rez de chaussée"
  size [32.000 22.000 0.800]
  pose [16.000 11.000 0 0]
  bitmap "bitmaps/cdta.png"
)

```

Nom de l'environnement simulé
 Taille de l'environnement simulé
 Position de l'environnement simulé
 Image de l'environnement simulé

Figure 3.4 : Définition de l'environnement dans *simple1.world*.

```

pioneer2dx
(
  # can refer to the robot by this name
  name "r0"
  pose [ 4.476 2.412 0 84.628 ]
)

sicklaser(
  # ctrl "lasernoise" # uncomment this line to run a laser noise generator
)

ctrl "wander"
)

```

Nom du robot simulé
 Position initiale du robot dans l'environnement simulé

Figure 3.5 : Modèle d'un robot dans *simple1.world*.

3.2. Fichier.inc

Le fichier.inc suit la même syntaxe et format d'un fichier .world, mais il peut être inclus par ce dernier. La création des modèles des objets simulés s'effectue en niveau de ce fichier utilisant des images PNG ainsi qu'un ensemble de propriétés (couleur, taille, etc.) pour les définir. Un exemple de modèle d'un objet simulé (table) que nous avons créé est illustré dans la figure suivante :

```

define table1 model
(
  bitmap "bitmaps/table1.png"
  color "gray"
  size [1.5 1.5 1]
)

```

Nom de modèle
 Image de l'objet simulé (table)
 Couleur de l'objet simulé (table)
 Taille de l'objet simulé (table)

Figure 3.6 : Un modèle d'un objet (table) dans *map.inc*

3.3. Fichier.cfg

Le fichier.cfg définit le driver (stage) et toutes les informations que Player a besoin d'utiliser afin d'interagir avec le robot. Ces informations représentent le nom de(s) robot(s) utilisé(s), les interfaces des capteurs simulés et le nom du fichier.world utilisé. Le fichier détermine au serveur Player comment interpréter les données à partir de driver pour qu'elles puissent être présentées par la suite dans notre propre code. La figure ci-dessous décrit le fichier.cfg que nous avons créé.

```
# load the Stage plugin simulation driver
driver
(
  name "stage"
  provides [ "simulation:0" ]
  plugin "stageplugin"

  # load the named file into the simulator
  worldfile "simple.world"
)

# Create a Stage driver and attach position2d and laser interfaces
# to the model "r0"
driver ( name "stage" provides [ "6665:position2d:0" "6665:laser:0" ]
model "r0" )
```

← Interfaces utilisées et le modèle du robot simulé

Figure 3.7 : *simple.cfg*.

Après la création des trois fichiers cités auparavant, notre environnement simulé est représenté comme suit :

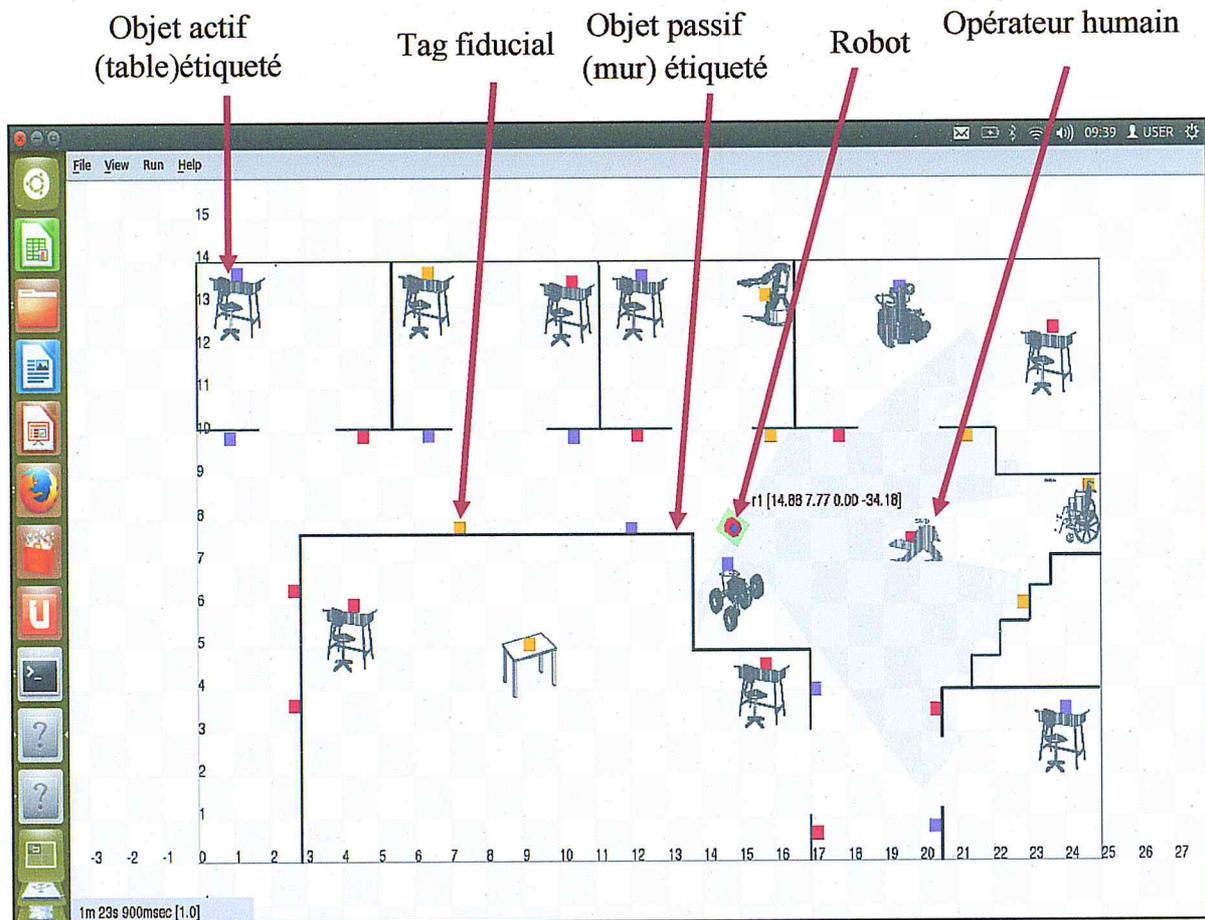


Figure 3.8 : Environnement simulé.

3.4. Fiducial et Player/Stage

Il n'est pas possible d'intégrer un système RFID dans ce simulateur ; ce dernier ne possède pas d'interface ou de modèle de détection et d'identification des différents objets de l'environnement. Pour cette raison, nous avons utilisé l'interface fiducial qui partage le même fonctionnement que le système RFID.

Le fiducial est un système qu'utilise le fiducial Finder (Lecteur) pour détecter les fiducials (tags) situés dans sa portée. Cette portée représente la distance et l'angle entre l'objet et le robot équipé par ce capteur.

Les fiducials (tags) sont généralement connus comme marqueurs, qui peuvent être facilement ajoutés à des objets. Un tag est un point fixe dans une image associé à un objet, dont chaque fiducial possède un identifiant. Le fiducial finder localise ce point fixe pour détecter le tag et récupérer son propre identifiant.

La figure ci-dessous illustre le modèle fiducial finder que nous avons créé :

```

fiducial(
  pose [0.030 0.150 0 0 7]
  range_max 9
  range_max_id 9
  fov 180
  name "fld"
)

```

Position du fiducial finder sur le robot
 Porté maximale de détection du capteur fiducial
 Nombre maximal d'objets détectés par le capteur fiducial
 Angle de détection du capteur
 Nom du capteur fiducial finder

Figure 3.9: Format du modèle Fiducial finder.

L'identifiant fiducial est une clé unique *fiducial_id* qu'on peut ajouter dans les propriétés de chaque modèle d'un objet défini dans le fichier.inc. Par la suite, cette clé est retournée lors de l'implémentation de fiducial finder.

Nous avons complété le fichier.cfg avec l'interface fiducial. En plus, la modification nécessaire que nous avons ajoutée dans les différents modèles est illustrée dans la figure suivante :

```

define table1 model
(
  bitmap "bitmaps/table1.png"
  color "gray"
  size [1.5 1.5 1]
  fiducial_return 50001
)

```

Identifiant fiducial

Figure 3.10 : Format d'un modèle « objet-fiducial ».

3.5. Langage de programmation choisi (C++)

Nous avons implémenté un programme en C++ utilisant le système fiducial pour réaliser la phase de détection avec différents scénarios.

Le C++ est un langage de programmation. Il sert donc à écrire des applications informatiques. Il s'agit d'ailleurs d'un des langages de programmation les plus utilisés aujourd'hui. Chaque programme en C++ doit être écrit en respectant des règles d'écriture très strictes [49].


```

fichier.TXT
The file /home/user/Documents/fichier.TXT changed on disk.
Do you want to reload the file?
Reload
Cancel

localhost: 6665
ID lecteur (fiducialfinder)
7000
7002 fp1 i=0 ID objet
7010 Objet: Identifiant : 50005 position x: 2 y: -7
7011 Robot0: position x0= 20.7478 y0= 11.9683 angle0= -22.5473 ID robot, position courante du robot
7012 Temps de détection: T= 1 Wed Jun 10 09:51:30 2015 Temps de détection
7000
7002 fp1 i=1
7010 Objet: Identifiant : 50008 position x: 0 y: 2
7011 Robot0: position x0= 20.7478 y0= 11.9683 angle0= -22.5473
7012 Temps de détection: T= 2 Wed Jun 10 09:51:30 2015

```

Figure 3.12 : Format fichier.txt.

5. Implémentation de la base de données RFID

Après l'implémentation de la première phase, l'étape suivante consiste à organiser l'ensemble des données résultantes dans une base de données centralisé. Par la suite, il est nécessaire de mettre à jour toutes les informations utiles du système. La gestion de cette base de données se fait au moyen d'un logiciel appelé SGBD.

5.1. Système de Gestion de Base de données (SGBD)

L'accès à une base de données par des utilisateurs ou des applications passe indirectement par un système connu comme étant un système de gestion de base de données (SGBD). Le SGBD et la base de données forment le système de base de données.

Le SGBD est un ensemble de services (applications logicielles) permettant de gérer les bases de données, c'est-à-dire [50] :

- Permettre l'accès aux données de façon simple.
- Autoriser un accès aux informations à de multiples utilisateurs.
- Manipuler les données présentes dans la base de données (insertion, suppression, modification).

Étant donné que notre choix s'est porté sur MySQL, ci-dessous une petite description de ses caractéristiques

5.2. MySQL

Le logiciel MySQL est un serveur de base de données SQL très rapide, multithread, multi-utilisateurs et robuste. Le serveur MySQL est destiné aux missions stratégiques et aux systèmes de production à forte charge, ainsi qu'à l'intégration dans des logiciels déployés à grande échelle. MySQL est développé, distribué et supporté par MySQL AB[51] :

- C'est un multithread, c'est-à-dire qu'il est facilement utilisé sur un serveur avec plusieurs processeurs.
- Application Programme Interface (API) pour C, C++, Java, PHP, Python et TLC.
- Fonctions SQL fortement optimisées.
- Les tables de différentes bases de données peuvent être utilisées dans une requête

5.3. Langage de programmation choisi (java)

Dans cette phase, nous avons opté pour Java comme langage de programmation et NetBeans comme environnement de développement intégré (EDI) pour le développement des différentes interfaces réalisées. Java est un langage de programmation moderne qui a été développé par Sun Microsystems, et qui est évolué en orienté objet.

Le JRE comprend le cœur même de java, celui-ci est la machine virtuelle (JVM, Java Virtual Machine), c'est cette machine virtuelle qui va faire le lien entre le code et l'ordinateur et qui va exécuter les programmes codés en java [52].

5.4. Environnement de développement intégré EDI NetBeans

NetBeans est placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2 (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, XML, Ruby, PHP et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web). [53]

6. Application développée

Nous avons réalisé un logiciel sécurisé qui se compose d'un ensemble d'interfaces utilisateur graphiques afin de gérer toutes les entités du système.

6.1. Interface d'accueil

Lors du lancement du logiciel, l'interface d'accueil donnée par la figure 3.13 apparaît, portant les champs d'authentification.

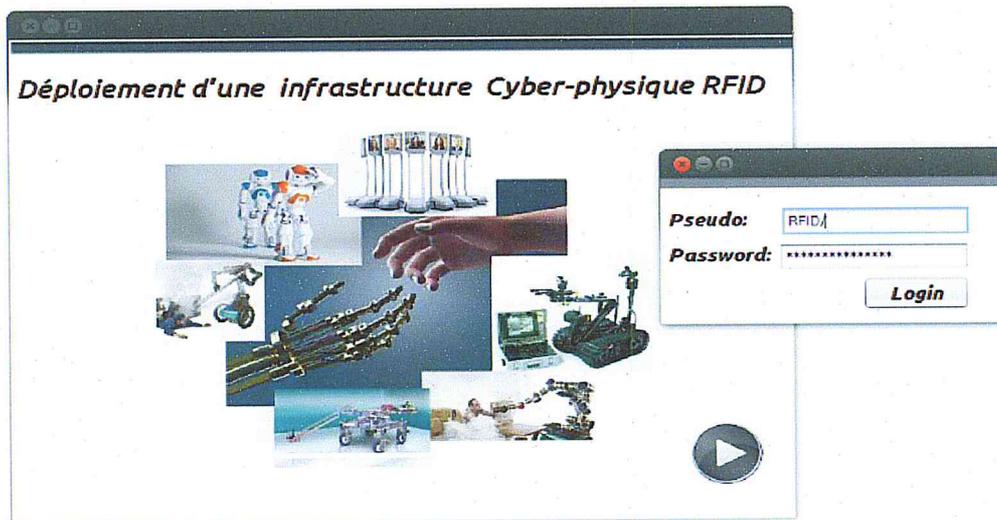


Figure 3.13 : Interface d'accueil.

6.2. Interface menu principal

Après l'authentification de l'utilisateur, l'interface suivante représentée par la figure 3.14) apparaît :



Figure 3.14 : Interface menu principal.

6.3. Interface « Traitement Fichier »

Cette interface donnée par la figure 3.15 permet de visualiser, d'analyser et d'intégrer les données du fichier « .txt ».

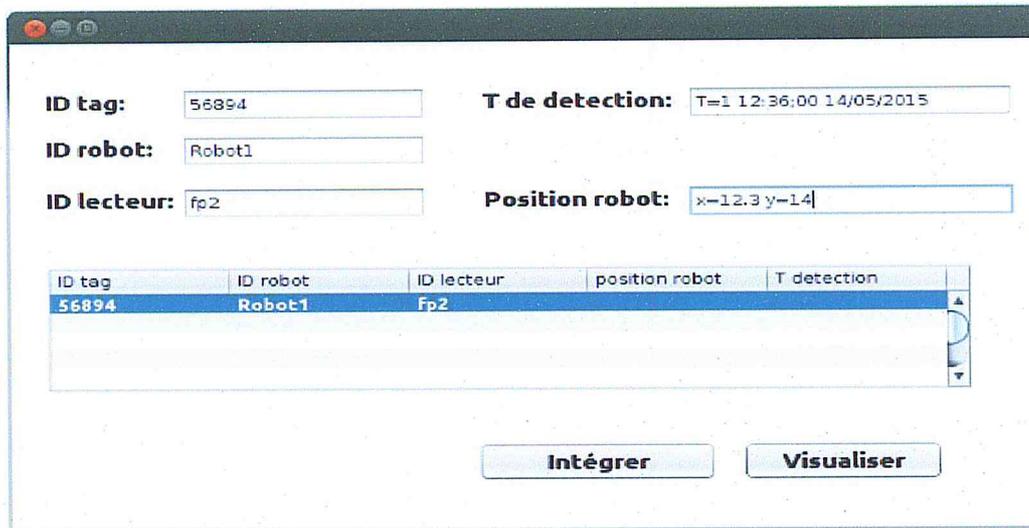


Figure 3.15 : Interface « Traitement Fichier ».

6.4. Interface « Tag »

La figure 3.16 ci-dessous représente un exemple d'un tag. Cette interface permet la gestion des tags. Cette dernière contient un formulaire qui représente les informations nécessaires d'un tag que nous pouvons sauvegarder dans une base de données. Les actions d'ajout, de modification, de suppression et de recherche peuvent être appliquées.

Il faut toutefois savoir qu'avant de cliquer sur les quatre boutons ajouter, modifier, rechercher et supprimer il est impératif que les champs de texte contiennent des informations correctes, dans le cas contraire une erreur survient.

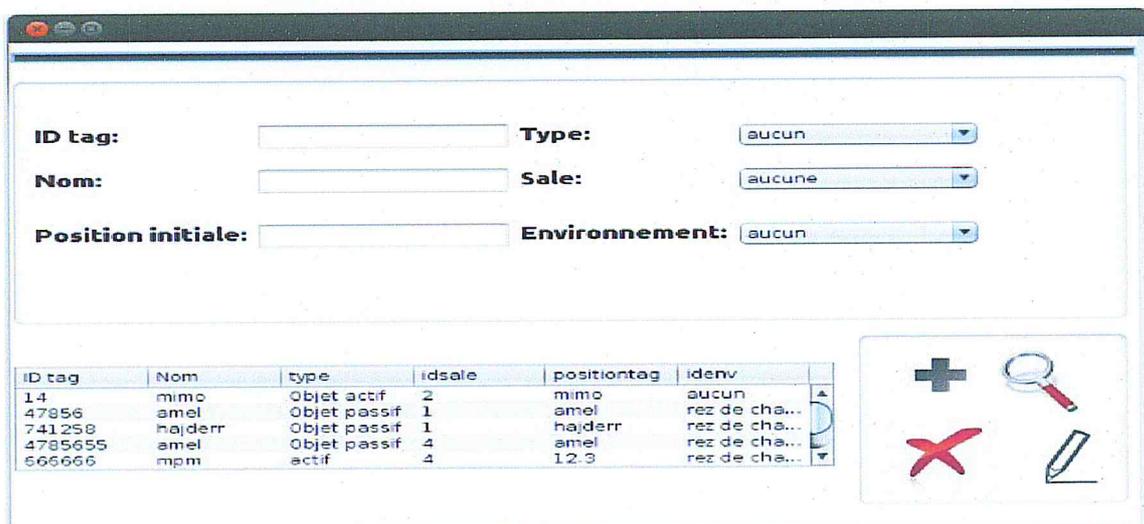
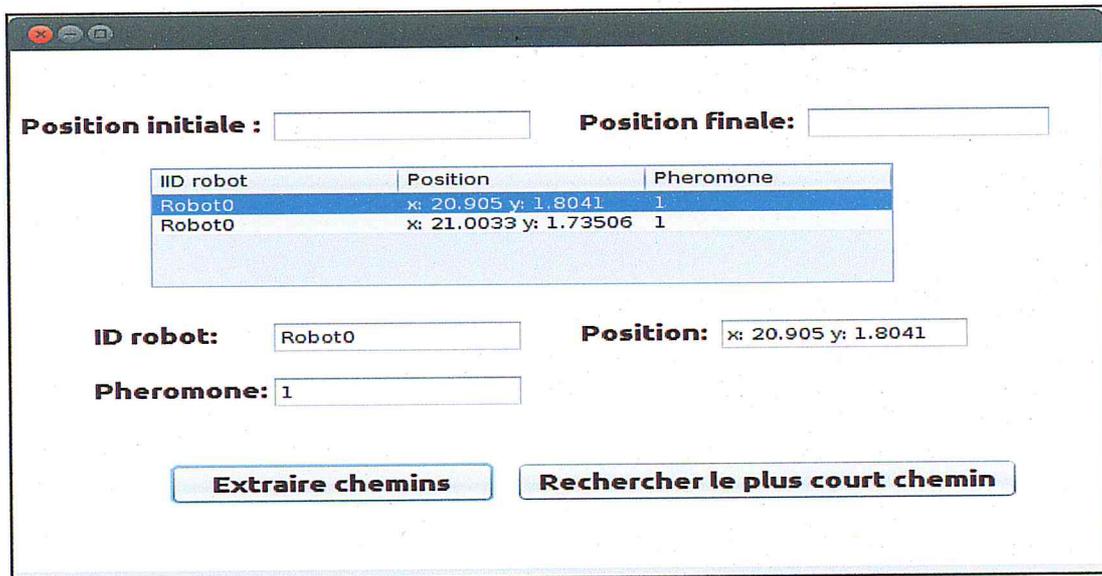


Figure 3.16 : Interface « Tag ».

6.5. Interface « Extraction Chemin »

L'interface de la figure 3.17 permet à l'utilisateur d'extraire les différents chemins existants en insérant la position initiale et finale des chemins. Par la suite, l'ensemble de tous les chemins ainsi que le chemin le plus court seront affichés.



IID robot	Position	Pheromone
Robot0	x: 20.905 y: 1.8041	1
Robot0	x: 21.0033 y: 1.73506	1

Figure 3.17 : Interface «Extraction Chemin ».

7. Conclusion

Dans ce chapitre nous avons présenté notre environnement de simulation et les différentes étapes de déploiement de notre solution. Nous avons aussi présenté brièvement l'environnement de programmation, et décrit les différentes interfaces de toutes les entités du système avec toutes les fonctionnalités qu'elle permette d'accomplir.

La validation de la solution que nous avons proposée et implémentée fera l'objet du quatrième et dernier chapitre.

1. Introduction

L'objectif de ce dernier chapitre consiste en la présentation et la discussion des résultats des différentes simulations que nous avons effectuées pour la validation de notre travail présenté dans les deux chapitres précédents. Toutes les expériences rapportées dans ce document ont été réalisées avec le simulateur Player/Stage.

À cet effet, nous avons organisé la validation en deux parties :

- Dans la première partie de validation, nous allons commencer par la description des résultats de détection des tags déployés dans l'environnement par les différents lecteurs RFID équipant les robots mobiles.
- Par la suite dans la deuxième partie, nous allons passer à la présentation des résultats de validation qui concerne l'extraction de chemins, basée sur les phéromones, menant les robots mobiles d'une position initiale à une position finale.

2. Scénarios de validation de la détection des tags

Lors de l'implémentation de la phase de détection, les différents scénarios traités sont présentés dans ce qui suit :

2.1. Premier scénario

Dans ce scénario, nous avons simulé un seul robot ($r0$) qui navigue dans son environnement d'une manière aléatoire, détecte et identifie au fur et à mesure les objets de l'environnement (figure 4.1).

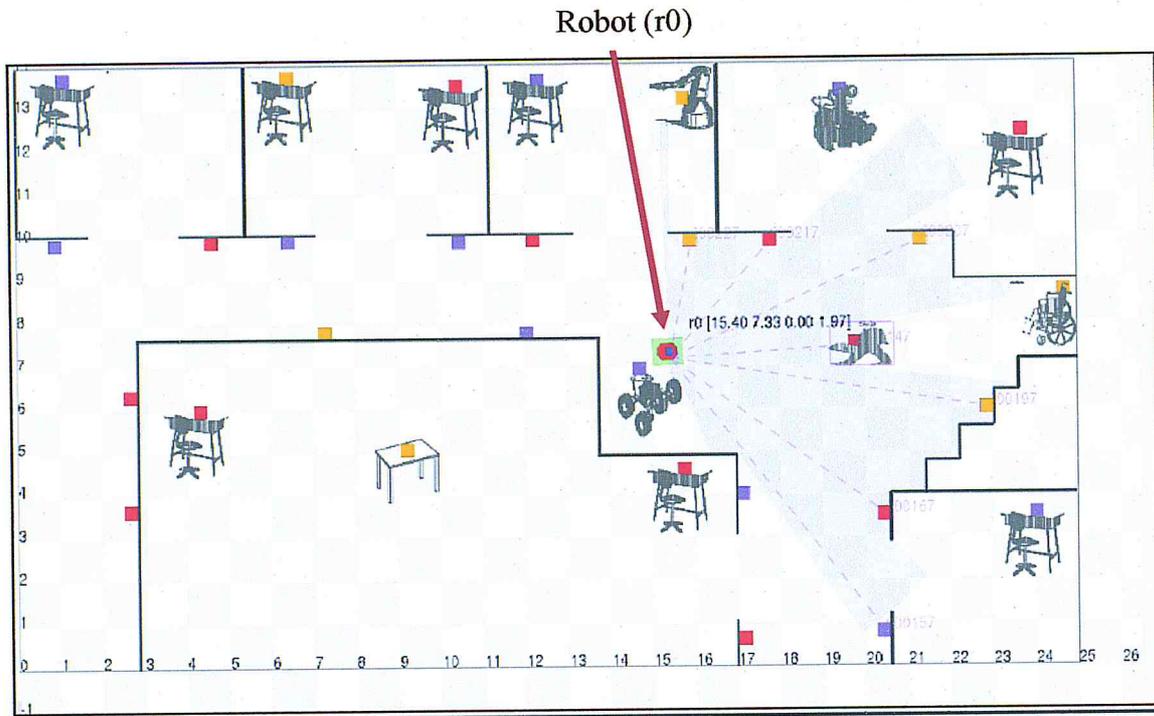


Figure 4.1 : Premier scénario.

2.2. Deuxième scénario

Dans ce scénario, nous avons simulé trois robots (r_0 , r_1 , r_2) qui naviguent dans leur environnement d'une manière aléatoire, détectent et identifient au fur et à mesure les objets éparpillés dans l'environnement ainsi que les autres robots mobiles eux même (figure 4.2).

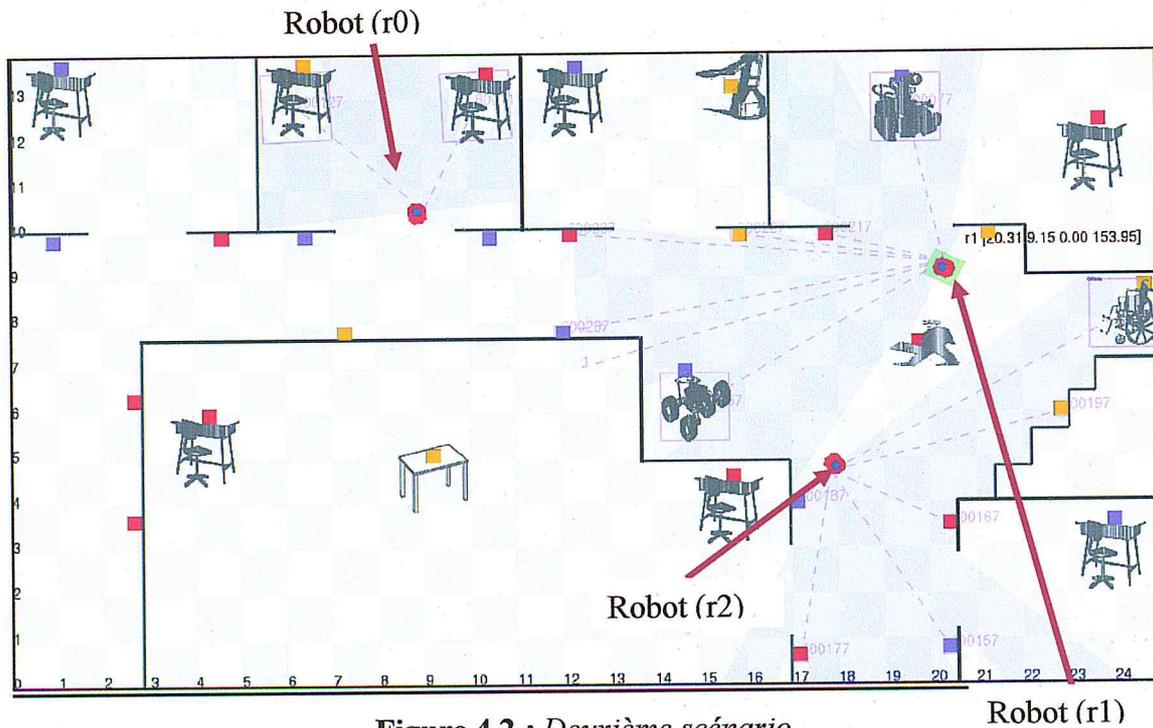


Figure 4.2 : Deuxième scénario.

2.3. Troisième scénario

Dans ce scénario, nous avons simulé plusieurs robots (r_0, \dots, r_{10}) qui naviguent dans l'environnement d'une manière aléatoire, détectent et identifient au fur et à mesure les objets éparpillés dans l'environnement ainsi que les robots eux même (figure 4.3).

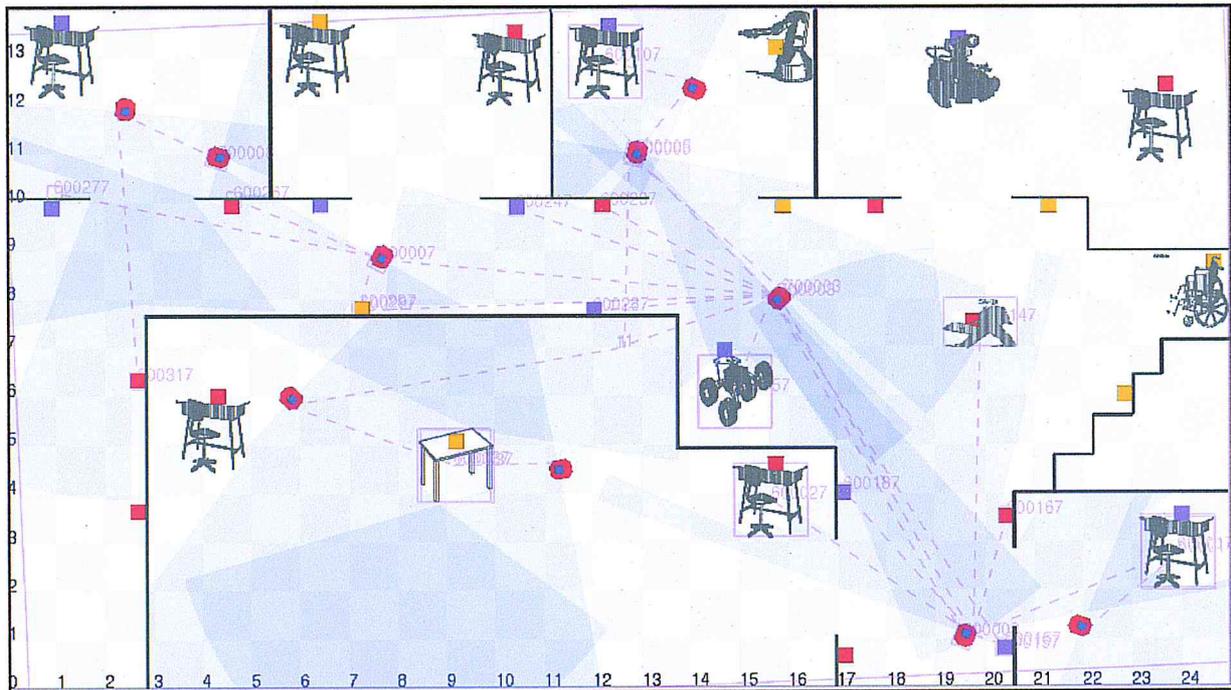


Figure 4.3 :Troisième scénario.

2.4. Discussion des résultats obtenus de détection des tags

Avec le simulateur Player/Stage, nous avons réalisé une série de simulations avec 10 robots, répartis aléatoirement sur un environnement de 32m x 22m avec une portée de détection maximale de 9m. Les robots sont placés initialement au hasard dans leur environnement. Notre infrastructure de simulation offre aux capteurs fiducial placés sur les robots un service leur permettant de détecter et d'identifier tous les objets situés dans leurs voisinages.

Le tableau 4.1 récapitule les résultats obtenus pour le premier scénario de validation. Au cours de ce scénario, qui simule un seul robot mobile, nous avons remarqué que le nombre maximum de tags détectés par seconde était inférieur ou égal (\leq) à trois (03) tags.

Temps de détection / s	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
Nombre de tags détectés	0	3	2	1	1	2	2	3	1	2	2

Tableau 4.1 : Résultats obtenus pour le premier scénario.

Le tableau 4.2 résume les résultats obtenus pour le deuxième scénario de validation considéré. Au cours de ce scénario, qui simule, trois robots mobiles, nous avons remarqué que le nombre maximum de tags détectés par seconde était inférieur ou égal (\leq) à quatorze (14) tags. Ce nombre représente la somme des tags détectés par les trois robots mobiles à chaque instant.

Temps de détection / s	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
Nombre de tags détectés	0	8	9	7	7	9	10	14	13	14	11

Tableau 4.2 : Résultat obtenu pour le deuxième scénario.

Le tableau 4.3 résume les résultats obtenus pour le dernier scénario de validation considéré. Au cours de ce scénario, qui simule plusieurs robots mobiles (r_0, \dots, r_{10}), nous avons remarqué que le nombre maximum de tags détectés par seconde inférieur ou égal (\leq) à vingt-et-quatre (24) tags. Ce nombre représente la somme des tags détectés par tous les robots mobiles à chaque instant.

Temps de détection / s	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
Nombre de tags détectés	0	18	18	20	17	15	15	24	19	20	21

Tableau 4.3 : Résultat obtenu pour le dernier scénario.

2.5. Comparaison entre les résultats obtenus

La figure ci-dessous consiste en une comparaison entre les résultats des différents tests effectués pour les trois scénarios.

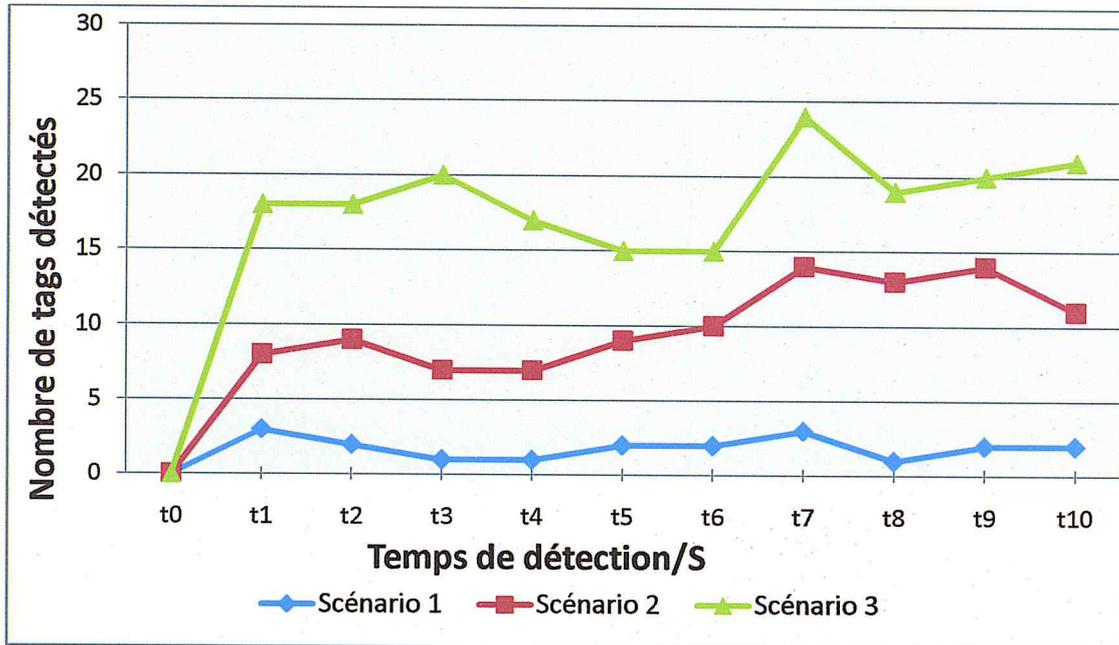


Figure 4.5 : Comparaison entre les scénarios.

Nous avons constaté que le nombre de tags détectés dans le premier scénario était très faible par rapport aux deux autres scénarios. Donc, la simulation d'un seul robot mobile ne nous a pas permis de collecter un grand nombre de tags. Par contre, nous avons constaté que l'accès en écriture au fichier « .txt » était facile et ne nécessite aucun traitement. De ce fait, les résultats de ce scénario sont insuffisants et ça prend beaucoup de temps pour avoir un grand nombre d'informations sur l'environnement.

En revanche avec les deux autres scénarios, le nombre de tags détectés était plus élevé. Cependant, l'accès en écriture au fichier « .txt » par ce système multi-robots s'avérait trop long et nécessite des considérations et des traitements de droits d'accès pour chaque robot. Donc, nous avons pu aboutir à des résultats meilleurs.

Comme il y avait des problèmes de collisions entre les différents robots mobiles considérés dans le dernier scénario, nous avons constaté que la vitesse des robots mobiles a diminué considérablement par rapport aux deux autres scénarios.

Enfin, nous pouvons conclure que le deuxième scénario était le meilleur pour notre déploiement.

3. Scénarios de validation de l'extraction de chemins

Nous avons testé cette partie sur le deuxième scénario (qui a donné de bons résultats de détection de tags RFID) pour valider l'extraction de chemins empruntés par les robots mobiles. Les résultats obtenus sont décrits comme suit :

- Chaque robot dispose d'un ensemble de chemins avec des phéromones.
- Les phéromones se diffèrent d'un chemin à un autre.
- Les mêmes positions initiales et finales peuvent apparaître dans plusieurs chemins.
- La possibilité d'en extraire le chemin optimal parmi tous les chemins existants.

3.1. Premier scénario

La figure 4.6 montre les différentes phéromones déposées par les différents robots mobiles évoluant dans l'environnement. Elle montre aussi les différents chemins faisables amenant les robots de la position initiale à la position finale.

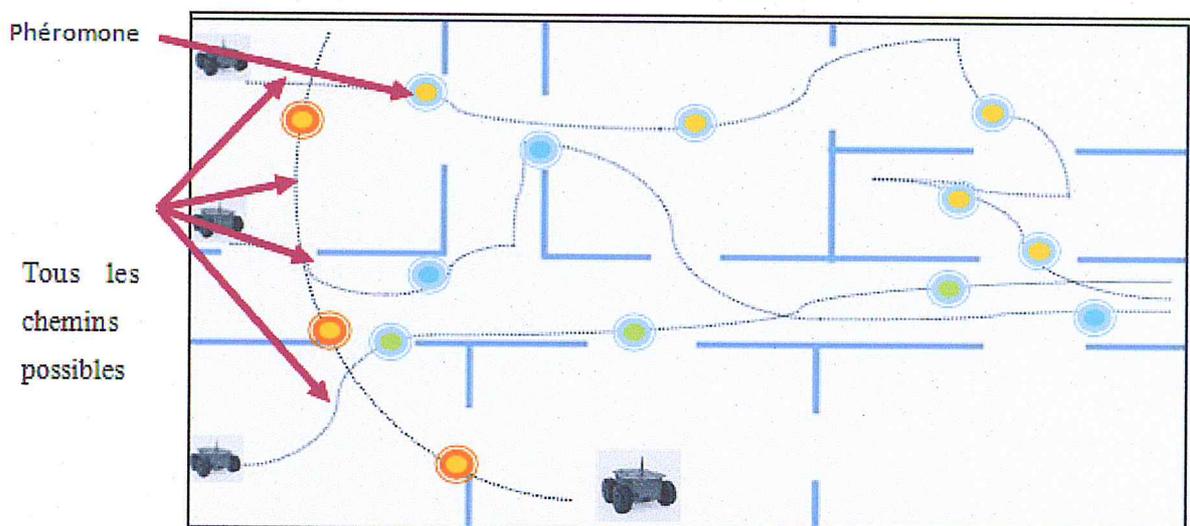


Figure 4.6 : Extraction de l'ensemble de chemins.

3.2. Deuxième scénario

La figure 4.7 montre le chemin unique existant amenant les robots d'une position initiale à une position finale. Ce chemin donc est considéré comme le plus court.

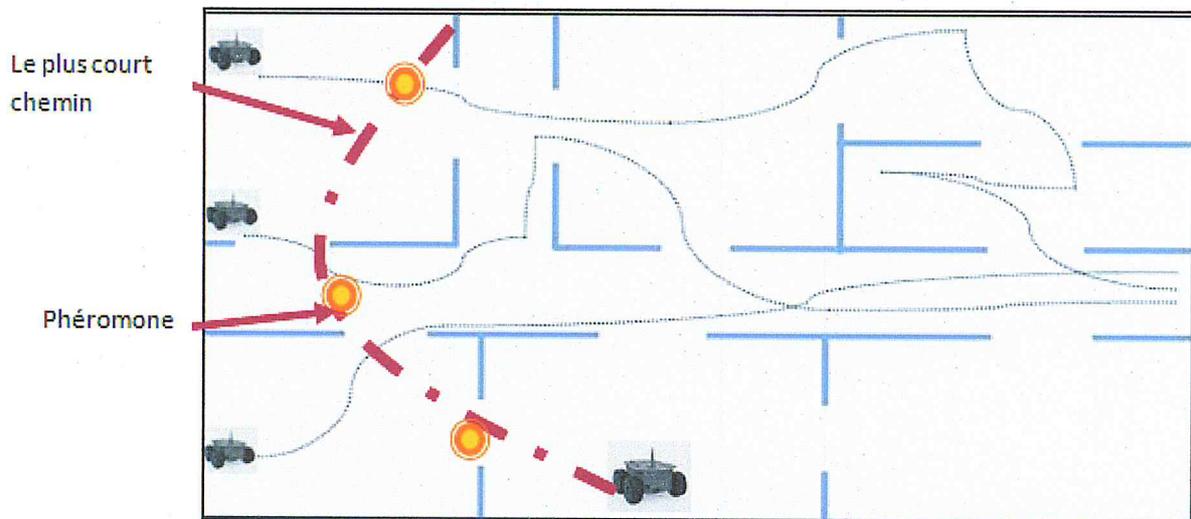


Figure 4.7 : *Extraction chemin.*

3.3. Troisième scénario

La figure 4.8 montre l'ensemble des chemins amenant les robots d'un point initial à un point final. Elle montre aussi le chemin optimal (le plus court en termes de distance).

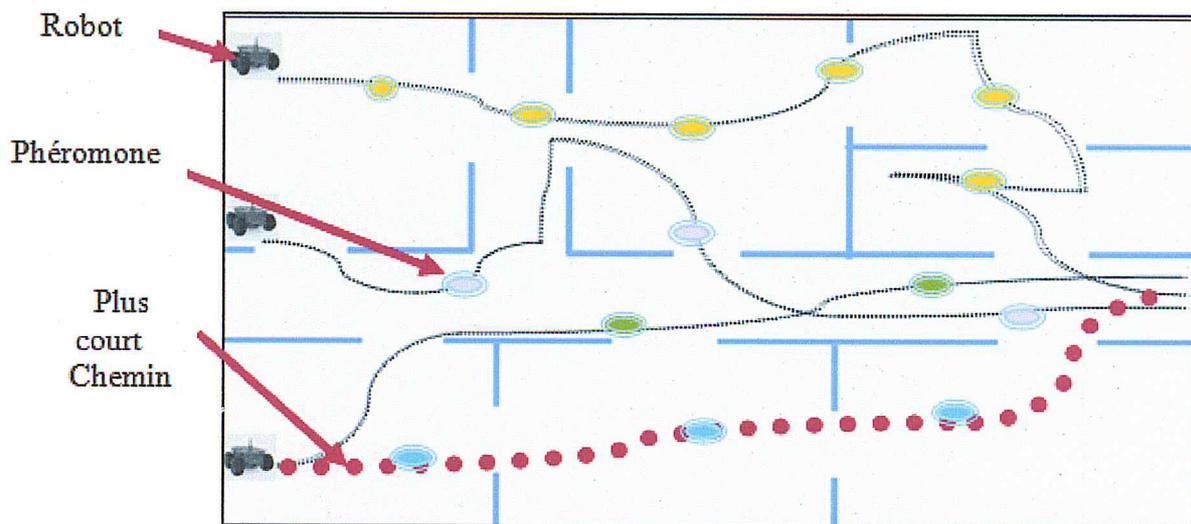


Figure 4.8 : *Extraction chemin optimal.*

4. Conclusion

Tout au long de ce chapitre, divers scénarios ont été réalisés avec le simulateur Player/Stage. En examinant les résultats obtenus, nous avons pu affirmer de la validité de

l'approche que nous avons proposée et implémentée pour le système multi-robots déployé dans l'environnement.

L'avantage principal d'un nombre important de robots mobiles dans un système multi-robots concerne le grand volume d'informations qu'on peut avoir sur l'environnement grâce au nombre important de tags RFID qu'on peut lire à chaque instant. Néanmoins, l'accès au fichier « .txt » partagé entre les différents robots mobiles devient de plus en plus lourd et, de plus, les vitesses de déplacement des robots mobiles diminuent considérablement.

Les derniers tests réalisés et les résultats obtenus montrent que la validité de l'algorithme que nous avons proposé pour extraire le plus court chemin amenant les robots mobiles d'une position initiale à une position finale.

Conclusion générale et perspectives

L'une des tâches les plus importantes en robotique mobile consiste en la détection et l'identification de n'importe quel type d'objets pendant les déplacements d'un robot mobile dans un milieu dynamique. Ainsi, la récupération des différentes informations liées à chaque objet présent dans l'environnement et les sauvegarder dans une base de données pour une utilisation ultérieure de ces informations dans différents contextes est une question nécessaire. À cet effet, de nombreux systèmes et technologies existent pour répondre à cette thématique. Parmi ces systèmes, le système d'identification par radio fréquence (RFID) s'impose comme un système très prometteur.

L'objectif de notre travail était de déployer une infrastructure cyber-physiques RFID dans un environnement pouvant contenir avec des opérateurs humains, des robots et des objets quelconques (mur, salles, etc.). Pour ce faire, nous avons simulé le système RFID par un système fiducial en utilisant le simulateur Player/Stage, de manière qu'un ensemble de tags fiducial soit attachés à tous les éléments de l'environnement. Aussi, les lecteurs fiducial sont placés sur les robots mobiles et peuvent détecter et identifier chaque objet étiqueté. Les informations obtenues sont enregistrées, par la suite, dans un fichier texte (.txt) accessible par tous les robots mobiles du système. Enfin, ces informations seront stockées dans une base de données centralisée.

Ainsi, nous avons proposé un algorithme pour garder la trace des différentes positions où ont passé les robots mobiles pendant la détection en s'inspirant de l'algorithme de colonie de fourmis. Ces phéromones ont été, à la fin, exploitées afin d'en extraire le plus court chemin pour les robots mobiles désirant se rendre d'une position initiale à une position finale.

Dans ce travail, nous avons présenté le système cyber-physique et ses applications avec la technologie RFID notamment dans le domaine de la robotique. Nous avons comparé aussi les différentes technologies d'identification avec le système RFID pour affirmer son efficacité, ses avantages et ses inconvénients.

Ensuite, nous avons décrit, d'une manière détaillée, le système de détection et d'identification ainsi que le traitement des différentes données avec les aspects conceptuels de ce système.

Après, nous avons utilisé le simulateur Player/Stage pour implémenter notre infrastructure cyber-physique avec ses différentes entités.

La conception et l'implémentation de la solution proposée ont été accompagnées, enfin, d'un ensemble de tests de simulation pour la validation de la solution que nous avons proposée.

La réalisation de ce travail a fait émerger plusieurs résultats importants qui répondent à notre problématique. Nous avons pu assimiler les différents objets statiques et mobiles dans un environnement inconnu avec un système multi-robots, collecter un grand nombre d'informations sur cet environnement et mettre à jour ces informations à chaque détection. En plus, les résultats ont permis de prouver l'efficacité de l'algorithme inspiré de colonie de fourmis pour obtenir le plus court chemin d'un robot navigant d'une position initiale à une position finale.

Durant l'implémentation de notre solution, nous avons rencontré plusieurs problèmes avec le simulateur Player/Stage. Son installation sous le système d'exploitation linux était fastidieuse et très difficile, la pluralité des bibliothèques existantes était aussi l'un des obstacles qui nous a compliqué la compréhension, la maîtrise et la manipulation de ce simulateur.

À l'issue du travail présenté dans ce mémoire, différentes pistes formant les directions futures à cette recherche, sont envisagées. Nous citons, dans ce qui suit, les plus importantes :

- Il est intéressant, que notre travail soit appliqué en réalité avec le matériel RFID dans des environnements différents de telle sorte que chaque robot sera équipé par plusieurs lecteurs et en prenant aussi en considération la nature des objets étiquetés (liquide, etc.).
- Une autre direction de recherche envisagée à travers ces études consiste à l'application de notre système intégré avec d'autres travaux : cartographie multi-robots, architecture de contrôle d'un système robotique cyber-physique.

Références bibliographiques

- [1] « Control Architectures of Robots 2015 », *CAR'15*. [en ligne]. <http://www.car-conference.fr/> [page consultée le 03 janvier 2015].
- [2] « Les Etats-Unis misent gros sur la cyber-physique », *L'Atelier-BNP Paribas*. [en ligne]. (03 septembre 2008). <http://www.atelier.net/trends/articles/etats-unis-misent-gros-cyber-physique> [page consultée le 03 janvier 2015].
- [3] CARDON, Alain. « Des systèmes cyber-physiques au Système Méta », *MEDIAPART*. [en ligne]. (04 décembre 2014). <http://blogs.mediapart.fr/blog/jean-paul-baquiast/041214/des-systemes-cyber-physiques-au-systeme-meta/> [page consultée le 28 décembre 2014].
- [4] E. A. Lee, S. A. Seshia, "Introduction to Embedded Systems - A Cyber-Physical Systems Approach", LeeSeshia.org, 2011.
- [5] E. A. Lee, "Computing foundations and practice for cyber-physical systems: A preliminary report", University of California, Berkeley, Technical Report, UCB/EECS-2007-72, 2007.
- [6] E. A. Lee, "Cyber-physical systems: Design challenges", The 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC), pp. 363-369, May 2008.
- [7] E. A. Lee, "CPS Foundations", The 47th ACM Design Automation Conference (DAC'10), pp. 737-742, Anaheim, CA, USA, 13-18 June 2010.
- [8] Lee, Edward A. «Cyber-Physical Systems», Cyber-Physical Systems. [enligne]. <http://cyberphysicalsystems.org/> [page consultée le 15 mars 2015].
- [9] R. R. Rajkumar, I. Lee, L. Sha, J. Stankovic, "Cyber-physical systems: the next computing revolution", The ACM 47th Design Automation Conference (DAC'10), pp. 731-736, Anaheim, CA, USA, 13-18 June 2010.
- [10] Y. Tan, S. Goddard, L. C. Perez, "A prototype architecture for cyber-physical systems", *ACM SIGBED Review*, 5(1), pp. 1-2, 2008.

- [11] C. Talcott, "Cyber-physical systems and events", *Software-Intensive Systems and New Computing Paradigms, Lecture Notes in Computer Science*, vol. 5380, pp. 101-115, 2008.
- [12] K. -J. Park, R. Zheng, X. Liu, "Cyber-physical systems: Milestones and research challenges", *Computer Communications*, 36(1), pp. 1-7, 2012.
- [13] A. Huebner, Ch. Facchi, M. Meyer, H. Janicke, "RFID Systems from a Cyber-Physical Systems Perspective", *The 11th Workshop on Intelligent Solutions in Embedded Systems (WISES 2013)*, pp. 1 – 6, Pilsen, Germany, 10-11 September 2013.
- [14] N. Wu, X. Li, "RFID Applications in Cyber-Physical System", *Deploying RFID-Challenges, Solutions, and Open Issues*, Cristina Turcu (Ed.), InTech, pp. 291-302, 2011.
- [15] «Cyber-Physical Systems Virtual Organization», CPS-VO. [en ligne]. (18 août 2014). <http://cps-vo.org/node/14159/> [page consultée le 20 avril 2015].
- [16] E. Eyisi, X. Koutsoukos, "Passivity-Based Self-Triggered Control: A Case Study on the Trajectory Tracking Control of a Robotic Manipulator Over Wireless Network", *SIGBED Review*, 8(2), pp. 15-18, New York, USA, June 2011.
- [17] C. Neuman, "Challenges in Security for Cyber-Physical Systems", DHS: S&T. Workshop on Future Directions in Cyber-physical Systems Security, Newark, NJ, USA, 22-24 July 2009.
- [18]]Krishna Venkatasubramanian, "Cyber-Physical Systems", [Partly adapted from slides by Prof. Insup Lee, Upenn], CS 525 C.
- [19]]A. Koubâa, B. Andersson, "A Vision of Cyber-Physical Internet", 8th International Workshop on Real-Time Networks, Dublin, Ireland, June 30, 2009.
- [20] Hua-Dong Ma, "Internet of Things: Objectives and Scientific Challenges", *Journal of Computer Science and Technology*, 26(6), pp. 919-924, Nov. 2011.
- [21] C. Benavente-Peces, V. M Moracho-Oliva, A. Domínguez-García, M. Lugilde-Rodríguez, "Global System for Location and Guidance of Disabled People: indoor and outdoor technologies integration", *Fifth International Conference on Networking and Services*, 2009.
- [22] « A Sensitive DIY Ultrasonic Range Sensor », Kerry D. Wong. [en ligne]. (22 janvier 2011). <http://www.kerrywong.com/21/01/22/a-sensitive-diy-ultrasonic-range-sensor/> [page consultée le 26 décembre 2014].
- [23] « Les transmissions sans fils ». [en ligne]. http://jaures-col.spip.ac-rouen.fr/IMG/pdf/Transmissions_sans-fil.pdf [page consultée le 12 janvier 2015].

- [24] « Mesure de température sans contact par infrarouge ». *Raytek A FluckComprany*. [enligne]. <http://www.raytek.fr/Raytek/fr-r0/IREducation/FonctionnementIR.htm>[page consultée le 03 février 2015].
- [25] « Document de consultation sur l'introduction des systèmes sans fil utilisant latechnologie à bande ultra-large », *Industrie Canada*. [enligne]. (28 septembre 2012). <https://www.ic.gc.ca/eic/site/smt-gst.nsf/fra/sf06060.html>[page consultée le 25 novembre 2014].
- [26] B. Noubel El Houssine, “Le WIFI Technologies & Enjeu”, TOPNETFournisseur DeService Internet, avril 2014.
- [27] SCHULER, Eric. PILLOU, Jean-François, «RFID (Radio FrequencyIDentification) », *Comment Ça Marche*. [en ligne]. (avril 2015). <http://www.commentcamarche.net/contents/1028-rfid-radio-frequency-identification/>[page consultée le 23 avril 2015].
- [28] A. Nemmiche, W. Dali youcef, “Etude et intégration d’un dispositif d’identification basé sur la technologie RFID”, Mémoire Master, Université AbouBakr Belkaid, Tlemcen, juin 2013.
- [29] KOUAS-BRAHMI, Vicky. LECHEVALIER, Pauline. « Les codes 2D », *Cerig*. [en ligne]. (12 juillet 2012). <http://cerig.pagora.grenoble-inp.fr/memoire/2012/codes-2D.htm>[page consultée le 10 décembre 2014].
- [30] F. Letient, “Etat de l’art et applications des RFID”, UE ELE114 - Epreuve TEST Travail d’Etude et de Synthèse Technique en ELECTRONIQUE, Grenoble, Juin 2008.
- [31] J. Sztipanovits, J. Stankovic, D. Corman, “Industry - Academy Collaboration in Cyber-Physical System (CPS)”, Research White Paper V. 1, George Mason University, May 2009.
- [32] G. Delille, W. Muzaffar, W. Hashim, “ La technologie RFID pour la santé: Lesapplications dans le domainemédical ”, RFID technology in health domain, research paper written in Internet of Things courses for Telecom Lille 1 University, Août 2013.
- [33] « Codes barres ou RFID dans les soins de santé et le secteur pharmaceutique », *PHIDATA*. [en ligne]. http://www.phidata.be/fr/marchs/soins_de_sant/[page consultée le 20 avril 2015].
- [34] ROUSSEAU, Michel. « RFID : l’armée américaine s’y met », *silicon*. [en ligne]. (10décembre 2005). <http://www.silicon.fr/rfid-larmee-americaine-sy-met-13594.html> [page consulté le 12 novembre 2014].

- [35] Y. Bachoti, B. BelhajSendague, J. G. Rodrigues Oliveira, "PROJET RFID", Projet défini d'étude (option RSM), Janvier 2011.
- [36] D. Work, A. Bayen, Q. Jacobson, "Automotive Cyber Physical Systems in the Context of Human Mobility", National Workshop on High-Confidence Automotive Cyber-Physical Systems, Troy, MI, April 3-4, 2008.
- [37] F. Thiesse, F. Michahelles, "Building the Internet of Things Using RFID, The RFID Ecosystem Experience", IEEE Computer Society, University of Washington, Seattle, pp. 48-55, June 2009.
- [38] T. Deyle, M. S. Reynolds, C. C. Kemp, "Finding and Navigating to Household Objects with UHF RFID Tags by Optimizing RF Signal Strength", 2014.
- [39] L. Chunag, "Object localization strategy for a mobile robot using RFID", Master's Thesis in Computational Science and Engineering, Umeå University, pp. 5-65, July 4, 2012.
- [40] « Da Vinci », *MB Robot*. [en ligne]. <http://mbrobot.e-monsite.com/pages/categories-robots/da-vinci.html> [page consultée le 15 février 2015].
- [41] « Robot chirurgical da Vinci », *Centre intégré universitaire de santé et de services sociaux du Nord- du -l'île-de-Montréal Québec*. [en ligne]. <http://www.hscm.ca/soins-et-services/les-soins-et-services-medicaux-chirurgicaux-psychiatriques-et-professionnels/c/chirurgie/robot-chirurgical-da-vinci/> [page consultée le 03 février 2015].
- [42] A. A. Mekonnen, F. Lerasle, A. Herbulot, A. Coustou, "Coopération entre un robot mobile et des caméras d'ambiance pour le suivi multi-personnes", RFIA 2012 (Reconnaissance des Formes et Intelligence Artificielle), Lyon, France, Jan 2012.
- [43] M. MAMEI, F. ZAMBONELLI, "Pervasive Pheromone-Based Interaction with RFID Tags", *ACM Transactions on Autonomous and Adaptive Systems*, University of Modena and Reggio Emilia, 2(2), article 4, June 2007.
- [44] Support de cours GL/LIG : Mme BOUSSAID, 2010.
- [45] S. BOULAKRINAT, "Gestion du Profil Utilisateur Pour Un Système De Filtrage D'information", Mémoire de Master Professionnel, Option concepteurs des systèmes informatiques.
- [46] J. Guyot, "Des diagrammes objets vers le modèle Relationnel", Université de Genève, pp. 1-16.

- [47] D. L. TRAN, “ Intégration du WiFiBoT dans l'architecture Player/Stage”, Mémoire de stage de fin d'étude, institut de francophonie pour l'informatique, pp. 1-50, 20 décembre 2009.
- [48] J. Owen, “How to Use Player/Stage”, pp. 3-56, July 10, 2009.
- [49] Merrheim, Zemarce, “ Programmation C++ (débutant)/Notions de base”.
- [50] « Les fonctions d'un SGBD », *Site personnel de Christian Braesch*. [en ligne]. <http://www.christian.braesch.fr/page/les-fonctions-dun-sgbd> [page consultée le 10 avril 2015].
- [51] Site officiel de MySQL. [En ligne]. <http://www.mysql.com> [page consultée le : 15mai2015].
- [52] Site officiel de Java. [En ligne]. <http://www.java.com/> [page consultée le:18 mai 2015].
- [53] Site officiel de NetBeans. [En ligne]. <https://netbeans.org/>[page consultée le:20 mai 2015].