

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

Université de Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj

Faculté des Sciences et de la technologie

Département Eléctronique

Mémoire

Présenté pour obtenir

LE DIPLOME DE MASTER

FILIERE : ELECTRONIQUE

Spécialité : Électronique des systèmes embarqués

Par

- **Ouchene Mokrane**
- **Lahouassa Anis**

Intitulé

*Développement d'une application Android de lecture/génération
de QR Codes*

Soutenu le : /.... / 2022

Devant le Jury composé de :

<i>Nom & Prénom</i>	<i>Grade</i>	<i>Qualité</i>	<i>Etablissement</i>
<i>Mlle. Nacira DIFFELLAH</i>	<i>MCA</i>	<i>Président</i>	<i>Univ-BBA</i>
<i>M. Abdelhakim LATOUI</i>	<i>MCA</i>	<i>Encadreur</i>	<i>Univ-BBA</i>
<i>M. M^{ed}Elhoussine DAACHI</i>	<i>MCA</i>	<i>Examineur</i>	<i>Univ-BBA</i>

Année Universitaire 2021/2022

Remerciements

Nous remercions en premier lieu le bon dieu qui nous a donné
la force et le courage pour terminer ce travail.

Le travail présenté dans ce projet de fin d'études a été effectué
dans le cadre de la préparation du diplôme du master en Eléctronique spécialité
« Électronique des systèmes embarqués » à l'Université de Mohamed El
Bachir El Ibrahimi de Bordj Bou Arréridj.

Nous souhaitons adresser nos remerciements les plus sincères
aux personnes qui nous ont apporté leur aide et qui ont
contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de
cette formidable année universitaire.

La première personne que nous tenons à remercier est notre encadreur

Monsieur Latoui Abdelhakim, pour

l'orientation, la confiance et la patience qui ont constitué un
apport considérable sans lequel ce travail n'aurait pas pu être
mené au bon port. Qu'il trouve dans ce travail un hommage
vivant à sa haute personnalité

Dédicace

Nous dédions ce modeste mémoire à :

Nos chers parents

Que nulle dédicace ne puisse exprimer ce que nous leurs devons, pour tous leurs sacrifices, leur bienveillance, leur amour, leur tendresse, leur soutien et leurs prières tout au long de nos études. Que ce travail soit témoignage de notre profond amour et notre grande reconnaissance « Que Dieu vous garde ».

Nos chères sœurs et nos chers frères

Pour leurs encouragements permanents, et leur soutien moral, Nous leur dédie ce modeste travail en témoignage de notre grand amour et notre gratitude infinie.

Toutes nos Familles

A tous les familles OUCHENE et LAHOUASSA, pour leur soutien tout au long de mon parcours universitaire, Que ce travail soit l'accomplissement de vos vœux tant allégués, et le fruit de votre soutien infailible.

Tous nos amis

Pour leur aide et leur soutien moral durant l'élaboration du travail de fin d'études

Résumé

L'utilisation très élargie du code-barres et du QR code dans tous les domaines de la vie oblige tout un chacun à posséder un moyen efficace pour les lire. Aussi, avec le développement gigantesque du marché du smartphone, l'outil idéal est sans doute l'application mobile dédiée à cet effet. Dans ce cadre, et dans le cadre de notre projet de fin d'études Master 2 électronique des systèmes embarqués, nous sommes amenés à réaliser et développer une application capable de générer et de lire ces deux types de codes. Cette application exécute les trois tâches essentielles à savoir : **Generate** qui permet de générer le QR code ou le code-barres correspondant au texte saisi ; **Save** qui permet de sauvegarder le QR code généré sous format d'une image JPEG et **Scan** qui permet de lancer la lecture (scan) du code-barre ou du QR code présents devant la caméra du smartphone et d'en afficher le texte correspondant.

Abstract

The widespread use of barcodes and QR codes in all domains requires everyone to have an effective method to read them. With the gigantic development of the Smartphone market, this ideal tool is undoubtedly the suitable one. In this paper, this project aims to create and develop an application capable of generating and reading these two types of codes. This application performs the three essential tasks, namely: Generate which allows you to generate the QR code or the barcode corresponding to the text entered; Save which can help you to save the generated QR code in the format of a JPEG image and Scan which means you may start reading (scanning) the barcode or QR code present in front of the smartphone camera and display and the corresponding text.

الملخص

يتطلب الاستخدام الواسع النطاق للرموز الشريطية (code-barres) ورموز الاستجابة السريعة (QR codes) في جميع مجالات الحياة أن يكون لدى جميع المستخدمين أداة فعالة لقراءتها. ومع التطور الهائل لسوق الهواتف الذكية، فإن الأداة المثالية بلا شك هي تطبيق الهاتف المحمول المخصص لهذا الغرض. في هذا السياق، وكجزء من مشروع تخرجنا قصد الحصول على شهادة الماجستير في الإلكترونيات تخصص الأنظمة المدمجة، فإننا قمنا بتطوير تطبيق محمول مهمته الأساسية إنشاء وقراءة هذين النوعين من الرموز. هذا التطبيق مبني على أساس يجعله قادرا على تنفيذ المهام الأساسية الثلاث وهي: إنشاء رمز الاستجابة السريعة (QR codes) أو الرمز الشريطي (code-barres) المقابل للنص الذي يتم إدخاله كما يتيح مهمة حفظ الرمز الذي تم إنشاؤه بتنسيق صورة JPEG والمسح الضوئي للرمز الشريطي أو رمز الاستجابة السريعة الموجود أمام كاميرا الهاتف الذكي وعرض النص المقابل.

SOMMAIRE

Introduction générale	01
Chapitre 1 : Introduction aux technologies sans contact	
1.1. Introduction	02
1.2. Définition	03
1.3. Différents types et formes de technologies sans contact	03
1.3-1. Le wifi	03
1.3-2. Le Bluetooth	04
1.3-3. La RFID	05
1.3-4. NFC	07
1.3-5. iBeacon ou la géolocalisation de proximité	07
1.3-6. R.I (Reconnaissance d'images)	09
1.3-7. QR code	10
1.4. Usages et domaines d'application des technologies sans contact	10
1.4-1. Le wifi	10
1.4-2. Le Bluetooth	11
1.4-3. La RFID	11
1.4-4. NFC	12
1.4-5. iBeacon ou la géolocalisation de proximité	12
1.4-6. R.I (Reconnaissance d'images)	13
1.4-7. QR code	13
1.5. Conclusion	15
Chapitre 2 : Généralités sur les smartphones et la plate forme de développement mobile sous Android	
2.1. Introduction	16
2.2. Les Smartphones	16
2.2-1. Définition	16
2.2-2. Systèmes d'exploitation	17
2.2-2-1. Android de Google	17
2.2-2-2. iOS d'APPLE	18
2.2-2-3. Windows Phone de Microsoft	19
2.2-2-4. BlackBerry OS de RIM	19
2.2-2-5. Autres systèmes d'exploitation pour mobiles	19
2-3. Plate forme de développement mobile sous Android	20
2-3-1. Définition	20
2-3-2. Architecture logicielle d'Android	20
2-3-2-1. Le noyau Linux	21
2-3-2-2. Les bibliothèques graphiques	21
2-3-2-3. Une Machine Virtuelle (Dalvik Virtual Machine)	21
2-3-2-4. Framework applicatif	21
2-3-2-5. Application	21
2-3-3. Kit de développement d'Android	22
2-3-3-1. Les outils	22
2-3-3-2. Les exemples de code	23
2-3-3-3. La documentation	23
2-3-3-4. Les API	23
2-3-4. Versions d'Android	23

2-4. Les applications mobiles	26
2-4-1. Définition	26
2-4-2. Stratégies de développement	26
2-4-2-1. Application native	26
2-4-2-2. Application Web	27
2-4-2-3. Application hybride	27
2-4-3. Environnement de développement	27
2-4-3-1. Android Studio	28
2-4-3-2. Eclipse	28
2-4-3-3. Netbeans	28
2-4-3-4. Autres environnements de programmation Android	28
2-5. Conclusion	29

Chapitre 3 : Conception et réalisation d'une application de lecture/génération de QR codes sous Android

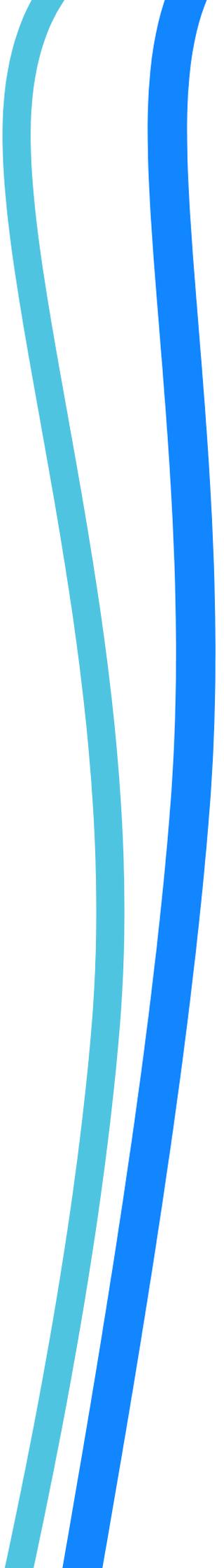
3.1. Introduction	30
3.2. Code-barres et QR code	30
3.2.1. Code-barres	30
3.2.2. Le QR Code	32
3.3. Environnement de développement Android Studio	33
3.3.1. Android Studio et ses bibliothèques	34
3.3.2. Structure d'une application Android	35
3.3.3. Composantes d'une application Android	36
3.3.3.1. Les Activités	36
3.3.3.2. Les Services	36
3.3.3.3. Les Broadcast Receivers	37
3.3.3.4. Les Content Providers	37
3.4. Développement de l'application	37
3.4.1. Installation des bibliothèques	37
3.4.2. Les différentes fonctions	38
3.4.3. Fonctionnalités de l'application	41
3.4.3.1. Fonctionnalité Qr Code	42
3.4.3.2. Fonctionnalité Bar Code	44
3.4.3.3. Fonctionnalité Scanner	45
3.4.3.4. Fonctionnalité Encrypt Qr	46
3.4.3.5. Fonctionnalité Decrypt Qr	47
3.4.3.6. Le partage des images QR code	48
3.4.3.7. La fonctionnalité « à propos »	49
3.5. Conclusion	51
Conclusion générale	52

LISTE DES FIGURES

	<i>Page</i>
Figure. 1.1. Le réseau WIFI	03
Figure.1.2 : Processus d'appairage via Bluetooth	05
Figure.1.3 : Schéma illustrant un système RFID UHF passif	06
Figure.1.4 : La technologie NFC et ses utilisations	07
Figure.1.5 : Balises iBeacon du marché	08
Figure.1.6 : Reconnaissance d'images (faciale) de Google	09
Figure.1.7 : QRcode renvoyant à un site web (lien)	10
Figure 2.1 : Vente mondiale de smartphones Android et iOS en 2010 et 2020	18
Figure 2.2 : Architecture logicielle de l'Android	22
Figure 3.1 : Structure d'un code-barres correspondant au GTIN 6 Algérie	31
Figure 3.2 : Structure générale du QR code	32
Figure 3.3 : Ecran d'accueil de Android Studio Ver 2021.2.1	33
Figure 3.4 : Environnement de travail d'Android Studio	34
Figure 3.5 : Structure générale d'une application Android	35
Figure 3.6 : Installation des librairies ZXing (Zebra Crossing) et Yuriy-Budiyev	38
Figure 3.7 : Code de la fonction QRScanner	39
Figure 3.9 : Code de la fonction SaveImage	40
Figure 3.8 : Code de la fonction QrGenerator.....	40
Figure 3.10 : Interface et principales fonctionnalités de l'application	41
Figure 3.11 : Ecran d'accueil de la fonctionnalité Qr Code	42
Figure 3.12 : Génération du Qr Code avec sauvegarde et partage	43
Figure 3.13 : Ecran d'accueil de la fonctionnalité Barcode	44
Figure 3.14 : Génération du code-barres avec sauvegarde et partage	45
Figure 3.15 : Ecran de scan QR code ou code-barres	46
Figure 3.16 : Ecran de génération de QR code crypté	47
Figure 3.17 : Ecran de scan et de décryptage d'un QR code crypté	48
Figure 3.18 : Ecran de scan et de décryptage d'un QR code crypté	49
Figure 3.19 : Ecran « about us »	50

LISTE DES TABLEAUX

	<i>Page</i>
Tableau 2.1 : Vente mondiale de smartphones selon l'OS de 2017 à 2021	17
Tableau 2.2 : Différentes versions de la plate forme Android.....	25



Introduction générale

Introduction générale

Introduction générale :

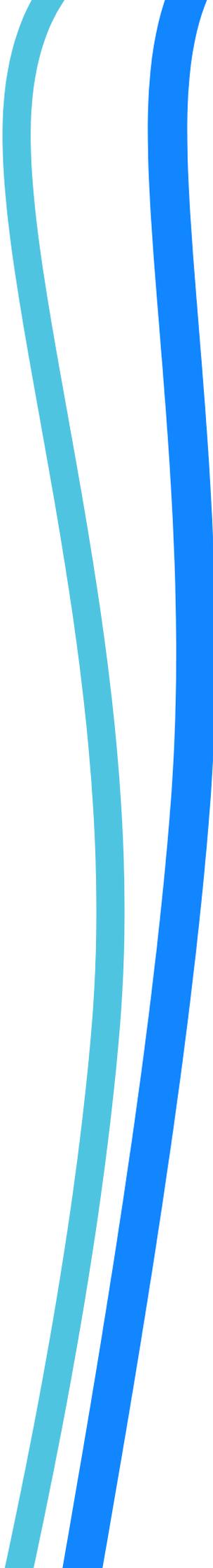
Avec l'avancée et l'émergence des technologies mobiles, les développements embarqués sont de plus en plus demandés sur le marché. Avoir un Smartphone est devenu incontournable car il est devenu très courant et de plus en plus accessible pour tout le monde.

De plus, les applications mobiles pénètrent dans tous les domaines de la vie, elles deviennent ainsi tant nécessaires qu'accessibles et multiplient leurs champs d'action. Elles sont d'ailleurs capables de satisfaire un large éventail de besoins. Grâce à leur rapidité d'accès à l'information ainsi que la possibilité de profiter de plusieurs fonctionnalités novatrices, ludiques et intuitives qui repoussent les limites de l'expérience utilisateur comme : l'appareil photo, vidéo, le GPS... en plus elles offrent une interface plus spécifique et plus harmonieuse qui les rend plus aisées que les sites mobiles dans leur utilisation.

Avec l'avènement du QR code, l'accès à l'information et son partage deviennent plus faciles. Cette nouvelle technologie offre en outre à l'utilisateur de nouvelles fonctionnalités faciles à utiliser et très rapides. Ainsi, en scannant un QR code, on peut facilement obtenir l'information souhaitée, avoir accès à un site internet ou avoir accès un à réseau pour ne citer que quelques possibilités offertes par l'utilisation devenue très large ces dernières années du QR code de par le monde.

Dans cette visée, et pour répondre aux besoins des utilisateurs en applications mobiles dans le domaine du QR code, notre projet porte sur le développement d'une application Android de lecture/génération de QR codes et codes-barres

Ce mémoire est organisé en trois (3) chapitres. Les deux premiers chapitres sont d'ordre purement théorique ; alors que le premier chapitre consiste en une introduction aux technologies sans contact ou sans fil, leurs différents types et domaines d'utilisation, le second présentera des généralités sur les smartphones et la plateforme de développement Android. Le troisième et dernier chapitre quant à lui est d'ordre pratique où nous réaliserons une application Android capable de générer et de lire des QR codes ou des codes-barres en utilisant le kit de développement Android Studio.



Chapier I

Introduction aux technologies sans contact

Chapitre 1 : Introduction aux technologies sans contact

1.1. Introduction :

La technologie ne cesse de progresser au jour le jour. Quel que soit le domaine, le but est la disparition des câbles ou tout au moins leur réduction. Nous nous dirigeons petit à petit vers un monde sans fil [1].

En effet, la technologie sans fil est désormais répandue sur la grande majorité des appareils électroniques. Que vous ayez besoin de faire usage d'un moniteur de santé, de partager des fichiers entre deux ou plusieurs appareils (imprimante, smartphone, etc.), de profiter d'une ambiance musicale (écouteur, enceinte, etc.), les câbles sont de moins en moins sollicités.

Depuis déjà plus de 20 ans, la technologie sans contact est utilisée pour nous aider à effectuer des transactions rapidement, facilement et en toute sécurité. Des dizaines de millions de personnes utilisent chaque jour la technologie sans contact dans les cartes de paiement bancaires, les passeports, les cartes d'identité et les cartes de transport en commun pour des transactions sécurisées, rapides et pratiques.

Durant la dernière décennie, et avec la révolution technologique engendrée par l'évolution des smartphones et autres tablettes, il est devenu très courant de vivre ce qu'auparavant était du pur imaginaire à savoir la vie en tant que connecté. Alors, se connecter à un site, faire des achats et des transactions bancaires, avoir accès à des lieux et autres sites ou encore prendre la température d'un malade sont devenus accessibles à tout porteur d'un smartphone ou d'une tablette tant que ces derniers sont équipés d'au moins trois technologies sans contact en embarqué et cela bien sûr sans aucune autre entrée de code quelconque ou d'adresse car il suffit de rapprocher le smartphone d'un terminal ou d'un lecteur pour effectuer la dite transaction en toute liberté , en toute simplicité et encore en toute sécurité.

1.2. Définition :

Les Technologies sans contact (en anglais *Contactless Technologies*) sont des technologies de communication sans-fil à courte ou à haute portée et basse ou haute fréquence, permettant l'échange d'informations entre des périphériques ou terminaux mobiles et un lecteur dans des distances Variables selon le type de technologie [2].

En d'autres termes, les technologies sans contact sont des moyens de communication permettant d'échanger des données entre un lecteur et n'importe quel terminal mobile ou entre les terminaux eux-mêmes sans pour autant toucher ce terminal [3].

1.3. Différents types et formes de technologies sans contact :

Etant donné que les technologies sans contact sont en quelque sorte les technologies sans fil, il existe plusieurs formes de ces technologies. On n'en citera que quelques-unes.

1.3-1. Le wifi :

Le Wi-Fi, ou Wifi, est un réseau local lancé en 1999 qui utilise des ondes radioélectriques pour relier entre eux, sans fil, plusieurs appareils informatiques dans le but de faciliter la transmission de données



Figure. 1.1. Le réseau WIFI

Le WiFi peut être compris comme une technologie liée au réseau Internet sans fil. Il est installé pour faire fonctionner les échanges sur l'Internet. Depuis certaines années, le WiFi est devenu un moyen de communication, un système d'accès haut débit à l'Internet. Il s'appuie sur la règle IEEE 802.11 (ISO/CEI 8802-11) [4].

Le terme WiFi est une combinaison de « Wireless » et « Fidelity » qui signifie le standard pour la fidélité sans fil ou le standard de la communication sans fil devenu le plus populaire chez les utilisateurs vus sa simplicité d'utilisation et son coût.

Ainsi, en installant un adaptateur sans fil sur n'importe quel périphérique, il devient très aisé de communiquer avec un routeur et de se faire partager une connexion internet ou différentes informations.

1.3-2. Le Bluetooth :

Le Bluetooth est un type de connexion sans fil qui a vu le jour dans l'objectif spécifique de prendre la place des câbles utilisés pour transférer des fichiers.

Le Bluetooth est une technologie sans fil de proximité, à la portée limitée à quelques mètres seulement (10 à 15 mètres) et au débit plus faible que le Wifi. Inventé en 1994 par Ericsson, le Bluetooth permet à deux appareils de communiquer très facilement entre eux. Cela a également permis d'accélérer le processus de transfert. Il existe un grand nombre de connexions Bluetooth dont la norme et la classe déterminent le nombre de liaisons simultanées et la portée. Le Bluetooth Low Energy (BLE) est celui qui est employé sur les petits appareils dont nous nous servons tous les jours. Cela inclut aussi nos téléphones portables. Cette technologie a une portée limitée à dix mètres au maximum et ne peut être connectée simultanément qu'à un nombre réduit d'appareils.

Le plus puissant Bluetooth peut lier Jusqu'à 215 dispositifs conjointement et sa portée peut atteindre les cent mètres. [5]

L'appairage est le processus permettant de lier, via Bluetooth, deux dispositifs. Ce processus consiste en un échange de chiffres pour coder la connexion. Ces chiffres sont envoyés au récepteur par l'émetteur. Une fois l'appairage effectué, les dispositifs peuvent se reconnaître aisément et effectuer une connexion automatique à la prochaine demande.

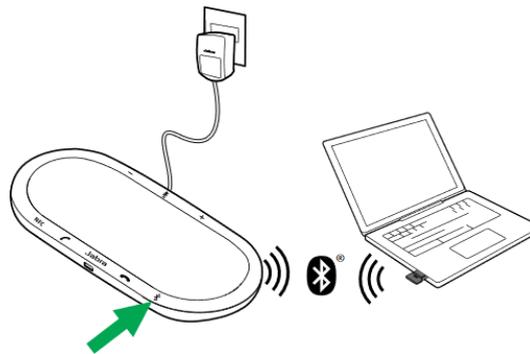


Figure.1.2 : Processus d'appairage via Bluetooth

La connexion Bluetooth est très facile à utiliser, et de ce fait représente avec le wifi les deux technologies sans contact les plus populaires mais par contre, ce mode de connexion n'est pas du tout sécurisé.

1.3-3. La RFID :

La RFID, pour "Radio Frequency Identification", est une technologie permettant de mémoriser, stocker, enregistrer des données sur un support et de les récupérer à distance. Elle existe depuis les années 1940 et servait, à l'époque, à identifier les avions de guerre entrant dans l'espace aérien du Royaume-Uni afin de les distinguer. D'abord utilisée par l'armée, la RFID s'est répandue dans différents secteurs industriels à partir des années 1980, de l'agroalimentaire à la santé, en passant par les transports [1].

Les étiquettes RFID, qui peuvent aussi prendre la forme de balises ou de tags, sont composées d'une puce RFID et d'une antenne et sont collées sur un produit. Elles enregistrent les données et un lecteur électromagnétique lit ensuite les ondes radio présentes sur la puce RFID grâce à l'antenne.

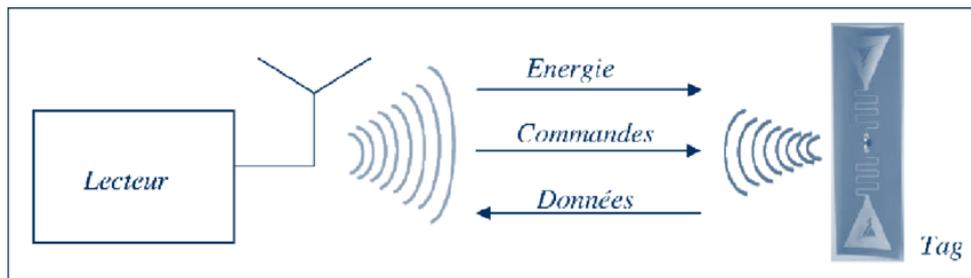


Figure.1.3 : Schéma illustrant un système RFID UHF passif

Les puces se différencient en grande partie par la fréquence utilisée et la distance de lecture. Plus la fréquence est élevée, plus la distance de lecture peut être élargie. Il existe par ailleurs trois types de puces RFID :

- Les puces actives qui émettent elles-mêmes un signal sur une portée de 100 mètres et ont leur propre source d'alimentation.
- Les puces semi-actives qui émettent un signal uniquement après en avoir reçu un.
- Les puces passives qui ne communiquent leurs informations en lecture seule que lorsque l'action est demandée dans un rayon de 25 mètres seulement par le lecteur, dans lequel elles puisent leur énergie.

1.3-4. NFC :

La technologie NFC (Near Field Communication) ou CCP (Communication en Champs Proche, en français) est une technologie qui permet à deux terminaux qui sont très proches (moins de 10 cm de distance) de s'envoyer des données de façon très rapide (moins d'une seconde). Ces terminaux peuvent être deux smartphones ou un téléphone et un autre appareil.



Figure.1.4 : La technologie NFC et ses utilisations.

Extension de la norme ISO/CEI 14443, dérivée de la technologie RFID, elle utilise des vitesses faibles (entre 106 et 414 kbps) et ne nécessite aucun appariement (contrairement au bluetooth par exemple), ce qui permet à deux périphériques de communiquer automatiquement dès qu'ils sont proches l'un de l'autre [1].

1.3-5. iBeacon ou la géolocalisation de proximité :

Le iBeacon est une petite balise, de quelques centimètres qui émet un signal relativement simple dans une norme dérivée du classique bluetooth appelée Bluetooth Smart Low Energy. Développée par Apple, cette technologie consiste en une balise caractérisée par un identifiant universel unique (UUID d'une longueur de 16 octets). Deux autres valeurs d'identification

Appelé Minor/Major qui permettent de différencier des balises avec le même UUID et d'imaginer un sous classement. [1]

La technologie iBeacon utilise principalement la Bluetooth pour communiquer entre une balise et le smartphone.



Figure.1.5 : Balises iBeacon du marché.

Les balises de type iBeacon émettent un signal de localisation avec le nom XX et le smartphone est capable de la reconnaître et d'apprécier sa distance selon trois zones de proximité : une première à moins de 10 centimètres, une seconde dans un rayon entre dix et cinquante centimètres et un autre au-delà de 50 centimètres jusqu'à 70 mètres théoriques (25 mètres constatés). On peut régler en usine la portée maximum.

Un beacon est capable de faire réagir un smartphone ou une tablette, écran verrouillé ou application éteinte pour les plus récents [1].

A la manière d'un GPS qui lui aussi ne fait qu'émettre un signal, le beacon, identifié, va pouvoir faire réagir le smartphone équipé d'une application dont le rôle principal est la localisation ou la géolocalisation en intérieur (aéroport, salle de spectacles, commerces divers ...etc.)

1.3-6. R.I (Reconnaissance d'images) :

La reconnaissance d'images est une technique d'intelligence artificielle permettant d'analyser des images captées par un équipement tel qu'une caméra. Concrètement, la Computer vision se présente comme un outil basé sur l'IA capable de reconnaître une image, de la comprendre, et de traiter les informations qui en découlent [1].

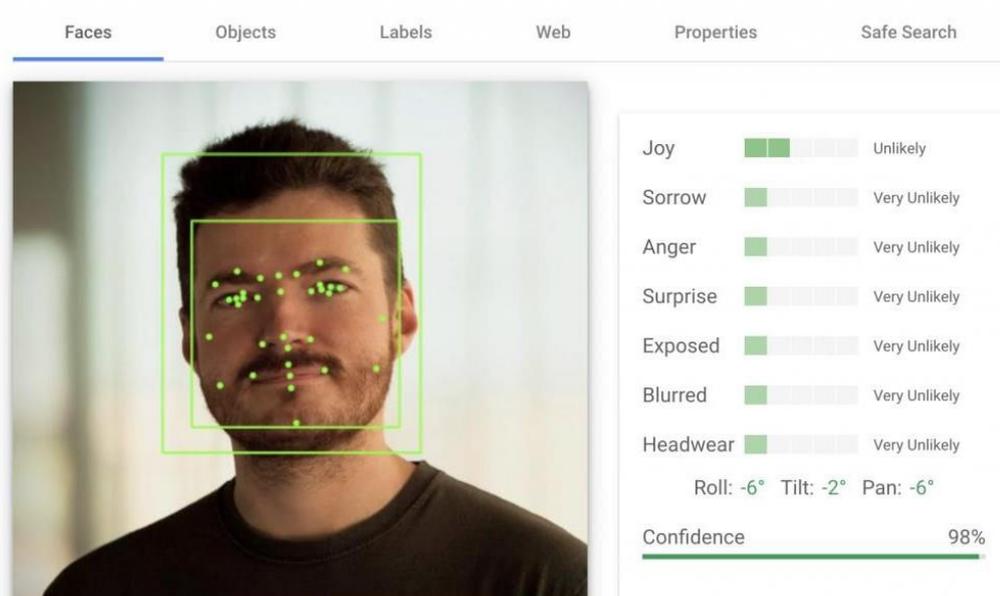


Figure.1.6 : Reconnaissance d'images (faciale) de Google.

Pour beaucoup, la vision par ordinateur est l'équivalent, en termes d'IA, des yeux humains et de la capacité de notre cerveau à traiter et analyser les images perçues. La reproduction de la vision humaine par des ordinateurs constitue d'ailleurs l'un des grands objectifs de la computer vision. La technologie de reconnaissance d'images permet d'identifier une image et de déclencher l'action qui lui est associée. Par exemple, le scan d'un logo peut renvoyer vers le site web de l'entreprise, ou encore le scan d'une photo peut permettre d'identifier le contact et d'accéder à ses coordonnées dans son téléphone. Beaucoup d'applications permettent déjà de reconnaître des images. La reconnaissance d'images est bien pratique pour le "m-shopping", le shopping sur smartphone. Des applications permettent de rechercher sur le web, en un scan de téléphone, l'objet repéré en vitrine, pour l'acheter sans attendre sur mobile. Au-delà du shopping, la reconnaissance d'images donne accès également à de nombreuses autres applications de reconnaissance faciale, traduction instantanée, et bien plus encore !

1.3-7. QR code :

Le QR Code, c'est cet ensemble de petits carrés noirs ou en couleur que l'on trouve imprimé un peu partout. C'est un véritable code informatique que l'on scanne avec une application QR Code pour obtenir l'information qu'il contient. Un QR Code renvoie en général vers un site Internet, mais il peut contenir bien d'autres informations : carte de visite, identifiants WiFi, coordonnées GPS, etc [1].

Son utilisation est très simple car Il suffit d'installer sur son téléphone une application de scan de QR Code. Le rôle de cette application est donc de scanner puis de décrypter le QR code pour en extraire les informations contenues.



Figure.1.7 : QRcode renvoyant à un site web (lien).

1.4. Usages et domaines d'application des technologies sans contact :

1.4-1. Le wifi :

- Créer un réseau local haut débit.
- Offrir un Accès Sans Fil à Internet (ASFI en français ou WIA en anglais), qui est aujourd'hui l'utilisation la plus courante du wifi.
- Transmettre des données entre plusieurs appareils informatiques.
- Relier deux ou plusieurs appareils ou périphériques informatiques sans utiliser de câblage.
- Augmenter les performances du réseau pour un plus grand nombre d'objet 4].

1.4-2. Le Bluetooth :

- Permettre aux appareils de former leurs propres petits réseaux sans fil rapidement et sans infrastructure supplémentaire
- L'utilisation la plus courante de Bluetooth est probablement l'audio sans fil. De nombreux autoradios modernes sont compatibles Bluetooth, les téléviseurs intelligents avec Bluetooth peuvent se connecter à des écouteurs Bluetooth et, bien sûr, les haut-parleurs Bluetooth sont incroyablement populaires [5].
- Permettre à d'autres périphériques tels que les souris et les claviers compatibles Bluetooth d'être connectés à un appareil informatique tel que l'ordinateur par exemple.
- Transférer des données entre deux appareils compatibles Bluetooth dans un champ très restreint.

1.4-3. La RFID :

- Identifier et localiser plus rapidement une marchandise, car la vision directe de l'étiquette par le lecteur n'est plus nécessaire.
- Contrôler de la traçabilité des produits.
- Réduire les déplacements du personnel à l'intérieur de l'entrepôt car l'intervention humaine est minimale dans le processus de lecture des étiquettes.
- L'intégration des terminaux radiofréquence dans le système de gestion d'entrepôt élimine les erreurs logistiques résultant de la gestion manuelle.
- En conclusion, la technologie RFID est orientée principalement dans la gestion des entrepôts grâce à l'utilisation des étiquettes et du lecteur ou terminal RFID. Les étiquettes RFID optimisent le temps et les coûts logistiques, car complétées par un système de gestion d'entrepôt, il est possible d'automatiser des dizaines de processus tels que l'enregistrement des entrées ou le contrôle de la production, sans qu'il soit nécessaire de procéder à des vérifications [1].

1.4-4. NFC :

Couplé à un smartphone, le **NFC** se met en relation avec la carte SIM, où peuvent être stockées des données chiffrées. Ainsi, le **NFC** permet de [1] :

- Transformer un appareil mobile en carte de paiement **NFC Android** sans contact, alors pour effectuer un paiement bancaire, il suffit donc d'approcher le mobile du terminal à cartes bancaires (qui doit être compatible) pour que le paiement soit validé.
- Grâce à NFC, un smartphone peut devenir facilement une carte de fidélité pour bénéficier des avantages fournis par une entreprise quelconque pour le détenteur de ce type de carte, en approchant le smartphone du lecteur NFC ou terminal dédié à cet effet.
- Il peut faire fonction aussi d'un titre de transport, et l'on n'aura plus besoin d'acheter un billet de train par exemple tant qu'on a validé son abonnement avec la carte SIM.

1.4-5. iBeacon :

L'utilisation du iBeacon (Balise) est orientée principalement vers le commerce en général et ceci selon quatre (4) niveaux d'engagement du Beacon [1] :

- La notification : Envoi de messages personnalisés aux clients à proximité de la balise.
- L'engagement : Suggestion d'informations contextuelles en fonction du comportement d'achat d'un client (Par exemple, un client intéressé par une paire de chaussures en magasin).
- La fidélisation : Actions sur le long terme (Programme de fidélité).
- Le suivi passif : Mesurer la fréquentation, le temps de passage, de manière anonyme ou personnelle.

Le commerce de détail est l'un des principaux secteurs où les beacons démontrent toute leur utilité. Cette technologie fournit une expérience exclusive aux consommateurs. Informations, événements, offres promotionnelles, coupons de réduction, affichage des produits... Pour recevoir les messages émis, l'utilisateur doit disposer de l'application mobile de l'enseigne, avoir activé le Bluetooth sur son smartphone et se situer dans le périmètre d'émission du beacon [1].

Le iBeacon fournit des informations qui permettent de parfaire une meilleure connaissance du client à travers la traçabilité de son comportement et de son mode de consommation. Cela passe par le temps d'occupation des magasins, l'affluence journalière ou la consultation des

informations liées aux produits. Le beacon est utile pour améliorer son offre et s'adapter aux évolutions du marché.

1.4-6. **R.I (Reconnaissance d'images) :**

- Renvoyer vers le site web de l'entreprise en scannant tout simplement le logo de cette dernière.
- Identifier un contact et accéder à ses coordonnées dans un smartphone en effectuant le scan de sa photo [1].
- La reconnaissance d'images est bien pratique pour le "m-shopping", le shopping sur smartphone. Des applications permettent de rechercher sur le web, en un scan de téléphone, l'objet repéré en vitrine, pour l'acheter sans attendre sur mobile.
- La reconnaissance faciale d'une personne et l'accès à plusieurs données la concernant [1].
- La traduction instantanée d'un texte photographié ou scanné grâce aux nouvelles technologies d'intelligence artificielle.

1.4-7. **QR code :**

D'une manière générale, le QR code offre des informations sur de nombreux sujets [1] :

- Archivage, Stockage : Il existe un QRCode pour archiver des dossiers, ranger des archives, ranger de cartons de stockage, ou encore optimiser les dossiers.
- Supports de communication générale : Le QRCode sert pour la publicité, le lancement de nouveautés, l'affichage de campagnes promotionnelles et les évènements... etc.
- Documentation et notice technique : un QRCode est fourni pour renseigner sur le montage, pour renvoyer à une vidéo, pour les interventions techniques, et enfin fournit un mode d'emploi.
- L'édition et les bibliothèques : QRCode renseigne sur un auteur, un éditeur, au dos d'un livre, pour le rangement et l'information en bibliothèque et pour le classement de livres.
- Emballage des produits : En apposant un QRCode sur l'emballage d'un produit, crée instantanément un lien entre le vendeur, les produits et les clients et donne alors des informations consommateurs, notices, coupons virtuels, jeux, vidéos...
- Evènements et culture : QRCode pour un salon, une foire et évènements en tous genres, pour les stands et affiches d'expos...

Un QRCode imprimé au dos d'un ticket, ou sur un programme, un magazine ciné donne accès à un contenu spécifique, comme les évènements, les remises, les offres commerciales de partenaires...

- Une plus grande distribution : Les QRcodes imprimés sur les produits, ou sur les catalogues permettent de promouvoir les ventes, renseigner les clients et gérer les opérations promotionnelles.
- Handicap et domaines connexes : Le QRcode est particulièrement adapté aux malvoyants : repéré puis scanné, il va renseigner oralement sur un accès, un train, des horaires, et permet de restituer oralement le contenu d'un texte écrit.
- Logistique, Expédition, Suivi : QRcode pour la gestion d'une logistique, pour le suivi des lots, pour lutter contre la contrefaçon, et pour une meilleure traçabilité.
- Musées et Expositions : Renseigne les visiteurs sur un musée, une œuvre, commente une exposition, raconte l'histoire d'un lieu, et donne des indications sur les horaires, les tarifs et les plans d'accès d'une exposition.
- Plan d'accès, Géolocalisation : QRcode renseigne sur un lieu, un plan. Il permet aussi de géolocaliser sa position et donne plus d'accessibilité aux non-voyants et mal voyants.
- Publicité, Panneau, Affiche : Les QRcodes peuvent être utilisés pour vos campagnes de publicité sur panneaux en tout genre, dans des vitrines et dans des magazines en guise de publicité.
- Santé et domaines connexes : Le QRcode donne des informations sur un médicament et sa composition et assure sa traçabilité, et celle des lots. Il limite et évite dans certains cas la contrefaçon.
- Transport, accessibilité : Un QRcode renseigne sur les horaires, les destinations, le trafic, et donne des informations en temps réel pour les voyageurs et dans leurs véhicules.
- Traçabilité, sécurité : L'authentification par QRcode permet de valider la réception d'un produit, suivre sa distribution. Le contenu web d'un QRcode alimentaire donne son origine, sa certification, sa composition.

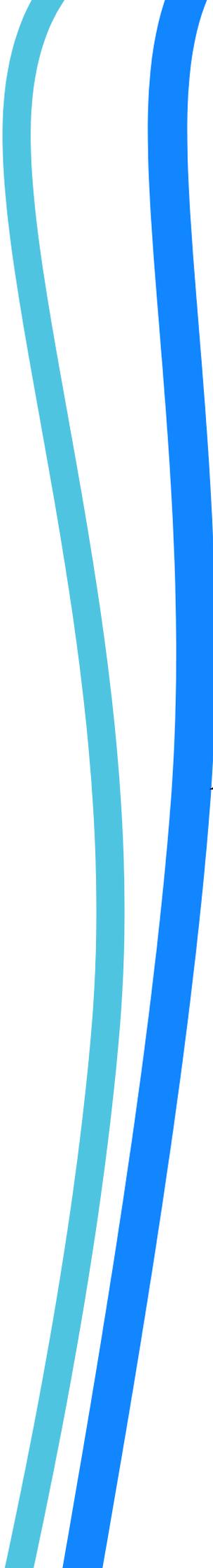
1.5. Conclusion :

L'usage des technologies sans contact dans la vie quotidienne devient de plus en plus large tant dans le taux d'utilisation que dans les domaines d'application.

Par la sécurité qu'offre le chiffrement ou codage de ces technologies, l'utilisation grand public s'avère imposée par la facilité et la vitesse d'exécution de ces modes de communication modernes. Grâce à ces nouvelles technologies, relier deux ou plusieurs appareils informatiques en réseau dépend maintenant de quelques clics introduisant un code simplifié ; se connecter à un site web se fait en scannant une image ou logo ou QRcode, faire ses achats sans présenter de carte ni de payer et même en faisant ceci en ligne, faire des transactions sans toucher à quelconque terminal ou faire une visite virtuelle d'un site devient possible sans oublier que ces mêmes opérations deviennent maintenant accessibles aux personnes handicapées surtout visuellement car ces technologies permettent maintenant la lecture oralisée d'un texte préalablement enregistré et présenté sous une forme iconique ou image.

L'avènement enfin du QRcode, la technologie sans contact la plus utilisée ces dernières années et dans tous les domaines, a permis l'élargissement du champ d'action et d'information en introduisant l'accès à l'information instantanément et dans n'importe quelle langue tant que la traduction est aussi rapide que le transfert de données.

Avec l'arrivée de la pandémie du 21^{ème} siècle, le sans contact et le sans-fil sont devenus l'arme secrète par laquelle l'humanité limite la propagation de la maladie tant que cette dernière est transmise au toucher, alors tout contact avec un terminal , une carte , de la monnaie , des billets de banque , de tickets et d'autres procédés de paiement, de transaction utilisés précédemment devient interdit et sera remplacé par le sans-fil, la technologie sans contact qui limite les contacts entre personnes entre elles ou entre personnes et différents appareils pouvant menacer la santé publique. On dira enfin que l'intelligence artificielle additionnée aux différentes technologies sans contact ouvrent d'autres perspectives et d'autres champs d'action dans le futur proche œuvrant pour le bien-être de l'humanité tout entière.



Chapier II

*Généralités sur les smartphones et
Les plateformes de développement*

Chapitre 2 : Généralités sur les Smartphones et la plateforme de développement mobile Android

1. Introduction :

De nos jours, le téléphone mobile n'est plus seulement utilisé pour les communications et l'échange de messages courts, de nouveaux usages sont apparus tels que la messagerie électronique, la navigation sur Internet, la musique, les vidéos, etc. Grâce à la généralisation des téléphones portables tactiles à écrans larges ainsi qu'au développement des logiciels et des réseaux, les applications mobiles sont capables de satisfaire un large éventail de besoins.

2. Les Smartphones :

2-1. Définition :

Le smartphone est un téléphone mobile que ses fonctions rapprochent de l'ordinateur portable et de l'appareil photo numérique. Un smartphone (également appelé téléphone intelligent) est un téléphone mobile disposant en général d'un écran tactile, d'un appareil photographique numérique, des fonctions d'un assistant numérique personnel et de certaines fonctions d'un ordinateur portable.

La saisie des données sur un smartphone se fait le plus souvent à l'aide d'un écran tactile ou, plus rarement, d'un clavier ou d'un stylet. Selon le principe d'un ordinateur, il peut exécuter diverses applications grâce à un système d'exploitation spécialement conçu pour mobiles. Il peut donc fournir des fonctionnalités en plus de celles des téléphones mobiles classiques comme : l'agenda, la télévision, le calendrier, la navigation sur le Web, la consultation et l'envoi de courrier électronique, la géolocalisation, le dictaphone/magnétophone, la calculatrice, la boussole, l'accéléromètre, le gyromètre, la messagerie vocale visuelle, la cartographie numérique, etc.

Les systèmes d'exploitation mobiles (OS) peuvent être définis comme les logiciels permettant à un smartphone ou un téléphone mobile basique de fonctionner. Ils permettent de ce fait aux utilisateurs de pouvoir passer un appel téléphonique, naviguer sur leurs téléphones parmi toutes les rubriques, télécharger des applications ou encore paramétrer et personnaliser leurs smartphones.

2-2. Systèmes d'exploitation :

Pour pouvoir fonctionner, tout smartphone doit posséder un système d'exploitation. Ce système se concentre essentiellement sur la gestion de la connectivité sans fil et celle des différents types d'interface.

Depuis les premiers smartphones, différents types et modèles de systèmes d'exploitation ont vu le jour. Certains sont délaissés progressivement faute d'utilisateurs, d'autres par contre, s'améliorent de jour en jour et gagnent du terrain parmi les utilisateurs vus leur gratuité (open source), facilité de développement et leur coût.

On distingue une vingtaine de système d'exploitation sur le marché, certains étant développés directement par et pour les fabricants de téléphones eux-mêmes, d'autres formant des conglomérats pour mutualiser les coûts et effectuer des synergies, et d'autres souhaitant uniquement se focaliser sur le développement de systèmes d'exploitation.

Plus généralement, on peut évoquer la présence de quatre OS sur le marché des systèmes d'exploitation, à savoir Android (Google), iOS (Apple Inc.), Windows Phone (Microsoft) et Blackberry OS (RIM).

Tableau 2.1 : Vente mondiale de smartphones selon l'OS de 2017 à 2021 [6].

ANNEE	ANDROID (%)	IOS (%)	AUTRES OS (%)	TOTAL (%)
2017	85.10	14.70	0.20	100.00
2018	85.10	14.90	0.00	100.00
2019	86.70	13.30	0.00	100.00
2020	86.60	13.40	0.00	100.00
2021	86.90	13.10	0.00	100.00
Prévision 2022	87.00	13.00	0.00	100.00
Prévision 2023	87.10	12.90	0.00	100.00

2-2-1. Android de Google :

Depuis maintenant de nombreuses années, Android, développé par les ingénieurs de Google, s'impose comme l'OS de référence du marché, loin devant ses concurrents. Il atteint 71.06 millions de smartphones équipés vendus en 2010, quand iOS à la seconde place ne dépasse pas les 47.8 millions. L'avantage de ce système d'exploitation se résume dans le fait

qu'il est mis à disposition de tous les fabricants de téléphones, à condition qu'ils respectent certaines conditions liées à la mise en avant de services Google. Pour le reste, l'OS est personnalisable à souhait, par les consommateurs comme par les fabricants. De nos jours, la majorité des téléphones vendus dans le monde tournent sous Android. On peut ainsi citer de très nombreuses marques de smartphones, parmi lesquelles Samsung, Acer, HTC, LG, Sony ou encore Huawei avec 1086.87 millions de smartphones tournant sous Android vendus en 2020 contre 205.42 millions tournant sous iOS.

2-2-2. iOS d'APPLE :

Avec Android, iOS de l'entreprise américaine Apple concentre une très large majorité des smartphones de la planète. L'iOS d'Apple est connu et présenté par la firme de Cupertino comme étant le système d'exploitation proposant la meilleure expérience utilisateur. Avec son flat design apparu à partir de l'iOS 7, Apple propose son logiciel uniquement sur ses propres produits. Il faut de ce fait passer par l'Apple Store pour installer sur son iPhone ou iPad les applications qui intéressent l'utilisateur parmi les centaines de milliers existantes. L'écosystème d'Apple rendant tous les produits de la marque compatibles entre eux, permet une ergonomie inégalée à ce jour, positionnant Apple comme fabricant éminent, et comme second plus gros système d'exploitation au monde en volume atteignant ainsi le nombre de 205.42 millions d'appareils tournant sous iOS vendus en 2020.

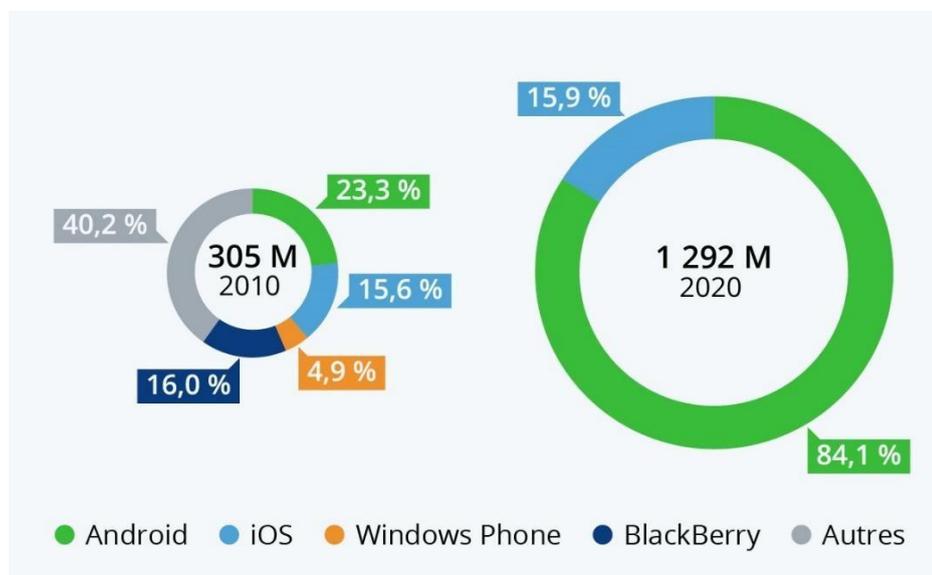


Figure 2.1 : Vente mondiale de smartphones Android et iOS en 2010 et 2020

2-2-3. Windows Phone de Microsoft :

Les smartphones tournant sous Windows Phone restent à ce jour limités face à Android, bien que toutefois présents en nombre. Microsoft, qui a pris le virage des smartphones en retard, souffre de son manque d'adoption par les constructeurs et consommateurs. Basé sur le design de Windows 8 pour PC, il n'est pas du goût de tout le monde. Il reste cependant aimé par les utilisateurs pour la possibilité de personnaliser de manière poussée la page d'accueil et sa simplicité de gestion des applis. Face aux géants que sont iOS et Android, il a toutefois de la peine à intéresser suffisamment les développeurs d'applications, certains d'entre eux préférant le délaissé au profit des OS beaucoup plus fréquentés, au grand déplaisir des utilisateurs.

2-2-4. BlackBerry OS de RIM :

Ancienne star de la téléphonie, prisé par le monde professionnel et particulièrement des cadres d'entreprise, l'OS Blackberry de RIM a perdu de ses parts de marché depuis la dernière décennie. Il n'empêche que les smartphones de la marque, avec leur design complètement revu, intéressent maintenant une partie ciblée des consommateurs. A l'attention des personnes avides de claviers physiques, Blackberry OS est tourné avant tout vers la lecture des e-mails et SMS. Fait identifiable, les applications Android fonctionnent maintenant sous Blackberry OS, rendant plus attrayant le système d'exploitation de RIM, en proposant un très large catalogue d'applications compatibles innovantes.

2-2-5. Autres systèmes d'exploitation pour mobiles :

D'autres smartphones proposent d'autres systèmes d'exploitations, mais représentent encore dans leur totalité une partie extrêmement infime des smartphones utilisés au niveau mondial. Parmi les près de 20 systèmes d'exploitation mobiles existants, on peut citer Linux (OS open source) et le plus récent Firefox OS de la fondation Mozilla. Depuis maintenant plusieurs années, les yeux sont tournés vers l'Asie, où les fabricants de smartphones prennent peu à peu de l'importance et de l'indépendance. Un conglomérat proposant des solutions innovantes pourrait à terme révoquer la domination d'Android et iOS. Mais ce n'est pas pour tout de suite, et les concurrents américains auront le temps de voir, analyser et contrecarrer leur stratégie.

2-3. Plateforme de développement mobile sous Android :

2-3-1. Définition :

Android est un système d'exploitation temps réel, mobile et open source conçue spécifiquement et pas exclusivement pour des systèmes embarqués légers comme les smartphones, PDA, tablettes...

Android désigne à la fois :

- Un environnement de développement (API)
- Un environnement d'exécution (le plus souvent) [7].

Android est un système d'exploitation OPEN SOURCE pour terminaux mobiles (smartphones, tablettes ... etc.). Conçu à la base par une startup (Android) rachetée par Google en 2005. C'est aujourd'hui le système d'exploitation mobile le plus utilisé à travers le monde [8].

2-3-2. Architecture logicielle d'Android :

Android est un système d'exploitation Open Source pour terminaux mobiles. Il se différencie principalement de ses concurrents, par le fait qu'il est ouvert. Le modèle économique de Google semble très pertinent, l'adoption d'Android par les fabricants sera plus rapide du fait de la gratuité d'utilisation pour le constructeur. Voilà pourquoi il est pertinent pour tout développeur de travailler sur cette plateforme. Une autre raison est que les applications Android sont développées en Java, un langage de programmation maîtrisé par un grand nombre de développeurs et utilisateurs des smartphones. Android dispose aussi d'un ensemble de bibliothèques qui incluent la plupart des fonctionnalités présentes dans java ainsi que des fonctionnalités supplémentaires pour la gestion de l'interface graphique par exemple. Cette plateforme mobile, bâtie autour du noyau Linux 2.6 a plusieurs versions avec une architecture organisée en plusieurs couches permettant ainsi de développer des applications innovantes :

2-3-2-1. Le noyau Linux :

Android repose sur un noyau Linux (version 2.6) qui gère les services du système, comme la sécurité, la gestion de la mémoire et des processus, la pile réseau et les pilotes. Il agit également comme une couche d'abstraction entre le matériel et la pile logicielle.

2-3-2-2. Les bibliothèques graphiques :

En interne, Android inclut un ensemble de bibliothèques C et C++ utilisées par de nombreux composants de la plateforme Android.

2-3-2-3. Une Machine Virtuelle (Dalvik Virtual Machine) :

Chaque application Android s'exécute dans son propre processus, avec sa propre instance de machine virtuelle Dalvik.

2-3-2-4. Framework applicatif :

Android offre aux développeurs la possibilité de créer des applications extrêmement riches et innovantes.

2-3-2-5. Application :

Android est fourni avec un ensemble d'applications natives permettant d'accéder à des fonctionnalités comme les courriels, le téléphone, le calendrier, les photos... Ces applications sont développées à l'aide du langage de programmation Java. Pour l'utilisateur final, c'est la seule couche accessible et visible.

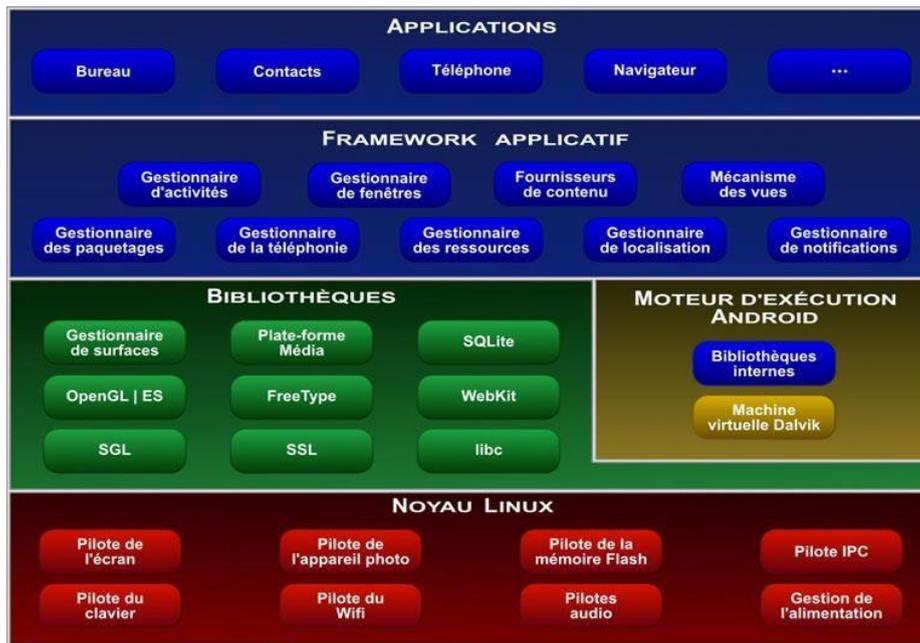


Figure 2.2 : Architecture logicielle de l'Android

2-3-3. Kit de développement d'Android :

Exploiter une nouvelle plate-forme n'est jamais été une chose aisée. C'est pourquoi Google fournit, en plus du système d'exploitation, un kit de développement (Software Développement Toolkit ou SDK).

Ce SDK est un ensemble d'outils qui permet aux développeurs et aux entreprises de créer des applications. Il est disponible gratuitement sur le site de Google.

Le SDK Android est composé de plusieurs éléments afin d'aider les développeurs à créer et à maintenir des applications :

- ✓ Des outils.
- ✓ Des exemples de code.
- ✓ De la documentation.
- ✓ Des API (interfaces de programme d'application).

2-3-3-1. Les outils :

Le SDK est livré avec un certain nombre d'outils couvrant différents aspects du cycle de développement d'une application Android. Le kit de développement propose une boîte à outils complète pour les tâches de compilation, de débogage, de génération de code AIDL et de la signature de l'application, etc.

L'émulateur Android : c'est un téléphone virtuel qui permet de tester les applications qui sont en train de se développer. Il est lancé par la commande "emulator ". Celle-ci prend en paramètre l'image AVD (Android Virtual Device) qui sera montée en mémoire. Il a des limitations par exemple : il n'est pas capable de supporter le Bluetooth ainsi qu'il ne permet pas le teste des applications de réalité augmentée.

2-3-3-2. Les exemples de code :

Le kit de développement est accompagné d'un certain nombre d'exemples illustrant les possibilités du SDK Android. Parmi ces exemples, on peut citer : un jeu du serpent et le projet qui couvre l'utilisation de plusieurs exemples de l'API Android comme les alarmes, les notifications et les menus.

2-3-3-3. La documentation :

La documentation du SDK Android est scindée en deux parties bien distinctes :

- ✓ Le guide du développeur qui est disponible en HTML (Hypertext Markup Language) dans le répertoire du SDK qu'on vient d'installer ;
- ✓ La documentation des API au format java doc est également située dans le répertoire docs et accessible grâce au chemin débutant du répertoire d'installation.

2-3-3-4. Les API :

Android offre plusieurs API (Application Program Interface) tel que :

- ✓ Google Maps : intègre et contrôle l'affichage d'une carte dans une interface graphique de l'application.
- ✓ Géolocalisation : permet d'accéder au service de localisation du système, de choisir le fournisseur en fonction des critères et de préciser la position actuelle du téléphone (latitude, longitude, vitesse, etc.).

2-3-4. Versions d'Android :

Android est une plateforme Open Source, puissante, moderne, sûre et ouverte. Grâce à l'ouverture du code source et des APIs, les développeurs obtiennent la permission d'intégrer, d'agrandir et de modifier les composants existants. Les utilisateurs peuvent

adapter les applications à leur besoin. Depuis sa création, il ne cesse de se développer avec l'apparition de nouvelles fonctionnalités à chaque nouvelle version.

Le tableau qui suit résume toutes les modifications et améliorations majeures portées sur les différentes versions d'Android [9].

Chapitre 2 : Généralités sur les Smartphones et la plateforme de développement mobile Android

Tableau 2.2 : Différentes versions de la plateforme Android

Nom de code	Version	Caractéristiques majeures ajoutées
Apple pie	1.0	Version connue uniquement ou presque des développeurs car c'est la version du SDK distribuée avant la sortie du premier téléphone Android
Bananas split	1.1	Ou Beta, version incluse dans le premier téléphone, le HTC G1/Dream
Cupcake	1.5	Nouvelles fonctionnalités et mises à jour de l'interface graphique
Donut	1.6	Nouvelles fonctionnalités et mises à jour de l'interface graphique
Eclair	2.0	Nouvelles fonctionnalités et mises à jour de l'interface graphique
Froyo	2.2.x	Vitesse améliorée, nouvelles fonctionnalités et mises à jour de l'interface graphique.
Gingerbread	2.3.x	Dernière version dédiée uniquement aux smartphones. Cette version est parfois utilisée sur de petites tablettes.
Honeycomb	3.x.x	Réservé aux tablettes tactiles et aux téléviseurs connectés ¹⁶ , cette mise à jour comprend de nombreux changements dans l'interface
Ice Cream Sandwich	4.0.x	Cette nouvelle version, fortement inspirée d'Honeycomb, unifiée ¹⁸ pour smartphones, tablettes et Google TV apporte de nombreux changements
Jelly Bean	4.1.x 4.2.x 4.3.x	Reconnaissance vocale sans connexion internet. Nouvelle interface de l'appareil photo, d'un système multi-compte. Support du Bluetooth SMART basse consommation et ajout de la norme AVRCP 1.3, gestion multi-utilisateur plus poussée, support d'OpenGL ES 3.0, nouvelle interface de l'appareil photo, mises à jour de sécurité et SlimPort.
KitKat	4.4.x	Consommation en ressource moins élevée nécessitant moins de RAM, nouvelles icônes plus soignées, la barre du bas et celle de statut deviennent transparentes sur certains menus et changent de couleur en fonction du contenu affiché.
Lollipop	5.0	Nouvelle interface / design ("Material design"), Amélioration de la rapidité Amélioration de la gestion de la batterie.
Marshmallow	6.0	Support de l'USB Type-C, Support de l'authentification par empreinte digitale. Amélioration de la durée de la batterie avec un mode "deep sleep", Panneau pour contrôler les permissions des applications. Android Pay. Améliorations de Google Now.
Nougat	7.0 7.1	Meilleur support du multitâche. Multifenêtrage. Mises à jour systèmes amélioré. Amélioration des performances et de la taille du code grâce à un nouveau compilateur JIT. Eclairage mode nuit. Amélioration du gestionnaire de stockage et des performances liées à la gestion de l'écran et tactile. Option pour activer les gestes sur le capteur d'empreinte. Mises à jour systèmes "transparentes".
Oreo	8.0 8.1	PIP : Picture-in-Picture avec fenêtre redimensionnable. Amélioration du système de notifications. Amélioration de la gestion des paramètres systèmes. Changement de l'interface pour 'Eteindre' et 'Redémarrer' Messages de notification plus lisibles.
Pie	9.0	Notifications plus riches avec des images larges et des réponses rapides type 'Smart Replies'. Authentification biométrique. Nouvelle transition entre les applications.
Queen Cake	10.0	Meilleure confidentialité des applications. Nouveau CODEC audio/vidéo Support du WPA3 et du WIFI. Support pour les téléphones pliables. La navigation par gestes.
Red Velvet Cake	11.0	Capture d'écran vidéo. Amélioration des permissions. Conversations sous forme de bulles. Détection de la 5G dans les applications.
Snow Cone	12.0	Interface utilisateur plus personnalisable. Amélioration de la gestion et de la sécurité des données personnelles. Langage pour décrire l'interface de type 'Material You'.

2-4. Les applications mobiles :

Les applications mobiles ont envahi notre quotidien : Consulter des mails, jouer, parler à un ami, faire du shopping, consulter les actualités ou la météo, retrouver un trajet ou un lieu bien précis, consulter des comptes est fait en un clic via un smartphone. Les entreprises et les particuliers ont tous (ou presque), aujourd'hui recours à ces applications qui font partie intégrante de leur vie et sans lesquelles, pour beaucoup, la vie serait presque plus compliquée.

Pour pouvoir exécuter une tâche bien précise non prise par les applications fournies avec le smartphone, ce dernier doit accéder à des applications externes dédiées à ces différentes tâches.

2-4-1. Définition :

Une application mobile est un logiciel applicatif développé pour un appareil électronique mobile, tel qu'un assistant personnel, un téléphone portable, un smartphone, un baladeur numérique, une tablette tactile, ou encore certains ordinateurs fonctionnant avec le système d'exploitation Windows Phone ou Chrome OS.

Une application mobile est un programme téléchargeable de façon gratuite ou payante et exécutable à partir du système d'exploitation d'un dispositif mobile (PDA, smartphone, tablette...). Elles sont pour la plupart distribuées depuis des plateformes de téléchargement (parfois elles mêmes contrôlées par les fabricants de smartphones) telles que l'App Store (plateforme d'Apple), le Google Play (plateforme de Google / Android), ou encore le Windows Phone Store (plateforme de Microsoft).

2-4-2. Stratégies de développement :

La conception d'applications mobiles peut se faire suivant trois stratégies de développement distinctes. Dans ce qui suit, nous donnons un bref aperçu de chaque stratégie [10].

2-4-2-1. Application native :

Une application native est une application mobile spécifiquement développée pour un système d'exploitation mobile. Elle est conçue avec le langage et les outils associés à son système d'exploitation, et installée directement sur le mobile. Cette installation se faisant soit au travers d'un téléchargement via Internet soit par déploiement depuis un ordinateur

connecté au mobile. Les tests pour vérifier le comportement de ces applications nécessitent des compétences techniques spécifiques et des appareils très coûteux.

2-4-2-2. Application Web :

Une application mobile Web est une application développée en HTML, accessible et exécutable par le biais d'un navigateur Internet pour téléphone mobile. Elle utilise le navigateur du Smartphone et ne nécessite pas forcément de télécharger l'application. Les applications mobiles Web complètent les applications natives qui sont développées spécifiquement pour un système d'exploitation et qui doivent être téléchargées et installées par les mobinautes. Elles s'adressent donc à l'ensemble des utilisateurs de mobiles, et non à une population spécifique utilisant une marque bien précise. Toutefois, les applications Web doivent être testées pour chaque navigateur, résolution et taille d'écran, à l'instar de n'importe quel site Web.

2-4-2-3. Application hybride :

L'application hybride est une application pour mobiles qui combine des éléments HTML5 sous forme d'application mobile Web et des éléments d'une application native permettant l'utilisation des fonctionnalités natives des Smartphones et d'être distribuée en tant qu'application sur les stores des systèmes mobiles (App Store, Play Store, etc.).

2-4-3. Environnement de développement :

De nos jours, Android est le système d'exploitation le plus utilisé dans le monde avec une part du marché qui atteint les 80 %, devant l'iOS d'Apple et Windows Phone de Microsoft. Donc, il n'est pas surprenant que de nombreux développeurs cherchent à créer de nouvelles applications fonctionnant sous ce système d'exploitation basé sur Linux.

Afin de réduire les coûts et les délais de réalisation d'une nouvelle application, de nouveaux environnements de travail qui rassemblent divers outils, sont lancés chaque année pour aider les développeurs dans leur labeur. Ainsi, les IDE, *Integrated Development Environment*, facilitent la compilation, l'écriture et le débogage du programme réalisé et permettent d'obtenir le fichier exécutable (.apk). En voici quelques-uns des plus utilisés par les différents développeurs :

2-4-3-1. Android Studio :

Cet outil sorti en 2013, dont l'installation est réalisable en même temps que le SDK sur tout type de systèmes d'exploitation, représente l'IDE privilégié par Google pour la création d'applications Android. Grâce à sa puissance, sa simplicité et sa gratuité, il a pu détrôner facilement tous les autres environnements utilisés jusqu'à lors. En se basant sur IntelliJ IDEA, cet utilitaire ne permet pas juste de créer des applications compatibles les smartphones, mais elles pourront fonctionner aussi les montres connectées, les téléviseurs connectés et tablettes. Les développeurs pourront aussi visualiser leur travail grâce à un émulateur intégré.

2-4-3-2. Eclipse :

Ce logiciel, signé IBM, était l'une des premières plates-formes de programmation dédiées à Android. Pour pouvoir l'utiliser correctement sur un ordinateur, il faut préinstaller plusieurs utilitaires ainsi que le plug-in ADT pour Eclipse. Il est vrai que ce dernier élément n'est plus maintenu depuis la fin de l'année 2015, après l'officialisation d'Android Studio, mais il reste un environnement très conseillé pour la programmation mobile.

2-4-3-3. Netbeans :

Netbeans est un IDE disponible sous forme d'une suite logicielle *Netbeans Mobile*. D'autres plug-ins peuvent être installés pour améliorer la programmation Android comme le Netbeans Mobility Pack ou encore le Java ME.

2-4-3-3. Autres environnements de programmation Android :

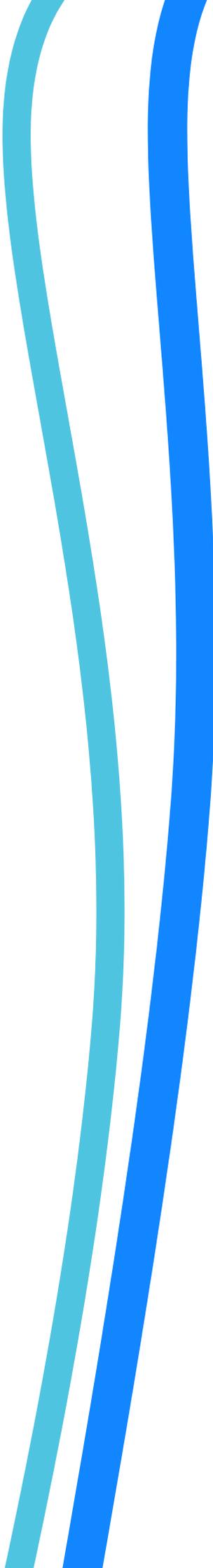
Il existe d'autres environnements de programmation Android développés par d'autres constructeurs, mais leur utilisation reste limitée car leur maniabilité ainsi que la facilité d'utilisation restent limitées par rapport à Eclipse et Netbeans et surtout Android Studio. On citera entre autres **xPerfect** sous Ubuntu, utilise une machine virtuelle appelée Virtual Box, pour réaliser la programmation et la compilation du programme conçu, l'environnement **Titanium** qui fonctionne avec le langage JavaScript orienté objet et permet de créer un même programme donnant lieu à une application fonctionnant sur des systèmes différents, et enfin **Xamarin** de Microsoft qui contrairement aux autres environnements de développement nécessite des compétences de langage C#. Xamarin est disponible gratuitement

sous Visual Basic Community pour les étudiants et les développeurs open source. Il permet de créer une application native pour plusieurs cibles différentes en générant un seul code.

2-5. Conclusion :

Avec la large utilisation quotidienne des smartphones, les applications mobiles deviennent une nécessité. Chaque tâche étant différente d'une autre, il convient alors d'utiliser l'application convenante ce qui donne une panoplie d'applications œuvrant dans le même but et réagissant différemment bien que tournant sous un même système d'exploitation.

L'environnement de développement Android Studio présente une facilité d'utilisation et un large choix de bibliothèques téléchargeables gratuitement, ceci le place le premier par rapport aux autres environnements de développement dédiés à Android et de ce fait la majeure partie des applications Android sont développées avec Android studio avec des résultats très satisfaisants.



Chapier III

*Conception et réalisation d'une
Application de Lecture/génération
De QR codes sous Android*

Chapitre 3 : Conception et réalisation d'une application de lecture/génération de QR codes sous Android

3.1. Introduction :

Notre projet est de réaliser une application Android capable de générer et de lire un QR code ou un code-barres. Cette application doit comporter les fonctionnalités suivantes :

- Générer un code-barres,
- Générer un QR code,
- Scanner un code-barres,
- Scanner un QR code,
- Lire un code-barres,
- Lire un QR code.

Pour parvenir à notre objectif, nous avons utilisé l'environnement de développement nommé Android Studio. Ce choix n'est pas arbitraire car Android Studio est l'environnement de développement le plus utilisé dans le monde car très facile à exploiter et de plus gratuit.

Pour la lecture des codes-barres et QR codes, il a fallu utiliser deux bibliothèques à savoir **ZXing (Zebra Crossing)** pour Java dont le rôle est de décoder ces deux types de codes, alors que la bibliothèque **Yuriy-Budiyev** sert à la mise au point (focus) des photos et images capturées par la caméra du smartphone en vue d'une utilisation optimale par l'application en question.

3.2. Code-barres et QR code :

Pour pouvoir générer ou lire un code-barres ou un QR code, il faut absolument savoir leurs formations ainsi que leurs modes de fonctionnement.

3.2.1. Code-barres :

Le **code-barres** est un **système de traçabilité** utilisé par la plupart des entreprises, tout secteur confondu. Le code-barres est la représentation graphique du code produit (GTIN). Il est représenté par des **barres noires et des espaces blancs** ou encore par des **traits noirs et blancs** destinés à être déchiffrés par un lecteur dans le but de renseigner sur l'origine, le producteur, la référence du produit...

L'identification est la première étape d'un système de traçabilité, l'identification automatique permet alors de tout connaître d'un produit, de suivre et gérer les flux ainsi que les stocks dans une entreprise. Les codes-barres dits **unidimensionnels (1D)** sont apposés sur les emballages et permettent le suivi des ventes ainsi que le stock des produits. Les codes-barres les plus répandus

sont utilisés dans le monde entier pour la quasi-totalité des articles de grande consommation. Comme son nom l'indique, le code GTIN 13 est composé de 13 chiffres qui ont chacun leur importance :

Les 2 ou 3 premiers chiffres correspondent au pays de fabrication du produit. Ainsi, les codes 000 à 019 correspondent aux USA ; 690 à 699 à la Chine et de 840 à 849 à l'Espagne.

Les chiffres suivants correspondent au préfixe entreprise attribué par GS1 (Organisme mondial chargé de la normalisation des méthodes de codage dans la logistique).

Les suivants correspondent à la référence (code produit) attribuée par la propriétaire de la marque commerciale.

Le dernier chiffre est une clé de contrôle qui permet de vérifier que le code-barres est bien valide.



Figure 3.1 : Structure d'un code-barres correspondant au GTIN 6 Algérie

Le code-barres est déchiffrable grâce à une lecture optique linéaire par faisceau laser c'est-à-dire en utilisant un scanner appelé dans le langage courant "**douchette**". Ainsi, l'identification

est automatique engendrant instantanément la prise en compte du prix de l'article mais également la gestion informatique des stocks.

La limitation de la quantité d'informations stockées dans un code-barres avec l'utilisation d'un scanner spécialisé forment les deux principaux inconvénients de l'utilisation du code-barres bien que des applications pour smartphones permettent aujourd'hui la lecture du code-barres mais à l'échelle personnelle.

3.2.2. Le QR Code :

Le QR code est un code bidimensionnel (code à deux dimensions). Il s'agit d'une évolution des codes-barres. Contrairement à son prédécesseur, le code-barres classique qui ne contient qu'une vingtaine de caractères, le QR code est formé de barres horizontales et verticales ce qui le rend capable de stocker un grand nombre de données, verticalement et horizontalement : jusqu'à 7089 caractères numériques et 4296 caractères alphanumériques.

Il existe trois (03) types de QR codes : le micro QR code petit par sa taille et sa capacité de données, le QR code complet qui est le plus fréquemment utilisé et le S QR code ou QR code sécurisé qui nécessite un mot de passe pour le lire. Néanmoins, la structure générale du QR code reste la même pour les trois types.

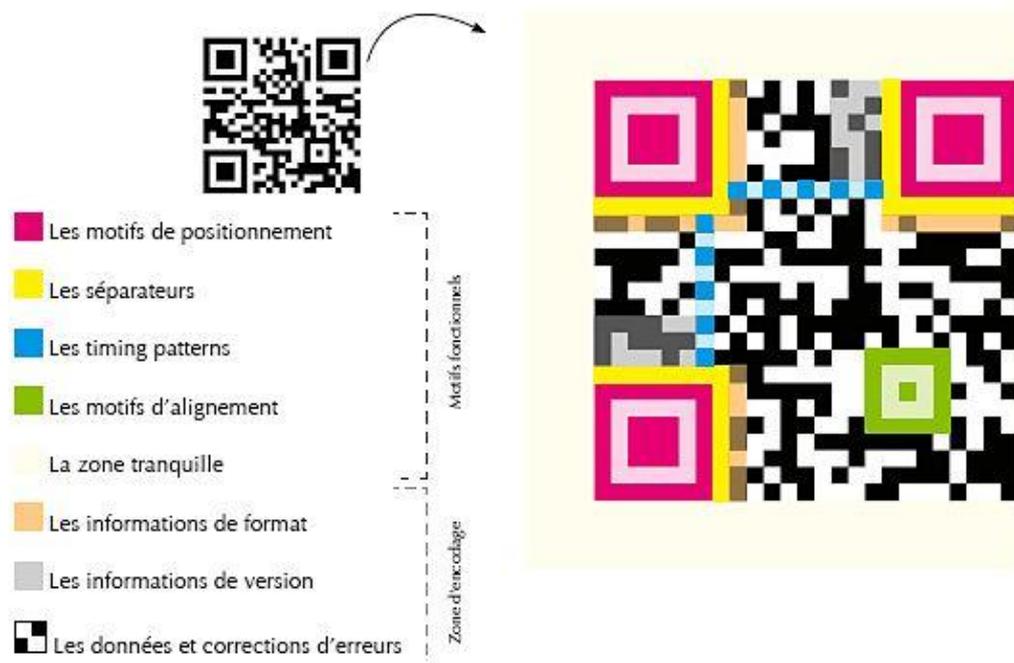


Figure 3.2 : Structure générale du QR code

Pour générer un QR code, il faudra passer par sept étapes essentielles : Analyse des données pour en déterminer le type (numériques, alphanumériques ou Kanji caractères Japonais), l'encodage des données selon la norme ISO18004, la Concaténation des données binaires (0 et 1), l'ajout des données binaires complémentaires selon le mode utilisé (numérique, alphanumérique...), la Formation des codewords (Les données binaires doivent être découpées par octets, soit 8 à 8), le Placement de ces codewords et enfin la création du masque qui donnera l'image finale du QR code.

La lecture du QR code codé ainsi généré se fera avec n'importe quel smartphone et cela explique clairement la large utilisation de ce type de code et dans tous les domaines.

3.3. Environnement de développement Android Studio :

Android Studio est un environnement de développement intégré en Anglais IDE (*Integrated Development Environment*) qui permet de réaliser des applications mobiles Android. Nous avons opté pour cet environnement car il est gratuit, simple d'utilisation et permet de développer des applications mobiles très performantes [13].

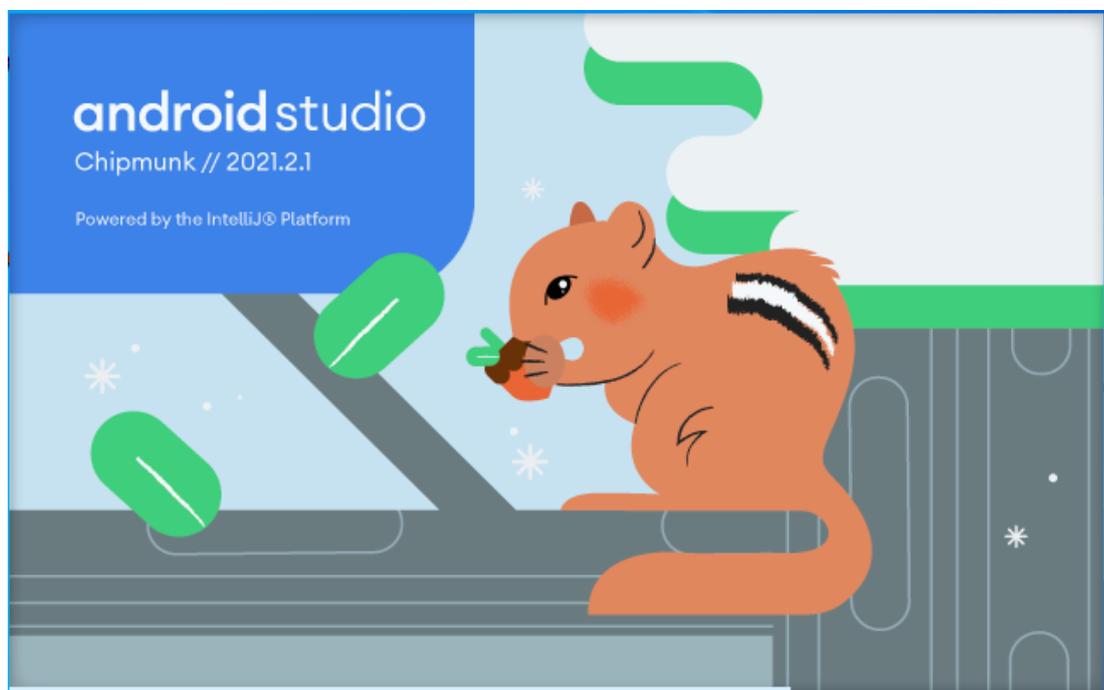


Figure 3.3 : Ecran d'accueil de Android Studio Ver 2021.2.1

3.3.1. Android Studio et ses bibliothèques [13] :

- Android Studio est basé sur l'environnement de développement intégré ou IDE (integrated development environment) d'IntelliJ IDEA.
- Android Studio est livré avec SDK ANDROID, un émulateur multiplateforme de système matériel Android qui est basé sur une machine virtuelle d'où l'émulation purement graphique sur le PC.
- Android Studio inclut les bibliothèques Android courantes dont la librairie CORE qui inclut une grande majorité de l'API (Application Programming Interface) JAVA ou l'interface de programmation des applications ainsi qu'un ensemble de bibliothèques spécifiques au système matériel sous Android.
- Android Studio inclut un processeur virtuel Java particulier, il s'agit du processeur virtuel nommé Dalvik Optimisé pour les systèmes matériels Android.
- Android Studio contient une documentation complète incluant de nombreux exemples d'applications Android.
- Android Studio comporte enfin un débogueur (Dalvik Debug Monitoring Service et Android Debug Bridge).

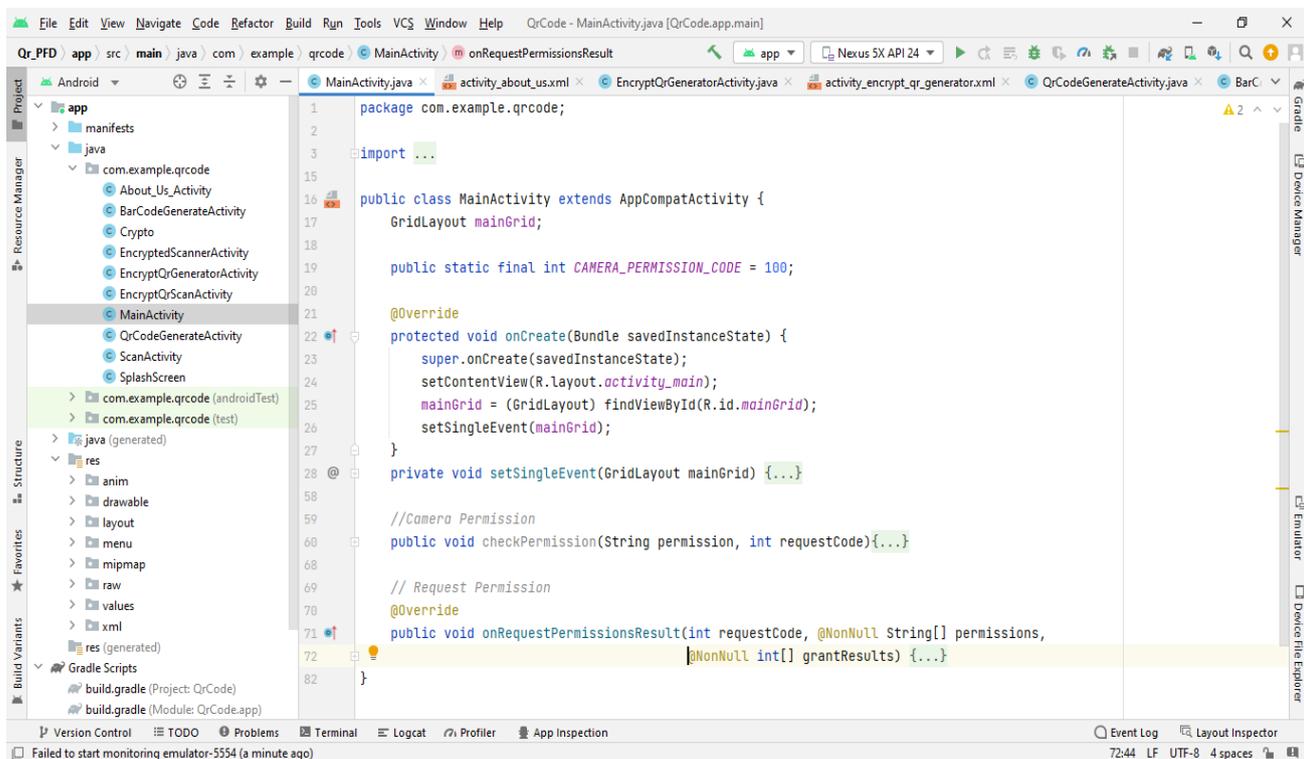


Figure 3.4 : Environnement de travail d'Android Studio

3.3.2. Structure d'une application Android :

Une application Android est constituée de composants à couplage, liés par un **manifeste**. Un projet Android contient essentiellement des fichiers sources `.java` et des fichiers de configuration XML [11].

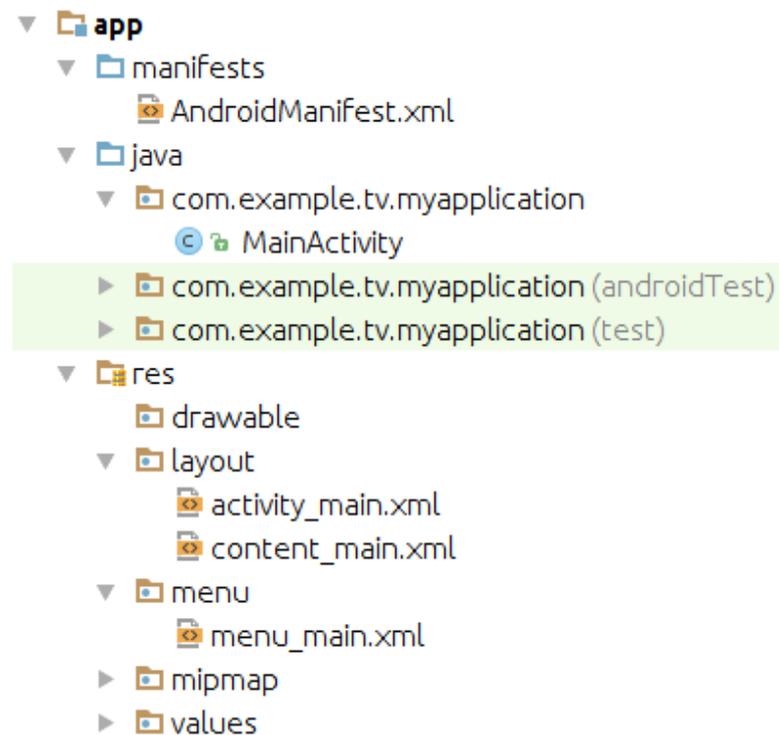


Figure 3.5 : Structure générale d'une application Android.

Tout comme bon nombre de technologies actuelles, les sources d'une application Android possèdent une structure bien définie qui doit être respectée. Ces arborescences permettent non seulement de rendre les projets plus lisibles et organisés, mais aussi de simplifier le développement. Lors de la création d'un nouveau projet, voici l'arborescence qui est automatiquement générée :

AndroidManifest.xml : fichier XML décrivant l'application et ses composants, tels que les activités, les services, etc. Lors de la création d'une activité, une erreur courante pour un premier projet Android est d'oublier de la déclarer dans le fichier Manifest. C'est une étape indispensable pour le fonctionnement de l'application. Le Manifest est, en quelque sorte, la carte d'identité de l'application, et permet d'autoriser l'exécution des activités et autres actions de l'application.

res : répertoire contenant toutes les ressources telles que les images, les vues de l'interface graphique, etc., nécessaires à l'application. Ce répertoire est structuré par défaut de la manière suivante :

res/drawable : contient les ressources de type image.

res/layout : contient les descriptions des interfaces graphiques au format XML (les vues).

res/xml : contient les fichiers XML supplémentaires (non présents par défaut).

res/menu : contient la description des menus, composants très courants d'une vue.

res/values : contient diverses ressources, telles que les textes, qui sont empaquetées sans aucun traitement.

Le fichier build.gradle: sert dans la configuration pour le nouveau moteur de production nommé Gradle, qui sera utilisé pour construire l'application afin de la déployer sur notre smartphone ou sur le Play Store.

3.3.3. Composantes d'une application Android :

Une application Android se compose de plusieurs éléments dont les plus importants sont :

3.3.3.1. Les Activités :

Une activité est la composante principale pour une application Android. Elle représente l'implémentation et les interactions des interfaces. Plusieurs choix se proposent pour mettre en place l'interface visuelle, soit d'utiliser un fichier XML pour décrire l'interface ou créer les éléments de l'interface à l'intérieur du code java.

3.3.3.2. Les Services :

Un Service est un programme tournant en tâche de fond et n'ayant pas d'interface graphique. Certains programmes ne nécessitent pas, pour la plupart du temps, d'interface graphique et doivent tourner en tâche de fond, laissant la possibilité aux autres applications de s'exécuter librement. Un service peut être lancé à différents moments ; soit au démarrage du smartphone, soit au moment d'un événement (arrivée d'un appel, SMS, mail, etc...), ou enfin lors du lancement ou au moment d'une action particulière dans l'application.

3.3.3.2. Les Broadcast Receivers :

Un Broadcast Receiver, comme son nom l'indique, permet d'écouter ce qui se passe sur le système ou sur l'application et de déclencher une action prédéfinie. C'est souvent par ce mécanisme que les services sont lancés.

3.3.3.2. Les Content Providers :

Les Content Providers sont des gestionnaires de données. Ils permettent de partager l'information entre les applications. Ils permettent d'accéder aux contacts stockés dans le téléphone ou dans l'agenda ; accéder aux photos ainsi que d'autres données depuis l'application.

3.4. Développement de l'application :

Pour pouvoir générer et lire un code-barres ou un QR code, l'application que nous avons développée repose sur deux librairies principales à savoir **ZXing (Zebra Crossing)** pour Java dont le rôle est de décoder et de générer ces deux types de codes ; ainsi que la librairie **Yuriy-Budiyev** sert à la mise au point (focus) pour optimiser la lecture des images. Donc, la première étape du développement est l'implémentation de ces deux librairies.

3.4.1. Installation des librairies :

Dans Android Studio, après avoir créé un nouveau projet, nous avons installé les deux librairies en question comme indiqué dans la figure 3.6 ci-dessous [12].

```
dependencies {
    implementation 'androidx.appcompat:appcompat:1.4.1'
    implementation 'com.google.android.material:material:1.6.1'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.3'
    implementation 'com.airbnb.android:lottie:5.0.2'
    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation 'androidx.test.ext:junit:1.1.3'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
    implementation 'com.google.android.gms:play-services-vision:20.1.3'
    //Zxing Library
    implementation 'com.journeyapps:zxing-android-embedded:4.3.0'
    coreLibraryDesugaring 'com.android.tools:desugar_jdk_libs:1.1.5'
    implementation 'com.google.zxing:core:3.3.0'
    //Yuriy Library
    implementation 'com.github.yuriy-budiyev:code-scanner:2.1.2'
    //CardView Library
    implementation 'androidx.cardview:cardview:1.0.0'
}
```

Figure 3.6 : Installation des bibliothèques **ZXing (Zebra Crossing)** et **Yuriy-Budiyev** .

3.4.2. Les différentes fonctions :

La seconde étape du développement consiste en l'écriture des codes correspondants aux différentes fonctions présentes dans notre application.

La fonction QrScanner :

Le rôle de la fonction QrScanner est de scanner en premier lieu un code-barres ou un QR code et de l'interpréter ensuite. Autrement dit, cette fonction sert à lire les différents codes pris en image par la caméra du smartphone.

```
// Scan QRCode
public void runCodeScanner() {
    codeScanner = new CodeScanner( context: this, scannerView);
    codeScanner.setAutoFocusEnabled(true); codeScanner.setFormats(CodeScanner.ALL_FORMATS);
    codeScanner.setScanMode(ScanMode.CONTINUOUS);codeScanner.startPreview();
    codeScanner.setDecodeCallback(new DecodeCallback() {
        @Override
        public void onDecoded(@NonNull Result result) {
            runOnUiThread(new Runnable() {
                @Override
                public void run() {
                    try {
                        String data = result.getText();
                        //Affiché la résultat
                        AlertDialog.Builder builder = new AlertDialog.Builder( context: ScanActivity.this);
                        builder.setTitle("Scan Result");
                        builder.setIcon(R.mipmap.app_logo);
                        builder.setMessage("" + result.getText());
                        AlertDialog alert1 = builder.create();
                        alert1.setButton(DialogInterface.BUTTON_NEUTRAL, text: "Close", new DialogInterface.OnClickListener() {...});
                        alert1.setButton(DialogInterface.BUTTON1, text: "copy", new DialogInterface.OnClickListener() {...});
                        alert1.setButton(DialogInterface.BUTTON2, text: "Share", new DialogInterface.OnClickListener() {...});
                        alert1.setCanceledOnTouchOutside(false);
                        alert1.show();
                    }catch (Exception e) { e.printStackTrace(); }
                }
            });
        }
    });
}
```

Figure 3.7 : Code de la fonction QRScanner

La fonction QrGenerator :

Contrairement à la précédente, cette fonction a pour rôle principale de générer un QR code à partir de n'importe quelle chaîne de caractères entrée à partir du clavier du smartphone.

En effet, se basant sur les fonctionnalités permises par la librairie **ZXing (Zebra Crossing)**, cette fonction fait appel à cette librairie qui analyse à son tour la chaîne de caractères entrée pour en déterminer le type (numériques, alphanumériques...), l'encode selon la norme ISO18004. Puis, vient la concaténation des données binaires (0 et 1) et l'ajout des données complémentaires selon le mode utilisé (numérique, alphanumérique...). Ces données binaires vont être découpées par octets, soit 8 à 8 appelés codewords et enfin le placement de ces derniers et la création du masque qui donnera l'image finale du QR code.

```
// Function to generate qrCode
MultiFormatWriter multiFormatWriter = new MultiFormatWriter();

public void qrCode() throws WriterException {
    BitMatrix bitMatrix = multiFormatWriter.encode(text.getText().toString(),
        BarcodeFormat.QR_CODE, width: 350, height: 350);
    BarcodeEncoder barcodeEncoder = new BarcodeEncoder();
    Bitmap bitmap = barcodeEncoder.createBitmap(bitMatrix);
    qrCodeView.setImageBitmap(bitmap);
}
```

Figure 3.8 : Code de la fonction QrGenerator.

La fonction SaveImage :

Comme tout le travail de capture et de scan des QR codes se fait par l'intermédiaire de l'appareil photo ou caméra du smartphone, il est indispensable de sauvegarder l'image de n'importe quel code-barres ou QR code scanné dans la bibliothèque des images avant de le décrypter ou de le partager. C'est pour cette raison que l'on a utilisé la fonction SaveImage dont le code est montré dans la figure 3.9 ci-dessous :

```
//Save Image To Gallery
private void saveImageToGallery(){
    FileOutputStream fileOutputStream = null;
    File file=getdisc();
    if (!file.exists() && !file.mkdirs())
    {
        Toast.makeText(getApplicationContext(), text: "sorry can not make dir",Toast.LENGTH_LONG).show();
        return;
    }
    SimpleDateFormat simpleDateFormat=new SimpleDateFormat( pattern: "yyyymmsshmmss");
    String date=simpleDateFormat.format(new Date());
    String name="img"+date+".jpeg";
    String file_name=file.getAbsolutePath()+"/"+name;
    File new_file=new File(file_name);
    try {
        fileOutputStream =new FileOutputStream(new_file);
        Bitmap bitmap = viewToBitmap(qrCodeView,qrCodeView.getWidth(),qrCodeView.getHeight());
        bitmap.compress(Bitmap.CompressFormat.JPEG, quality: 100,fileOutputStream);
        Toast.makeText(getApplicationContext(), text: "succes", Toast.LENGTH_LONG).show();
        fileOutputStream.flush();
        fileOutputStream.close();
    } catch (FileNotFoundException e) {
    } catch (IOException e) {} refreshGallary(file);
}
```

Figure 3.9 : Code de la fonction SaveImage.

3.4.3. Fonctionnalités de l'application :

L'application que nous avons réalisée comporte cinq (5) fonctionnalités principales représentée chacune par un bouton. Le sixième bouton donne des informations complémentaires sur l'application.

L'interface est comme représentée dans la figure qui suit :

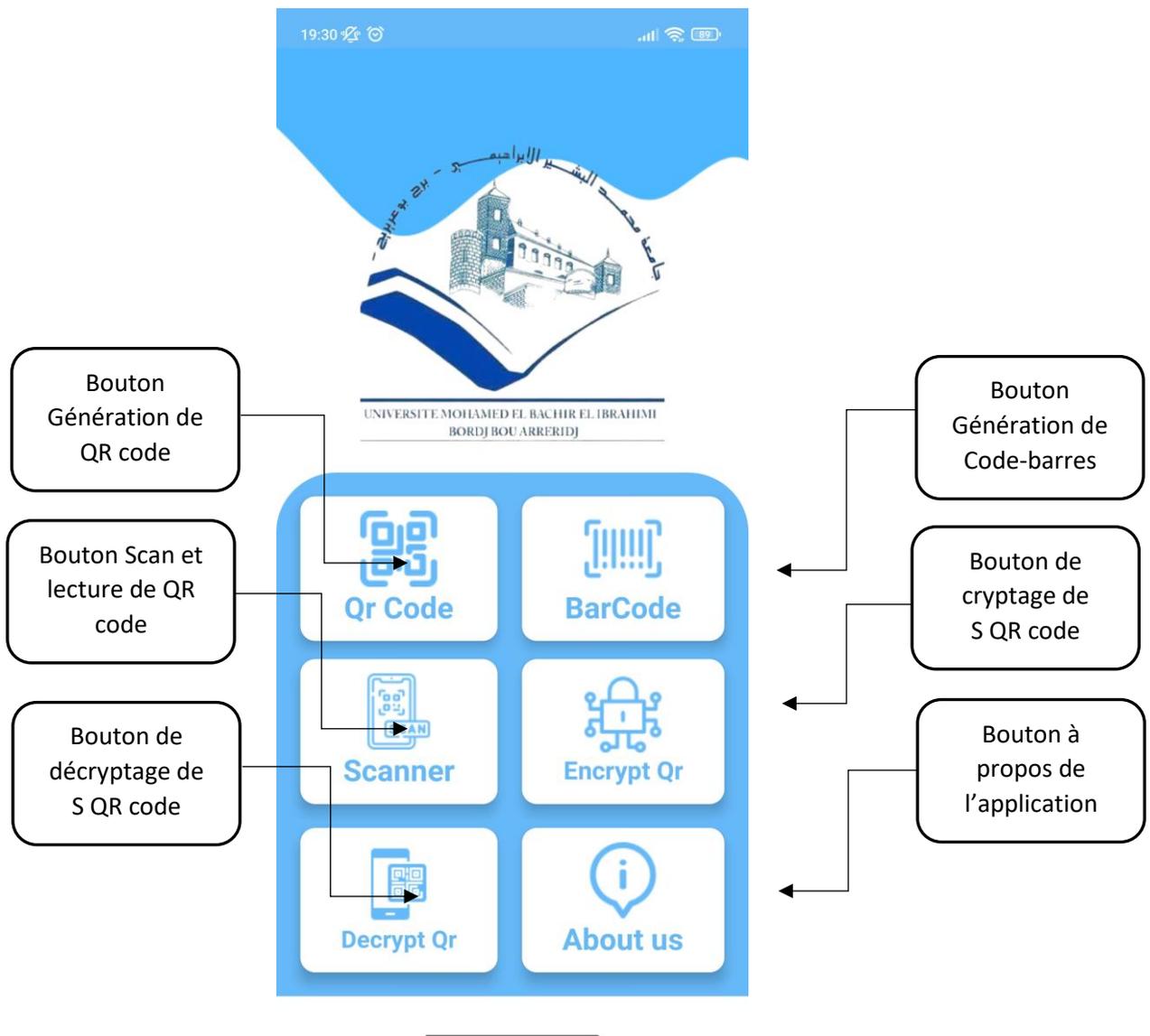


Figure 3.10 : Interface et principales fonctionnalités de l'application.

3.4.3.1. Fonctionnalité Qr Code :

Ce bouton exécute la fonctionnalité de génération de QR code. Il renvoie à l'écran de saisie de la chaîne de caractères (voir **figure 3.11**). Une fois Cette chaîne de caractères saisie, on pourra alors générer le code QR correspondant en appuyant sur le bouton « Generate ». Une fois le code QR généré, on pourra sauvegarder son image en appuyant sur « Save » ou la partager par le bouton « Share ».

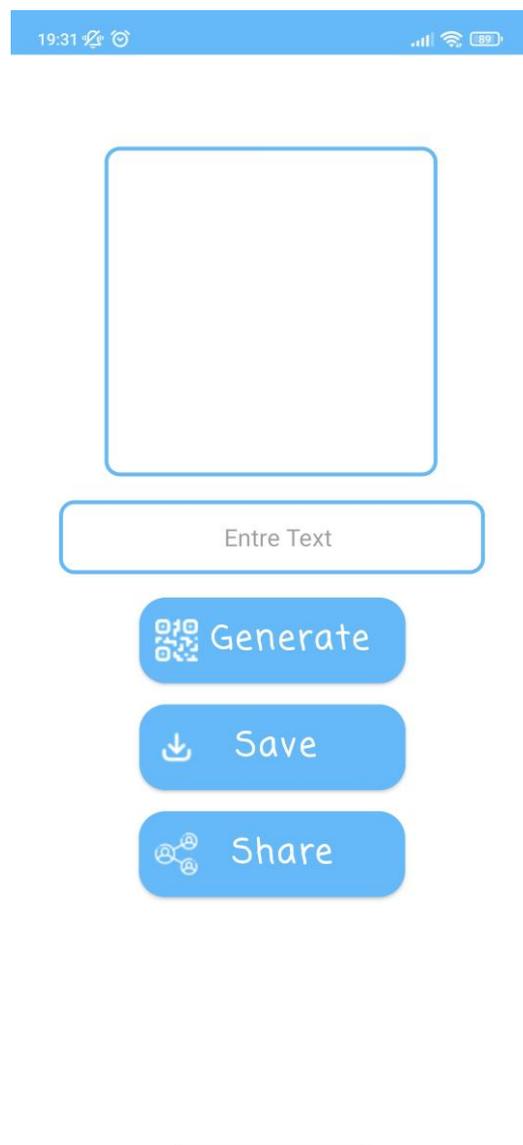


Figure 3.11 : Ecran d'accueil de la fonctionnalité Qr Code.

Une fois la chaîne de caractères entrée dans la case de saisie, la génération du QR code correspondant se fait grâce au bouton « Generate ». On pourra alors sauvegarder l'image dans la bibliothèque ou la partager sur d'autres appareils. (Voir la **figure 3.12**)



Figure 3.12 : Génération du Qr Code avec sauvegarde et partage.

3.4.3.2. Fonctionnalité Bar Code :

Ce bouton exécute la fonctionnalité de génération de code-barres. Il renvoie à l'écran de saisie de la chaîne de caractères (voir **figure 3.13**). Une fois Cette chaîne de caractères saisie, on pourra alors générer le code-barres correspondant en appuyant sur le bouton « Generate ». Une fois le code-barres généré, on pourra sauvegarder son image en appuyant sur « Save » ou la partager par le bouton « Share ».



Figure 3.13 : Ecran d'accueil de la fonctionnalité Barcode.

Une fois la chaîne de caractères entrée dans la case de saisie, la génération du code-barres correspondant se fait grâce au bouton « Generate ». On pourra alors sauvegarder l'image dans la bibliothèque ou la partager sur d'autres appareils. (Voir la **figure 3.14**)



Figure 3.14 : Génération du code-barres avec sauvegarde et partage.

3.4.3.3. Fonctionnalité Scanner :

Ce bouton exécute la fonctionnalité de scan et lecture d'un QR code ou d'un code-barres. Avant de renvoyer à la fonction Qrscan destinée au scan d'une image QR code ou code-barres, il active l'appareil photo du smartphone, fait appel à la librairie **ZXing (Zebra Crossing)** chargée de transformer une chaîne de caractères en QR code ou code-barres et vice versa. L'écran de scan d'image est alors ouvert (voir **figure 3.15**). Dès que le code scanné est supporté par la librairie, la chaîne de caractères correspondante s'affiche. Il est à noter que l'on pourra utiliser l'appareil photo pour capter différents codes, les sauvegarder dans la galerie pour

ensuite être ouverts dans l'application juste en appuyant sur Gallery et choisir l'image du QR code ou code-barres à lire.

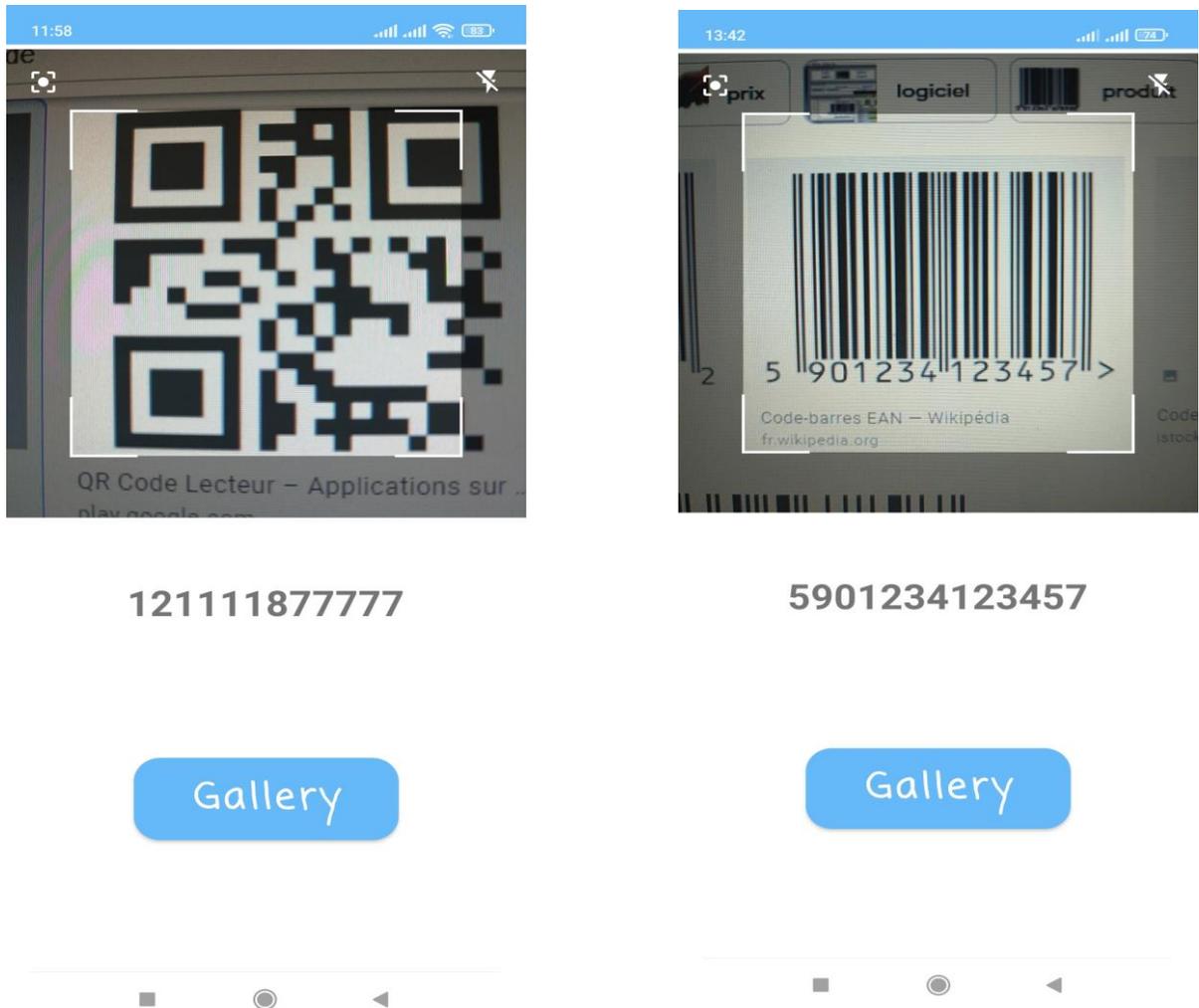


Figure 3.15 : Ecran de scan QR code ou code-barres.

3.4.3.4. Fonctionnalité Encrypt Or :

Ce bouton exécute la fonctionnalité de cryptage d'un QR code produisant ainsi un S QR code. Il renvoie à l'écran de saisie du code choisi sous forme de chaîne de caractères (voir **figure 3.16**). La chaîne de caractères entrée est transformée en une autre chaîne de caractères selon un mode de cryptage choisi produisant ainsi une chaîne de caractères plus longue et illisible. C'est cette dernière chaîne de caractères produite qui sera transformée en un QR code crypté en appuyant sur le bouton « Generate ». L'image obtenue pourra être sauvegardée en appuyant sur « Save » ou partagée par le bouton « Share ». Il est à noter enfin que la lecture de

ce QR code par la méthode classique renverra un résultat incompréhensible et nécessitera donc l'utilisation de la fonction de décryptage propre à notre application.



Figure 3.16 : Ecran de génération de QR code crypté.

3.4.3.5. Fonctionnalité Decrypt Qr :

Ce bouton exécute la fonctionnalité de décryptage d'un QR code produit à l'aide de notre application car il renferme un mode de cryptage spécifique. Il renvoie à l'écran de scan du QR code crypté (voir **figure 3.17**). Une fois le QR code crypté scanné, il renverra à la chaîne de caractères produite par le système de cryptage choisi. Cette chaîne de caractères est inversement décodée donnant la chaîne de caractères principalement saisie représentée par le QR code produit.

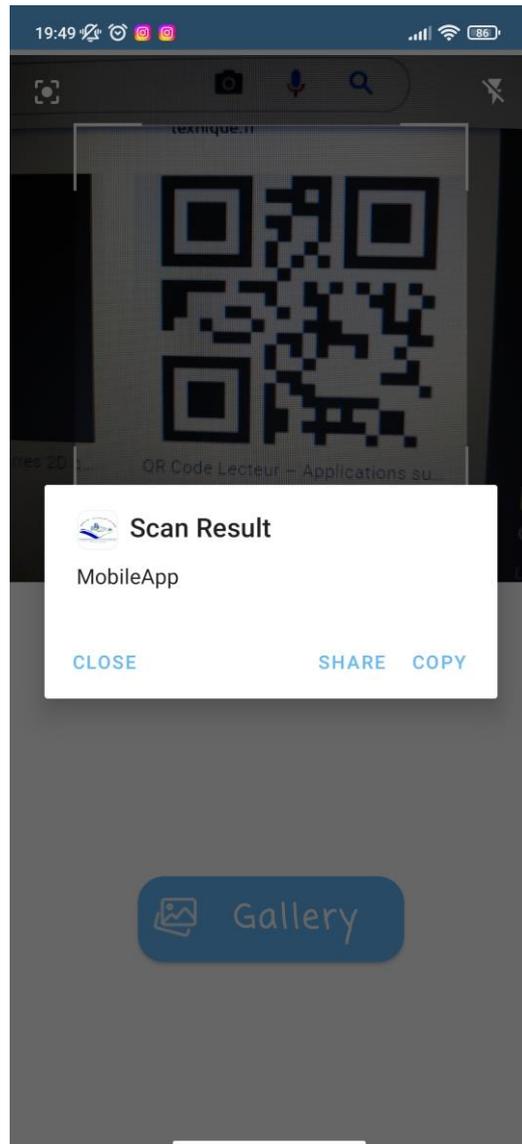


Figure 3.17 : Ecran de scan et de décryptage d'un QR code crypté

3.4.3.6. Le partage des images QR code :

Dans les trois opérations précédentes de génération ou lecture des codes-barres et QR codes classiques et cryptés, la fonction de partage des images produites est assurée dans les différentes plates-formes grâce au bouton « share » (Voir **Figure 3.18**)

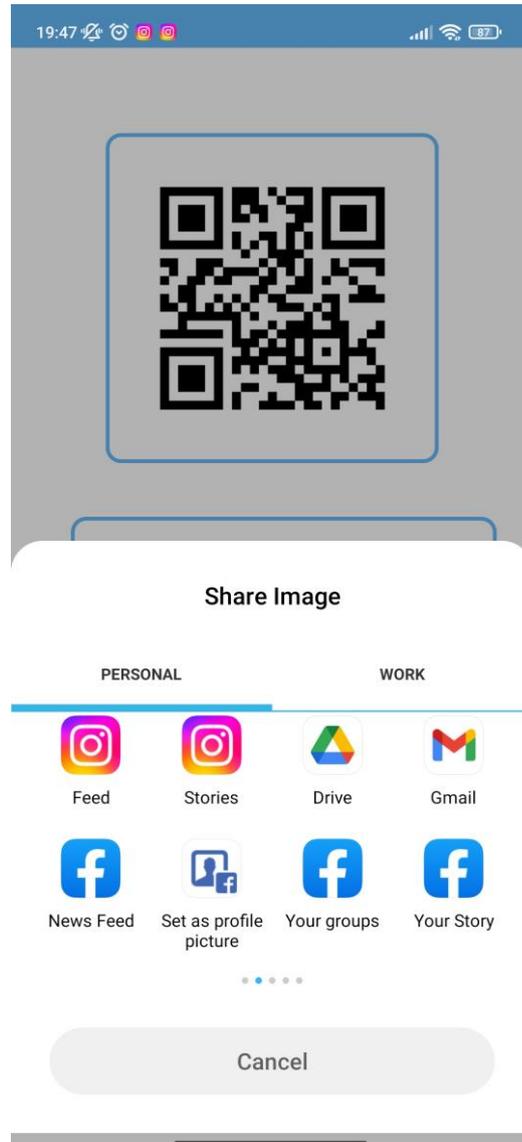
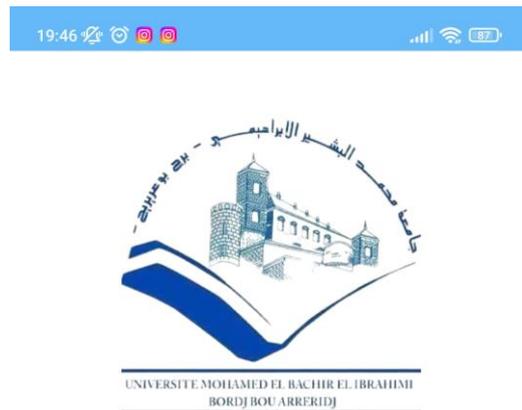


Figure 3.18 : Ecran de scan et de décryptage d'un QR code crypté

3.4.3.7. La fonctionnalité « à propos » :

Ce bouton exécute la fonctionnalité « about us », qui renvoie à l'écran présentant des informations sur l'application ainsi que les droits d'auteur comme indiqué dans la figure qui suit :



Université Mohamed Elbachir Ibrahimy

Faculté Science et Technologie

Département: Électronique

Spécialité:

Électronique des Systemes Embarquée

Projet fin d'etude

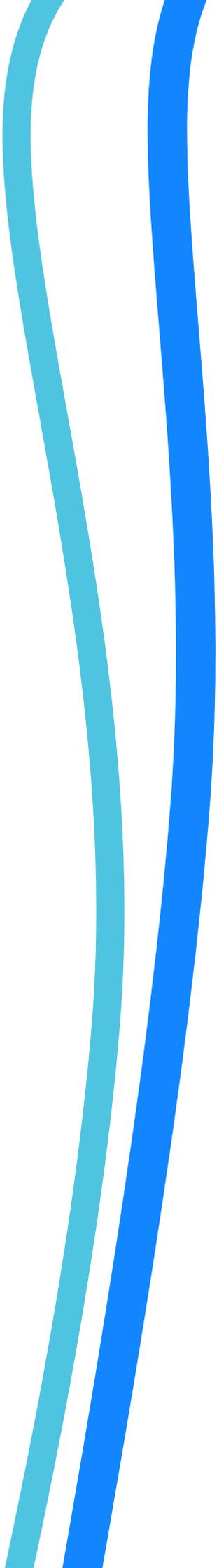
Figure 3.19 : Ecran « about us ».

3.5. Conclusion :

Grâce à l'utilisation de l'environnement de développement intégré Android Studio, nous avons réalisé une application mobile qui gère les opérations de lecture et de génération des deux types de codes à savoir le code-barres et le QR code.

Ce travail a été facilité par la richesse des bibliothèques compatibles avec Android Studio et aussi par l'aisance d'écriture du code dans cet environnement de développement. Cette facilité nous a poussés à ajouter une autre fonctionnalité à notre application, celle de produire des QR codes cryptés et de les lire.

Cette application fonctionne correctement et nous espérons enfin qu'elle puisse rendre service à ses utilisateurs futurs et pourquoi pas concurrencer avec ses semblables du commerce. Toutefois, on notera bien qu'elle reste ouverte à tout développement futur.



Conclusion générale

Conclusion générale

Conclusion générale :

L'utilisation du QR code et du code-barre étant généralisée dans tous les domaines. En effet, les applications gérant ces deux types de codes tant pour la génération que pour la lecture, ne cessent de s'améliorer de jour en jour : facilité d'utilisation, rapidité d'exécution et finesse des résultats obtenus quel que soit la plateforme de développement utilisée.

Notre projet consistait en fait en la conception et la réalisation d'une application Android de lecture / génération de QR codes et de barres-codes.

L'application réalisée après l'étude des différentes plateformes et types d'applications, permet de générer aisément les deux types de codes et de les sauvegarder sous forme d'image et ils pourront ultérieurement lus et interprétés par n'importe quelle application mobile dédiée à cet effet ou un terminal de lecture de QR codes et codes-barres.

Notre application est capable en outre de lire les codes-barres et les QR codes générés par elle-même ou scannés avec la même vitesse d'exécution et la facilité d'utilisation fournies par ce genre d'applications.

Pour enrichir notre application et ses domaines d'utilisation, nous avons opté pour l'ajout d'une fonction génération et lecture des QR codes cryptés ou chiffrés poussant ainsi l'emploi de l'application dans un niveau de sécurité d'emploi des QR codes plus innovatif, plus créatif, plus sécurisé offrant un large éventail d'emploi dans le futur.

Enfin, nous pourrions dire que notre application est d'une simplicité d'utilisation qui lui permet de prendre une belle place parmi ses semblables grâce à son efficacité. Elle reste néanmoins ouverte à tout développement ultérieur vu la large utilisation de ce type d'application et des innovations qui voient le jour dans ce domaine chaque instant.

Références

REFERENCES

- [1] <https://htpratique.com/wifi-bluetooth-nfc-rfid/>
- [2] CHRISTOPHE COQUIS, NFC, QR Codes, iBeacon, RI : tout savoir sur les technologies sans contact, 2014
- [3] <https://justaskthales.com/en/10-things-to-know-about-contactless-technology/>
- [4] [https://htpratique.com/guide-wifi-En termes simplifies/le WiFi haut débit et internet /](https://htpratique.com/guide-wifi-En termes simplifies/le WiFi haut débit et internet/)
- [5] <https://definir-tech.com/app2/30013/est-ce-que-le-bluetooth-fonctionne-sans-wifi#>
- [6] <https://doi.org/10.3991/ijim.v14i09.11548>
- [7] Imene Sghaier, Développement d'applications mobiles (Support de cours)
- [8] Amosse EDOUARD, Introduction au Développement d'Application Android pp 8
- [9] <https://socialcompare.com/fr/comparison/android-versions-comparison>
- [10] Guillet O., Les différents types d'applications mobiles : natives, Web apps, hybrides, Flash, [http ://olivierguillet.com/2012/02/les-differents-types-dapplications-mobiles-natives-webapps-hybrides-flash/](http://olivierguillet.com/2012/02/les-differents-types-dapplications-mobiles-natives-webapps-hybrides-flash/), Consulté le 05 juin 2022.
- [11] Dylan TEIXEIRA, Créer une application Android
- [12] Abdelhak-Djamel Seriai, Développement d'Applications Mobiles sous Android 2015
- [13] Collectif, Support de cours, Android Studio – SDK Présentation de l'outil.