

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université Mouloud Mammeri de Tizi-Ouzou

Faculté de Génie Electrique et d'Informatique

Département d'Informatique



Memoire

De fin d'études

En vue de l'obtention du diplôme de Master en Informatique

Option : Réseaux, Mobilités et Systèmes Embarqués

Thème

**Gestion des commandes par les
Tags RFID**

Proposé et dirigé par :

M^{elle} CHAMEK Linda

Réalisé par :

M^{elle} ALILI Thanina

M^{elle} AIT IZEM Radhia

Promotion : 2013/2014

Dédicaces

A l'homme de ma vie, mon exemple éternel, mon soutien moral et source de joie et de bonheur, celui qui s'est toujours sacrifié pour me voir réussir ; à toi mon père.

A la lumière de mes jours, la source de mes efforts, la flamme de mon cœur, ma vie et mon bonheur ; maman que j'adore.

*Aux personnes qui m'ont soutenue tout au long de ce projet, à mes deux frères **Amazigh** et **Ithri**, je dédie ce travail dont le grand plaisir leurs revient en premier lieu pour leurs conseils, aides, et encouragements.*

*Aux personnes qui m'ont toujours aidé et encouragé, qui étaient toujours à mes côtés, et qui m'ont accompagné durant mon chemin d'études supérieures, mes aimables amies **Soumia** et **Lydia**, mon binôme **Radhia** et sa famille.*

Thanina

Dédicaces

Je dédie ce modeste travail à :

*A mes parents .Aucun hommage ne pourrait être à la hauteur de
l'amour Dont ils ne cessent de me combler. Que dieu leur procure
bonne santé et longue vie.*

*A mes frères **Amine** et **Amirouche**,*

*A toute ma famille, tous mes amis en particulier **Lila**,*

*A mes cousines **Celia** et **Ryma**,*

*A mon binôme **Thanina** et sa famille,*

*A **Mouri**,*

*Et à tous ceux qui ont contribué de près ou de loin pour que ce
projet soit possible, je vous dis merci.*

Radhia

Remerciements

C'est avec un grand plaisir que nous réservons ces quelques lignes en signe de gratitude et de profonde reconnaissance à tous ceux qui, de près ou de loin, ont contribué à la réalisation et l'aboutissement de ce travail.

Nous tenons tout d'abord à remercier Melle CHAMEK Linda pour son soutien, son sérieux sa gentillesse et surtout pour son aide précieuse tout au long de l'élaboration de ce travail.

Un grand merci à RABIA Nouredine de nous avoir envoyé le tag sans lequel notre travail n'aurait pas été complet, et pour son aide, son soutien et ses conseils.

Nous tenons aussi à exprimer l'honneur qui nous est fait par les membres du jury, en acceptant de juger notre travail.

Nous nous acquittons, enfin, volontiers d'un devoir de gratitude et de remerciements à tous nos enseignants pour la qualité de l'enseignement qu'ils ont bien voulu nous prodiguer durant nos études afin de nous fournir une formation efficiente.

Résumé

Dans ce projet, notre tâche est de montrer la façon de gérer les commandes d'un restaurant en utilisant la technologie RFID. Une application Android est mise à la disposition du client pour qu'il puisse effectuer sa commande et l'envoyer au tag. La transmission de données se fera en utilisant le NFC qui assure la communication du tag au Smartphone et vice versa.

Tout au long du projet, nous étions amenés à développer une application Android sur l'environnement eclipse.

Afin de réaliser une simulation, nous avons utilisé une application disponible sur le Google store.

Mots-clés : Smartphone, Android, NFC, RFID, systèmes embarqués.

Projet de fin d'études Gestion des commandes par les tags RFID

Abstract

In this project, our task is to show how to handle the controls of a restaurant using RFID technology. An Android application is available to the customer so that he can make the order and send it to the tag. The data will be using NFC that enables communication from the tag to the smartphone and vice versa.

Throughout the project, we were required to develop an Android application on Eclipse environment.

To perform a simulation, we used an application available on the Google store.

Keywords: Smartphone, Android, NFC, RFID, embedded systems.

Graduation Project Order Management using RFID tags

Liste d'abréviations

A

ADT : **A**ndroid **D**evloppement **T**ools

API : **A**pplication **P**rogramming **I**nterfaces

ASIC : **A**pplication **S**pecific **I**ntegrated **C**ircuits

AVD : **A**ndroid **V**irtual **D**evice

B

BF : **B**asse **F**requency

E

EDI: *Electronic Data Interchange*

EPC: **E**lectronic **P**roduct **C**ode

F

FPGA: **F**ield **P**rogrammable **G**ate **A**rray

H

HF : **H**igh **F**requency

HI-FI: **H**igh **F**idelity

I

IFF : **I**dentifie **F**riendly**F**oe

IHM: **I**nterface **H**omme **M**achine

ISO: **I**nternational **S**tandard **O**rganisation

J

JDK : **J**ava **D**evloppement **K**it

JVM : **J**ava **V**irtual **M**achine

L

LCD: **L**iquid **C**rytal **D**isplay

LED: **L**ight **E**mitter **D**iode

M

MIT: **M**assachusetts **I**nstitute of **T**echnology

N

NFC: Near Field Communication

O

OS: Operating System

P

PC: Personal Computer

PDA: Personal Digital Assistant

R

RF: Radio Frequency

RFID: Radio Frequency Identification

RTF: Reader Talks First

S

SDK: Software Development Kit

T

TTF: Tag Talks First

U

UII : Unique Item Identifier ou code

UHF: Ultra Haute Fréquence

W

Wi-Di: WI-FI Direct

WORM: Write Once Read Multiple

Sommaire

Introduction	1
---------------------------	----------

Chapitre I : Généralités sur les systèmes embarqués

I.1. Définition et domaine d'utilisation	1
I.1.1. Définition	1
I.1.2. Domaines d'applications	3
I.2. Composition d'un système embarqué	5
I.3. Caractéristiques spécifiques	7
I.4. Systèmes d'exploitation embarqués	9
I.4.1 Symbian OS	9
I.4.2. Palm OS	10
I.4.3. Ios (iPhone OS).....	10
I.4.4. Windows mobile, Windows CE et Windows Phone 7.....	10
I.4.5. BlackBerry OS.....	11
I.4.6. Bada	11
I.4.7. Androïde.....	11
a) Versions.....	12
b) Architecture du système d'exploitation Android.....	15
c) Moteur d'exécution d'Android	16
Conclusion	19

Chapitre II : Généralités sur les RFID

Introduction	20
II.1. Définition de la RFID	20
II.2. Historique de la RFID	21
II.3. Fonctionnement	22
II.4. Composants et fonctionnement du système	24
II.4.1. Le tag (étiquette).....	24
II.4.1. 1. Les différents types de tags et leurs spécificités techniques	25
II.4.2. Le lecteur	27

II.4.3. L'intergiciel (middleware).....	29
II.5. Communication tag	30
II.6. Les gammes de fréquences RFID.....	32
II.6.1. La RFID dans le spectre radio	32
II.6.2. Les tags RFID UHF, HF, LF	33
II.7. Standard RFID.....	34
II.7.1. Les standards de données	35
II.7.2. Les standards de technologie	35
II.8. Applications de la RFID	36
II.8.1. L'application aux entreprises.....	36
II.8.1.1 Expédition.....	36
II.8.1.2 Réception	37
II.8.1.3. En transit	37
II.8.1.4. Local.....	37
II.9. Avantages et inconvénients.....	38
II.9.1. Avantages	38
II.9.2. Inconvénients	40
II.10. Communication en champ proche (NFC)	42
II.10.1. Modes de fonctionnement du NFC	43
II.10.2 Quelques exemples d'usages	44
Conclusion	45

Chapitre III : Conception de l'application

Introduction.....	46
III. Présentation et objectif de l'application	46
III.1.1. Présentation	46
II.1.2. Objectif	46
III.2. Description du système	46
III.3. Description des composants du système.....	48
III.3.1. Le Tag RFID	48
III.3.2. Le Lecteur	48
III.3.3. Le serveur de base de données intégré.....	49
III.3.4. Rôle des activités, classe	50
Conclusion	51

Chapitre IV : Réalisation de l'application

Introduction	52
IV.1 environnement et outils de développement	52
IV.1.1. matériel	52
IV.1.1.1. le tag RFID	52
IV.1.1.2. Le Smartphone	53
IV.1.2. logiciel	53
IV.1.2.1. le langage JAVA	53
IV.1.2.2. l'environnement Andoid	56
IV.1.2.3. SQLite serveur de base de données intégré	59
IV.2. Présentation de notre application	59
IV.2.1. L'application client	60
IV.2.2. partie de base de donnée	64
Conclusion	64
Conclusion générale	65

Liste des figures

Chapitre I

Figure I.1 Composition d'un système embarqué	2
Figure I.2 Système de contrôle des gouvernes de l'Airbus A320	5
Figure I.3 Système électronique embarqué	6
Figure I.4 Aperçu des versions d'Android.....	12
Figure I.5 Architecture d'Android	15
Figure I.6 Architecture Java.....	17
Figure I.7 Etapes de compilation et d'exécution d'un programme Android	18

Chapitre II

Figure. II.1 principaux éléments d'un système	23
Figure. II.2 Composants d'un système RFID	24
Figure. II.3 Tag RFID.....	25
Figure. II.4 Types de tags RFID.....	26
Figure. II.5 Lecture d'un tag	27
Figure. II.6 la communication entre un transpondeur et une station de base.....	31
Figure.II.7 Fréquences de la RFID	33
Figure. II.8 Exemple d'utilisation pour les fréquences RFID	34
Figure.II.9 EPC 96 bits	35
Figure.II.10 Distributeur de billets de train	42

Chapitre III

Figure III.1 modèle embarqué	47
Figure III.1 Architecture de notre système	47
Figure III.2 fonctionnement du lecteur	49
Figure III.3 diagramme de l'application.....	50

Chapitre IV

Figure IV.1 smart tag AIOI	52
Figure IV.2 Samsung Galaxy S2 Cityzi.....	53
Figure IV.3 interface eclipse	56
Figure IV.4 SDK Manager	57
Figure IV.5 émulateur Android.....	58
Figure IV.6 fenêtre principale.....	60
Figure IV.7 fenêtre menu	61
Figure IV.8 liste des desserts.....	62
Figure IV.9 choix du client.....	63

Liste des Tableaux

Tableau II.1 Standards ISO	36
Tableau III.1 table plat.....	49

Introduction générale

INTRODUCTION GENERALE

Les progrès conjoints de la microélectronique, des technologies de transmission sans fil et des applications embarquées ont permis de produire à coût raisonnable des terminaux mobiles de haute technologie comme les Smartphones et les tablettes PC.

Mis à part le développement proprement dit de l'application, la première étape consistait à nous familiariser avec la technologie RFID et l'environnement Android, puis de choisir les outils conviviaux et envisageables à l'aboutissement du projet et par la suite, nous entamerons la modélisation et le développement de l'application.

Ce rapport peut ainsi être subdivisé en quatre parties. La première partie « généralités sur les systèmes embarqués » consiste à la présentation des systèmes embarqués en se basant sur les systèmes mobiles. Dans la seconde partie « généralités sur les RFID » présentons le fonctionnement et les composants de la RFID. le fonctionnement de l'application sera décrit dans la troisième partie « conception de l'application » .Enfin dans la dernière partie « réalisation de l'application » nous verrons les outils logiciels et matériels utilisés pour le développement de notre application ainsi que ses différentes interfaces.

CHAPITRE I

Généralités sur les systèmes embarqués

Introduction

Aujourd'hui, les applications embarquées sont de plus en plus variées et elles prendront plus de place dans notre vie quotidienne de demain. Réservés, il y a quelques années, aux applications industrielles, les systèmes embarqués font leur apparition dans beaucoup d'autres secteurs tels que le transport, le multimédia, les consoles de jeux. En termes de complexité, les systèmes embarqués couvrent un large spectre allant du simple microcontrôleur (contrôle de la fermeture/ouverture d'une vanne), jusqu'aux systèmes répartis (contrôle du trafic aérien). Les enjeux économiques et les intérêts scientifiques liés aux systèmes embarqués sont multiples. C'est la raison pour laquelle on assiste, depuis les années soixante-dix, à une profusion de langages, de méthodes, d'algorithmes, de protocoles de communication, pour l'embarqué. Dans ce premier chapitre nous allons présenter les systèmes embarqués de façon générale, puis en se basant sur les systèmes mobiles.

I.1. Définition et domaine d'utilisation**I.1.1. Définition**

Il est difficile de donner une définition précise d'un système embarqué, mais pour commencer, on peut citer quelques définitions qui nous clarifient cette situation :

1. Un système embarqué est une combinaison de matériels et logiciels permettant de remplir une ou plusieurs fonctions spécifiques avec des contraintes plus ou moins sévères tel que la consommation, la température, la taille, les performances ...etc. [3]
2. Un système embarqué est susceptible d'être utilisé dans un environnement matériel de faibles performances (si l'on compare au PC de bureau d'aujourd'hui). Si l'ajout de quelques Mo de mémoire sur un PC de bureau n'a pas une grande influence sur le budget d'un utilisateur, le gain de quelques Mo pour un produit de grande consommation (téléphone, équipement auto mobile, organisateur personnel) à une énorme influence sur le cout final. [21]

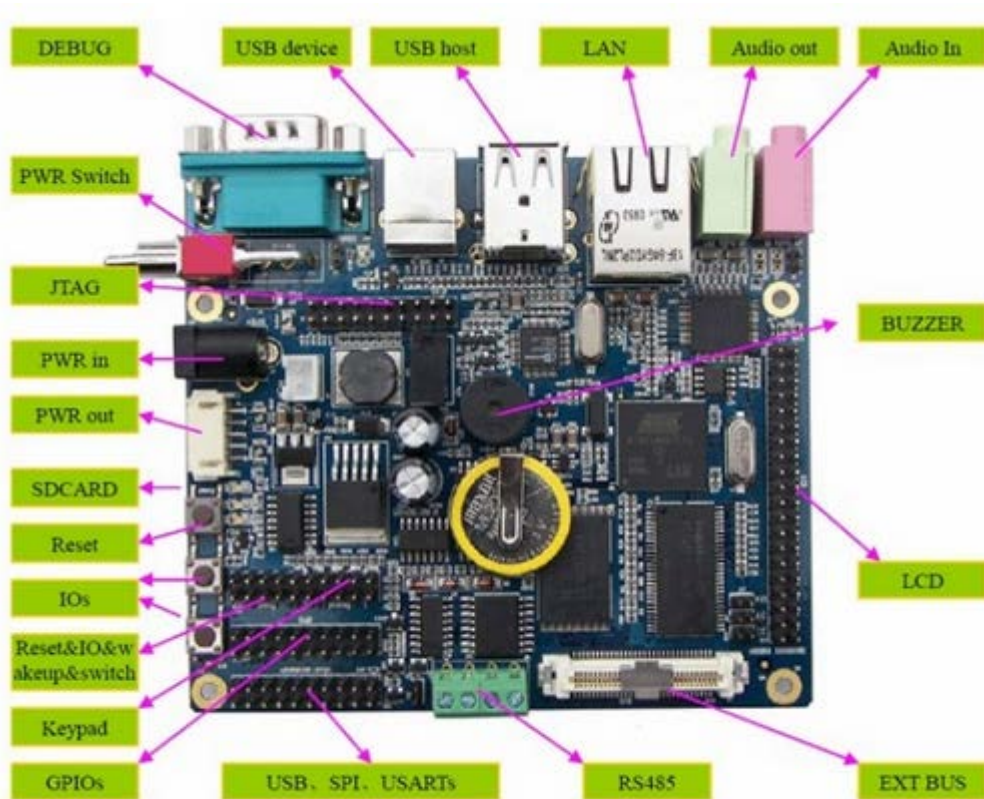


Figure I.1. Composition d'un système embarqué

- **Globalement**

Un système embarqué est un système complexe qui intègre du logiciel et du matériel conçus ensemble afin de fournir des fonctionnalités données. Il contient généralement un ou plusieurs microprocesseurs destinés à exécuter un ensemble de programmes définis lors de la conception et stockés dans des mémoires. Le système matériel et l'application (logiciel) sont intimement liés et immergés dans le matériel et ne sont pas aussi facilement discernables comme dans un environnement de travail classique de type ordinateur de bureau PC (Personal Computer).

Un système embarqué est autonome et ne possède pas des entrées/sorties standards tels qu'un clavier ou un écran d'ordinateur. Contrairement à un PC, l'interface IHM (Interface Homme machine) d'un système embarqué peut être aussi

simple qu'une diode électroluminescente LED (Light Emitter Diode) qui clignote ou aussi complexe qu'un système de vision de nuit en temps réel ; les afficheurs à cristaux liquides LCD (Liquid Crystal Display) de structure généralement simple sont couramment utilisés.

Afin d'optimiser les performances et la fiabilité de ces systèmes, des circuits numériques programmables FPGA (Field Programmable Gate Array), des circuits dédiés à des applications spécifiques ASIC (Application Specific Integrated Circuits) ou des modules analogiques sont en plus utilisés.

Le logiciel a une fonctionnalité fixe à exécuter qui est spécifique à une application. L'utilisateur n'a pas la possibilité de modifier les programmes. Bien souvent, il n'a pas conscience d'utiliser un système à base des microprocesseurs.

Les systèmes embarqués sont désormais utilisés dans des applications diverses tels que le transport (avionique, espace, automobile, ferroviaire), dans les appareils électriques et électroniques (appareils photo, jouets, postes de télévision, électroménager, systèmes audio, téléphones portables), dans la distribution d'énergie, dans l'automatisation, ..., etc.

I.1.2. Domaines d'applications

Jusqu'à une date récente, les systèmes temps réel et embarqués étaient destinés quasi-exclusivement aux applications de contrôle/commande de procédés (ou de phénomènes) physiques (tels que des laminoirs ou des usines de fabrication de voitures) et applications militaires. La notion de temps réel se confondait alors avec ce type d'applications. Le développement de l'informatique aidant, les applications temps réel et embarqués sont présentes aujourd'hui dans des domaines très variés comme le montre la liste suivante, même si elle n'est pas exhaustive : [22]

- Télécommunications (transport de la parole, systèmes de commutation, ...),
- Domaine médical (assistance et supervision de malades, ...),
- Contrôle de différents équipements dans les voitures, bus, trains, avions, ...,
- Contrôle et régulation de trafic en milieu urbain,

- Guidage de véhicules en milieu urbain,
- Industrie (contrôle/commande de procédés manufacturiers ou continus, ...),
- Domaine militaire (suivi de trajectoires de missiles, ...)
- Aérospatial (suivi de satellites, simulation de vols, pilotage automatique, ...),
- Multimédia (transport d'images et de voix, téléconférences, ...),
- Finance (suivi du cours des devises et actions, ...),
- Loisirs (consoles de jeu, ...),
- Domotique (sécurité d'habitations, ...),
- Contrôle/commande d'appareils électroménagers.

Exemple

Le système de contrôle des gouvernes de l'Airbus A320 est un système embarqué dont les exigences sont :

- le système doit traduire en ordres de déflexion des gouvernes, les ordres de pilotage venant du pilote ou du pilote automatique (facteur de charge)
- Le système doit maintenir l'avion dans son domaine de vol quelles que soient les commandes du pilote ou du pilote automatique
- Compte tenu de la dynamique de l'avion, les ailerons et la gouverne de direction doivent être asservis à une période minimale de 10ms
- Compte tenu de la dynamique de l'avion, la gouverne de profondeur doit être asservie à une période minimale de 30ms
- La perte de contrôle à la fois des ailerons et des spoilers est catastrophique (taux de panne de 10^{-9})
- La perte de contrôle de la profondeur est catastrophique (taux de panne de 10^{-9})
- La perte de contrôle des spoilers, des ailerons, de la direction et de la profondeur ne doit pas être causée par une panne unique

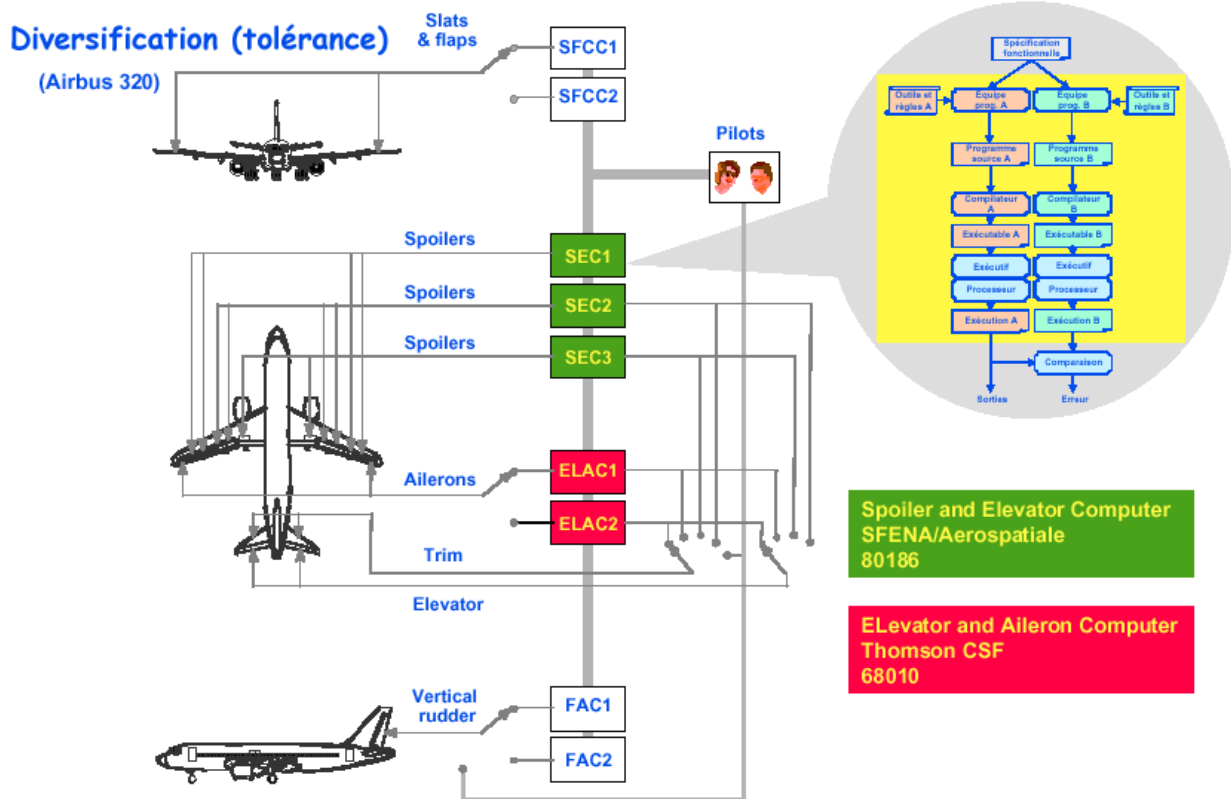


Figure I.2. Système de contrôle des gouvernes de l'Airbus A320

I.2. Composition d'un système embarqué [22]

Un système embarqué est composé de trois couches : Application, OS, matériels.

C'est la même architecture en couche qu'un ordinateur. Chaque couche à la même fonctionnalité qu'un système normal. Mais, il y a des différences de sous composants de chaque couche.

Les deux premières couches forment la partie « soft » et la troisième couche est la couche « hard », il s'agit d'un ensemble d'éléments physiques cohabitant sur une faible surface et employés pour le traitement de données. Ces composants matériels sont limités et peuvent avoir des composants dédiés et des traitements spécifiques.

Equipements permanents :

- UC : unité centrale formée de :
- CPU : microprocesseur(s) ou microcontrôleurs

- RAM : mémoire centrale

Equipements supplémentaires :

Ceux-ci dépendent de l'application dédiée :

- La mémoire de masse : (Disque dur, mémoire flash, utilisation de la ROM, disque à distance)
- Les entrées : (Capteurs, clavier, lecteurs de tags)
- Les sorties : (écrans, systèmes d'alarme, imprimantes)
- IHM : dispositif qui sert à communiquer entre l'humain et la machine. Un exemple réaliste de IHM est l'écran avec les dispositifs « TouchScreen »

Quelle que soit la nature et la complexité du système, on décompose un système embarqués en :

- système contrôlé
- système de contrôle

Le système contrôlé : environnement (procédé) équipé d'une instrumentation qui réalise l'interface avec le système de contrôle

Le système de contrôle : éléments matériels (microprocesseurs...) et logiciels dont la mission est d'agir sur le procédé via les actionneurs en fonction de l'état de ce procédé indiqué par les capteurs de manière à maintenir ou conduire le procédé dans un état donné.

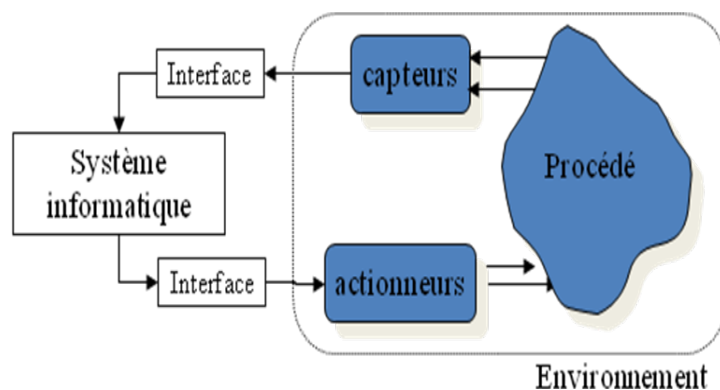


Figure I.3. Système électronique embarqué

Un système électronique embarqué ou enfoui est un élément constitutif d'un système plus complexe pour lequel il rend des services bien précis (contrôle, surveillance, communication...). Il est constitué de parties matérielles et logicielles qui sont conçues spécifiquement pour réaliser une fonction dédiée.

- **Comparaison aux systèmes informatiques standards : [3]**

Informatique :	Embarqué :
<ul style="list-style-type: none"> ▪ Processeur standard <ul style="list-style-type: none"> • Multiples unités fonctionnelles (flottant) • Vitesse élevée (> GHz) • Consommation électrique élevée • Chaleur • Taille ▪ MMU (mémoire virtuelle) ▪ OS ▪ Cache ▪ Grand nombre de périphériques 	<ul style="list-style-type: none"> ▪ Processeur dédié (contrôleur) <ul style="list-style-type: none"> • Architecture adaptée • Vitesse faible (~200 MHz) • 8-32bits : mémoire limitée • Basse consommation • Petite taille, grand volume => faible coût ▪ Processeur DSP (traitements) <ul style="list-style-type: none"> • Très puissants ▪ Quelques Mo de mémoire ▪ RTOS

I.3 Caractéristiques spécifiques [3]

Les systèmes embarqués fonctionnent généralement en Temps Réel (TR) : les opérations de calcul sont alors faites en réponse à un événement extérieur (interruption matérielle). La validité et la pertinence d'un résultat dépendent du moment où il est délivré. Une échéance manquée induit une erreur de fonctionnement qui peut entraîner

soit une panne du système (plantage), soit une dégradation non dramatique de ses performances.

Lorsque les systèmes embarqués sont utilisés dans les produits de grande consommation, ils sont fabriqués en grande série. Les exigences de coût se traduisent alors en contraintes sur les différentes composantes du système : utilisation de faibles capacités mémoires et de petits processeurs (4 bits ou 8 bits), mais en grand nombre. Ainsi, les systèmes embarqués sont particulièrement sensibles au coût de production. Il existe des applications dans lesquelles les contraintes de coût de production et de maintenance ont une importance de même niveau que les performances envisagées.

Dans les systèmes embarqués autonomes, la consommation d'énergie est un point critique pour le coût. En effet, une consommation excessive augmente le prix de revient du système embarqué, car il faut alors des batteries de forte capacité.

Ainsi, un système embarqué :

- Dispose de ressources limitées
- Ne possède pas toujours de systèmes de fichiers
- Doit être le moins cher possible
- Ne doit pas consommer de l'énergie inutilement
- Une puissance de calcul limitée
- Une capacité de communication limitée
- Exécute un logiciel dédié aux fonctionnalités spéciales

Faible cout :

Les concepteurs des systèmes embarqués cherchent des solutions optimales entre le prix et la performance pour proposer des produits aux marchés à la portée de toute personne n'ayant pas les moyens pour acheter des produits performants et chers.

C'est pour cette raison que les ressources utilisées sont minimales et qu'un système embarqué n'as que peu de mémoire.

Faible consommation :

Les systèmes embarqués utilisent une batterie d'emmagasinassions d'énergie. Pour rester autonome le plus longtemps possible, il faut bien gérer la consommation.

Une consommation excessive augmente le prix de revient du système embarqué car il faut alors des batteries de forte capacité.

Faible encombrement et faible poids :

Il est très important de minimiser la taille et le poids pour un système embarqué. Ce qui revient à ce que les composants électroniques (analogiques et/ou numériques) doivent cohabiter sur une faible surface.

Fonctionnement en Temps Réel :

Généralement, les applications embarquées doivent répondre rapidement aux événements internes ou externes. Dans des applications de systèmes de contrôles, le résultat peut être néfaste au système s'il ne réagit pas dans l'immédiat à un événement du système.

Environnement :

Un système embarqué est soumis à de nombreuses contraintes d'environnement auxquelles il doit s'adapter et fonctionner avec. On peut citer :

- La température
- L'humidité
- Les vibrations
- Les chocs
- Les variations d'alimentation, les interférences RF, la corrosion, l'eau, le feu, les radiations...etc.

I.4. Systèmes d'exploitation embarqués

Les principaux systèmes d'exploitation sont :

I.4.1. Symbian OS

Est le système d'exploitation le plus utilisé pour des Smartphones et PDA, qui a été conçu par Symbian Ltd. Symbian OS fournit les fonctionnalités essentielles du système d'exploitation, notamment le cœur du système, ainsi que les API communes et une interface utilisateur de référence. Il a été adopté par différents fabricants de téléphones portables de 2G et 3G (Nokia, Sony Ericsson, Samsung etc.). Les

principales interfaces utilisateurs sont S60 série (pour clavier numérique), UIQ (pour écran tactile), S80 série (pour clavier alphanumérique), etc.

I.4.2. Palm OS

(Aussi connu sous le nom de Garnet OS) est un système d'exploitation embarqué développé initialement par U.S. Robotics Corp., propriétaire de la société Palm Computing, Inc. A partir de 1995, pour des PDA (assistance personnels numériques) en 1996. L'interface utilisateur graphique de Palm OS est prévu pour être utilisée avec un écran tactile. Il est fourni avec une suite d'applications de base pour gestionnaire d'informations personnelles. Plus tard les versions de cet OS ont été améliorées pour fonctionner sur des Smartphones. Plusieurs autres sociétés ont reçu une licence pour construire des appareils électroniques fonctionnant avec Palm OS. ACCESS est le propriétaire de la version actuelle de la licence et l'a renommé en Garnet OS.

En 2007, ACCESS a sorti le successeur de Garnet OS, appelé Access Linux Platform. En 2009, la principale société utilisant la License de Palm OS, Palm, Inc., informa qu'elle passera de Palm OS à Web OS pour ses prochains appareils électroniques.

I.4.3. Ios (IPhone OS)

Anciennement appelé « IPhone OS » est un système d'exploitation conçu par Apple, qui est dérivé de Max OSX. Il fonctionne sur IPhone et IPadTouch et a pris une part significative du marché.

I.4.4. Windows mobile, Windows CE et Windows Phone 7

Sont développés par Microsoft pour les Smartphones et Pocket PC. Il propose des applications basiques comme email, Internet, chat et multimédia, etc., les applications tierces développées avec .NET Compact Framework sont également supportées.

Le 15 février 2010 Microsoft a lancé un nouveau système d'exploitation pour mobile, Windows Phone 7. Il intègre des fonctionnalités media sociaux tel Facebook et Twitter. Comme Windows Phone 7 est une nouvelle plate-forme, il n'existe aucune compatibilité avec les applications Windows Mobile.

I.4.5. BlackBerry OS

Est un système d'exploitation qui fonctionne sur le Smartphone BlackBerry. Il permet aux développeurs de mettre en place des applications en utilisant les APIs BlackBerry, mais toute application doit être signée numériquement par le compte RIM du développeur.

I.4.6. Bada

Est le système d'exploitation pour Smartphones haut de gamme et milieu de gamme de Samsung basé sur le système d'exploitation propriétaire SHP OS. Le système d'exploitation est, au 27 avril 2011, le 3eme en part de marché en France et à vendu plus d'un million de Smartphones.

Tous les téléphones sous Bada ont un nom commençant par Samsung Wave.

I.4.7. Androïde [7]

Est un système d'exploitation open source utilisant le noyau Linux, pour Smartphones, PDA et terminaux mobiles conçu par Androïde, une startup rachetée par Google, et annoncée officiellement le 5 novembre 2007. D'autres types d'appareils possédant ce système d'exploitation existant, par exemple des téléviseurs et des tablettes.

Androïde est un système d'exploitation fondé sur un noyau Linux, il comporte une interface spécifique, développée en JAVA, les programmes sont exécutés via un interpréteur JIT, toutefois il est possible de passer outre cette interface, en programmant ses applications en C, mais le travail de portabilité en sera plus important. Si la majorité des périphériques Androïde sont basés sur l'architecture

ARM, l'hétérogénéité des versions de des coprocesseurs peut varier grandement d'un constructeur à l'autre.

Androïde a été conçu pour intégrer au mieux des applications existantes de Google comme le service de courrier Gmail, celui de cartographie, Google Maps, ou encore Google Agenda, Google Talk, Youtube. Un accent particulier est mis sur la géolocalisation avec Google latitude de la météo correspondant à la ville la plus proche disponible sur le menu principal (vu sur HTC Sense).

a) Versions d'Android :

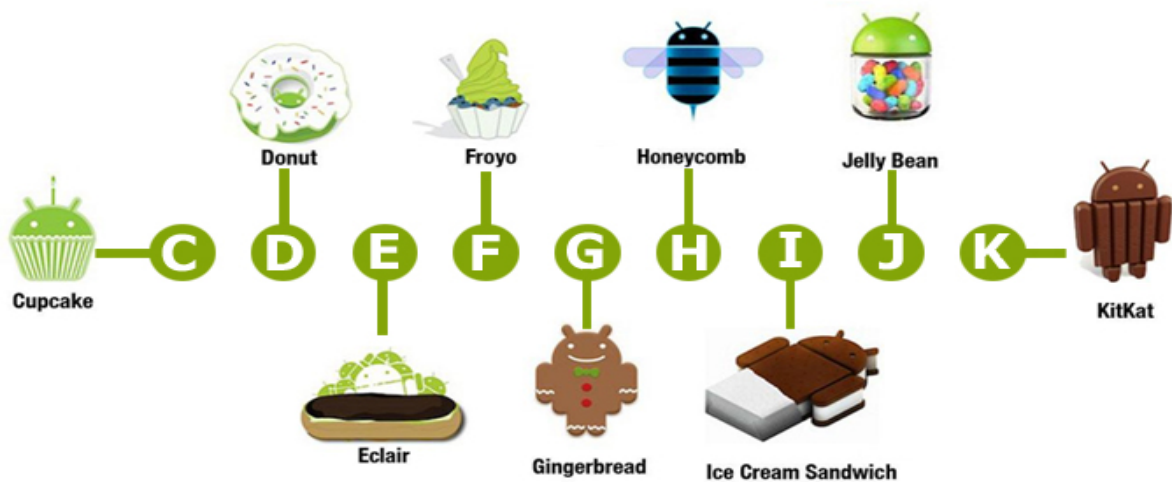


Figure I.4: Aperçu des versions d'Android

- Android 2.3 Gingerbread

La version 2.3 Gingerbread est sortie en décembre 2010. Elle a apporté de nombreux changements qui ont fait de l'Android un bien meilleur système d'exploitation. Les améliorations les plus marquantes incluent le support pour de multiples capteurs photos, l'amélioration de la fluidité, un meilleur clavier, le support NFC et des améliorations au niveau de la batterie.

Le Samsung Nexus S est le premier téléphone à fonctionner avec Android 2.3 Gingerbread. On retrouve aussi d'autres dispositifs populaires comme le Samsung Galaxy S2, HTC Sensation et le Motorola DroidRazr.

- **Android 3.0 Honeycomb**

Le mois de février 2011, la version Honeycomb est sortie. Celle-ci fut la première version Android optimisée pour les tablettes. A cette époque, c'était aussi la seule version capable de supporter les tablettes. Elle a notamment apporté de nouvelles fonctionnalités comme l'interface utilisateur surnommée «Holo». Elle est aussi dotée d'un clavier repensé, d'une fonction multitâche simplifiée, plusieurs onglets possibles dans le navigateur Web, des widgets redimensionnables, et d'autres ajouts majeurs.

Etant donné que seules les tablettes pouvaient fonctionner sous Android 3.0 Honeycomb, il y avait donc très peu d'appareils compatibles, et la plupart d'entre eux n'ont pas tardé à être mis à jour vers les versions les plus récentes. Néanmoins, on citera quelques dispositifs populaires fonctionnant sous Android Honeycomb, comme le Motorola, AsusEeePad Transformer, et le Samsung Galaxy Tab 10.1.

- **Android 4.0 IceCream Sandwich**

La version IceCream Sandwich est apparue en octobre 2011. Fortement inspirée d'Honeycomb, elle fut la première version à se retrouver à la fois sur des Smartphones et des tablettes. Elle s'est également vêtue de l'interface utilisateur «Holo», et a apporté des améliorations ainsi que de la stabilité au système d'exploitation. La création de dossiers est aussi devenue plus simple, tout comme le débloqué du visage. Elle a également apporté une nouvelle application appareil photo.

Pour les dispositifs populaires ayant été lancé avec AndroidIceCream Sandwich, on retrouve le plus populaire Samsung Galaxy S3, ensuite HTC One X, Samsung GalaxyNexus, Motorola DroidRazr HD et le Sony Xperia T.

- Android 4.1 Jelly Bean

Sortie en juillet 2012, la version Android 4.1 Jelly Bean est une mise à jour importante, apportant une foule de nouveautés et d'améliorations. Elle a apporté de nouvelles fonctionnalités comme le Project Butter, visant à rendre Android encore plus souple, l'extension des notifications, le Google Now, un assistant personnel intelligent, Google Chrome comme navigateur par défaut, la reconnaissance vocale sans connexion Internet, etc...

Les dispositifs les plus populaires fonctionnant sous Android 4.1 Jelly Bean sont la Nexus 7, le Samsung Galaxy Note 2, HTC One, le Sony Xperia Z et le LG Optimus G Pro.

- Android 4.2 Jelly Bean

En novembre 2012, la version Android 4.2 Jelly Bean est sortie. Celle-ci s'inspire bien entendu de la version précédente, et s'inscrit ainsi dans une continuité de cette dernière. C'est d'ailleurs pour cela qu'elle partage le même «nom de code». Mais celle-ci apporte encore de nombreuses et nouvelles caractéristiques, comme le nouveau clavier avec le *gesturetyping* permettant d'écrire avec le clavier rien qu'en glissant le doigt dessus, un nouveau *lockscreen* ou écran de verrouillage, la possibilité d'avoir plusieurs comptes utilisateurs (sur tablette)...

La grande caractéristique de la version Android 4.2 Jelly Bean est aussi l'introduction de Photosphère, un mode photo panoramique à 360° dans le genre de Google Maps Street View. Sinon, les dispositifs populaires embarquant la version Android 4.2 Jelly Bean sont entre autres le LG Nexus 4, le Samsung Nexus 10, et le légendaire Samsung Galaxy S4.

- Android KitKat 4.4

Sortie en novembre 2013, la version Android 4.4 (KitKat) apporte des nouveautés majeures qui sont une empreinte mémoire plus faible, un mode immersion,

des styles translucides, l'impression d'écran, plusieurs frameworks (printing, storage, transitions) et une WebView Chromium.

Parmi les dispositifs populaires ayant été lancé avec Android KitKat 4.4 on trouve Nexus 4, Nexus 5 et Nexus 7, Samsung Galaxy S4 Google Edition.

b) Architecture du système d'exploitation Android :

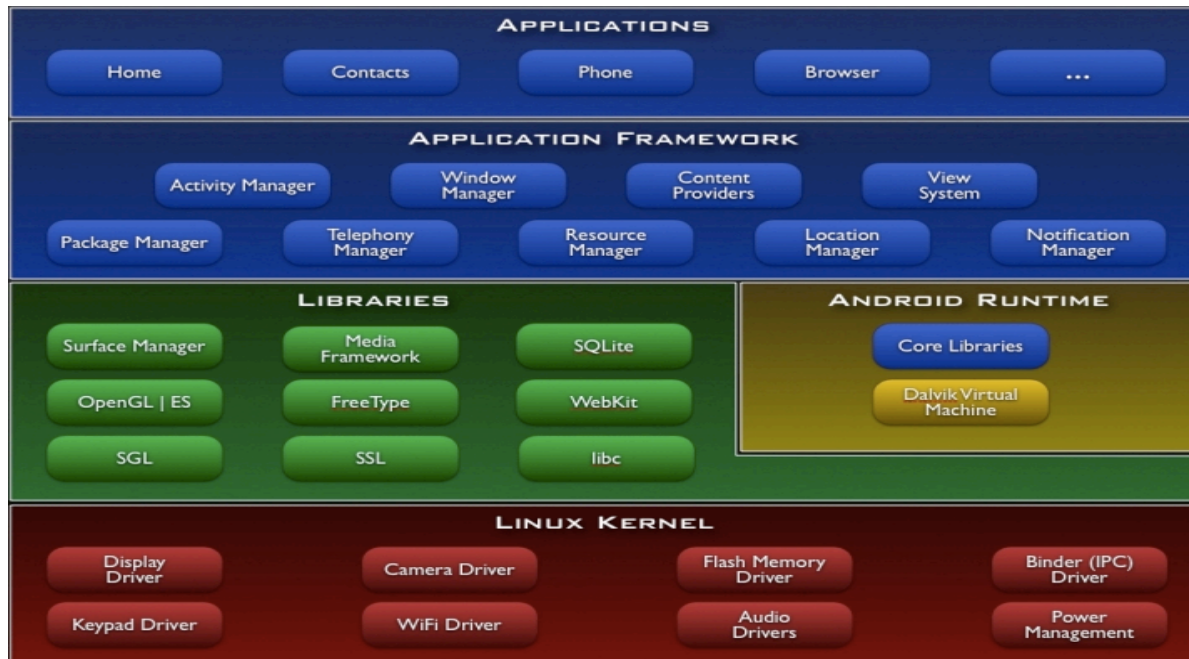


Figure I.5 : Architecture d'Android

La figure précédente schématise l'architecture d'Android. On peut y observer toute une pile de composants qui constituent le système d'exploitation. Le sens de lecture se fait de bas en haut, puisque le composant de plus bas niveau (le plus éloigné des utilisateurs) est le noyau Linux et celui de plus haut niveau (le plus proche des utilisateurs) est constitué par les applications.

Nous savons que le système d'exploitation d'Android se base sur Linux. Si on veut être plus précis, c'est le noyau « *kernel* » de Linux qui est utilisé. Le noyau est

l'élément du système d'exploitation qui permet de faire le pont entre le matériel et le logiciel.

La version du noyau utilisée avec Android est une version conçue spécialement pour l'environnement mobile, avec une gestion avancée de la batterie et une gestion particulière de la mémoire. C'est cette couche qui fait en sorte qu'Android soit compatible avec tant de supports différents.

Si on regarde attentivement le schéma, nous remarquerons que cette couche est la seule qui gère le matériel. Android en soi ne s'occupe pas de ce genre de détails, ça ne veut pas dire qu'il n'y a pas d'interactions entre Android et le matériel, juste que, quand un constructeur veut ajouter un matériel qui n'est pas pris en compte par défaut par Android, il doit travailler sur le kernel et non sur les couches au-dessus, qui sont des couches spécifiques à Android.

c) Moteur d'exécution d'Android :

Un moteur d'exécution « runtime system » est un programme qui permet l'exécution d'autres programmes. Pour utiliser des applications développées en Java sur ordinateur nous avons besoin du JRE « Java Runtime Environment ». Il s'agit du moteur d'exécution nécessaire pour lancer des applications écrites en Java.

C'est cette couche qui fait qu'Android n'est pas qu'une simple « implémentation de Linux pour portables ». Elle contient certaines bibliothèques de base du Java accompagnées de bibliothèques spécifiques à Android et la machine virtuelle «Dalvik».

La figure suivante est un schéma qui indique les étapes nécessaires à la compilation et à l'exécution d'un programme Java standard.

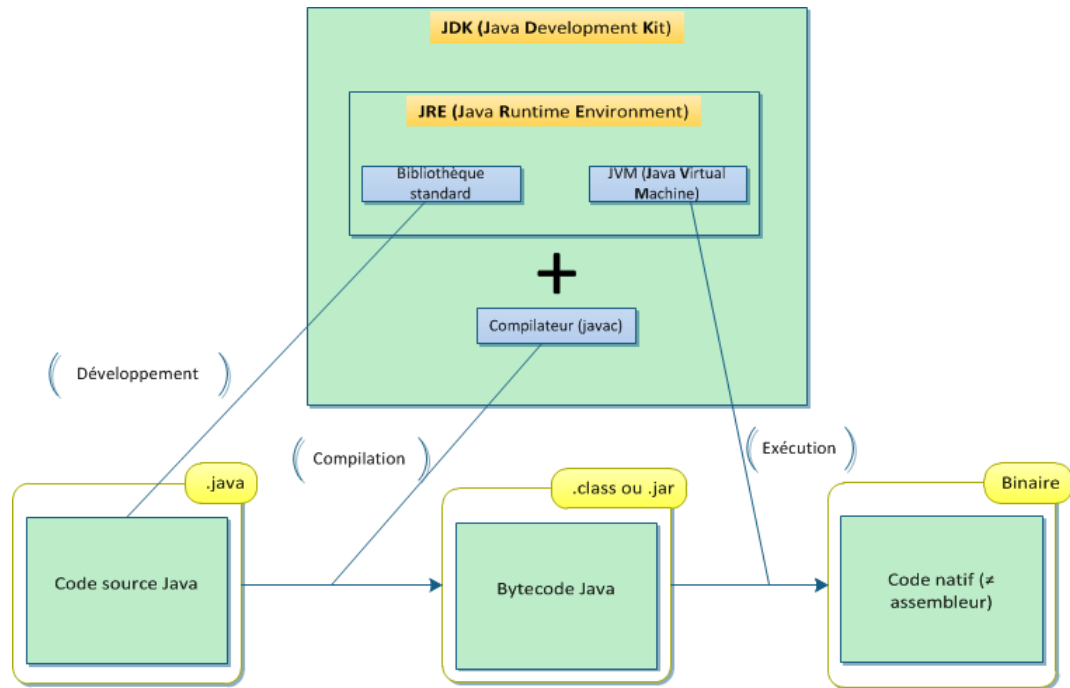


Figure I.7 Architecture Java

Ce code est une suite d'instructions que l'on trouve dans un fichier `.java` qui sera traduit en une autre suite d'instructions dans un autre langage que l'on appelle le «bytecode». Ce code est contenu dans un fichier `.class`. Le bytecode est un langage spécial qu'une machine virtuelle Java peut comprendre et interpréter. Les différents fichiers `.class` sont ensuite regroupés dans un `.jar`, et c'est ce fichier qui est exécutable. En ce qui concerne Android, la procédure est différente. Pour développer des applications pour Android, on n'utilise pas Java SE. Ce n'est pas non plus « Java ME » que l'on utilise qui est une version spéciale de Java destinée au développement mobile, mais pas pour Android.

La version de Java qui permet le développement Android est une version réduite amputée de certaines fonctionnalités qui n'ont rien à faire dans un environnement mobile. Par exemple, la bibliothèque graphique Swing n'est pas supportée, on trouve à la place un système beaucoup plus adapté. Mais Android

n'utilise pas une machine virtuelle Java ; une machine virtuelle tout étudiée pour les systèmes embarqués a été développée, et elle s'appelle « Dalvik ». Cette machine virtuelle est optimisée pour mieux gérer les ressources physiques du système. Elle permet par exemple de laisser moins d'empreinte mémoire (la quantité de mémoire allouée à une application pendant son exécution) ou d'utiliser moins de batterie qu'une machine virtuelle Java.

La plus grande caractéristique de Dalvik est qu'elle permet d'instancier (créer une occurrence de) chaque programme a sa propre occurrence de Dalvik et elles peuvent vivre sans se perturber les unes les autres. La figure suivante est un schéma qui indique les étapes nécessaires à la compilation et à l'exécution d'un programme Android standard.

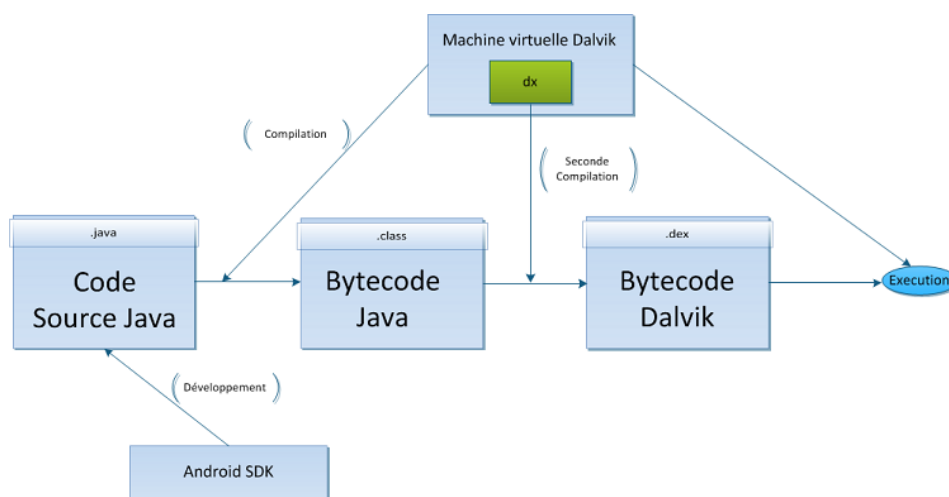


Figure I.8. Compilation et exécution d'un programme Android

On voit bien que le code Java est ensuite converti en bytecode Java comme auparavant. Sachant que le bytecode Java ne peut être lu que par une machine virtuelle Java, mais que Dalvik n'est pas une machine virtuelle Java., Il faut donc procéder à une autre conversion à l'aide d'un programme qui s'appelle « dx » qui s'occupe de traduire les applications de bytecode Java en bytecode Dalvik, qui, lui, est compréhensible par la machine virtuelle.

La puissante machine virtuelle Dalvik est destinée uniquement à Android, mais il est possible de développer une machine virtuelle similaire et qui ne fonctionne pas sous Android. Par exemple, le Nokia N9 pourra exécuter des applications Android sans utiliser ce système.

Conclusion :

L'évolution des technologies des systèmes embarqués offre aujourd'hui des perspectives d'intégration négligeables. Nous avons vu dans cette partie toute l'importance et l'étendue de ces systèmes. Dans le chapitre suivant nous présentons des technologies qui utilisent des systèmes embarqués : RFID et NFC.

CHAPITRE II

Généralités sur les RFID

Introduction :

RFID fait partie des technologies d'identification automatique, au même titre que la reconnaissance optique de caractères ou de codes barre. Cette technologie permet d'identifier un objet ou une personne, d'en suivre le cheminement et d'en connaître les caractéristiques à distance grâce à une étiquette émettant des ondes radio, attachée ou incorporée à l'objet ou à la personne. La technologie RFID permet la lecture des étiquettes même sans ligne de vue directe et peut traverser de fines couches de matériaux (peinture, neige, etc.).

Après une définition et un petit historique de la technologie RFID, nous présentons son fonctionnement et ses composants, puis nous donnons la gamme de fréquence des systèmes RFID et quelques exemple d'application de cette technologie. Enfin, nous présentons la technologie NFC (extension de la RFID) qui sera utilisé dans notre application.

II.1. Définition de la RFID [10]

Insérer une clé pour démarrer un véhicule, badger pour accéder à un bâtiment ou une salle, valider un titre de transport dans le bus ou le métro sont des gestes entrés dans le quotidien de bon nombre d'entre nous. Nous utilisons, sans en être toujours conscient, des technologies de capture automatique de données basées sur les ondes et rayonnements radiofréquence. Cette technologie est connue sous le nom de RFID pour Identification par RadioFréquence. Ce que chaque être humain fait dans sa vie quotidienne, les objets le font également depuis leur lieu de fabrication jusqu'au point de vente en passant par les lieux de stockage. Ils sont, comme nous, porteurs d'étiquettes RFID. La différence entre les objets et nous, c'est qu'ils ne présentent pas «volontairement» leur étiquette ou badge RFID lorsqu'on leur demande. Les conditions de lecture de ces étiquettes sont donc différentes et demandent généralement des distances de détection plus importantes.

On peut donner la définition suivante à la RFID (RadioFrequency Identification) : Technologie d'identification automatique qui utilise le rayonnement

radiofréquence pour identifier les objets porteurs d'étiquettes lorsqu'ils passent à proximité d'un interrogateur.

Ceci dit, la RFID ne peut pas se résumer à une seule technologie. En effet, il existe plusieurs fréquences radio utilisées par la RFID, plusieurs types d'étiquette ayant différents types de mode de communication et d'alimentation. Pour transmettre des informations à l'interrogateur (encore appelé station de base ou plus généralement lecteur), un tag RFID est généralement muni d'une puce électronique associée à une antenne. Cet ensemble, appelé inlay, est ensuite packagé pour résister aux conditions dans lesquelles il est amené à vivre. L'ensemble ainsi formé est appelé tag, label ou encore transpondeur. Les informations contenues dans la puce électronique d'un tag RFID dépendent de l'application. Il peut s'agir d'un identifiant unique (UII, Unique Item Identifier ou code EPC, Electronic Product Code, etc.). Une fois écrit dans le circuit électronique, cet identifiant ne peut plus être modifié mais uniquement lu (WORM Write Once Read Multiple). Certaines puce électroniques disposent d'une autre zone mémoire dans laquelle l'utilisateur peut écrire, modifier, effacer ses propres données. La taille de ces mémoires varie de quelques bits à quelques dizaines de kilobits.

II.2. Historique de la RFID [20]

1940

Le principe de la RFID est utilisé pour la première fois lors de la Seconde Guerre Mondiale pour identifier/authentifier des appareils en vol (IFF : Identifie FriendlyFoe). Il s'agissait de compléter la signature RADAR des avions en lisant un identifiant fixe permettant l'authentification des avions alliés

1970

Durant les années 1960-1970, les systèmes RFID restent une technologie confidentielle, à usage militaire pour le contrôle d'accès aux sites sensibles, notamment dans le nucléaire.

1980

Les avancées technologiques permettent l'apparition du tag passif. Le tag RFID rétro module l'onde rayonnée par l'interrogateur pour transmettre des informations. Cette technologie permet de s'affranchir de source d'énergie embarquée sur l'étiquette réduisant de ce fait son coût et sa maintenance

1990

Début de la normalisation pour une interopérabilité des équipements RFID.

1999

Fondation par le MIT (Massachusetts Institute of Technology) de l' Auto-ID center : centre de recherches spécialisé en identification automatique (entre autre RFID).

2004

L'auto-ID du MIT devient "EPCglobal", une organisation chargée de promouvoir la norme EPC (Electronic Product Code), extension du code barre à la RFID.

A partir de 2005

Les technologies RFID sont aujourd'hui largement répandues dans quasiment tous les secteurs industriels (aéronautique, automobile, logistique, transport, santé, vie quotidienne, etc.). L'ISO (International Standard Organisation) a largement contribué à la mise en place de normes tant techniques qu'applicatives permettant d'avoir un haut degré d'interopérabilité voire d'interchangeabilité.

2009

Création du Centre National de Référence RFID (France).

II.3. Fonctionnement [19]

Le système RFID fonctionne de la manière suivante :

- **L'étiquette RFID (ou transpondeur ou tag)** est elle-même équipée d'une puce reliée à une antenne, l'antenne permet à la puce de transmettre les informations (numéro de série, poids...) qui peuvent être lues grâce à un lecteur émetteur-récepteur.

- Une fois les informations transmises au lecteur RFID équipé d'une antenne intégrée ou externe, celui-ci n'a plus qu'à convertir les ondes-radio en données et celles-ci pourront être lues par un logiciel RFID.

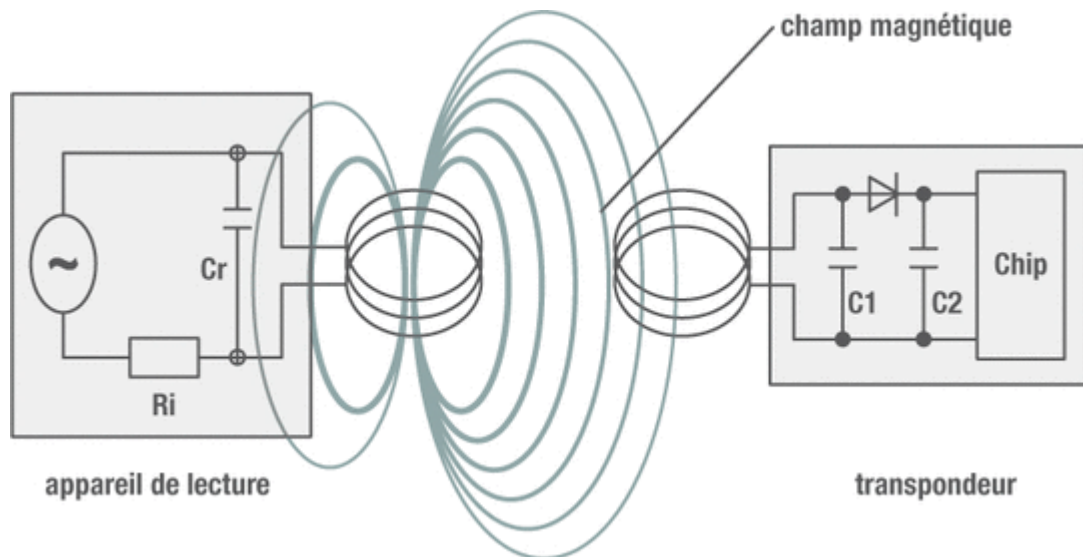


Figure. II.1 principaux éléments d'un système

- L'illustration montre les deux principaux éléments d'un système RFID : le **lecteur** et le **transpondeur**. Le transpondeur est un mot dérivé de l'anglais composé à partir de « transmitter » (transmetteur) et « respond » (répondre).
- Dans le lecteur, un générateur émet un signal d'appel qui est transmis à une bobine. La bobine produit un champ magnétique alternatif, matérialisé dans la fig. II.1 par les lignes de champ ovales qui pénètrent et ressortent de l'ensemble de la bobine. Une partie de ces lignes de champ traverse la « bobine antenne » du transpondeur.
- Selon la loi d'induction, une tension est générée dans la bobine du transpondeur qui est ensuite traitée par l'électronique (puce) raccordée. Un signal de réponse est émis et retransmis au lecteur : le transpondeur est ainsi

déecté et appelé. Ce mécanisme décrit de manière simplifiée est à la base de toutes les variantes de RFID. Le principal domaine d'utilisation est la logistique de marchandises, mais les systèmes d'accès basés sur la RFID sont également répandus. La technologie de transpondeur n'est pas encore très répandue dans le domaine de la sécurité au travail, il s'agit d'un domaine d'application récent.

II.4. Composants et fonctionnement du système

Une solution complète de RFID comprend les étiquettes, les lecteurs et encodeurs et l'intergiciel (middleware).

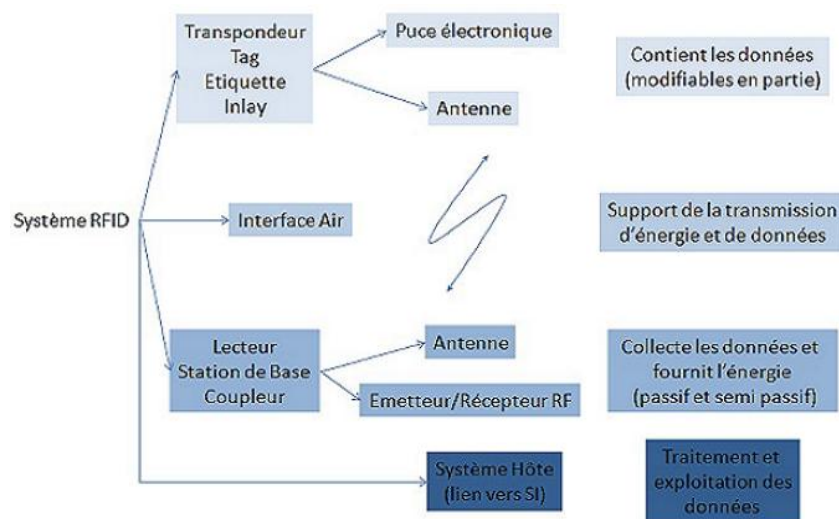


Figure. II.2 Composants d'un système RFID

II.4.1. Le tag (étiquette)

Une des méthodes d'identification les plus utilisées est d'abriter un numéro de série ou une suite de données dans une puce (chip) et de relier cette dernière à une petite antenne. Ce couple (puce silicium + antenne) est alors encapsulé dans un support

(RFID Tag ou RFID Label). Ces "tag" peuvent alors être incorporés dans des objets ou être collés sur des produits.

Le tout est alors imprimé sur un support pliable, souvent adhésif. Le format des données inscrites sur les étiquettes est standardisé à l'initiative d'EPC Global (Electronic ProductCode).

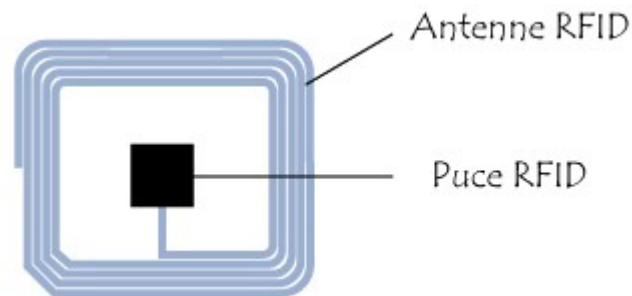


Figure. II.3 Tag RFID

II.4.1. 1. Les différents types de tags et leurs spécificités techniques

Pour exploiter les informations contenues dans ces étiquettes, il faut impérativement disposer du lecteur approprié. Celui-ci émet des ondes radios en direction de la capsule ce qui permet de l'alimenter en énergie (alimentation par induction électromagnétique), en d'autres termes de l'activer (la puce renvoie alors des données), pour en extraire les informations qu'elle renferme. Ces puces ne sont pas capables d'effectuer des traitements dynamiques mais seulement de renvoyer des données statiques.

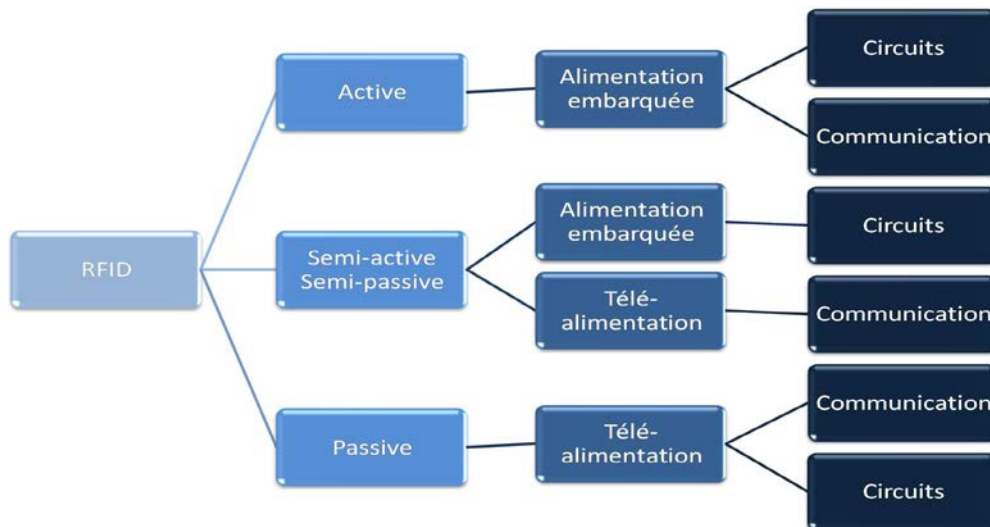


Figure. II.4 Types de tags RFID

1. Tags passifs (sans batterie)

Ne disposant d'aucune alimentation externe, ils dépendent de l'effet électromagnétique de réception d'un signal émis par le lecteur. C'est ce courant qui leur permet d'alimenter leurs microcircuits. Ils sont peu coûteux à produire et sont généralement réservés à des productions en volume. Ce sont eux que l'on trouve plus particulièrement dans la logistique et le transport. Ils utilisent différentes bandes de fréquences radio selon leur capacité à transmettre à distance plus ou moins importante et au travers de substances différentes (air, eau, métal). La distance de lecture est inférieure à un mètre. Les basses et hautes fréquences sont normalisées au niveau mondial. Ces puces sont collées sur les produits pour un suivi allant jusqu'aux inventaires. Elles sont jetables ou réutilisables suivant les cas. Les puces avec une antenne de type "papillon" ont une portée courante de 1 à 6 mètres. Ces puces UHF (Ultra Haute Fréquence) sont utilisées pour la traçabilité des palettes dans les entrepôts. Par contre, la tolérance aux obstacles est moyenne. Pour les très hautes

fréquences (UHF), l'Europe, l'Asie et les Etats-Unis se distinguent par des fréquences et des réglementations différentes

2. Tags semi-passifs

Ces tags sont similaires aux cartes d'identification passive. Ils emploient des technologies proches, mais avec quelques différences importantes. Ils disposent en effet eux aussi d'une petite batterie qui fonctionne en permanence, ce qui libère l'antenne pour d'autres tâches, notamment la réception de signaux de retour. Ces tags sont plus robustes et plus rapides en lecture et en transmission que les tags passifs, mais ils sont aussi plus chers.

3. Tags actifs

Les étiquettes actives sont les plus chères car elles sont plus complexes à produire et assurent, outre des fonctions de transmission, des fonctions soit de captage soit de traitement de l'information captée, soit les deux. De ce fait, elles ont besoin d'une alimentation embarquée et sont donc caractérisées par la durée de vie de celle-ci. Si le prix est un facteur discriminatif, il faut savoir que ces étiquettes s'avèrent particulièrement bien adaptées à certaines fonctions, dont notamment la création de systèmes d'authentification, de sécurisation, d'antivol, etc. Bref, elles sont idéales pour tout ce qui concerne le déclenchement d'une alerte ou d'une alarme. Elles peuvent émettre à plusieurs centaines de mètres. Le dernier cri est le tag «insensible à l'orientation du produit».

II.4.2. Le lecteur



Figure. II.5 Lecture d'un tag

Le lecteur/enregistreur est constitué d'un circuit qui émet une énergie électromagnétique à travers une antenne, et d'une puce électronique qui reçoit et décode les informations envoyées par le transpondeur et les envoie au dispositif de collecte des données, il permet même d'écrire leur contenu. Le lecteur RFID est l'élément responsable de la lecture des étiquettes radiofréquence et de la transmission des informations qu'elles contiennent (code EPC ou autre, informations d'état, clé cryptographique...) vers le niveau suivant du système (middleware). Cette communication entre le lecteur et l'étiquette s'effectue en quatre temps :

- 1) Le lecteur transmet par radio l'énergie nécessaire à l'activation du tag ;
- 2) Il lance alors une requête interrogeant les étiquettes à proximité ;
- 3) Il écoute les réponses et élimine les doublons ou les collisions entre réponses ;
- 4) Enfin, il transmet les résultats obtenus aux applications concernées.

La communication entre le lecteur et l'étiquette s'effectue via les antennes qui équipent l'un et l'autre, ces éléments étant responsables du rayonnement radiofréquence. Les antennes dont dispose le lecteur sont plus ou moins standardisées, mais offrent les mêmes différences que les haut-parleurs d'une chaîne stéréo d'un modèle à l'autre. Pour continuer ce paradigme, la logique de la chaîne stéréo s'applique tout aussi bien ici puisque la lecture ne sera bonne que si l'antenne est de bonne facture. D'où l'importance de ce composant dans le choix de la solution. De même, si le lecteur s'avère de qualité insuffisante, le traitement des données en souffrira. Il y a donc là un équilibre à trouver entre ces deux composants. La puissance du lecteur est donc à combiner avec l'antenne adéquate, ceci permettant de déterminer la portée optimale de la lecture. Généralement, on distingue quatre modalités :

- Lecture de proximité : entre 10 et 25 cm ;
- Lecture de voisinage : jusqu'à 1 mètre ;
- Lecture à moyenne distance : de 1 à 9 mètres ;
- Lecture longue portée : jusqu'à plusieurs centaines de mètres.

Par ailleurs, le terme de lecteur RFID est en fait une impropriété, puisque ce dernier est également capable d'écrire des informations sur l'étiquette. Car, si bon nombre d'étiquettes sont en lecture seule (le code qu'elles contiennent ayant été «imprimé» en même temps que l'étiquette elle-même, d'autres contiennent, au-delà du code de base, une zone mémoire pouvant contenir des données variables.

On peut opérer la classification suivante à propos de la lecture de l'étiquette :

- Lecture seule : le lecteur prélève le code du tag émettant le signal le plus fort
- Lecture multiple : le lecteur explore le champ de lecture pour prélever les codes de toutes les étiquettes en émission rf

En ce qui concerne en revanche le fonctionnement du lecteur:

- Autonome: le lecteur active le signal rf après avoir reçu une entrée ou une commande du logiciel ;
- Interactif : le lecteur lit lorsqu'il reçoit une requête d'une autre application à un autre niveau.

II.4.3. L'intergiciel (middleware)

C'est un logiciel servant comme intermédiaire de communication entre plusieurs applications, généralement complexes ou distribuées sur un réseau informatique. L'intergiciel offre des services de haut niveau liés aux besoins de communication des applications.

Nous expliquerons ici le concept de l'intergiciel de RFID et montrerons le rôle crucial qu'il joue.

Un système de RFID se compose du matériel, y compris des lecteurs de RFID, et du logiciel. Le logiciel fonctionne sur les PCs ou les serveurs ordinaires et se compose de l'intergiciel, qui contient la logique de l'application de RFID, et d'un système de base de données principal (par exemple, Oracle, serveur de SQL, Postgres, MySQL) pour stocker des informations sur les étiquettes. Typiquement, les étiquettes contiennent un numéro d'identification et probablement de l'information sur un article spécifique. Pour une application de supermarché, le scanner de contrôle pourrait

simplement indiquer le numéro d'identité et le regarder dans la base de données pour voir combien le produit coûte. Cependant, dans une application de bagages d'aéroport le système pourrait écrire le nombre du billet du passager et la destination définitive pour le sac sur l'étiquette au temps d'enregistrement (parce que le prochain aéroport que le sac visite peut ne pas avoir accès à la base de données de l'aéroport d'origine).

Dans l'Internet des objets, le middleware RFID constitue le cerveau de la chaîne logicielle qui vient donner de l'intelligence aux données récupérées sur les produits étiquetés d'une puce radiofréquence. Un point central dans l'implémentation en entreprise, qui a d'importantes quantités de données à traiter et, surtout, à partager avec l'ensemble de sa chaîne de valeur et de partenaires. Un distributeur doit fournir à ses partenaires des données compréhensibles par les systèmes en place. Et c'est bien là que réside l'enjeu du middleware. Dans la RFID, la couche intergicielle est confrontée à de fortes contraintes. D'abord liées au matériel et à l'environnement dans lequel ce même matériel est déployé. Le choix de lecteurs et d'étiquettes RFID (Tag) adaptés correspond au premier enjeu, d'autant que les analystes s'accordent à dire que la qualité de lecture de ces capteurs est assez hétérogène. Puis vient ensuite celle de la législation sur l'utilisation des plages de fréquences autorisées qui diffèrent d'un pays à l'autre.

Suivent ensuite le filtrage des flux de données ainsi récupérées, leur gestion centralisée et enfin la gestion de la qualité des services. Ce sont les tâches premières du middleware RFID. Suivront ensuite les procédures de transformation des données liées à l'intégration aux processus métier de l'entreprise. Une chaîne complexe, en somme. Aujourd'hui, les offres middleware dédiées aux RFID se positionnent comme des solutions à moyen/long terme permettant aux entreprises de mieux profiter des apports des RFID.

II.5. Communication tag

Voici une représentation basique de la communication entre un transpondeur et une station de base :

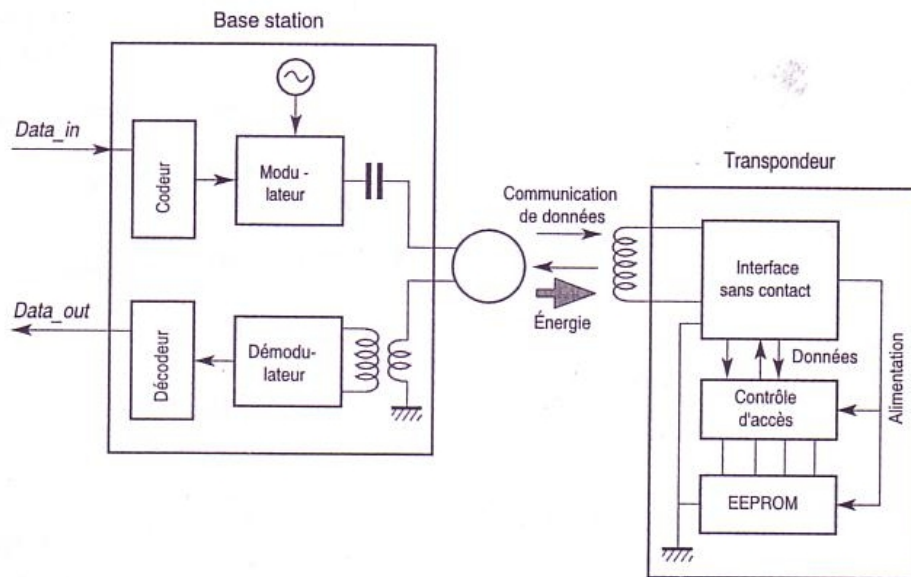


Figure. II.6 la communication entre un transpondeur et une station de base

Le schéma ci-dessus présente deux éléments dans la composition de la communication : tout d'abord, les données qui sont effectivement échangées entre les deux équipements, mais aussi de l'énergie transmise du lecteur vers le transpondeur. Cette énergie est la source d'alimentation utilisée dans le cas de télé alimentation.

Les communications RFID, comme dans la plupart des communications sans fil, sont half-duplex. Cela signifie que chacun des interlocuteurs (ici la station de base et le ou les transpondeurs) communiquent à tour de rôle. Il existe cependant deux modes de communication liés à la présence des deux types d'information transmis :

- **simultané** : les données et l'énergie sont transmises simultanément au transpondeur,
- **non simultané** : les données et l'énergie sont fournies alternativement au transpondeur.

Comme dans toute conversation, l'un des deux interlocuteurs doit nécessairement initialiser la communication. Pour cela, il existe deux modes :

- TTF (Tag Talks First) : dans ce mode, le tag annonce sa présence à son arrivée dans le champ d'un lecteur. Ce mode peut poser des conflits lorsque plusieurs tags annoncent leur présence simultanément.

- RTF (Reader Talks First) : dans ce mode, le lecteur interroge constamment son environnement afin de détecter la présence de nouveaux arrivants. Une requête est propagée régulièrement et, lorsqu'un transpondeur entre dans le champ et est capable de répondre, il renvoie une réponse annonçant sa présence.

Bien évidemment, l'utilisation simultanée des deux modes implique des conflits importants. Pour ces raisons, il faut veiller à appliquer un mode unique dans des secteurs fermés employant la technologie RFID. Certains pays d'Extrême-Orient ont même complètement banni l'utilisation du mode TTF.

Toute communication RFID est basée sur des protocoles simples ou complexes. Les protocoles permettent, par exemple, d'inclure des mécanismes de gestion de collisions, de sécurisation, d'authentification, etc., en plus du simple mécanisme d'échange des données.

II.6. Les gammes de fréquences RFID

II.6.1. La RFID dans le spectre radio

L'utilisation de ressources radio est soumise à des autorisations et suit des règlements nationaux ou internationaux :

- LF : 125 kHz - 134,2 kHz : basses fréquences,
- HF : 13,56 MHz : hautes fréquences,
- UHF : 860 MHz - 960 MHz : ultra hautes fréquences,
- SHF : 2,45 GHz : super hautes fréquences.

Voici un aperçu des fréquences de la RFID dans le spectre radio :

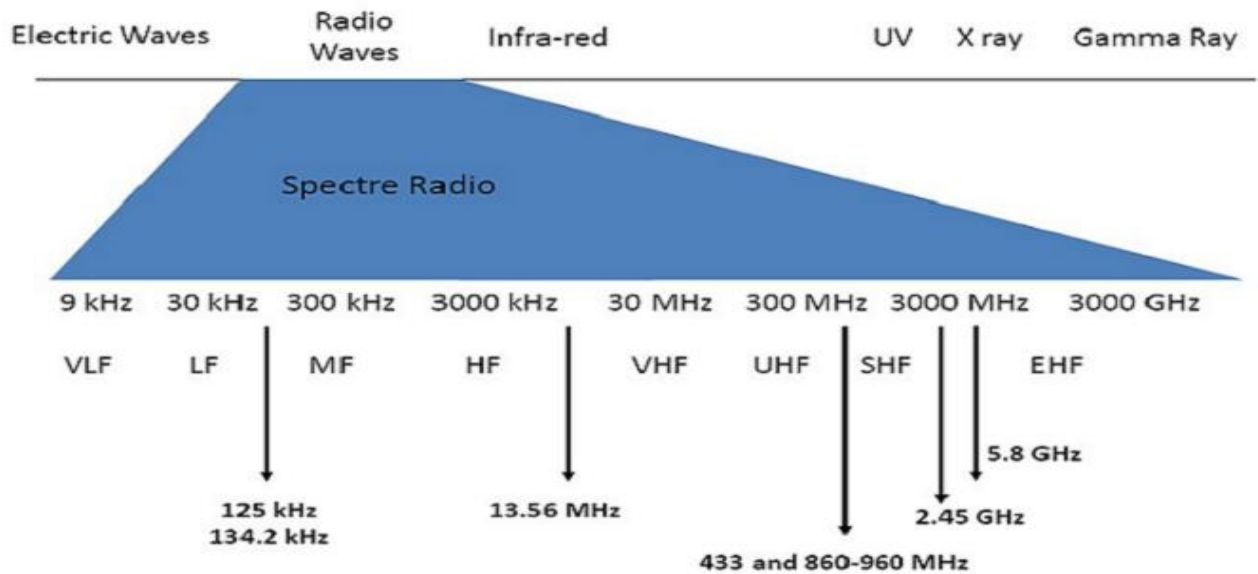


Figure.II.7 Fréquences de la RFID

II.6.2. Les tags RFID UHF, HF, LF

Voici les trois fréquences de tags RFID:

- Les tags **RFID UHF à 900 MHz** possèdent des antennes imprimées ou gravées. En technologie passive, ils peuvent être lus à plusieurs mètres. Ils sont plus sensibles à l'environnement (métal, eau) du fait de la fréquence utilisée mais des design particuliers d'antenne et de packaging permettent de les utiliser sur des supports métalliques. Les fréquences UHF réservées à la RFID n'étant pas harmonisées dans toutes les régions du monde (entre 860 et 960 MHz), les tags doivent généralement présenter des bandes passantes importantes qui réduisent leurs performances.

- Les tags **RFID HF 13.56 MHz** sont utilisés dans des applications de logistique et de traçabilité. Les antennes boucle peuvent être imprimées ou gravées ce qui rend les tags particulièrement fins. Ils sont largement répandus dans les applications de transport et d'identité (passeport, passe Navigo, cartes sans contact).

Cette technologie est à la base des applications NFC (Near Field Communication) que l'on trouve dans de plus en plus de smartphones.

- Les tags **RFID LF 125 kHz** sont adaptés aux applications de logistique et traçabilité. Les caractéristiques physiques de ces tags, d'un poids et une taille réduits, font d'eux des candidats idéals pour être intégrés dans tout type de matériaux, textiles, métaux, plastiques, etc.

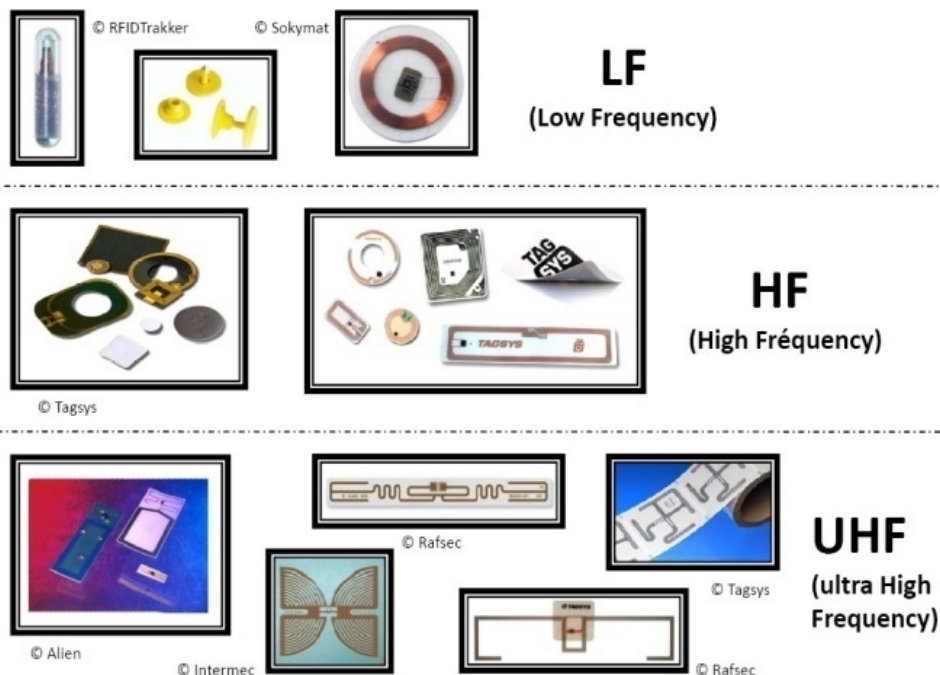


Figure. II.8 Exemple d'utilisation pour les frequences RFID

II.7. Standard RFID

Afin d'obtenir des performances maximales de la technologie RFID, les étiquettes, lecteurs et fréquences doivent fonctionner en harmonie totale même en cas de fabricant différents, a cette effet plusieurs organisations ont établies des standards

internationales permettant le développement considérable d'applications basés sur cette technologie. On peut distinguer deux types de standards.

II.7.1. Les standards de données

Ils définissent le contenu des mémoires des étiquettes RFID, ainsi que le format de stockage. EPC (Code Produit Electronique) est l'un des standards de données. Il a été développé par EPCGlobal afin d'assurer une identification des objets d'une manière unique dans le monde. Un EPC est un code unique pour chaque produit, il peut contenir plusieurs informations telles que le fabricant et le type du produit...etc. actuellement, il existe trois format d'EPC : 64 bits, 96 bits ou 128 bits. La figure suivante présente un EPC-96 bits.

Nombre de codage	Entreprise	Type de produit	Numéro de série du produit
01	0000A89	00016F	000169DC0

Figure.II.9 EPC 96 bits

II.7.2. Les standards de technologie [13]

Ils se focalisent sur les protocoles de communication entre le tag et le lecteur. L'une des organisations qui travaille sur les standards de technologie est l'ISO (International Organization for Standardization). Grâce à son partenariat avec des organismes internationaux, des gouvernements, des industries et des représentants des consommateurs, elle a fourni des standards internationaux dans environ 148 pays. Plusieurs normes ont été développées par l'ISO, le tableau suivant présente quelques normes de l'ISO.

Les standards ISO	Caractéristiques
ISO 11785	Pour les applications BF
ISO 15693	Pour les applications HF
ISO 18000-6	Pour les applications UHF
ISO 15693	Pour les tags passifs, Lecture/Ecriture et qui travaillent avec HF

Tableau. II.1 Standards ISO

II.8. Applications de la RFID

Les applications RFID s'appuient sur différents standards dépendant des fonctionnalités exigées par les processus métier et par certaines contraintes locales. La RFID permet de répondre à un grand nombre de besoins. Elle se développe bien en intra entreprise et dans la logistique. Les principales difficultés auxquelles le standard RFID doit faire face sont en voie d'être surmontées : coût de l'étiquette (plus de 0,10€ pour les moins chères), gestion de l'anticollision en cas de lecture de nombreuses étiquettes en simultané, lecture au travers des fluides, adoption ou convergence trop lente de certains standards, remise en cause de processus anciens, problèmes de sécurité et d'éthique.

II.8.1. L'application aux entreprises

Les applications des étiquettes RFID sont déjà très nombreuses, il s'agit simplement ici de donner quelques exemples des possibilités offertes par cette technologie.

La gestion de la chaîne d'approvisionnement en matière logistique, quatre niveaux d'applications peuvent être distingués:

II.8.1.1 Expédition

L'étiquette peut faciliter le picking des produits, la constitution des palettes, leurs tris par destination, le contrôle du chargement. L'étiquette peut contenir, outre 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 collectées au moment du chargement peuvent être stockées dans l'étiquette RFID du conteneur ou du moyen de transport afin de faciliter les contrôles en cours de transit, douane.

II.8.1.2. Réception

Les données pourront automatiquement être collectées pour effectuer les contrôles et mettre à jour les stocks et effectuer les rapprochements avec documents commerciaux ou les messages EDI.

II.8.1.3. 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.

II.8.1.4. Local

Les étiquettes permettent les inventaires de produit, mais aussi la gestion des supports de manutention et des équipements (bouteilles de gaz...). La collecte des déchets Aussi bien en Europe qu'au Japon et aux Etats Unis les sociétés de collecte de déchets ménagers ou industriels, se préoccupent d'améliorer la répartition de la charge du ramassage et du traitement des ordures. Le principe est d'équiper chaque poubelle ou container d'une étiquette RFID, et les camions de collecte de lecteurs et de système de pesage, afin qu'à chaque opération puisse être automatiquement identifié le «producteur» et mesurer le poids des matières collectées.

De nombreuses sociétés de gestion d'autoroute à péage ont mis en place des péages automatiques qui sont des systèmes d'abonnement basés sur des étiquettes RFID, hyperfréquence, placées dans les véhicules. Les abonnés bénéficient ainsi de voie particulière de passage aux barrières de péage. Le paiement s'effectue sans arrêt du véhicule, par simple lecture de son identification.

Les étiquettes RFID sont déjà utilisées pour le contrôle d'accès des immeubles ou des parkings. Certaines stations-services américaines expérimentent, la délivrance

automatique de carburant au propriétaire de véhicule muni d'une étiquette RFID à 2,45 GHz qui leur permet d'être identifiés lors du stationnement devant les pompes.

Des applications de plus en plus nombreuses de traçabilité des animaux se développent, que ce soit les étiquettes auriculaires sur les animaux d'élevage ou les étiquettes sous cutanées pour les chevaux ou les animaux domestiques. Dans tous les cas, il s'agit d'assurer une traçabilité des animaux dans un but de contrôle sanitaire ou de la qualité des cheptels.

La gestion de la traçabilité des bouteilles de gaz pour des raisons de sécurité est très strictement réglementée. Des entreprises, comme l'Air Liquide, utilisent des étiquettes RFID pour tracer la vie des bouteilles de gaz. Il s'agit essentiellement d'en assurer la gestion des stocks et de la distribution, et les opérations de maintenance obligatoires. Dans cette application, il faut noter que la composition métallique des bouteilles a été surmontée en isolant les étiquettes de la bouteille.

Pour le nettoyage des vêtements de travail, des entreprises mettent en place des systèmes d'identification des uniformes basé sur une étiquette RFID d'un diamètre de 20mm et d'une épaisseur de 2,5 mm, la fréquence 13,56 MHz en lecture / écriture à 20 cm. Ces étiquettes sont fixées au vêtement, elles résistent aux opérations de lavage. Elles permettent un suivi des opérations de lavage et une identification aisée du porteur de l'uniforme.

Quelques applications originales et parfois controversées: le suivi des écoliers au Japon grâce à des étiquettes placées sur les cartables ou vêtements, l'implantation d'étiquettes directement sous la peau de militaires au Mexique, etc.

II.9. Avantages et inconvénients

II.9.1. Avantages

- **La capacité de mise à jour du contenu par les intervenants**

A la différence du code à barres pour lequel les données sont figées une fois imprimée ou marquée, le contenu des données stockées dans une étiquette radio

fréquence va pouvoir être modifié, augmenté ou diminué par les intervenants autorisés (étiquettes en lecture et écriture multiple).

- **Une plus grande capacité de contenu**

Dans une étiquette radiofréquence une capacité de 1 000 caractères est aisément stockable sur 1mm², et peut atteindre sans difficulté particulière 10 000 caractères. Dans une étiquette logistique apposée sur une palette, les différentes unités contenues et leurs quantités respectives pourront être enregistrées et lues.

- **La vitesse de marquage**

Le code à barres dans un contexte logistique nécessite le plus souvent l'impression d'un support papier. La manipulation et la pose des étiquettes restent des opérations manuelles ou mécaniques. Les étiquettes radio fréquence peuvent être incluses dans le support de manutention ou dans les conditionnements dès l'origine. Les données concernant les objets contenues ou transportées sont écrites en une fraction de seconde au moment de la constitution de l'unité logistique ou de transport, sans manipulation supplémentaire.

- **Une sécurité d'accès au contenu**

Comme tout support numérique, l'étiquette radio fréquence peut être protégée par mot de passe en écriture ou en lecture. Les données peuvent être chiffrées. Dans une même étiquette, une partie de l'information peut être en accès libre, et l'autre protégée. Cette faculté fait de l'étiquette RF, un outil adaptée à la lutte contre le vol et la contrefaçon.

- **Une plus grande durée de vie**

Dans les applications où un même objet peut être utilisé plusieurs fois, comme l'identification des supports de manutention, ou la consignation du contenant, une étiquette radio fréquence peut être réutilisée 1 000 000 de fois.

- **Une plus grande souplesse de positionnement**

Avec l'étiquette radio fréquence, 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. De plus leur principe de fonctionnement ne les rend pas sensibles aux souillures, ou taches diverses qui nuisent à l'utilisation du code à barres.

II.9.2. Inconvénients

- **Le coût**

Les prix restent nettement supérieurs à ceux des étiquettes code à barres pour des unités consommateurs. Utiliser les étiquettes radio fréquence en lieu et place du code à barres sur les produits de grande consommation, n'est donc pas aujourd'hui économiquement réaliste. Cela le devient pour lutter contre le vol ou la contrefaçon sur les produits à forte valeur ajoutée, ou pour tracer les produits dans le cadre du service après-vente, comme l'électroménager ou la HI-FI. Par contre au-delà du conditionnement unitaire, le coût de l'étiquette radio fréquence peut devenir marginal par rapport à la valeur des produits contenus. C'est pourquoi dans le domaine des produits de grande consommation, les premières applications de ces étiquettes peuvent voir le jour sur les cartons, sur les palettes et sur les unités de transport. Par ailleurs, si la comparaison se fait au niveau du système d'identification et de traçage, il faut prendre en compte les coûts des lecteurs, favorables à la RFID, ainsi que le gain de temps venant de la non obligation de manipuler les objets pour présenter le code à barres devant le lecteur.

- **La perturbation par l'environnement physique**

La lecture des étiquettes radio fréquences est perturbée par la présence, par exemple, de métaux dans leur environnement immédiat. Des solutions doivent être étudiées au cas par cas pour minimiser ces perturbations, comme cela a été fait par exemple pour l'identification des bouteilles de gaz.

- **Les perturbations induites par les étiquettes entre elles**

Dans de nombreuses applications, plusieurs étiquettes radio fréquences peuvent se présenter en même temps dans le champ du lecteur volontairement ou involontairement. Ceci peut être voulu en magasin, au moment du passage à la caisse ou entre les portiques antivols.

- **La sensibilité aux ondes électromagnétiques parasites**

Les systèmes de lecture RFID sont dans certaines circonstances sensibles aux ondes électromagnétiques parasites émises par des équipements informatiques (des écrans d'ordinateurs) ou des systèmes d'éclairages plus généralement par les équipements électriques. Leur emploi doit donc être testé en tenant compte de l'environnement.

- **Les interrogations sur l'impact de la radio fréquence sur la santé**

Cette question fait débat depuis quelques années, en particulier concernant les portiques antivols et les téléphones portables. Les étiquettes passives 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 tels que les pacemakers, mais aussi sur l'organisme humain.

II.10. Communication en champ proche (NFC) [18]

Figure.II.10 Distributeur de billets de train

La communication en champ proche (en anglais nearfield communication, NFC) est une technologie de communication sans-fil à courte portée et haute fréquence, permettant l'échange d'informations entre des périphériques jusqu'à une distance d'environ 10 cm. Cette technologie est une extension de la norme ISO/CEI 14443 standardisant les cartes de proximité utilisant la radio-identification (RFID), qui combinent l'interface d'une carte à puce et un lecteur au sein d'un seul périphérique.

Un périphérique NFC est capable de communiquer avec le matériel ISO/CEI 14443 existant, avec un autre périphérique NFC ou avec certaines infrastructures sans-contact existantes comme les valideurs des transports en commun ou les terminaux de paiement chez les commerçants. La NFC équipe aujourd'hui des cartes utilisées dans les transports, dans le commerce ou pour l'accès à certains services publics et de plus en plus de terminaux mobiles. La NFC équipait en effet en 2011, 50 millions de tablettes tactiles ou téléphones mobiles. Dotés d'un écran, d'un clavier et d'une connexion internet, ces terminaux NFC ont un fort potentiel d'usages en favorisant les interactions entre les machines, les objets et un contexte.

II.10.1. Modes de fonctionnement du NFC

Il y a trois modes de fonctionnement qui sont :

1. Le mode émulation de carte

Dans le mode émulation de carte, dit passif, le terminal mobile se comporte comme une carte à puce sans-contact. Dans le cas où le terminal mobile est un téléphone mobile compatible, la carte SIM de l'opérateur peut être utilisée comme élément de sécurité en stockant des informations chiffrées. Les usages sont multiples: paiement, billetterie spectacle ou transport (ex. : Navigo), couponing, contrôle d'accès. Le mobile, par ses fonctionnalités étendues (IHM, connexion réseau, capacité de traitement), enrichit considérablement les services basés sur des cartes.

2. Le mode lecteur

Le terminal mobile devient un lecteur de cartes sans-contacts (mode actif) ou de « radio-étiquettes » (étiquettes électroniques). Ce mode permet de lire des informations en approchant son mobile devant des étiquettes électroniques disposées dans la rue, sur des abris bus, des monuments, des affiches ...etc, ou sur des colis, des produits ou sur sa carte de visite (vCard)

3. mode pair-à-pair

Ce mode permet à deux terminaux mobiles d'échanger de l'information, par exemple des vCard, des photos, des vidéos, de l'argent, des tickets, etc. Un appareil doté de la technologie NFC est capable d'échanger des informations avec des cartes à puces sans contact mais également avec d'autres appareils dotés de cette technologie.

Ce mode peut également être couplé avec une technologie permettant un transfert plus rapide. Le système d'exploitation mobile Android, permet par exemple depuis la version IceCream Sandwich de transférer les données via une connexion bluetooth, une fois la communication NFC établie; Samsung a amélioré le système en remplaçant, dans S-Beam, la liaison bluetooth par une liaison Wi-Di (ou Wi-Fi Direct)

II.10.2 Quelques exemples d'usages

Les services NFC commencent à voir le jour un peu partout dans le monde et notamment en Corée, en France et en Turquie. Voici quelques exemples d'usages :

- Paiement en utilisant une carte bancaire sans contact, ou un appareil mobile (téléphone portable, smartphone, ordinateur portable, tablette numérique...) sur un terminal de paiement sans contact ;
- Paiement du parking à une borne acceptant le paiement sans contact à l'aide de son terminal mobile NFC ;
- Achat et validation sans contact d'un titre de transport ou d'un billet d'entrée à un spectacle avec son mobile ;
- Gestion de coupons de réduction dans un magasin, gestion de points de fidélité chez les commerçants (couponing) ;
- Accès et démarrage d'un véhicule à l'aide de son téléphone mobile ;
- Lecture d'informations produites (prix, composition, allergie, etc.) dans un magasin ;
- Contrôle d'accès à des locaux en accès réservé (salle de réunion, entreprise, salle de cours, etc.) ;
- Échange de profils entre deux utilisateurs d'un réseau social ou de niveaux de jeux en rapprochant ("tapant") les deux téléphones (mode pair-à-pair) ;
- Lecture d'une carte de visite électronique avec un PDA ;
- Synchronisation de signets Internet et de contacts entre un PDA et un téléphone portable ;
- Récupération de la clef WiFi d'un point d'accès en approchant son «périphérique NFC» de la borne de diffusion ;
- Accès aux fonctionnalités domotique d'un bâtiment.

Conclusion :

La technologie RFID fait partie de notre quotidien et s'utilise dans beaucoup de domaines de la vie courante (transport, gestion des stocks, antivols...). Nous avons vu dans ce chapitre la définition de la RFID et son fonctionnement, sa plage de fréquences, ses avantages et inconvénients et la technologie NFC utilisé dans notre projet. Le chapitre suivant sera consacré pour détailler la phase conception de l'application.

CHAPITRE III

Conception de l'application

Introduction

Maintenant que nous avons parlé du principe de la technologie RFID (NFC) dans le chapitre précédent, nous allons maintenant présenter dans ce chapitre l'objectif de l'application et ses composants. Par la suite nous allons décrire le fonctionnement de l'application.

III. Présentation et objectif de l'application**III.1.1. Présentation**

Pour faciliter la gestion d'un restaurant, une application mobile est utilisée pour prendre la commande du client. Cette application utilise la technologie NFC et permet d'afficher la commande du client sur le tag, celui-ci est transmis au serveur (homme) pour qu'il puisse lui apporter sa commande.

Le tag RFID permet donc :

- De prendre la commande du client lorsque celui-ci touche le tag avec son Smartphone.
- De lui afficher le total à payer.
- De retransmettre la commande du client vers le Smartphone pour qu'il puisse la visualiser.

Nous avons utilisé un tag RFID avec afficheur et un Smartphone compatible avec la technologie NFC qui permet la lecture/écriture sur le tag.

II.1.2. Objectif

En se basant sur la technologie NFC (mode lecture écriture), notre projet consiste à concevoir et réaliser une application Android permettant :

- De transmettre le plat choisi au tag RFID et d'y afficher celui-ci avec son prix
- De modifier la commande en ajoutant ou en supprimant des plats
- D'afficher le total à payer sur le tag RFID

III.2. Description du système

Le but de l'application est de montrer l'utilité du tag RFID dans la gestion du restaurant. Nous avons utilisé une base de données avec un ID pour chaque plat pour

qu'il puisse être enregistré sur le tag. La base de donnée permet de stocker et restituer les informations du plat correspondant à l'ID. Nous nous sommes servi du serveur de base de données par défaut contenu dans Eclipse qui est le SQLite.

Notre système utilise un modèle embarqué, il se limite à la communication entre le tag et l'application client :

- L'application client : l'application Android qui s'exécute sur un Smartphone compatible avec la technologie NFC responsable de la manipulation du tag et de la transmission et réception des données vers/depuis le tag, avec un système de gestion de base de données SQLite, intégrée dans l'application.

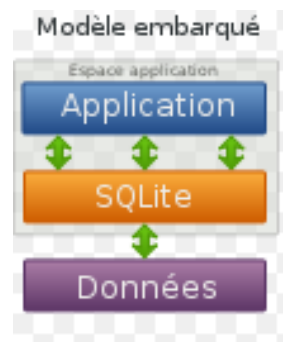


Figure III.1 : modèle embarqué

- Le tag : envoie ou reçoit les données de l'application.

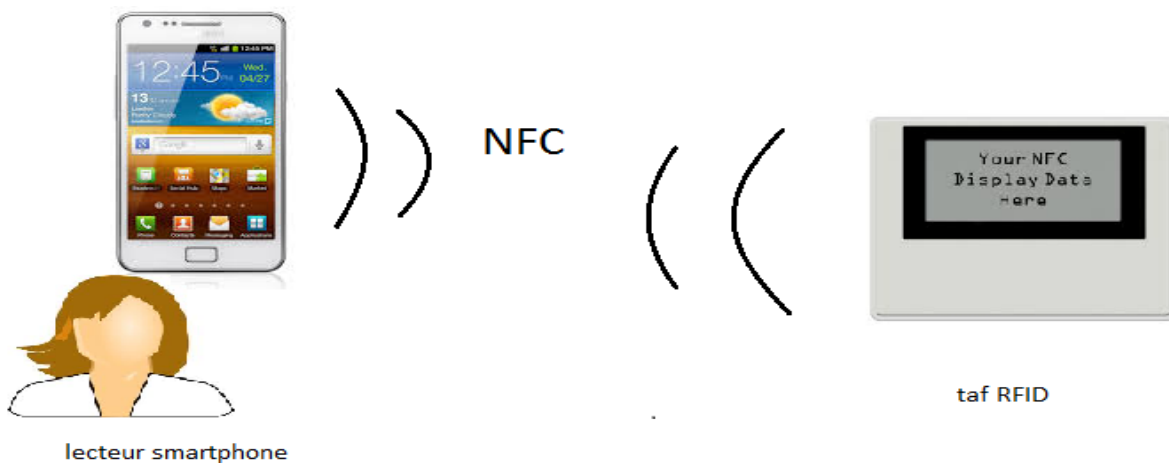


Figure III.2 : Architecture de notre système

III.3. Description des composants du système :

Nous allons décrire dans cette partie les composants de notre système en détail.

III.3.1. Le Tag RFID

Le tag RFID est le cœur de notre système, il est utilisé pour identifier Les informations qu'on lui transmet par Smartphone. Il se suffit à lui-même car il dispose d'un afficheur. Il contiendra le nom du plat, le prix et le total à payer.

Exemple : Omelette espagnole 350 DA

Ces informations (nom du plat, prix) sont introduites dans le tag par l'application Android implémentée au niveau du Smartphone en utilisant la technologie NFC.

III.3.2. Le Lecteur

Il assure la communication et l'affichage d'information dans notre système, il représente un Smartphone compatible avec NFC. La fonction principale de notre lecteur est :

- échange de donnée avec le tag : En utilisant la technologie NFC, le lecteur permet au client d'écrire des données (nom du plat, prix) dans la mémoire du tag, aussi il interroge le tag afin de récupérer ce qui est affiché sur son écran. Pour ce faire le lecteur reçoit l'identifiant du tag qui se trouve dans son champ de lecture NFC. Il récupère les informations de la base de données SQLite qui lui renverra la réponse.

Les requêtes envoyées par l'application vers le serveur de base de données SQLite ont la forme :

- [ID_plat, « nom_plat »] : pour récupérer les informations correspondantes au nom du plat.
- [ID_plat, « prix_plat »] : pour récupérer les informations correspondantes au prix du plat.

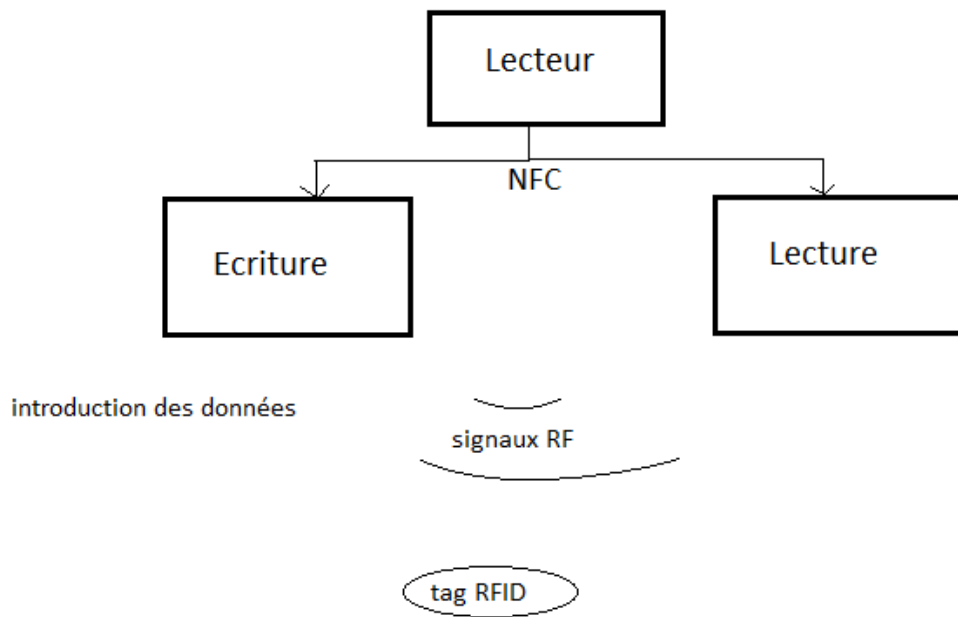


Figure III.3 : fonctionnement du lecteur

III.3.3. Le serveur de base de données intégré

Stocke une base de données dans la table suivante :

Table plat :

Elle stock les données de tous les plats du restaurant, elle est sollicitée par le tag lors de la commande du client.

Non de colonne	Désignation	Type
<u>PL_ID</u>	Identifiant du plat	Caractère
PL_NAME	Nom du plat	Caractère
PL_PRIX	Prix du plat	Caractère

Tableau III.1 : table plat

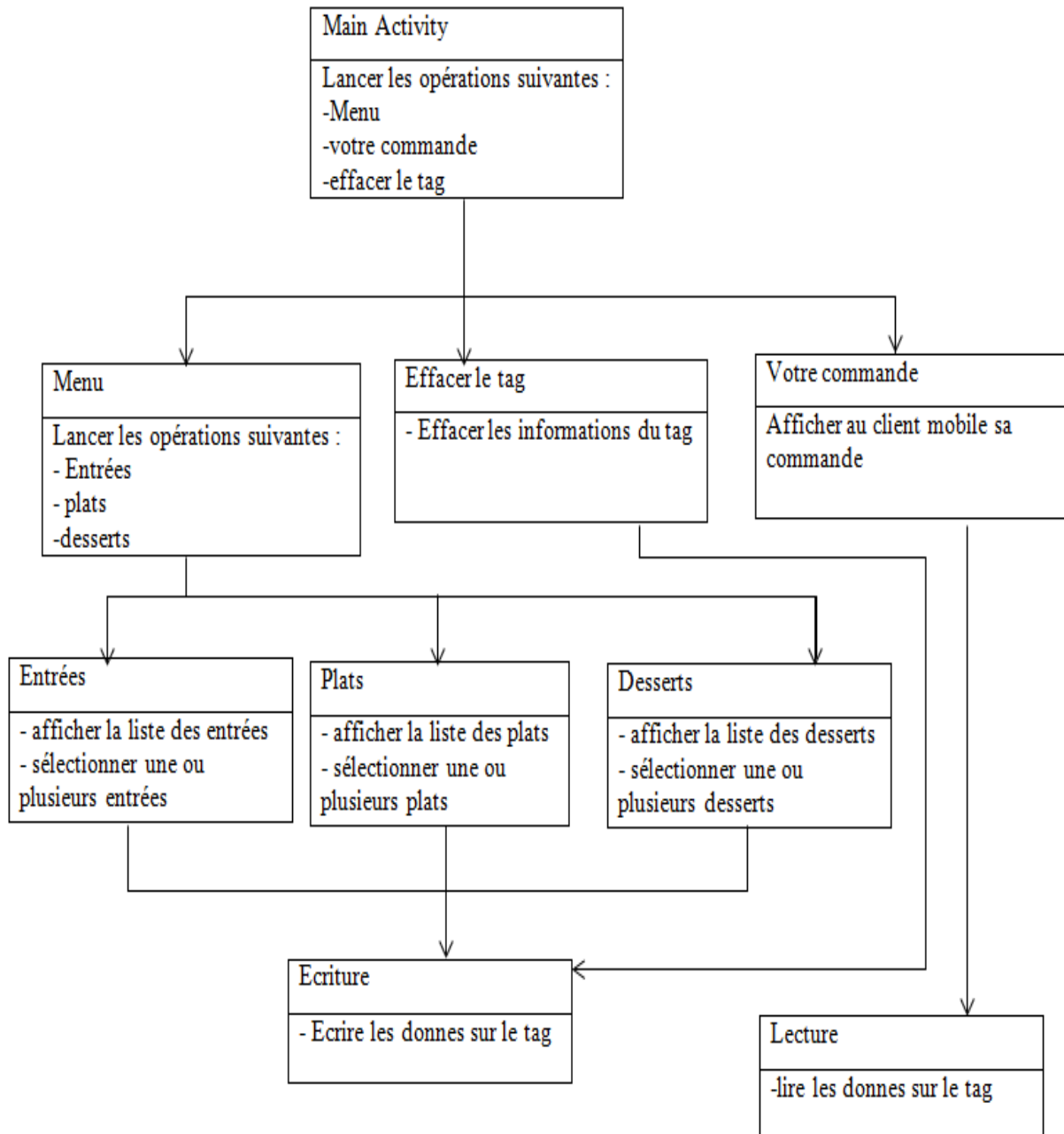


Figure III.4 : diagramme de l'application

III.3.4. Rôle des activités, classe

Main activity : c'est l'activité principale qui gère les autres activités.

Menu : permet à l'utilisateur de choisir entre entrées, plats ou desserts.

Entrées, plats, dessert : affiche une liste de plats, l'utilisateur en choisi un, rapproche le tag du téléphone et sa commande s'affiche sur le tag.

Votre commande : récupère la liste des plats choisis à partir du tag.

Effacer tag : efface les données du tag.

Conclusion

Dans ce chapitre nous avons tout d'abord présenté les objectifs de notre application et par la suite l'architecture de notre système en expliquant ces composants et son fonctionnement. Dans le chapitre qui suit nous allons présenter les outils matériels et logiciels utilisés dans le développement de notre application.

CHAPITRE IV

Réalisation de l'application

Introduction

Nous allons présenter dans ce chapitre les outils logiciels et matériels utilisés pour le développement de notre application ainsi que ses différentes interfaces.

IV.1 environnement et outils de développement

IV.1.1 matériel

Constitué d'un ensemble de ressources utilisées pour mettre en marche notre application.

Nous avons donc :

IV.1.1.1. le tag RFID

Pour permettre au client de transmettre sa liste de commande au serveur (homme) nous avons utilisé un tag RFID fourni par AIOI qui est un tag unique en son genre. Il utilise un type 3 Felica IC. Le smart tag permet au donnée interne de s'afficher via l'écran e-paper. Son espace optimal est de 3KO. Il a aussi les caractéristiques suivantes :

Dimension	3x58x8.5mm
Poids	30g
Affichage	200x96 (pixel)
Interface	Felica
Durée de vie d'affichage	10,000 reads/writes or 5 years
Intervalle de température	0 ~40 Degrees C
Durée de la réécriture	Approx 2 Sec

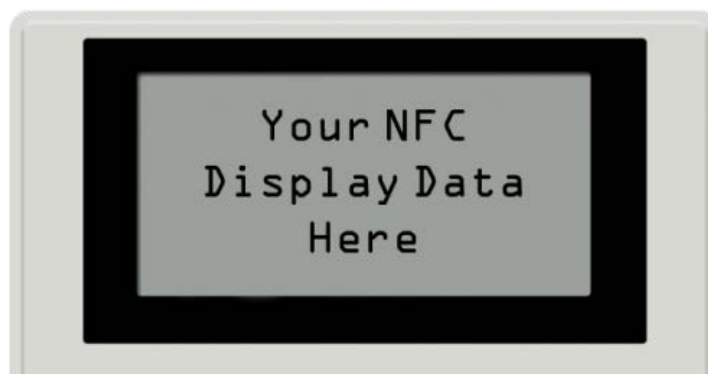


Figure IV.1 : smart tag AIOI

IV.1.1.2. Le Smartphone :

Pour interroger le tag RFID notre application client est exécutée sur un Samsung Galaxy S2 Cityzi qui a les caractéristiques suivantes :

Dimensions	125 x 66 x 8,89 mm
Processeur Samsung Exynos	4210 - 1.2 GHz
Système d'exploitation (OS)	Android 4.1.2 (JellyBeans)
Mémoire interne	16 Go, 1 Go RAM
Compatible avec la technologie	NFC



Figure IV.2: Samsung Galaxy S2 Cityzi

IV.1.2. logiciel

IV.1.2.1. le langage JAVA [2]

C'est le langage utilisé pour programmer notre application mobile.

- **Description :**

Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java.

Le langage Java reprend en grande partie la syntaxe du langage C++, très utilisée par les informaticiens. Néanmoins, Java a été épuré des concepts les plus subtils du C++ et à la fois les plus déroutants, tels que les pointeurs et références, ou l'héritage multiple contourné par l'implémentation des interfaces. Les concepteurs ont privilégié l'approche orientée objet de sorte qu'en Java, tout est objet à l'exception des types primitifs (nombres entiers, nombres à virgule flottante, etc.).

Java permet de développer des applications client-serveur. Côté client, les applets sont à l'origine de la notoriété du langage. C'est surtout côté serveur que Java s'est imposé dans le milieu de l'entreprise grâce aux servlets, et plus récemment les JSP (JavaServer Pages) qui peuvent se substituer à PHP, ASP et ASP.NET.

Java a donné naissance à un système d'exploitation (JavaOS), à des environnements de développement (eclipse/JDK), des machines virtuelles (MSJVM (en), JRE) applicatives multiplate-forme (JVM), une déclinaison pour les périphériques mobiles/embarqués (J2ME), une bibliothèque de conception d'interface graphique (AWT/Swing), des applications lourdes (Jude, Oracle SQL Worksheet, etc.), des technologies web (servlets, applets) et une déclinaison pour l'entreprise (J2EE). La portabilité du bytecode Java est assurée par la machine virtuelle Java, et éventuellement par des bibliothèques standard incluses dans un JRE. Cette machine virtuelle peut interpréter le bytecode ou le compiler à la volée en langage machine. La portabilité est dépendante de la qualité de portage des JVM sur chaque OS.

Les caractéristiques les plus intéressantes de java sont :

- Portabilité
 - Langage indépendant de la machine cible.
- Sécurité
 - interprétation dans la machine virtuelle
- véritable langage objet.

- Un mariage entre smalltalk et C++, « plus pur » en quelque sorte que C++
- intégration de concept présent dans UML comme la notion d'interface.
- l'héritage multiple de code est abandonné, par contre l'héritage privé n'est pas implémenté.
- Multi threaded
- Présence d'un « Ramasse Miettes » (garbage collector).
- Environnement de programmation JDK
 - Complet de l'image au son en passant par le réseau.
 - * AWT
 - * Son
 - * Mécanisme de réflexion (les objets, les types, les services sont considérés comme des objets de première classe).

- **IDE Eclipse [12]**

Eclipse est un projet, décliné et organisé en un ensemble de sous-projets de développements logiciels, de la Fondation Eclipse visant à développer un environnement de production de logiciels libre qui soit extensible, universel et polyvalent, en s'appuyant principalement sur Java.

Son objectif est de produire et fournir des outils pour la réalisation de logiciels, englobant les activités de programmation (notamment environnement de développement intégré et frameworks) mais aussi d'AGL recouvrant modélisation, conception, testing, gestion de configuration, reporting... Son EDI, partie intégrante du projet, vise notamment à supporter tout langage de programmation à l'instar de Microsoft Visual Studio.

Bien qu'Eclipse ait d'abord été conçu uniquement pour produire des environnements de développement, les utilisateurs et contributeurs se sont rapidement mis à réutiliser ses briques logicielles pour des applications clientes classiques. Cela a conduit à une extension du périmètre initial d'Eclipse à toute production de logiciel : c'est l'apparition du framework Eclipse RCP en 2004.

Figurant parmi les grandes réussites de l'Open source, Eclipse est devenu un standard du marché des logiciels de développement, intégré par de grands éditeurs logiciels et sociétés de services. Les logiciels commerciaux Lotus Notes 8, IBM Lotus Symphony ou WebSphere Studio Application Developer sont notamment basés sur Eclipse.

Dans notre projet nous avons utilisé la version eclipse INDIGO

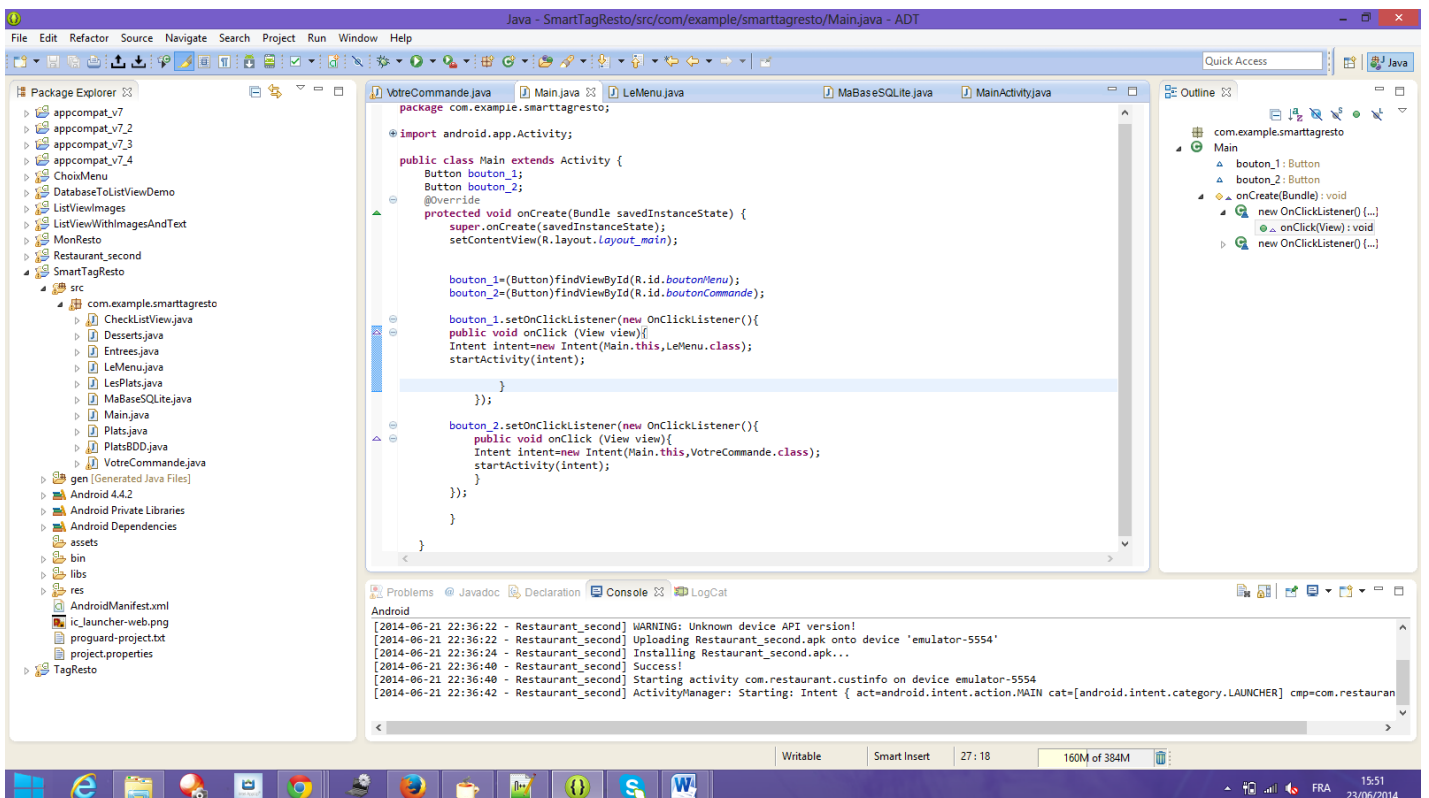


Figure IV.3: interface eclipse

IV.1.2.2. l'environnement android

- **Le kit de développement (SDK) [9]**

La SDK d'Android est un ensemble complet d'outils de développement. Il inclut un débogueur, des bibliothèques logicielles, un émulateur basé sur QEMU, de la documentation, des exemples de code et des tutoriaux. Les plateformes de

développement prises en charge par ce kit sont les distributions sous Noyau Linux, Mac OS X 10.5.8 ou plus, Windows XP ou version ultérieure. L'IDE officiellement supporté est Eclipse combiné au plugin d'outils de développement d'Android (ADT). Les développeurs peuvent utiliser n'importe quel éditeur de texte pour modifier les fichiers Java et XML, puis utiliser les outils en ligne de commande (Java Development Kit et Apache Ant sont obligatoires) pour créer, construire et déboguer des applications Android ainsi que contrôler des périphériques Android (pour déclencher un redémarrage, installer un logiciel à distance ou autre).

Notre application a été développée pour la version d'Android 4.1.2. Cette version permet de manipuler la NFC du smart phone.

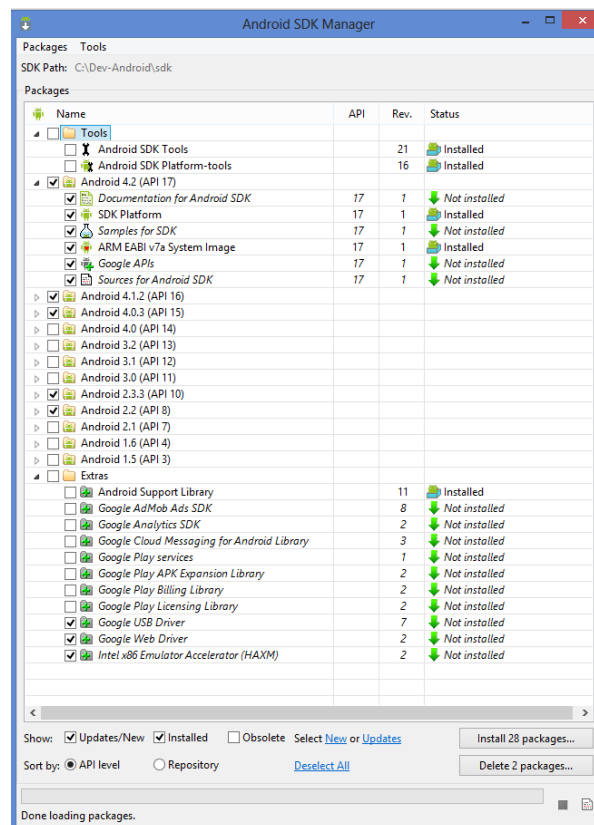


Figure IV.4 : SDK Manager

- **ADT pour Android :**

ADT (Android Developer Tools) est un plugin pour Eclipse qui fournit une suite d'outils qui sont intégrés à l'IDE Eclipse. Il offre un accès à de nombreuses fonctionnalités qui permettent de développer des applications Android rapidement.

ADT propose l'accès à l'interface graphique de la plupart des outils du SDK ainsi que d'un outil de conception d'interface utilisateur pour le prototypage rapide, la conception et la construction de l'interface utilisateur de l'application.

Parce que ADT est un plugin pour Eclipse, on obtiens les fonctionnalités d'un IDE bien établi, ainsi que des fonctionnalités spécifiques à Android qui sont regroupés avec ADT.

- **L'émulateur Android (AVD) [8]**

Un Android Virtual Device (AVD) est une configuration de l'émulateur qui permet de modéliser un dispositif réel en définissant des options matérielles et logicielles émuls par l'émulateur Android.

Un AVD se compose de :

Un profil matériel : Définit les caractéristiques matérielles de l'appareil virtuel. Par exemple, vous pouvez définir si l'appareil dispose d'un appareil photo, si elle utilise un clavier QWERTY physique ou un clavier de composition, la quantité de mémoire qu'il a, et ainsi de suite.

Une cartographie à une image du système: pour définir la version de la plate-forme Android qui sera exécutée sur le périphérique virtuel. Vous pouvez choisir une version de la plate-forme Android standard.

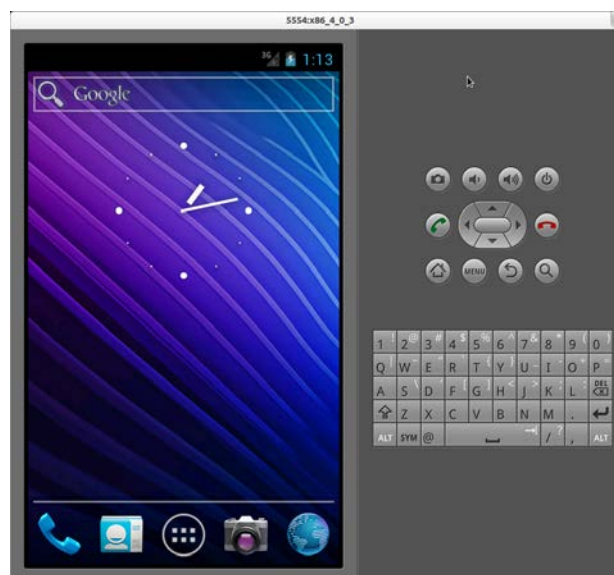


Figure IV.5 : Emulateur Android

IV.1.2.3. SQLite serveur de base de données intégré [5]**Description :**

SQLite est une bibliothèque écrite en C qui propose un moteur de base de données relationnelle accessible par le langage SQL. SQLite implémente en grande partie le standard SQL-92 et des propriétés ACID.

Contrairement aux serveurs de bases de données traditionnels, comme MySQL ou PostgreSQL, sa particularité est de ne pas reproduire le schéma habituel client-serveur mais d'être directement intégrée aux programmes. L'intégralité de la base de données (déclarations, tables, index et données) est stockée dans un fichier indépendant de la plateforme.

D. Richard Hipp, le créateur de SQLite, a choisi de mettre cette bibliothèque ainsi que son code source dans le domaine public, ce qui permet son utilisation sans restriction aussi bien dans les projets open source que dans les projets propriétaires. Le créateur ainsi qu'une partie des développeurs principaux de SQLite sont employés par la société américaine Hwaci.

SQLite est le moteur de base de données le plus distribué au monde, grâce à son utilisation dans de nombreux logiciels grand public comme Firefox, Skype, Google Gears, dans certains produits d'Apple, d'Adobe et de McAfee et dans les bibliothèques standards de nombreux langages comme PHP ou Python. De par son extrême légèreté (moins de 300 Ko), il est également très populaire sur les systèmes embarqués, notamment sur la plupart des Smartphones modernes : l'iPhone ainsi que les systèmes d'exploitation mobiles Symbian et Android l'utilisent comme base de données embarquée³. Au total, on peut dénombrer plus d'un milliard de copies connues et déclarées de la bibliothèque.

IV.2. Présentation de notre application :

Notre application se compose de la partie client qui est l'application Android, elle communique avec le serveur de base de donnée SQLite pour récupérer ou insérer des informations. Ces informations sont ensuite transmises au tag.

IV.2.1. L'application client :

Elle représente l'application Android qui se déroule comme suit :

- En lançant l'application la fenêtre principale apparaît, elle contient les boutons qui mènent vers 3 différentes interfaces.

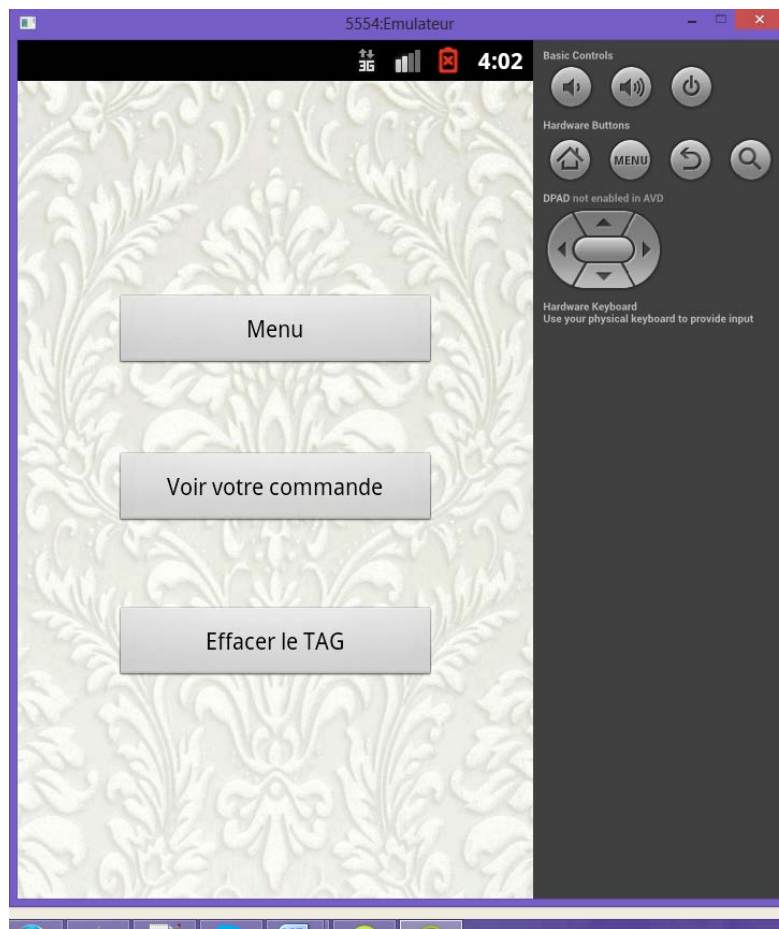


Figure IV.6 fenêtre principale

- Lorsqu'on choisit de cliquer sur le bouton Menu, on accède à la fenêtre proposant un choix.



Figure IV.7 fenêtre menu

- Si on choisit les desserts par exemple on accède à une liste de choix contenant différents desserts.



Figure IV.8 liste des desserts

- Quand le client sélectionne un ou plusieurs desserts, un toast le prévient du choix qu'il vient d'effectuer.

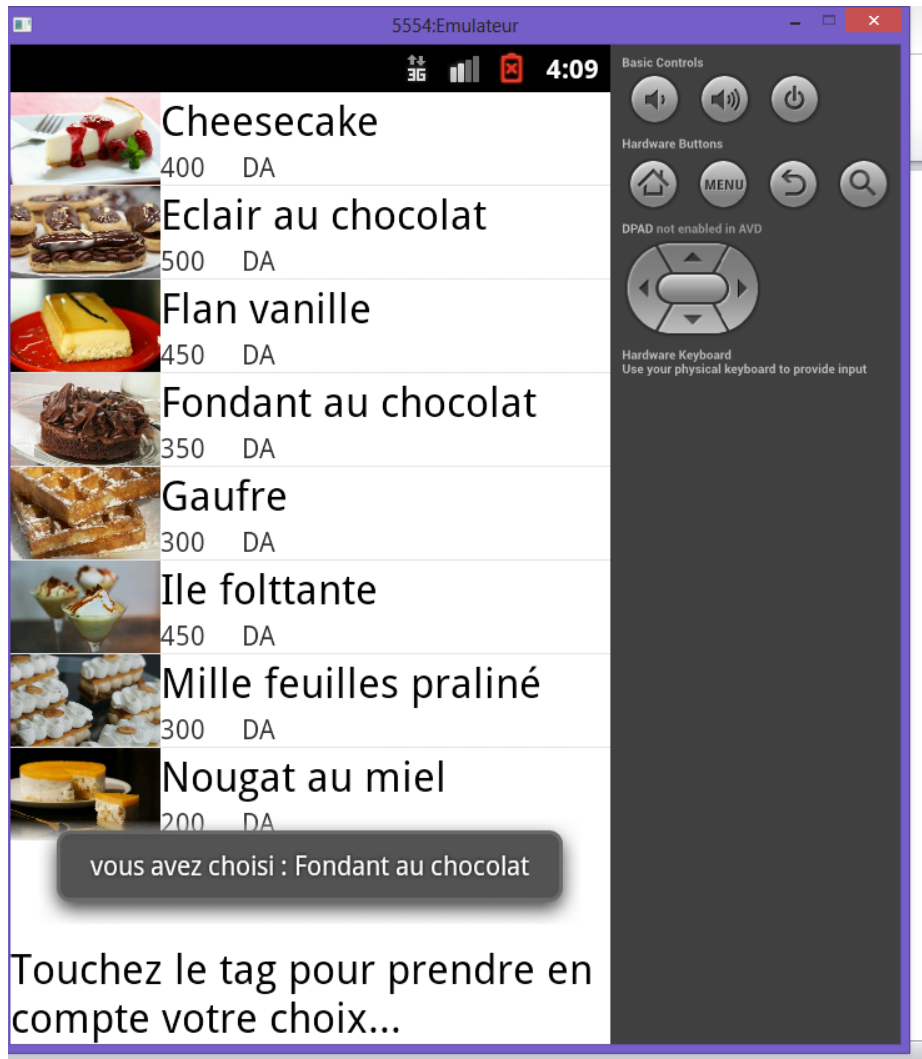


Figure IV.9 choix du client.

IV.2.2. Partie base de donnée :

Contient une base de données SQLite contenant la table de notre application.

Conclusion :

Dans ce chapitre, nous avons d'abord présenté l'environnement et les outils utilisés pour implémenter notre application. Ensuite nous avons vu quelques interfaces de l'application réalisée.

Conclusion générale

CONCLUSION GENERALE

A l'issue de la réalisation de ce travail, nous pouvons affirmer que notre projet nous a été d'une grande utilité dans la mesure où il nous a permis d'enrichir nos connaissances sur les systèmes embarqués, la technologie RFID et de nous familiariser avec le travail sur la plate-forme Android.

Le but de notre travail est d'étudier la technologie RFID à travers une application permettant la gestion des commandes d'un restaurant. Une fois les objectifs de notre application fixés, nous avons enchaîné avec la conception afin de mener à bien notre projet. Au cours de la phase de réalisation, nous nous sommes familiarisés avec le langage de programmation JAVA.

Pour conclure, notre travail peut être sujet à des améliorations et éventuellement des extensions. En effet, nous envisageons de modifier les parties sujettes à des craches et d'ajouter à notre application ce qui lui manque pour son bon fonctionnement.

Bibliographie

Bibliographie

- [1] **Learn Java for Android Development** Par *Jeff Friese*
- [2] **Teach Yourself Java in 24 Hours** Par *Rogers Cadenhead*
- [3] **Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems** Par *Peter Marwedel*
- [4] **Introduction pratique aux bases de données relationnelles** Par *Andreas Meier*
- [5] **Using SQLite** Par *Jay A. Kreibich*
- [6] **L'Art du développement Android** Par *Mark L. Murphy*
- [7] **Développement d'applications professionnelles avec Android 2** Par *Reto Meier*
- [8] **Developpez Pour Android** Par *Ludovic Perrier et Cyril Mottier*
- [9] **Android Application Development All-in-One For Dummies** Par *Bur*
- [10] **RFID-Enabled Sensor Design and Applications** Par *Amin Rida, Li Yang, Manos M. Tentzeris*
- [11] **Near Field Communication (NFC): From Theory to Practice** Par *Vedat Coskun, Kerem Ok, Busra Ozdenizci*
- [12] **Android Apps with Eclipse** Par *Onur Cinar*
- [13] **Mémoire de fin d'étude sur le contrôle d'accès et gestion de la bibliothèque par les tags RFID**, par Sid ali Mourad et Slimani Khaled

Webliographie

- [14] <http://dept-info.labri.fr/ENSEIGNEMENT/programmation2/Cours/Generalites.pdf>
- [15] <http://www.android.com/>
- [16] <http://fr.wikipedia.org/wiki/Android>
- [17] <http://android-france.fr/>
- [18] <fr.wikipedia.org/wiki/Radio-identification>
- [19] <http://www.centrenational-rfid.com/definition-de-la-rfid-article-71-fr-ruid-17.html>
- [20] <http://rfid.comprendrechoisir.com/comprendre/systeme-rfid>
- [21] http://fr.wikipedia.org/wiki/Système_embarqué
- [22] <www.irit.fr/~Zoubir.Mammeri/Cours/M2PConcSTR/Chap2STRbases.pdf>
- [23] <http://fr.openclassrooms.com/informatique/cours/creez-des-applications-pour-android/1-architecture-d-android>