



Rapport de Projet de SFE

Préparé par :

KHALIL Mohamed

JENJARE Yassine

Pour l'obtention du diplôme

Diplôme Universitaire de Technologie (D.U.T)

Option : Génie Informatique

Projet intitulé

Smart Logistique utilisant un système robotique pour la gestion des entrepôts.

Encadré par : PR. Hicham BELHEDDAOUI

PR. Nadia AFIFI

PR. Mounir RIFI

Année Universitaire 2020/2021









Remerciement

Premièrement nous remercions Dieu source de toute connaissance. Nous tenons à remercier toute l'équipe Pédagogique de l'école supérieure de technologie Casablanca, le centre d'innovation et de recherche et les Intervenants professionnels responsables de la formation génie Informatique. Nous remercions aussi les membres du jury qui nous font honneur en jugeant ce travail. Nous tenons à remercier toutes les personnes qui ont contribué au succès de notre projet et qui nous ont aidé lors de la rédaction de ce rapport.

Tout d'abord, nous adressons notre remerciement à notre Parent, qui nous ont beaucoup aidé. Leur écoute et leur conseil nous ont permis d'être plus concentré au projet, et d'être motiver tout le temps.

Nous tenons à remercier vivement nos professeurs encadrants, **Mme AFIFI** et **Mr BELHADAOUI**, pour leur accueil, le temps passé ensemble et le partage de leur expertise au quotidien. Grâce aussi à leur confiance nous avons pu nous accomplir totalement dans nos tâches. Ils furent d'une aide précieuse dans les moments les plus délicats.

Nous remercions également, notre professeur **Mr RIFI Mounir** qui nous a offert beaucoup d'aide, soutien et conseils afin de surmonter les difficultés rencontrées.

Enfin, Nous tenons à remercier toutes les personnes qui nous ont conseillé et relu lors de la rédaction de ce rapport de stage de fin d'étude : nos familles, nos camarades de promotion.





Résumé

Le présent rapport synthétise le travail effectué dans le cadre du projet de fin d'études au sein du labo RITM. Ce projet a pour but final le développement et l'intégration d'une solution de smart-logistique visant à faciliter la gestion des entrepôts et la traçabilité des produits entreposés.

Notre Projet a pour objectif de développer et déployer une architecture de smart-warehousing pour une automatisation optimale de la gestion des entrepôts.

La plate-forme conséquente a pour objectif de faciliter, auprès des entrepôts, des opérations encore très couteuses en terme de ressources tel que l'inventaire, la traçabilité des produits etc.

Notre solution Smart-Logistique permet :

- La lecture des Tag RFID assurée par des Robots autonomes afin d'identifier les articles
- Le transfert des données vers un serveur Cloud
- La gestion de stocks

L'automatisation de tous le processus de la lecture des Tag RFID jusqu'au suivi des articles va permettre une réduction des coûts par l'optimisation des ressources humaines déployés et l'augmentation de la productivité.





Abstract

This report summarizes the work carried out as part of the end of studies project within the RITM lab. The final goal of this project is the development and integration of a smart-logistics solution aimed at facilitating warehouse management and the traceability of stored products.

Our Project aims to develop and deploy a smart-warehousing architecture for optimal automation of warehouse management.

The purpose of the resulting platform is to facilitate, with warehouses, operations that are still very costly in terms of resources such as inventory, product traceability, etc.

Our Smart-Logistics solution allows:

- RFID tag reading provided by autonomous robots to identify items
- Data transfer to a Cloud server
- Inventory management

Automating the entire process from RFID tag reading to item tracking will reduce costs by optimizing the human resources deployed and increasing productivity.





SOMMAIRE

Reme	ercie	ement	3
Résu	mé		4
Abstı	ract		5
SOM	MAI	IRE	6
Table	e de	figures	9
INTR	ODL	JCTION GÉNÉRALE	.11
Chap	itre	1 : Contexte du travail et cahier de charge	. 12
1.	Pr	ésentation de l'organisme d'accueil	. 13
	1.	Objectifs	. 14
	2.	Equipes De Recherches	. 14
2.	De	escription du projet :	. 15
	1.	Problématique	. 15
	2.	Objectif Du Projet	. 16
	3.	Planning prévisionnel	. 16
Chap	itre	2 : Etat de l'art	. 18
1.	Qı	u'est-ce que la logistique	. 19
2.	De	éfinition de la Technologie RFID	. 20
3.	Po	ourquoi l'univers de la logistique s'intéresse à	
la R	RFID		. 21
4.	Qı	uelles applications dans le domaine de la Supply Chain?	.22
5.	Ľá	avenir du Supply Chain Management lié à la RFID	. 23
6.	Di	agramme de F.A.S.T	. 23





Cha	pitre 3 : Conception et Modélisation de la solution	. 25
1.	La Gamme de Fréquence	. 26
2.	Lecteur RFID ID ISC.LRU1002	. 27
	Informations techniques :	. 28
3.	Antenne ID ISC.ANT.U270/270	. 29
4.	Les étiquettes Radio-identifications	.31
	Quels types de Tags à Choisir ?	.31
	Le modèle de tag passif qu'on possède est EPC Class1 Gen2	.34
5.	Conception :	.36
	1. Language UML:	.36
	2. Diagramme de cas d'utilisation :	.37
Cha	pitre 4 : Réalisation Du Projet	.39
1.	Les outils utilisés	.40
	ID ISOStart 2017 :	.40
	Apache NetBeans :	.40
	Microsoft SQL Server :	.41
	Microsoft Azure :	.41
	Java :	.42
	Java Data base Connectivity :	.42
2.1	Partie technique	.43
	1. ISOStart	.43
	2. Application de gestion des tags	.50
	3. Partie Code :	.51
Con	clusion	.57









Table de figures

Figure 1 : Diagramme de Gantt	. 17
Figure 2 Digarmme de F.A.S.T	. 24
Figure 3: Table de frequence	. 26
Figure 4: Lecteur ID.ISC LRU1002	. 27
Figure 5: Caracteristique du lecteur	. 28
Figure 6: Antenne ID ISC.ANT.U270/270	. 29
Figure 7: Carcateristique de l'antenne	.30
Figure 8: Tags active vs Tags passive	.31
Figure 9: Tags passive	.32
Figure 10: Tags Active	. 34
Figure 11: Diagramme de cas d'utilisation	.38
Figure 12: Diagramme de cas d'utilisation	.38
Figure 13: ISOStart	. 40
Figure 14: Netbeans	.40
Figure 15: MSSQL Server	.41
Figure 16: Microsoft Azure	.41
Figure 17: Java	.42
Figure 18: JDBC	. 42
Figure 19: Configuration initiale du leteur	. 44
Figure 20: Buffered read mode	. 45
Figure 21: Configuration du mode	. 46
Figure 22: EPC Globale	. 47
Figure 23: Read Multiple Blocks	
Figure 24: Write Multiple Blocks	
Figure 25:Interface de l'Applicarion desktop	.50
Figure 26: Connexion avec base de donnees	
Figure 27: Insertion des donnees	.52





Figure 28: Modification des donnees	53
Figure 29: Suppresion des donnees	53
Figure 30: Classe du TAG	54
Figure 31: Autres fonctionnalite de l'application	56





INTRODUCTION GÉNÉRALE

Dans le but de compléter notre formation et d'acquérir notre diplôme Universitaire Technique au sein de l'école Supérieure de Techniques de Casablanca, on est amené à effectuer un Stage de fin d'étude pendant lequel on doit réaliser un projet proposé par le centre d'innovation, Le projet s'étend sur une durée de 2 mois Mai et Juin.

Ce projet a pour but l'exploitation et la mise en œuvre des connaissances et compétences acquises durant notre formation, ainsi l'auto-formation et l'amélioration des notions de travail d'équipe, de coordination et de gestion primordiaux pour s'introduire dans le monde professionnel.

Pour notre part, on a pour objectif de notre stage est La lecture des Tag RFID assurée par des Robots autonomes afin d'identifier les articles.

Parce qu'une bonne structure est nécessaire pour une bonne compréhension, Nous avons adopté le plan suivant :

Le premier chapitre sera consacré au Contexte du travail et cahier de charge Avec une description détaillée de ce qu'on va faire, divisé en une description graphique et un autre fonctionnelle et technique.

Dans le deuxième chapitre, nous présentons l'état de l'art relatif à la relation du RFID avec la logistique.

Le troisieme chapitre , nous utilisons les différents diagrammes pour modéliser les fonctionnalités de la lecture RFID

Le dernier chapitre, où on va présenter les outils utilisés pour la réalisation ainsi que des captures d'écrans de l'application avec une description, ainsi on va donner une conclusion générale de notre projet et des perspectives.





Chapitre 1 : Contexte du travail et cahier de charge

La présentation du cadre général du projet a pour but de situer le projet dans son environnement organisationnel et contextuel. Ce chapitre introduit donc l'organisme d'accueil, décrit la problématique à traiter et présente la démarche suivie pour la réalisation du projet.





1. Présentation de l'organisme d'accueil

L'école supérieure de technologie de Casablanca compte parmi les grands établissements publics d'enseignement supérieur à finalité professionnalisant, elle a été créée en 1986 par le ministre de l'enseignement supérieur de la formation des cadres et de la recherche scientifique.

L'école supérieure de technologie de Casablanca et une composante de l'Université Hassan sur sa vocation et de former des techniciens supérieurs polyvalent hautement qualifié et immédiatement opérationnels après leur sortie de l'école en tant que collaborateurs d'ingénieurs et de managers.

L'école supérieure de technologie Casablanca dispose de plusieurs départements qui pilotent et gèrent les formations de leurs étudiants, chaque département se charge de plusieurs disciplines.

Ce stage a été réalisé au sein du centre d'innovation et de recherche dont font partie les équipes de recherche du département génie informatique.

Dirigé par Monsieur Mounir RIFI professeurs de l'enseignement supérieur à l'ESTC même et qui travaille avec l'équipe de recherche réseau et télécommunication (R&T).

Le laboratoire est intégré dans le CED (Centre d'Etudes Doctoral) science de l'ingénieur à l'ENSEM. Il comporte plus d'une trentaine de membres composés des enseignants chercheurs et des doctorants rattachés à quatre équipes.





1. Objectifs

Les activités de recherche du laboratoire sont centrées autour de son projet fédérateur les technologies de l'information et de la communication. L'ensemble des 7 équipes du RITM participe de façon équilibrée à la valorisation des travaux de recherche. Le labo RITM entretient de nombreuses relations au niveau national et international.

Le labo RITM couvre des thématiques scientifiques larges regroupant 7 équipes :

- Technologies de l'Information et de la Communication,
- Sécurité de l'information,
- Réseaux informatiques et systèmes distribués,
- Sciences de données et ingénierie des systèmes,
- Télécommunications,
- Théorie des systèmes,
- Modélisation en Optoélectronique et Electronique,
- Energies renouvelables,
- Efficacité énergétique,
- Smart cities,
- Smart grids,
- Compatibilité électromagnétique dans les dispositifs de conversion d'énergie.

2. Equipes De Recherches

- RSD : Réseaux et Systèmes Distribués
- SDIS : Sciences de Données et Ingénierie des Systèmes
- MODEC : Modélisation Optoélectronique et électronique
- R3E : Réseaux Électriques & Efficacité Énergétique
- RT: Réseaux & Télécommunications





SIASR: Security of Information and Advanced Software Research

• SC: Smart Cities SC

2. Description du projet :

1. Problématique

Petit à petit, la technologie RFID fait son chemin en entreprise.

Si certaines sont carrément en avance (avec l'implantation de puce RFID en sous-cutanée), d'autres commencent à peine à se familiariser avec le sujet.

Au-delà de la méfiance légitime liée à la protection et l'exploitation des données, la RFID est surtout un outil à considérer pour améliorer les procédures au sein d'une entreprise. Et cela, sans avoir à greffer une puce sur les employés.

Toutes les entreprises de productions s'appuient sur le principe du stockage physique, avec l'objectif d'assurer une réponse à la forte demande toujours aussi grandes de nos sociétés de surconsommation. D'un point de vue organisationnel, il s'agit de faciliter la transmission et le suivi d'informations tout en maintenant une fiabilité impeccable.

Les Tags RFID sont une bonne solution pour presque tous les types de produits, permettant d'introduire beaucoup d'informations sur le produit, son prix est très compétitif et permet de garder une trace de l'unité de stock.





2. Objectif Du Projet

Les objectifs du projet, tels qu'ils ont été établis lors des premières phases, ont été fixés autour de l'objectif principal qui est la mise en place d'une architecture de smart- Warehousing pour une automatisation optimale de la gestion des entrepôts en utilisant le procédé de lecture RFID ainsi qu'un robot autonome.

Ce Processus consiste à munir un robot d'une self-awareness géographique le rendant capable de s'adapter aux environnements d'entreposage grâce à une technique de mappage d'où l'aspect autonome, ensuite On va se charger de le programmer pour qu'il puisse lire des tags RFID qui est l'innovation apporté à l'ancien procédé du code à barres. Il faudra finalement lier toutes ces technologies pour pouvoir donner un géolocalisation précis et propre à chaque produit. Les données seront ensuite envoyées vers une autre équipe chargée de répertorier le tout dans une base données flexible usant de la technologie de block Chain afin d'assurer la traçabilité de chaque produit.

3. Planning prévisionnel

La planification du projet est une phase indispensable pour préparer les projets. Pour cette planification on a utilisé le diagramme de Gantt pour diviser les différentes tâches en fonction de leur période temporelle.





La planification du projet est une phase indispensable pour préparer les projets. Pour cette planification on a utilisé le diagramme de Gantt pour diviser les différentes tâches en fonction de leur période temporelle.

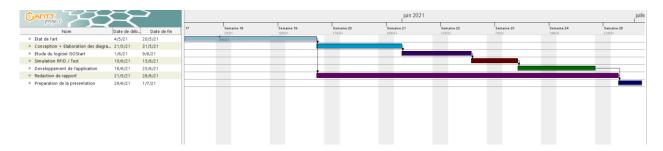


Figure 1 : Diagramme de Gantt

Le diagramme de Gantt, couramment utilisé en gestion de projet, est l'un des outils les plus efficaces pour représenter visuellement l'état d'avancement des différentes activités (tâches) qui constituent notre projet comme nous montre la figure ci-dessus.





Chapitre 2 : Etat de l'art

Une étape essentielle de tout cycle de développement logiciel ou conceptuel consiste à effectuer une étude préalable. Le but de cette phase est de comprendre le contexte du système. Il s'agit d'éclaircir au mieux les besoins fonctionnels et non fonctionnels, apparaître les acteurs et identifier les cas d'utilisation.

Dans ce chapitre seront exposés les résultats de la phase d'étude, d'analyse et de cadrage du projet. La spécification des besoins se fera à l'aide de la description des différentes sources de données, des indicateurs retenus et leurs axes d'analyse.





1. Qu'est-ce que la logistique

La logistique consiste à piloter des flux physiques de produits à destination du client final en respectant un cahier des charges précis : dans les meilleurs délais, avec la meilleure qualité de service possible, tout en optimisant les stocks et les flux.

Cette gestion des flux passe par des outils innovants, offrant de la traçabilité ou une image du stock en temps réel.

Gérer les flux physiques/données, mettre à disposition les ressources nécessaires à l'atteinte des objectifs, maîtriser les coûts, sont autant de critères à prendre en compte pour optimiser sa stratégie logistique.

La logistique peut-être soit intégrée (gérée par la marque comme un service interne) soit externalisée (confiée à un prestataire spécialisé de type 3PL).

On constate différents types de logistique selon le marché/secteur de la marque, qu'elle soit de prêt-à-porter ou de cosmétiques / accessoires :

- Logistique de distribution
- Logistique textile ou logistique prêt-à-porter
- Logistique maroquinerie
- Logistique cosmétique
- Logistique luxe ou logistique grande distribution





Les flux logistiques se différencient selon des notions d'amont et aval

- Amont avec l'organisation de l'approvisionnement, le pilotage des stocks, l'entreposage et l'optimisation des flux du fournisseur jusqu'à un stock central
- Aval avec la préparation des commandes, l'expédition, le SAV, la gestion des retours et l'optimisation de ces flux

L'optimisation des coûts et des délais est la base d'une logistique maîtrisée. Elle concerne l'optimisation du stock sans atteindre la rupture et les coûts transport par exemple, afin d'améliorer la qualité de service et de garantir une forte satisfaction client. Elle doit permettre d'améliorer la productivité des opérateurs, passant également par une communication transversale.

2. Définition de la Technologie RFID

La radio-identification ou RFID (Radio Fréquence Identification) est une méthode issue de l'aéronautique qui est destinée à mémoriser et récupérer des données à distance grâce à des marqueurs appelés « radio-étiquettes ». Développée à la fin des années 40 afin de répondre à une problématique d'identification des appareils circulants dans l'espace aérien, cette technologie a ensuite été utilisée par les militaires avant de faire son apparition dans le domaine du pilotage de la chaîne logistique il y a maintenant une dizaine d'années.





Malgré un coût élevé et la nécessité d'une adaptation technique des processus non négligeable, on constate aujourd'hui, grâce à de nombreux retours d'expérience, des gains non négligeables pour les entreprises ayant décidé d'inclure la RFID dans leur solution logistique.

3. Pourquoi l'univers de la logistique s'intéresse à la RFID

Standardisée dans les années 90, la RFID est une « technologie d'identification automatique qui utilise le rayonnement radiofréquence pour identifier les objets porteurs d'étiquettes lorsqu'ils passent à proximité d'un interrogateur » (Source Centre national de la RFID). Pour faire plus simple, la RFID permet de stocker une information dans une puce électromagnétique. Ensuite cette puce pourra être interrogée à l'aide d'un terminal radiofréquence (on parle alors de puce passive) ou sera à même de communiquer ses informations (on parle dans ce cas de puce active). A l'ère du Big Data, on imagine aisément l'intérêt de pouvoir suivre avec précision des flux de matières ou de produits dans de nombreux domaines tels que l'industrie, le secteur pharmaceutique, les bibliothèques, la logistique ou la grande distribution...





4. Quelles applications dans le domaine de la Supply Chain ?

Bien que posant certaines contraintes techniques du point de vue de l'intégration dans les logiciels de logistique, la RFID permet de travailler sur les aspects suivants de la Supply Chain :

La visibilité des flux logistiques :

Acquisition de données en temps réel ou à des points-clés de la Supply Chain.

L'amélioration du pilotage de l'entrepôt :

Connaissance en temps réel de l'emplacement, amélioration du taux de fiabilité des stocks grâce une « communication » automatique avec la puce active. Nettement plus efficace que le classique travail de recherche / comptage de l'opérateur avec de simples codes-barres.

La réduction des coûts d'inventaires :

Les procédures d'inventaires gagnent en rapidité et en fiabilité.

La traçabilité :

Connaissance en temps réel du positionnement des articles équipés de puces qui sont stockés ou qui transitent dans l'entrepôt. La RFID se met au service du Warehouse Management.

L'amélioration de la productivité :

Identification plus rapide (pas d'étape de scanning d'un code-barres), réduction des opérations de comptage des articles et potentiel d'optimisation de tâches à faible valeur ajoutée (ex : aiguillage d'article sur une machine de tri).





La réduction du risque d'erreur :

La lecture de puce permet d'éviter des erreurs de tri et garantit l'exactitude de la donnée.

5. L'avenir du Supply Chain Management lié à la RFID

Avec la volatilité de l'économie mondiale et le besoin toujours plus marqué d'accroître sa visibilité afin d'améliorer son efficacité opérationnelle, la RFID n'en est certainement qu'à ses débuts et devrait bientôt envahir l'univers de la chaîne logistique. Et lorsque l'on observe le développement ultra rapide de l'Internet des objets, on s'imagine que de nombreuses révolutions risquent d'arriver bien plus vite que prévu!

6. Diagramme de F.A.S.T

FAST (en anglais function analysis system technic) est un type de diagramme qui présente une manière de penser, d'agir, ou de parler. Le diagramme FAST se construit de gauche à droite, dans la logique suivante : du "pourquoi" au "comment". Grâce à sa culture technique et scientifique, l'ingénieur développe les fonctions de service du produit en fonctions techniques. Il choisit des solutions pour construire finalement le produit. Le diagramme FAST constitue alors un ensemble de données essentielles permettant d'avoir une bonne connaissance d'un produit complexe et ainsi de pouvoir améliorer la solution proposée





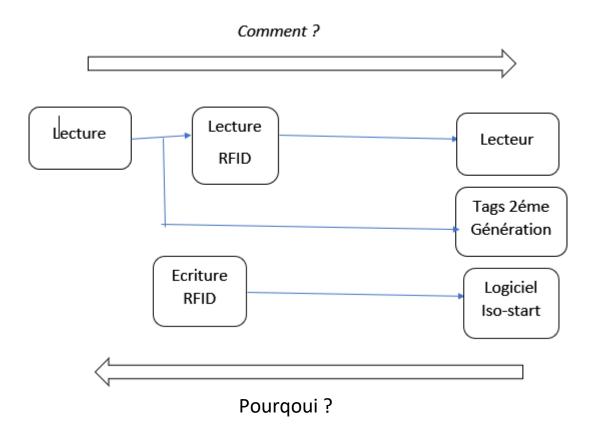


Figure 2 Digarmme de F.A.S.T





Chapitre 3 : Conception et Modélisation de la solution

Nous présentons dans ce chapitre la phase de conception de notre projet. En y incluant les deux axes majeurs de ce dernier, matérialisés en la partie lecture et écriture dans les Tags RFID. Il sera question de présenter la solution théoriquement idéale à chaque solution et une autre pratique. Puis nous présenterons différents diagrammes de consolidation du plan de travail.





1. La Gamme de Fréquence

Choisir un système de lecture, c'est d'abord choisir une fréquence de communication et un type d'étiquette. Le principe est simple : le champ doit « éclairer » ni trop ni trop peu, sous peine de manquer des objets ou de toucher des produits circulants ailleurs.

La fréquence est la caractéristique qui permet d'établir la communication entre la puce et l'antenne. Cette fréquence utilisée est variable, selon le type d'application visé et les performances recherchées

Frequency Range	Low frequency RF	High frequency RF ISO/IEC 15693	High frequency RF ISO/IEC 14443	Ultra High frequency RF	
	125/135 KHz	13,56 MHz	13,56 MHz	902 – 928 MHz	
Technology name	RFID	RF enabled contactless	RF enabled contactless	RFID	
rechnology fiame	KID	smart card	smart card	RFID	
	Proprietary for access		ISO/IEC 14443 and		
Standard (for commu- nications link)	control. ISO/IEC 11784 and ISO/IEC 11785	ISO/IEC 15693	ISO/IEC 7816, parts 4	ISO 18000-6 for inven- tory control tags	
			& above for applica- tion/security standards		
Operational Range	Medium: < 20 - 60 cm	Medium: < 70 cm	Short: < 10 cm	Long: 3,5 - 10 m	
Data transfer rates	< 10 Kbps	26 Kbps	106 - 848 Kbps	20 - 100 Kbps	
Chip types supported	Memory only	Memory, Fixed Logic	Memory, Fixed Logic, Microcontroller, Crypto processors	Memory only	
	Hundreds to low kilo-				
Memory capacity range	bytes	256 bytes and 2K bytes	64 Kbytes and more	Hundreds of bits today	
Read/write ability	Read/write	Read/write	Read/write	Read/write	
Factory affecting secu- rity	Longer range, Fixed logic chip, Limited flexibility in communications proto- cols	Longer range, Fixed logic chip, Limited flexibility in communications proto- cols	Short range, Programmable micro- controller, Counterfeiting and tamper resistance fea- tures, More Flexibility in communications proto- cols	Longer range, Fixed logic chip, Limited flexibility in communications proto- cols	
	Tags,	Tags,	Plastic card,		
Available form factors	Plastic card, Key fob, Watch	Plastic card, Key fob, Watch	Key fob, Watch,	Tags, Plastic card	
			Mobile phone		
			Secure ID cards and		
Applications	Security, Access control, Asset tracking, Animal tracking,	Inventory tracking, Physical access control systems	documents (ePassport), Credit and debit card payment,	Transportation vehicle Inventory tracking, Supply chain	
	Automobile immobi- lizer		Transit payment, Physical access control systems		

Figure 3: Table de frequence





Une fois déterminée la fréquence idéale, quel lecteur choisir ? Plusieurs systèmes peuvent cohabiter sur un site, pour permettre l'identification des mêmes marchandises dans plusieurs situations. Un lecteur est associé à une ou plusieurs antennes, dont le choix dépend des conditions de lecture, de l'orientation des tags et de l'environnement.

2. Lecteur RFID ID ISC.LRU1002



Figure 4: Lecteur ID.ISC LRU1002





Le lecteur UHF LRU1002 longue portée peut être utilisé pour de multiples applications et convainc par son excellent rapport qualité-prix. La sensibilité du récepteur prend en charge une gamme élargie mais homogène d'étiquettes et la plage de lecture peut aller jusqu'à 12 mètres, même dans des environnements difficiles.

Informations techniques:

Mechanical Data			
Mechanical Data Housing	Aluminum, powder coated	Features	
riousing	Aluminum, powder coated	Supported transponder types	EPC Class1 Gen2
Dimensions	260 mm x 157 mm x 65 mm		EPC Class1 Gen2 V2
Dimensions	(10,23 x 6,18 x 2,56 inch)		ISO 18000-6-C (Upgrade Code)
Weight	1.800 g	Indicator	16 LEDs for diagnosis of reader
. roight	1.000 g		operation and antenna status
Protection Class	IP 53,		7777222079
	IP 64 (with protection cap)*	Other Features	Anti-Collision, Output of RSSI values and phase
Color	RAL9003 Signal white		angle,
			Battery assisted Real Time Clock
			Supports encrypted transponder
Electrical Data			communication,
Power Supply	24 V DC (± 20 %)		Secure Key Storage, "Config Cloning" function
Power Consumption	max. 24 VA**		
Operating Frequencies		Environmental Conditions	
- Version EU:	865 MHz to 868 MHz	Temperature	
- Version FCC:	902 MHz to 928 MHz	- Operation	-25 °C to 55 °C
		- Storage	-25 °C to 85 °C
Output Power	100 mW to max. 2 W	117702600	72.234.522.72.72.74.74.74.74.75.74.7
	configurable in steps of 100 mW	Humidity	5 % to 95 % (non-condensing)
Antenna Connector	4 x SMA-Female (50 Ohm),	Vibration	EN 60068-2-6
Antenna Connector	integrated Multiplexer,		10 Hz to 150 Hz: 0,075 mm / 1 g
	support of external Multiplexer	Ch - d	EN 00000 0 07
	ID ISC.ANT.UMUX	Shock	EN 60068-2-27 Acceleration: 30 g
			Acceleration: 30 g
RF-Diagnosis	RF-channel monitoring,		
	Antenna SWR control,	Applicable Standards	
	internal overheating control	Radio Regulation	
		- Europe	EN 302 208
Outputs		- USA	FCC 47 CFR Part 15
- 2 Optocoupler - 2 Relays	max. 24 V DC / 20 mA max. 24 V DC / 1 A switching	- Canada	IC RSS-GEN, RSS-210
32. TO 6 1. S.	current, 2 A permanent current	EMC	EN 301 489

Figure 5: Caracteristique du lecteur





- Fréquence de fonctionnement : 860 MHz ~ 960 MHz
- Plage de lecture : 20m ~ 30m (dépend du tag et de L'environnement)
- Vitesse de lecture : Lecture simultanée de 200 à 300 tags
- Distance d'écriture : 10m ~ 15m (dépend du tag et de l'environnement)
- Vitesse d'écriture : 8bits moins de 30ms
- Consommation d'énergie : 6W

3. Antenne ID ISC.ANT.U270/270



Figure 6: Antenne ID ISC.ANT.U270/270





L'antenne ID ISC.ANT.U270/270 de l'OBID i scan UHF

La série est spécialement conçue pour une utilisation dans les différents UHF Bandes de fréquence. Une version séparée est disponible pour le Bande de fréquence européenne dans la gamme de 865 MHz à 868 MHz ainsi que pour la bande de fréquence FCC de 902 MHz à 928 MHz

	ID ISC.ANT.U270/270
Dimensions *	270 x 270 x 57 mm ³
Housing	ASA-ABS
Colour	White
Weight	1.210 g
Protection class	IP65
Gain	approx. 9 dBic
3 dB beam width	65° x 65°
Polarization	circular
Antenna connection	SMA socket (50 Ohm)
Temperature range	
Operation	 25 °C up to 55 °C
Storage	- 25 °C up to 80 °C

Figure 7: Carcateristique de l'antenne





4. Les étiquettes Radio-identifications



Figure 8: Tags active vs Tags passive

Les étiquettes RFID permettent de reconnaître un produit et d'obtenir toutes ses caractéristiques. Ce procédé s'est notamment vu sur des étiquettes authentifiant chaque produit dans le but d'éviter le vol et la contrefaçon. Pour optimiser au mieux sa chaîne logistique, des entreprises comme Décathlon utilisent la technologie RFID pour reconnaître les produits et les échanger sans tickets de caisse.

Quels types de Tags à Choisir?

Pour faire le choix de la bonne technologie RFID pour le tag que vous allez employer pour votre application, il convient de faire un résumé des technologies disponibles sur le marché :





Le tag RFID passif: comme son nom l'indique, ce tag RFID est purement passif, c'est à dire qu'il n'intègre pas d'émetteur radiofréquence. Le tag passif utilise l'onde (magnétique ou électromagnétique) issue de l'interrogateur (lecteur RFID + antenne) pour alimenter le circuit électronique embarqué. Il peut contenir des informations dans sa mémoire. C'est le type de tag le plus répandu sur le marché.

Ils sont peu coûteux à fabriquer : leur coût moyen de 2007 à 2016 se situe entre 0.10€ et 0.20€, et varie de 0,05€ au minimum à 1,5€. Ils sont généralement réservés à des productions en volume.

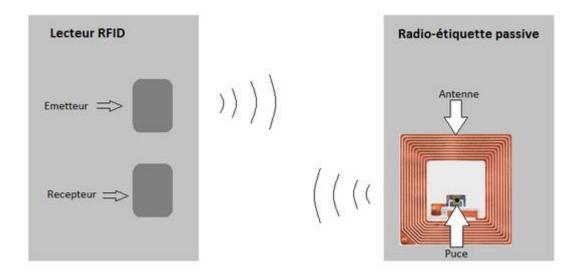


Figure 9: Tags passive





Le tag RFID semi-passif ou assisté par batterie (BAP Battery Assisted Passive) : ce tag embarque une alimentation (batteries).

L'alimentation n'est pas utilisée pour apporter de l'énergie à un émetteur puisque le principe de communication reste équivalent au tag RFID passif. Cette énergie est utilisée en général pour alimenter le circuit électronique du tag ou un capteur connecté au circuit de base par exemple (température, intensité, accélération, gyroscope, etc.). Ce type de tag propose théoriquement des performances plus élevées en comparaison au tag RFID passif. Ce type de tag est utilisé généralement pour des applications particulières : relevé de température, flux de personnes, etc. Ces tags nécessitent de la maintenance liée au changement de batteries. Ils représentent moins de 15% des ventes de tags RFID en 2012.

Le tag RFID actif: ce tag embarque un émetteur radiofréquence et par conséquent une alimentation (batteries, piles). Tout comme le tag semi-actif, il peut être équipé de capteurs (température, gyroscope, GPS, etc.) et embarquer de l'intelligence sous format d'un microcontrôleur par exemple. Ce tag peut interagir de manière autonome et en temps réel avec son environnement grâce à sa batterie: envoi de sa position, prise de température, etc. Ces tags nécessitent la mise en place d'une structure de communication (nœuds, antennes, lecteurs) afin de « quadriller » une zone, un bâtiment, etc. Ils nécessitent également de la maintenance liée au remplacement des batteries, plus fréquent qu'un tag semi-actif. Ils représentent moins de 15% des ventes de tags RFID en 2012.





Les tags actifs les plus représentatifs tournent autour de 30€ l'unité. Bien sûr, si vous avez le budget et le besoin, vous pouvez toujours opter pour des balises actives pouvant coûter plus de 100 € l'unité, celles-ci sont généralement livrées avec un étui de protection spécial, une autonomie extra longue ou des capteurs.



Figure 10: Tags Active

Le modèle de tag passif qu'on possède est EPC Class1 Gen2

Les tags EPC Gen2 sont disponibles en plusieurs saveurs différentes - EPCglobal - ce qui signifie que le tag peut être lu à n'importe quelle fréquence entre 860 MHz et 960 MHz, EPC Gen2 860 MHz. ~ 868 MHz, également connu sous le nom de Gen2 européen et Gen2 902 MHz ~ 928 MHz connu sous le nom de Gen2 nord-américain. Une excellente saveur de la balise EPCglobal est une balise mondiale EPC - cette balise mondiale a une déviation de lecture maximale de 1,5 dBi sur l'ensemble du spectre de lecture de 860 à 960 MHz, ce qui le rend parfait pour les solutions mondiales/mondiales.





En effet, vous pouvez utiliser une seule étiquette dans le monde entier qui identifie positivement l'article étiqueté directement à la source, d'où une gestion et une visibilité de la chaîne d'approvisionnement de bout en bout.

Les balises Gen2 sont disponibles dans une très grande variété de types, de formes et de tailles qui sont généralement conçues pour une application spécifique.

Les étiquettes RFID Gen2 sont composées d'une antenne et d'une puce (plus précisément appelée circuit intégré ou IC). Les circuits intégrés pour les balises Gen2 contiennent **quatre types** de mémoire :

Reserved Memory:

Cette banque de mémoire stocke le mot de passe de suppression et le mot de passe d'accès (chacun de 32 bits). Le mot de passe kill désactive définitivement le tag (très rarement utilisé) et le mot de passe d'accès est défini pour verrouiller et déverrouiller les capacités d'écriture du tag. Cette banque de mémoire n'est accessible en écriture que si vous souhaitez spécifier un certain mot de passe.

EPC Memory:

Cette banque de mémoire stocke le code EPC ou le code de produit électronique. Il a un minimum de 96 bits de mémoire inscriptible. La mémoire EPC est ce qui est généralement utilisé dans la plupart des applications si elles n'ont besoin que de 96 bits de mémoire.





TID Memory:

Cette mémoire est utilisée uniquement pour stocker le numéro d'identification d'étiquette unique par le fabricant lors de la fabrication du circuit intégré. Typiquement, cette portion de mémoire ne peut pas être modifiée.

User Memory:

Si l'utilisateur a besoin de plus de mémoire que la section EPC n'en dispose, certains circuits intégrés ont une mémoire utilisateur étendue qui peut stocker plus d'informations. En ce qui concerne la mémoire utilisateur, il n'y a pas de norme sur le nombre de bits de mémoire pouvant être écrits sur chaque étiquette.

5. Conception:

Afin de détailler les besoins précédemment spécifiés, une bonne réflexion autour du développement de notre application par un langage de modélisation comme l'UML (Unified Modeling Language) s'avère nécessaire. Nous utilisons alors dans la suite les différents diagrammes.

1. Language UML:

La réussite de notre projet nécessite de faire une bonne conception qui permet d'assurer la cohérence et l'intégrité entre les données et aussi les biens structurés. Pour cela on doit passer par l'une des méthodes de conception comme l'UML (Unified Modeling Language) qui est un langage utilisé pour montrer la relation entre les tables de





la base des données et leurs fonctionnalités. Il a été construit afin de standardiser les artéfacts de développement (modèles, notation, diagrammes) sans standardiser le processus de développement. Ce langage de modélisation unifie également les notations nécessaires aux différentes activités d'un processus de développement et offre, par ce biais, le moyen d'établir le suivi des décisions prises, depuis la définition des besoins jusqu'au codage. Il spécifie treize types de diagrammes de modélisation des systèmes dans ce projet on va s'intéresser au :

• Diagramme du cas d'utilisation : Décrit les fonctions du système selon le point de vue de ses futurs utilisateurs.

2. Diagramme de cas d'utilisation :

Les cas d'utilisation permettent de représenter le fonctionnement de l'application vis-à-vis de l'utilisateur : c'est donc une vue de l'application dans son environnement extérieur.

Les fonctionnalités principales que l'application doit assurer, ainsi raffinées nous passons à leur modélisation en langage UML à travers le digramme de cas d'utilisation :





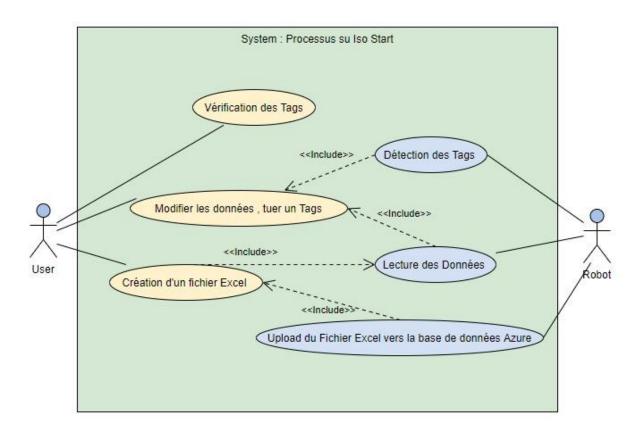


Figure 11: Diagramme de cas d'utilisation

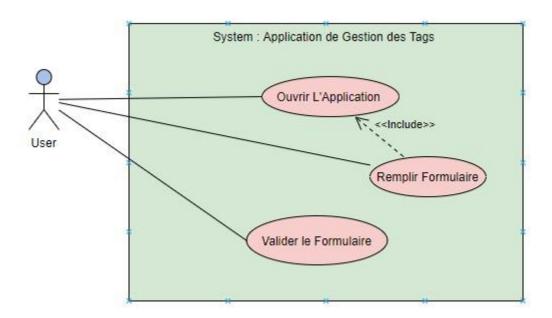


Figure 12: Diagramme de cas d'utilisation





Chapitre 4 : Réalisation Du Projet

Au sein de ce dernier chapitre, nous allons présenter l'environnement de travail, les différentes étapes de mise en place de notre système de lecture et de l'écriture, et après avoir achevé la conception orienté objet de notre application, il est indispensable de se lancer dans la phase de développement. Nous avons opté les outils nécessaires tels que le choix du langage de programmation...





1. Les outils utilisés ID ISOStart 2017 :

ISOStart





Figure 13: ISOStart

Le programme de démonstration ID ISOStart a été développé pour vous familiariser avec les fonctionnalités des lecteurs OBID i-scan® et OBID® classic-pro. Grâce à ce logiciel, vous pouvez :

- Tester la communication avec les transpondeurs HF et UHF.
- Lire et modifier la configuration de Lecteurs OBID iscan®undOBID®classic-pro.
- Communication avec des unités fonctionnelles comme le multiplexeur ou le syntoniseur d'antenne dynamique
- Activer une mise à niveau du micrologiciel.

<u>Apache NetBeans :</u>



NetBeans est un environnement de développement intégré, placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2. En plus de Java, NetBeans permet la prise en charge native de divers langages tels le C, le C++, le JavaScript, le XML, le Groovy, le PHP et le HTML, ou d'autres par l'ajout de greffons.





Microsoft SQL Server:



Microsoft SQL Server est un système de gestion de base de données (SGBD) en langage SQL incorporant entre autres un SGBDR (SGBD relationnel ») développé et commercialisé par la société Microsoft. Il fonctionne sous les OS Windows et Linux (depuis mars 2016), mais il est possible de le lancer sur Mac OS via Docker, car il en existe

une version en téléchargement sur le site de Microsoft

Microsoft Azure:



Figure 16: Microsoft Azure

Microsoft Azure (Windows Azure jusqu'en 2014) est la plate-forme applicative en nuage de Microsoft. Son nom évoque le « cloud computing », ou informatique en nuage (l'externalisation des ressources informatiques d'une entreprise vers des datacenters distants).





Java:



Figure 17: Java

Java est une technique informatique développée initialement par Sun Microsystems puis acquise par Oracle à la suite du rachat de l'entreprise. Défini à l'origine comme un langage de programmation, Java a évolué pour devenir un ensemble

cohérent d'éléments techniques et non techniques.

Java Data base Connectivity:



Figure 18: JDBC

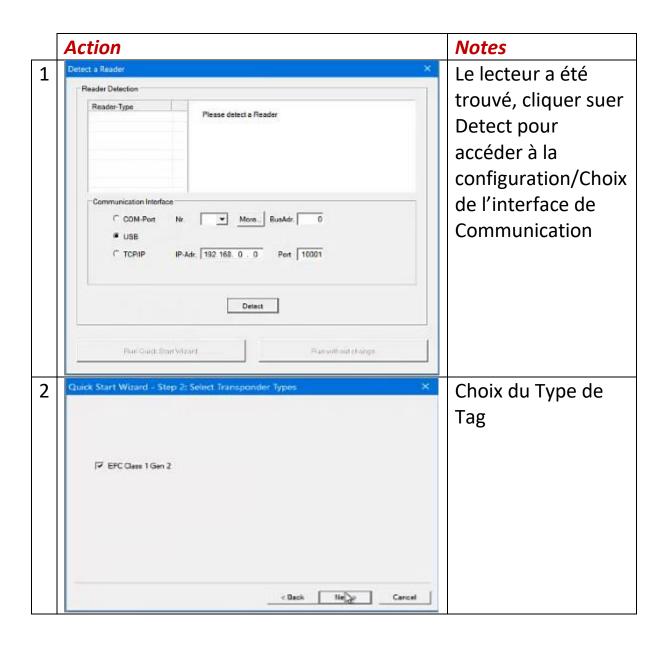
JDBC est une interface de programmation créée par Sun Microsystems — depuis racheté par Oracle Corporation — pour les programmes utilisant la plateforme Java. Elle permet aux applications Java d'accéder par le biais d'une interface commune à des sources de données pour lesquelles il existe des pilotes JDBC.





2.Partie technique

1. ISOStart







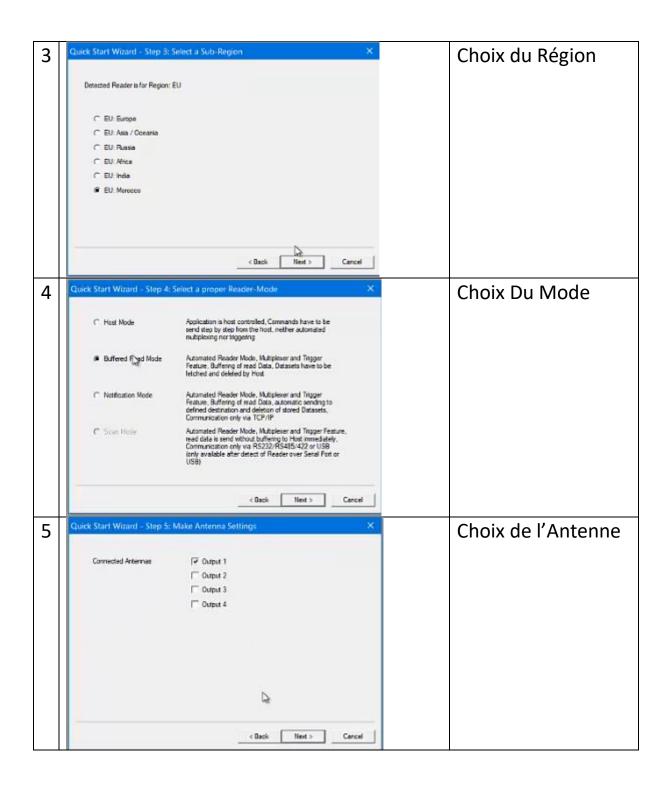


Figure 19: Configuration initiale du leteur





Buffered Read Mode

Le mode de lecture tampon est un mode de fonctionnement de haut niveau pour détecter les transpondeurs qui se trouvent dans la plage de détection du lecteur. Ce mode de fonctionnement traite toutes les données de lecture du transpondeur et les opérations de filtrage pour rendre l'interface utilisateur transparente pour les données du transpondeur et pour minimiser les transferts de données entre le lecteur et l'hôte.

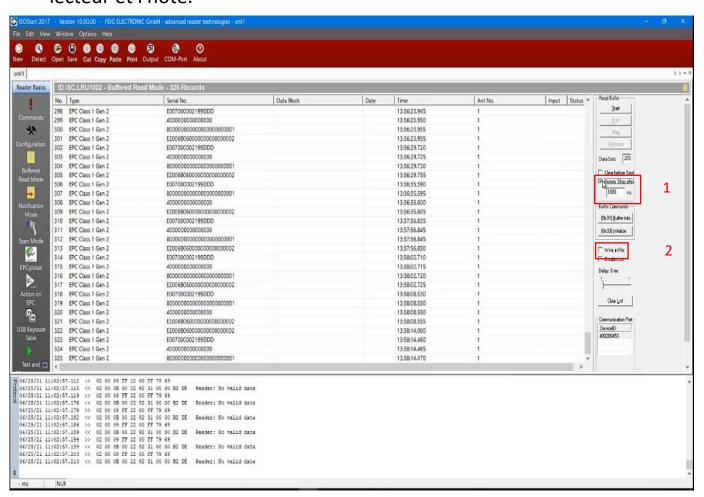


Figure 20: Buffered read mode





Ce Mode offre la possibilité de visualiser les différentes informations concernant les Radio-étiquettes qui sont dans le champ de lecture de l'antenne ainsi que la possibilité d'écrire ces derniers dans un fichier Excel du format .csv (Figure 2), il y on a ainsi le droit de manipuler le temps du Travail en ms (Figure 1)

Changement du Mode

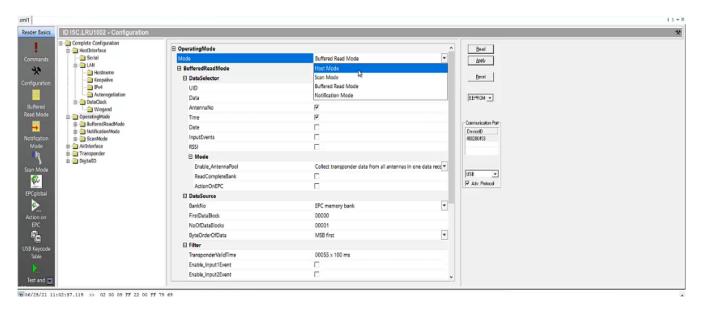


Figure 21: Configuration du mode

Cette interface consiste à changer le mode de configuration du Lecteur sans le redémarrer.





Host Mode

Les commandes d'hôte permettent l'échange de données entre un hôte et des transpondeurs via le lecteur tant que le transpondeur reste dans la zone de détection du lecteur.

1. EPC Global

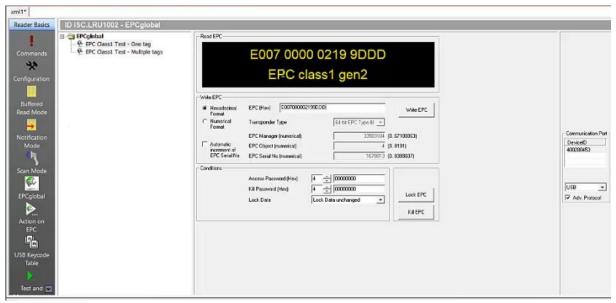


Figure 22: EPC Globale

Cette interface vous permet de spécifier et modifier l'EPC de chaque Tag avec la possibilité de le verrouiller (Lock EPC) ou le détruire(Kill EPC)





2. Phase de Lecture

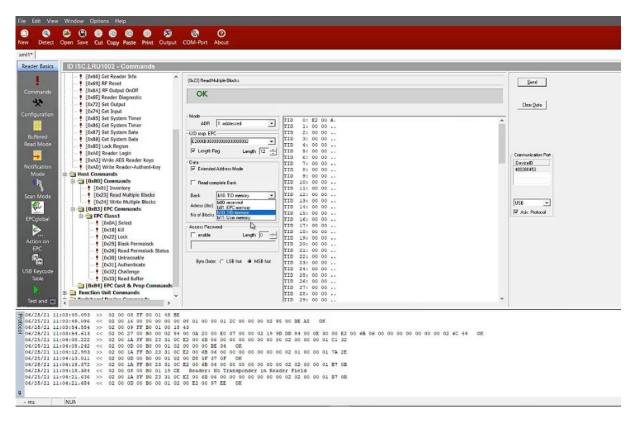


Figure 23: Read Multiple Blocks

En ce qui concerne la lecture dans le logiciel ISOStart, il vous affiche les différents blocs de données déjà stocké dans la deuxième section, Ainsi qu'il vous donne le droit de spécifier s'il est addressed (le cas où on va indiquer le nom du Tag) ou non-addressed (Concerne tous les Tags généralement)





3. Phase d'écriture

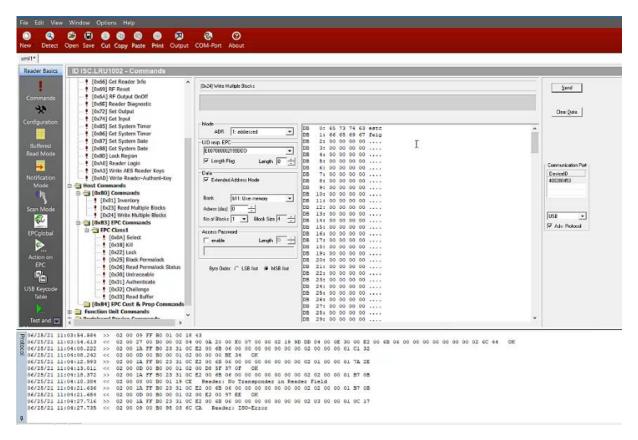


Figure 24: Write Multiple Blocks

Cette phase d'écriture consiste à insérer les données dans les Tags en choisissant un parmi les quatre modes (voir Page 32), on doit ainsi remplir le champ « Adress », Ce champ nous permet d'indiquer le ligne dont va commencer l'écriture, En plus, la spécification des nombres du bloc est exigé pour fournir l'espace convenable dans la mémoire, et finalement il est nécessaire de spécifier le mot de pass du Tag afin qu'on puisse modifier et éditer sur le Tag .





2. Application de gestion des tags

Pour la deuxième partie d'écriture, les balises qui nous sont fournies ne contiennent pas la zone mémoire utilisateur ce qui rend le logiciel ISOStart inutile pour l'écriture car nous ne pouvons même pas y écrire d'informations supplémentaires dans les balises. Alors comme solution nous avons pensé développer une application bureautique (logiciel) qui nous permettra de préciser pour chaque EPC (code produit électronique ou code de série de l'étiquette) des informations mais cette fois en utilisant la base de données : MSSQL Database.

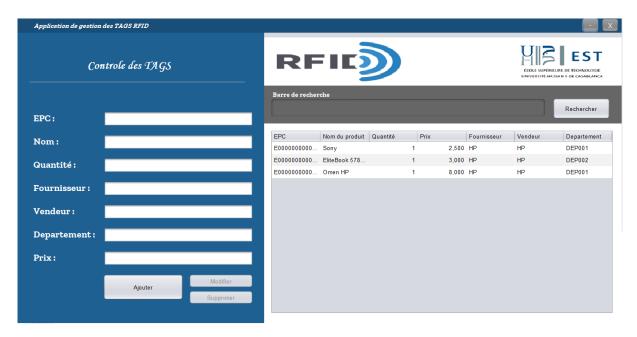


Figure 25:Interface de l'Applicarion desktop





La figure ci-dessus montre les différentes options qu'offre l'application :

- Un formulaire pour ajouter de nouvelles étiquettes ainsi que ses informations.
- Un tableau qui affiche ce qu'il y a dans la base de données pour mieux manipuler les choses
- La possibilité de modifier ou de supprimer les étiquettes de la base de données.
- La possibilité de rechercher des balises à l'aide de leurs EPC.

N.B: L'application n'était qu'une solution alternative. Certes, il n'est pas pratique de manière équitable pour la mémoire utilisateur embarquée dans les balises. Mais malheureusement les tags que nous avons utilisés pendant le stage ne contenaient pas cette zone, nous avons donc pensé à une autre solution.

3. Partie Code:

1. Assurer la connectivite

La première chose à faire pour démarrer le fonctionnement de notre application est d'assurer la connexion avec la base de données MSSQL

```
String driver="com.microsoft.sqlserver.jdbc.SQLServerDriver";
String url="jdbc:sqlserver://iotser.database.windows.net;databaseName=guest;user=iot;password=Admin123";

public ArrayList<Tag> tagList() {
    ArrayList<Tag> tagSlist = new ArrayList<>();

    try{
        Class.forName(driver);
        Connection con=DriverManager.getConnection(url);
```

Figure 26: Connexion avec base de donnees





Donc avant de commencer à programmer chaque fonction qui interagit avec la base de données La connexion ci-dessus doit obligatoirement être présente.

2. Inserer les donnees

```
private void addBtnActionPerformed(java.awt.event.ActionEvent evt) {
        String epc=epcField.getText();
        String name=nameField.getText();
        int quantity=Integer.parseInt(quantityField.getText());
       String manufacturer=manufacturerField.getText();
String vendor=VendorField.getText();
        String departement=depField.getText();
        float price=Float.parseFloat(priceField.getText());
            Class.forName(driver);
            Connection con=DriverManager.getConnection(url);
            Statement stm=con.createStatement();
            stm.executeUpdate("insert into Tags values('"+epo+"','"+name+"','"+quantity+"','"+manufacturer+"','"+vendor+"','"+departement+"','"+pri
            DefaultTableModel model =(DefaultTableModel)prodTable.getModel();
            model.setRowCount(0);
            show tag();
            JOptionPane.showMessageDialog(null, "insert success");
            con.close();
         catch (ClassNotFoundException ex) {
            ex.printStackTrace();
        } catch(SQLException e) {
        e.printStackTrace();
```

Figure 27: Insertion des donnees

Pour l'insertion des données, le code ci-dessus reprend les valeurs insérées dans les champs du formulaire puis après s'assuré de la connexion avec la base de données, il lance des requêtes SQL à écrire dans les tables du Database. à la fin, il affiche soit un message de réussite, soit un message d'erreur en spécifiant le type d'erreur.

3. Modifier les donnees





```
private void modifyBtnActionPerformed(java.awt.event.ActionEvent evt) (
            TODO add your handling code here:
             Class.forName(driver):
             Connection con=DriverManager.getConnection(url);
             int row = prodTable.getSelectedRow();
             String value = (prodTable.getModel().getValueAt(row,0).toString());
String query = "UPDATE Tags SET EPC=?,Name=?,Quantity=?,Manufacturer=?,Vendor=?,Departement=?,Price=? WHERE EPC='"+value+"'";
             try (PreparedStatement pst=con.prepareStatement(query)){
                 pst.setString(l,epcField.getText());
pst.setString(2,nameField.getText());
                 pst.setString(3,quantityField.getText());
                 pst.setString(4,manufacturerField.getText());
                 pst.setString(5,VendorField.getText());
                 pst.setString(6,depField.getText());
                 pst.setString(7,priceField.getText());
                 pst.executeUpdate();
                 DefaultTableModel model = (DefaultTableModel)prodTable.getModel();
                 model.setRowCount(0);
                 show_tag();
                 JOptionPane.showMessageDialog(null, "UPDATED !"+query);
          catch(Exception e) {
        JOptionPane.showMessageDialog(null, e);
```

Figure 28: Modification des donnees

Concernant la modification des balises, assurant toujours la connectivité, il faut d'abord sélectionner la balise à modifier pour que le programme puisse mémoriser la clé primaire afin de mettre à jour les données ultérieurement dans la base de données.

4. Supprimer les donnees

```
private void deleteBtnActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
     try{
       Class.forName(driver);
        Connection con=DriverManager.getConnection(url);
        int row = prodTable.getSelectedRow();
        String value = (prodTable.getModel().getValueAt(row,0).toString());
        String query="DELETE from Tags where EPC='"+value+"'";
        PreparedStatement pst= con.prepareStatement(query);
        pst.executeUpdate();
        DefaultTableModel model = (DefaultTableModel) prodTable.getModel();
        model.setRowCount(0);
        show_tag();
        JOptionPane.showMessageDialog(null, "DELETED !");
        } catch(Exception e){
    JOptionPane.showMessageDialog(null, e);
```

Figure 29: Suppresion des donnees





Le button supprimer suit le meme concept de la modification mais cette fois ci avec une requete SQL diffrent . il faut bien noter que a chaque fois on change des informations dans la base de donnees la fonctions show_tag(); rafraishit le tableaux .

5. Afficher les donnees

```
class Tag {
  private String epc, name, manufacturer, vendor, departement;
   private int quantity;
   private float price;
   public Tag(String epc, String name, String manufacturer, String vendor, String departement, int quantity, float price) {
    this.name = name:
   this.manufacturer = manufacturer;
   this.vendor = vendor;
   this.departement = departement;
   this.quantity = quantity;
   this.price = price;
   public String getepc() {
       return epc;
    public String getname() {
       return name;
    public String getmanufacturer() {
       return manufacturer:
   public String getvendor() {
       return vendor;
   public String getdepartement() {
       return departement;
   public int getquantity(){
      return quantity;
   public float getprice(){
       return price;
```

Figure 30: Classe du TAG

La classe TAG qui sera ensuite instancier pendant la creation des nouveaux tag.





```
public ArrayList<Tag> tagList(){
    ArrayList<Tag> tagList = new ArrayList<>();

    try{
        Class.forName(driver);

        Connection con=DriverManager.getConnection(url);

        String queryl="SELECT * FROM Tags";
        Statement st=con.oreateStatement();
        ResultSet rs = st.executeQuery(queryl);
        Tag tag;
        while(rs.next()){
        tag=new Tag(rs.getString("EPC"),rs.getString("Name"),rs.getString("Manufacturer"),rs.getString("Vendor"),rs.getString("Departement"),rs.
        tagsList.add(tag);
    }
        con.close();
} catch(Exception e) {
        JOptionPane.showMessageDialog(null, e);
}
        return tagsList;
}
```

La fonction tagList() qui recuppere les donnees du database, puis elle cree des nouveaux objets qui font refernece a la calsse TAG pour affecter des valeurs initiales aux données membres de la même classe.

```
public void show_tag() {
    ArrayList<Tag> list = tagList();
    DefaultTableModel model = (DefaultTableModel)prodTable.getModel();
    Object[] row = new Object[7];
    for (int i=0;i<list.size();i++) {
        row[0]=list.get(i).getepc();
        row[1]=list.get(i).getname();
        row[2]=list.get(i).getquantity();
        row[3]=list.get(i).getprice();
        row[4]=list.get(i).getmanufacturer();
        row[5]=list.get(i).getvendor();
        row[6]=list.get(i).getdepartement();
        model.addRow(row);
}</pre>
```

La fonction show_tag() prends les valeurs stocker dans la liste et puis il les affichent en respectant l'ordre des champs donnes a l'aide d'une boucle For .





6. Options supplementaires

```
private void closeBtnMouseEntered(java.awt.event.MouseEvent evt) { ...4 lines }

private void closeBtnMouseClicked(java.awt.event.MouseEvent evt) { ...5 lines }

private void closeBtnMouseClicked(java.awt.event.MouseEvent evt) { ...4 lines }

private void minimizeBtnMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    this.setExtendedState(NewJFrame.ICONIFIED);
}

private void minimizeBtnMouseEntered(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    minimizeBtn.setBackground(Color.cyan);
    minimizeBtn.setForeground(Color.black);
}

private void minimizeBtnMouseExited(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    minimizeBtn.setBackground(Color.darkGray);
    minimizeBtn.setForeground(Color.white);
}
```

Figure 31: Autres fonctionnalite de l'application

Les deux figures ci-dessus montrent du code supplémentaire pour les boutons pour quitter et minimiser l'application pendant les trois états (lorsque le curseur est sur le bouton, pour quitter le bouton, cliquez sur le bouton).

```
int x,y;
private void jPanellMousePressed(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    x=evt.getX();
    y=evt.getY();
}

private void jPanellMouseDragged(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int xx=evt.getXOnScreen();
    int yy=evt.getYOnScreen();
    this.setLocation(xx-x,yy-y);
}
```

Ce code offre la possibilité de changer le positionnement de l'application sinon l'application sera fixe (non positionnable à l'écran).





Conclusion

Pendant la période de notre stage de fin d'études, nous avons été chargé de programmer les tags RFID (lecture / ecriture)

Nous avons commencé dans un premier lieu par comprendre le contexte général de notre tache et identifier les différentes exigences de notre futur système. On a préparé par la suite notre planning de travail en respectant les priorités de nos besoins.

Après l'analyse des besoins et la conception du système, On a pu réaliser une application web qui permet de :

- Lire les tags RFID
- Ecrire des informations concernant les tags

Nous avons trouvé l'expérience très intéressante et enrichissante, puisqu'il nous a permis de découvrir un nouveau domaine de travail et de s'éloigner des projets traditionnels de gestion, et de découvrir la partie opérationnelle du monde professionnel de développement informatique.





Références

https://www.est-uh2c.ac.ma/recherche/laboratoires-de-recherche/r%C3%A9seaux-informatique-t%C3%A9l%C3%A9communications-multim%C3%A9dia/

https://www.industrie-techno.com/article/12-systemes-de-lecture-rfid.19491

https://www.nexess-solutions.com/fr/choisir-un-tag-rfid-pour-des-applications-industrielles/

https://www.atlasrfidstore.com/rfid-insider/types-of-memory-ingen-2-uhf-rfid-tags

https://www.cipam.com/produits/lecteurs-fixes-uhf-54/lecteur-lru1002-231

https://www.cipam.com/uploads/produits/documentation/Feig Elec tronic Lecteur UHF LRU1002.pdf

https://sbedirect.com/fr/blog/article/comprendre-la-rfid-en-10-points.html

https://skyrfid.com/Mid-Range RFID.php

https://fccid.io/PJMLRU1002A/User-Manual/manual-3394151

https://fr.scribd.com/document/470160783/H30810-8e-ID-B-pdf

https://www.feig.de

https://manualzz.com/doc/13673576/id-isostart-handbuch-v7.02obid#p4