

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOHAMED BOUDIAF - M'SILA

**FACULTE MATHÉMATIQUE ET
INFORMATIQUE**
DEPARTEMENT INFOFRMATIQUE

N° :



DOMAINE : Mathématique et Informatique
FILIERE : INFORMATIQUE
OPTION : OMID

**Mémoire présenté pour l'obtention
Du diplôme de Master Académique**

Par : Laifa Abdelhadi

Intitulé

**Conversation Multimédia dans des réseaux
domestique large sous Android**

Soutenu devant le jury composé de :

L.Sayad	Université de M'sila	Président
Mr Debbi Imadeddine	Université de M'sila	Rapporteur
F.Mezrag	Université de M'sila	Examineur

Année universitaire : 2016 /2017

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

UNIVERSITE MOHAMED BOUDIAF - M'SILA

**FACULTE MATHÉMATIQUE ET
INFORMATIQUE**

DEPARTEMENT INFOFRMATIQUE

N° :.....



DOMAINE : Mathématique et Informatique

FILIERE : INFORMATIQUE

OPTION : OMID

**Mémoire présenté pour l'obtention
Du diplôme de Master Académique**

Par : Laifa Abdelhadi

Intitulé

**Conversation Multimédia dans des réseaux
domestique large sous Android**

Soutenu devant le jury composé de :

L.Sayad

Université de M'sila

Président

Mr Debbi Imadeddine

Université de M'sila

Rapporteur

F.Mezrag

Université de M'sila

Examineur

Année universitaire : 2016 /2017



Dédicace :

Je dédie ce travail

A mon père son soutien et pour ses encouragements et ses précieux conseils durant toute ma vie. A ma mère pour son amour inconditionnel et sa présence à mes côtés dans les moments difficiles.

Ma grand-mères et Mes Grands-pères.

Mes chers frères et Mes chères sœurs.

Mes chères amis.

LaiFa AbdElHadi



Remerciement :

Tout d'abord, Je tiens à remercier le grand dieu tout puissant qui nous a donné la force et la volonté pour arriver au maximum de notre travail.

Je remercie mon encadreur Mr. Debbi Imad pour sa précieuse assistance et sa disponibilité pour ce travail.

Je tiens à remercier les membres de jury pour l'honneur qu'ils nous font en jugeant ce modeste travail.

Enfin, mes remerciements vont à toute personne ayant contribué, de près ou de loin, à l'aboutissement de ce travail.

Table des matières

INTRODUCTION GÉNÉRALE.....	1
CHAPITRE 1 VILES INTELLEGENES (SMART CITY).....	2
Introduction.....	3
1.2 Villes intelligentes.....	3
1.3 Le concept anglo-saxon de smart city.....	4
1.4 Réseaux pour villes intelligentes.....	4
1.5 Le défi : l'intégration des trois composantes.....	6
1.6 Les trois modèles de la ville intelligente.....	6
1.7 La ville intelligente, une solution	7
1.8 Critiques	7
1.9 Étude sur la « maturité » des villes européennes.....	8
1.10 Expérimentation par des villes	9
Conclusion.....	11
CHAPITRE2 : SMARTPHONE,ANDROID ,Wifi-direct...12	12
Introduction	13
2.1.2 Définition.....	13
2.1.3 Histoire du Smartphone.....	14
2.1.4 Les fonctionnalités d'un Smartphone	15
2.1.5 Les systèmes d'exploitation mobiles.....	15
2. Android.....	20
2.2.1 Description.....	20
2.2.2 La naissance d'Android.....	20
2.2.3 Historique d'Android	21
2.2.5 Android face à la concurrence	23
2.2.6 Architecture Android.....	24
3.Wifi Direct	30
Introduction.....	30

2.3.1	Présentation technique	30
2.3.2	Architecture	30
2.3.3	Sécurité	32
2.3.5	Avantages	32
Conclusion	33
CHAPITRE 3 : CONCEPTION DU SYSTEME	34
Introduction	35
3.1	ETUDE DE L'EXISTANT.....	35
3.2	Critique les applications.....	36
-Solution	36
3.3	Présentation du projet.....	36
3.4	Domaine couvert par le projet.....	36
3.5	Les limites de l'application.. ..	36
3.6	Architecture fonctionnelle	37
UML	37
3.7	Définition	38
3.8	L'Histoire d'UML	38
3.9	les diagrammes	40
3.1	Les diagrammes de l'Application	41
-Diagramme de cas d'utilisation	41
-Diagramme de séquence	42
-Définition du diagramme de classes	49
-Définition du diagramme d'activité	50
Conclusion:	52
CHAPITRE 4: LA REALISATION	45
Introduction	54
3.1	Les outils utilisés.....	54
3.1.1	Android studio.....	54
3.1.2	Le plugin ADT.....	55

3.1.3 Software Development Kit (SDK)	55
3.1.4 L'émulateur.	55
3.1.5 Les langages de programmation.....	56
1.13 Captures d'écrans	57
Conclusion générale	60
Bibliographie	61

Liste des Figures

Figure1.1: les trois composantes de la ville intelligente	6
Figure1.2: Ville de Lyon.....	9
Figure1.3 : Ville de Nice.....	9
Figure1.4 : Ville d'Amsterdam	10
Figure 1.5 : Ville de Paris.....	10
Figure1.6 : Ville de Madrid.....	10
Figure1.7 : Ville de Singapore	11
Figure1.8 : Ville de Montréal.....	11
Figure2.1 : Les Smartphones.....	13
Figure 2.2 : BlackBerry	15
Figure2.3 : Windows Mobile	17
Figure 2.4 : IPhone	18
Figure 2.5 : La collections IPhone	19
Figure2.6 : Concurrence d'Android.....	23
Figure 2.7 : Architecture Android.....	24
Figure 2.8 : Applications.....	25
Figure 2.9 : Framework de développement.....	25
Figure 2.10 : Bibliothèques	26
Figure2.11 : Android Runtime	28
Figure 2.12 : Linux Kernel.....	28
Figure 2.13 : Architecture de Wifi-Direct.....	31
Figure 3.1 : Historique d'UML.....	39
Figure 3.2 : Architecture UML	40
Figure 3.3 : Le diagramme de cas d'utilisation.....	41
Figure3.4 : Le diagramme de Séquence de l'application	42
Figure3.5 : Le diagramme de Séquence de l'envoi un message	44
Figure 3.6 : Le diagramme de Séquence de l'envoi un fichier (Audio, vidéo, photo, PDF.....	45
Figure 3.7 : Le diagramme de Séquence de l'appel Vidéo/Audio	47
Figure 3.8 : diagramme de classes	49
Figure 3.9 : diagramme d'activité.....	51
Figure 4.1 : Android studio	54
Figure 4.2 : SDK	55
Figure.4.3 : L'Envoie des messages	57
Figure 4.4 : Première interface de l'application.....	57
Figure 4.5 : L'Envoie des Photos	57
Figure 4.6 : Message vidéo	58
Figure 4.7 : Message vidéo	58
Figure4.8 : L'envoi des fichiers.....	59

Liste des tableaux :

Tableau 2.1 Historique d'Android.....	20
Tableau 3.1 Description de diagramme de séquence de L'application	43
Tableau 3.2 Description de diagramme de séquence Envoi Message/Fichier.....	46
Tableau 3.3 Description de diagramme de séquence Appel Video/Audio	48

INTRODUCTION GÉNÉRALE

Dans la vie humaine les sources d'informatique sont évolués et Progressés plus vite que le passé parce que Ya beaucoup plus de moyens disponible pour le développement soit des logiciels , Matériels et technologiques, ce dernier développe de plus en plus par rapport aux années l'une de ces choses que nous trouvons les villes intelligente(Smart city).

L'être humain ont toujours eu besoin de communiquer malgré les distances. Autrefois, les porteurs de messages remplissaient cette mission. Mais il a fallu trouver des moyens de communication plus rapides et plus sûrs, quels que soient les obstacles à franchir

Dans le cadre d'une 2eme année Master Outils et Méthodes pour L'Informatique Décisionnelle (OMID) de l'Université de M'sila, et en vue de la réalisation d'un projet technique « Conversation multimédia dans des réseaux domestiques larges sous Android» consiste à développer une application mobile permettant de rendre spontané, directes et en temps réels les conversations textuel ,partage des fichiers et faire des appels audio et vidéo. Cette application tourne sur des réseaux locaux sans fil et couvre donc les agglomérations qui sont à portée de ces réseaux. L'application a l'avantage d'être très sécurisée, elle permet de renforcer de manière fiable et efficace le dialogue spontané entre individus qui se partage un intérêt dans un périmètre spécifique. Elle peut faire l'objet d'une utilisation domestique large. Dans ce cas le service est fourni en privée et d'une façon plus rassurée. Elle peut être déployé dans un campus, un département ou un siège délimité où des membres coopèrent pour une activité particulière. Cette application est intégrée dans un Smartphone tactile de type Android

Notre travail est organisé comme quatre chapitres ou le premier présent la ville intelligence ensuite, le deuxième chapitre présent généralités sur les smartphone, Android et Wifi-direct . Ensuite en a fait l'étude du domaine comme une troisième chapitre qui contient deux partie l'un pour l'étude préalable (étude de l'existent)et l'autre pour l'analyse et conception général du système. Enfin, l'explication des outils utilisés dans le développement de l'application avec des captures écran dans la quatrième chapitre.

CHAPITRE 1
VILLE INTELLEGENTE (SMART CITY)

Introduction:

Dans ce chapitre nous présenterons d'une façon générale la Ville intelligente, sa définition sa caractéristique, les solutions obtenus et des exemples d'expérimentations dans le monde.

1.1 Ville intelligente :

Une ville est un espace vivant en constant changement qui s'adapte et qui se modifie au fil du temps. La forme des villes change aussi, notamment au courant de la moitié du dernier siècle avec l'augmentation de la concentration de population dans les villes. Pour répondre aux nouveaux défis urbains des villes, les penseurs, les différents acteurs du milieu public et les professionnels ont imaginé et conçu différents modèles et concepts de villes à travers les années.

Les nouvelles technologies de l'information et des communications (NTIC) sont l'élément qui a inspiré le concept de ville intelligente. Ce terme a été adopté en 2005 par plusieurs compagnies du milieu des technologies informatiques (Harrison et Donnelly, 2011). Les géants de l'informatique, dont IBM, Siemens, CISCO, ainsi que plusieurs autres collaborateurs s'affairent à développer des technologies d'information et de communication ainsi que des applications informatiques. Au-delà des bases de données ouvertes, des technologies avancées et des applications qui permettraient de faire évoluer la ville, il faut se pencher sur les défis de ce concept en matière de développement durable pour la ville.

De nos jours, les performances urbaines ne dépendent plus seulement de la dotation de la ville d'infrastructures (son capital physique) mais aussi, et de plus en plus, de la disponibilité et la qualité de la communication du savoir et l'infrastructure sociale (capital social et capital intellectuel). La dernière forme de capital est décisive pour la compétitivité urbaine. C'est sur ces bases que le concept de « ville intelligente » a été introduit comme outil pour englober les facteurs de production urbaine moderne dans un cadre commun et pour mettre en valeur l'importance grandissante des technologies de l'information et de la communication (TIC) et des capitaux social et environnemental pour définir la compétitivité des villes. L'importance de ces deux atouts (capital social et environnemental) suffit par elle-même à différencier la ville intelligente de ses homologues plus orientées vers la technologie, marquant la frontière entre elles et celles que l'on appelle

« villes en réseau », même si cette différence n'est pas toujours clairement perçue en France [1].

1.2 Le concept anglo-saxon de smart city

Selon Giffinger, les villes intelligentes peuvent être identifiées et classées selon six dimensions principales. Ces critères sont : une « économie intelligente », une « mobilité intelligente », un « environnement intelligent », des « habitants intelligents », un « mode de vie intelligent » et, enfin, une « administration intelligente ». Ces six critères se connectent avec les traditionnelles théories régionales et néoclassiques de la croissance et du développement urbain. Ils sont respectivement basés sur les théories de la compétitivité régionale, l'économie des transports et des technologies de l'information et de la communication, les ressources naturelles, les capitaux humains et sociaux, la qualité de vie et la participation des citoyens à la vie démocratique de la ville. La notion d'intelligence inclut ici les concepts d'efficience énergétique, d'habitat intelligent, de réseaux intelligents et de mobilité douce.

Pour Jean Bouinot, la ville intelligente est aussi « celle qui sait à la fois attirer et retenir des entreprises employant de la main-d'œuvre hautement qualifiée ».

Au sujet de l'utilisation efficiente et intégrée des TIC, en 1992 G. Dupuy parle d'« informatisation des villes ».

Dans ce cadre se développe aujourd'hui le concept de réalité augmentée à travers la téléphonie mobile.

Le marché des Villes Intelligentes est un marché en constante évolution. Selon un rapport du cabinet Grand View Research, le marché de la ville intelligente atteindra 1 400 milliards de dollars d'ici à 2020. [2]

1.3 Réseaux de capteurs sans fil pour les villes intelligentes

Un réseau de capteurs sans fil est une technologie spécifique qui aide à la création de villes intelligentes. Leur but est de créer un réseau réparti de noyaux de capteurs intelligents qui peuvent mesurer plusieurs paramètres intéressants pour une meilleure gestion de la ville.

Toutes les données sont transmises en temps réel aux citoyens ou aux autorités concernés.

Par exemple, les citoyens peuvent surveiller le niveau de pollution dans chaque rue de la ville ou encore recevoir une alerte quand le niveau de radiations atteint dépasse une certaine limite. Cela offre aussi la possibilité aux autorités d'optimiser l'irrigation des parcs ou l'éclairage de la ville. De plus, les fuites d'eau peuvent être facilement détectées, et des cartes dressant l'état

de la pollution sonore peuvent être créées. Les poubelles peuvent aussi être plus intelligentes, des capteurs permettant de déclencher une alarme lorsqu'elles sont presque pleines.

Le trafic routier peut être contrôlé pour modifier l'éclairage urbain de manière dynamique. De même, le trafic peut être réduit grâce à des systèmes détectant la place de parking la plus proche. Les automobilistes sont informés en temps réel et peuvent rejoindre rapidement une place libre, économisant ainsi du temps et du carburant. Tout cela réduit la pollution et les embouteillages tout en améliorant la qualité de vie.

De même des capteurs peuvent permettre aux autorités de la ville de contrôler en temps réel l'occupation des places de stationnement en dépassement sur de la zone bleue. Une première réalisation d'envergure (près de 300 capteurs) a été mise en service aux Mureaux en juin 2014, réalisation suivie par les villes du Touquet, du Havre, de Rosay-en-Brie, de Verrieres-le-Buisson, de Calais, etc. À noter que les villes de Zoug (Suisse), Madrid, Verone, Moscou, Courtrai (Belgique), Arrezzo (Italie), New-York, etc. ont également été équipées avec ces mêmes capteurs à raison de plusieurs milliers d'exemplaires pour tous types d'applications: emplacements type "Arrêts-Minutes" ("Shop & Go" en Belgique), Optimisation du Stationnement payant, Guidage Dynamique de Voirie, Parkings "Poids-Lourds" aux entrées de villes, contrôle des dépose-minutes (Gares), surveillance des places spécifiques: PMR / Livraisons / Ambulance / Pompiers / Police etc.. [3]

1.4 Le défi : l'intégration des trois composantes de la ville intelligente

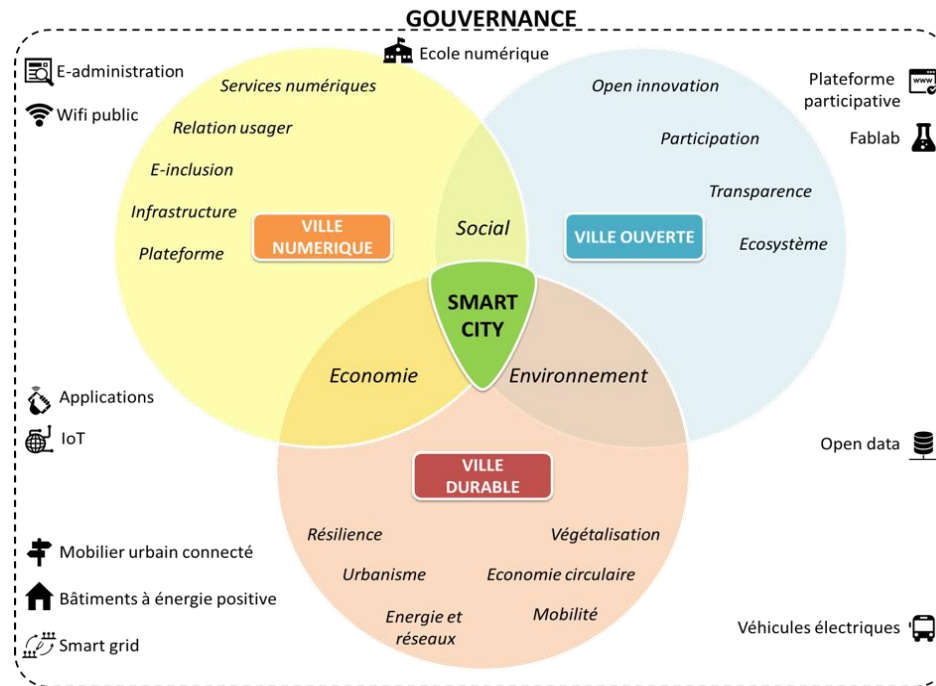


Figure 1.1 :les trois composantes de la ville intelligente [4]

La ville intelligente ne se définit plus par rapport aux technologies employées, mais selon une dimension humaine. La notion de ville intelligente n'a cessé de changer pour s'adapter aux nouvelles réalités ; il n'est plus possible de la cantonner à une seule acception, puisqu'elle recouvre désormais des champs très larges.

1.5 Les trois modèles de la ville intelligente:

- **La ville numérique** : qui met à disposition des habitants les ressources numériques leur permettant d'être connectés.
- **La ville ouverte** : à laquelle tout le monde peut participer .
- **La ville durable (ou « éco-cité »)** : qui prend en compte les infrastructures de la ville dans une perspective de durabilité.

Au croisement de chaque de ces modèles se trouve un axe de travail :

- L'axe social .
- L'axe économique .

- L'axe environnemental (à entendre au sens large : à la fois l'environnement de vie et le développement durable).

La Figure 1.1 proposé intègre également des exemples concrets de réalisations (FabLab, mobilier urbain connecté, etc.) qui constituent un aperçu des « services » de la ville intelligente et connectée ; le prochain billet de blog de la série smart city sera l'occasion de détailler ces services.

La ville intelligente et connectée comporte donc des facettes multiples ; par ailleurs, il convient de souligner que la smart city s'inscrit dans un écosystème « smart et digital » plus large et global. Elle peut ainsi être le prolongement des technologies au service des individus (par exemple les maisons connectées ou les téléservices médicaux).

1.6 La ville intelligente, une solution :

La ville intelligente, une solution qui permet :

- D'éviter le développement chaotique des villes afin d'en l'attractivité et en soutenir le développement.
- D'assurer une gestion efficace des ressources.
- De valoriser la consultation citoyenne afin de stimuler l'innovation et la création collaborative;
- De maintenir une qualité de vie appréciable pour ses citoyens et citoyennes.
- De miser sur le développement durable.[4]

1.7 Critiques :

Les principaux arguments contre l'utilisation superficielle de ce concept dans le domaine politique sont les suivants :

Un parti pris dans des intérêts stratégiques peut amener à ignorer des voies alternatives de développement urbain prometteur.

Le fait de se concentrer sur le concept de ville intelligente peut entraîner une sous-estimation des effets négatifs possibles découlant du développement des nouvelles infrastructures technologiques en réseau dont une ville a besoin pour devenir intelligente.

Le risque d'imposer des solutions technologiques toutes faites, sans en penser leur intégration dans l'existant, menant à des villes censurées.

L'idée d'espaces urbains néo-libéraux a été critiquée pour les risques potentiels encourus lorsque l'on compte sur les seules valeurs économiques comme vecteur de développement urbain.

Étant donné que les modèles d'entreprises sont basés sur la mobilité du capital, suivre un modèle ainsi orienté peut entraîner un échec de la stratégie sur le long terme.[5]

1.8 Étude sur la « maturité » des villes européennes

En janvier 2014, la commission « Industrie, recherche et énergie » du Parlement européen a publié l'étude Mapping Smart cities in the EU20. Inscrite dans le contexte de la stratégie Europe 2020, cette étude a passé au crible les 468 villes européennes de 100 000 habitants ou plus afin de les classer selon une échelle à quatre niveaux de maturité des villes intelligentes²¹, sur base de leur situation en 2011 :

Niveau 1 : la ville dispose d'une politique ou d'une stratégie de ville intelligente, ce qui suffit selon le rapport à la classer comme ville intelligente .

Niveau 2 : en plus du niveau 1, la ville s'appuie pour sa politique ou sa stratégie sur une planification ou une vision .

Niveau 3 : en plus des niveaux précédents, des initiatives-pilotes de ville intelligente sont implémentées .

Niveau 4 : en plus des niveaux précédents, une initiative de ville intelligente au moins a été complètement lancée ou implémentée.

Au total, selon l'étude du Parlement européen :

240 villes européennes de 100 000 habitants ou plus atteignent le niveau 1 et sont donc identifiées comme villes intelligentes²² .

6 villes se distinguent par leur niveau de maturité le plus élevé : Amsterdam, Barcelone, Copenhague, Helsinki, Manchester et Vienne.

Plus en détail, l'étude identifie comme ville intelligente :

43 % des villes européennes de 100 000 à 200 000 habitants .

près de 90 % des villes européennes de plus de 500 000 habitants. [6]

1.9 Expérimentation par des villes :[4]

En Europe, en Amérique et en Asie, plusieurs initiatives sont mises de l'avant par les administrations publiques, par des entreprises ou par des citoyens. Par exemple, des initiatives visant à :

- ✓ Attirer des entreprises innovantes et diversifier l'économie.
- ✓ Favoriser un urbanisme actif, (analyse de la march_abilité, préservation du patrimoine, transport durable, éco_quartiers, consultation citoyenne.
- ✓ Réaliser la gestion intégrée des infrastructures souterraines (égouts, aqueducs, électricité, fibre optique.
- ✓ Permettre un meilleur contrôle environnemental (qualité de l'air, qualité des eaux, gaz à effet de serre, gestion des déchets.

- Ville de Lyon :

Utilisation de capteurs pour le monitoring de la qualité des eaux et la présence de poissons dans l'East River et la Bronx River.

ville de Lyon



Figure 1.2 : Ville de Lyon

- Ville de Nice :

Horodateurs intelligents, doublés d'une application iPhone et Android permettant de cibler rapidement les stationnements libres dans le centre-ville.

Ville de Nice



Figure 1.3 : Ville de Nice

- Ville d'Amsterdam :

Service de location de bureaux partagés et encouragement du télétravail pour diminuer la circulation automobile au centre-ville.

Ville d'Amsterdam



Figure 1.4 : Ville d'Amsterdam

- Ville De Paris :

40 initiatives liées au mobilier intelligent sont en développement depuis 2011.

Par exemple : abribus digitaux, panneaux quartiers en 3D, borne à double-face tactile, boîtes à vélos, etc.

Ville de Paris



Figure 1.4 : Ville de Paris

- Ville de Madrid :

Système de sécurité intégré qui permet aux autorités de réagir avec plus d'efficacité et de rapidité lors de situations d'urgence.

Ville de Madrid

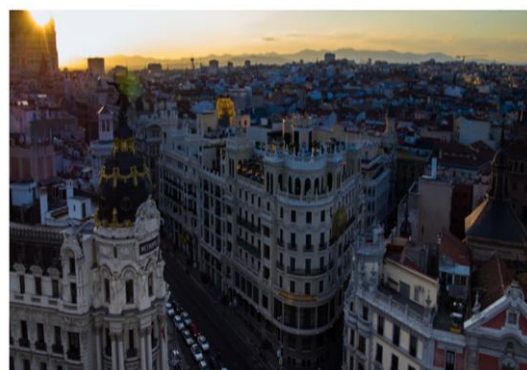


Figure 1.4 : Ville de Madrid

- Ville de Singapour :

Plutôt que d’investir dans la construction d’autoroutes, choix de systèmes de transport en commun intelligents ayant permis la prospérité et le développement durable de l’île.

Ville de Singapour



Figure 1.4 : Ville de Singapour

- Ville de Montréal :

Valorisation de l’offre culturelle du Quartier des Spectacles grâce au numérique.[4]

Ville de Montréal



Figure 1.4 : Ville de Montréal

Conclusion :

Pour développer une ville intelligente, il doit y avoir volonté, convergence et soutien :

- de l’administration publique, des entreprises et acteurs économiques et de la population.
- Pour développer une ville intelligente, il faut mettre en place des infrastructures pour supporter et intégrer les systèmes numériques.
- Le défi demeure l’intégration harmonieuse dans la ville intelligente des composantes économique, sociologique et environnementale.

Dans ce chapitre nous avons fait une étude globale sur les villes intelligentes ses caractéristiques, ses composants, ses critiques, ses solutions, et ses Expérimentation dans le monde.

CHAPITRE 2
SMARTPHONE, ANDROID et Wifi-direct

1.Smartphone

Introduction :

Le Smartphone est un compagnon de tous les instants et un véritable ordinateur de poche pour gérer le quotidien, s'informer , se divertir. Il comporte aussi un ensemble de capteurs embarqués ou connectés. Le Smartphone a donc une place à part dans la galaxie des appareils.

Pourtant, les utilisateurs savent très peu de chose sur ce qui se passe l'intérieur de ces boites.

Ainsi, dans ce chapitre nous allons essayer de voir les concepts de base des Smartphone en nous intéressant plus précisément a la plateforme de Google Inc. Android .

2.1.1 Définition

C'est quoi un Smartphone, [7] nommé aussi ordiphone ou téléphone intelligent, comme définition smart phone est un téléphone mobile disposant des fonctions d'un assistant numérique personnel. La saisie des données se fait par moyen d'un écran tactile (clavier virtuel) ou d'un clavier ordinaire téléphonique (normal ou complet "azerty - qwerty") Il fournit des fonctionnalités et options basiques comme répertoire des contacts, l'agenda, le calendrier, le navigateur web, la consultation de courrier électronique (E-mail), de messagerie instantanée en plus de SMS - MMS, le GPS, alarme réveil, et plus. À partir de fin 2007, le marché de commercialisation de Smartphones s'étend considérablement jusqu'à dépasser en quelques années celui des téléphones mobiles classiques.



Figure 2.1 Les Smartphones

2.1.2 Histoire du Smartphone:

- Et bien tout commence en 1993 Bellsouth et IBM produisent l'IBM Simon. Le 1er Smartphone de l'histoire. Un IBM donc. Un peu téléphone, un peu PDA, un peu pager et un peu fax. Gros et peu pratique ...
- 1996 : le Nokia 9000 « Communicator » présageait de tout le savoir-faire que la société Suédoise pouvait (et allait) investir dans ce tout nouveau territoire. Toujours pas très pratique ni très attirant. mais sans aucun doute l'un des modèles ayant contribué à l'apparition des Smartphones actuels.
- 1997 : L'Ericsson GS 88, en fait le 1er téléphone bénéficiant du terme Smartphone.
- En 2000, l'année du « bug » : L'Ericsson R 380, c'est le 1^{er} à avoir utilisé un OS (Operating System) propre, le Symbian.
- 2002 c'est un grand cru : C'est d'abord le rebond d'un phénomène né beaucoup plus tôt. Le Palm Pilot lancé en 1996 mais qui n'était qu'un PDA et qui est lancé en version téléphone et doté d'un clavier à touches. C'est le Palm OS treo. Et surtout le BlackBerry 5810 lancé par les canadiens de RIM. Premier matériel optimisé pour les envois & réception de mails sans fils.
- 2007 est l'année à marquer d'une pierre blanche : La naissance de l'iPhone. La révolution complète. Les fameuses « apps », iTunes, un écran tactile, l'an 2 des Smartphones !
- Et 2008 voit la naissance du premier téléphone équipé de l'OS Android, le HTC Dream.
- 2010, c'est l'année de l'explosion des Smartphones avec une multitude de produits dans toutes les gammes des fabricants de matériels électroniques.
- 2011 voit quant à elle une nouvelle révolution apparaître avec des appareils de plus en plus puissants et capables de se transformer en GPS, appareils photos ou vidéos de qualité, plateformes de jeux, commandable à la voix, ...[7]

2.1.3 Les fonctionnalités d'un Smartphone :

Les fonctionnalités d'un Smartphone sont diverses: [8]

- la gestion de l'emploi du temps.
- La consultation du courrier électronique
- La création et la consultation de documents de bureautique que l'on peut conserver, rechercher et consulter.
- La synchronisation des données avec celles de l'ordinateur de bureau.
- La possibilité de disposer d'applications multimédia (appareil photo, caméra vidéo, baladeur mp3, jeux,.....).
- La communication grâce aux technologies sans fil.
- Avec le Bluetooth, qui est une norme sans fil à courte portée, il se connecte avec les périphériques compatibles (ordinateur, téléphone mobile, imprimante,.....).
- Avec le WIFI, il joue le rôle d'un modem pour assurer une connexion internet.
- La navigation GPS.

2.1.4 Les systèmes d'exploitation mobiles

Un système d'exploitation mobile est un système d'exploitation conçu pour fonctionner sur un appareil mobile. Ce type de système d'exploitation se concentre entre autres sur la gestion de la connectivité sans fil et celle des différents types d'interface. [17]

❖ BlackBerry

BlackBerry a été créé par Research In Motion (RIM), une société Canadienne. Il est une plateforme très populaire particulièrement dans Amérique du Nord. Selon un rapport de la société Gartner, dans le marché des États-Unis, BlackBerry a maintenu 1^{er} rang n°1 avec 42% de marche. Mais cette plateforme seulement marche sur les appareils du BlackBerry.



Figure 2.2 BlackBerry

Au commencement, BlackBerry est développé pour le businessman. Il y a donc des applications pour le business comme: les messages électronique, les messages PIN, les messages texte (SMS), les messages MMS, le BlackBerry Messenger, le web navigateur, la tâche, le mémo, etc.

BlackBerry fournit le service de messagerie électronique pour les sociétés, les entreprises en utilisant BlackBerry Enterprise Server. De plus, le fournisseur offre aux abonnés des fonctions push de messagerie et d'accès mobile pratique à du contenu Internet pour leurs terminaux.

Il soutient aussi la technologie de la pièce jointe qui soutiens beaucoup de types de pièce jointe comme les fichiers avec les extensions .zip, .html, .doc, .dot, .ppt, .pdf, etc. C'est pourquoi son service de messagerie électronique est plus meilleur que les autres plateformes. D'autre part, le système de messagerie électronique est plus sûr que les autres.

Maintenant, BlackBerry est supporté dans beaucoup pays. Alors, l'utilisateur peut accéder au système de courriel de son société dans ses pays via la technologie du roaming. C'est pourquoi BlackBerry est " Standard d'or " pour la messagerie électronique. BlackBerry devient de plus en plus populaire dans le monde. Il non seulement retient les bons caractéristiques mais encore offre les fonctionnalités pour les consommateurs comme le multimédia (l'appareil photo, le camera, etc.), la carte Média, etc.

En général, malgré qu'il y a plusieurs applications tierces pour Blackberry mais le nombre d'applications tierces est moins que les applications pour Palm OS, Microsoft et Symbian.

En termes d'utilisation, la molette, le clavier QWERTY et l'interface d'utilisateur sont les excellents appareils pour faciliter d'utiliser le système de courriel.

La technologie BlackBerry Connect permet les appareils qui marche sur les autres plateformes comme Symbian OS, Palm OS et Windows Mobile peuvent connectent avec BlackBerry Enterprise Server et BlackBerry Internet Server pour fournir les services comme la synchronisation de la messagerie électronique, le visionnement de l'attachement, etc..

En conclusion, généralement pour les sociétés, BlackBerry est optimal pour les grands déploiements, mais pour les moyens ou les petits déploiements on peut choisir les autres plateformes. [17]

❖ Windows Mobile

Windows Mobile est développé par la corporation Microsoft. Alors, Windows Mobile a la capacité de fonctionnement des logiciels sur « Windows » (seulement Windows), compatible avec tous les logiciels de « Windows ». Windows Mobile soutient beaucoup de types d'audio, vidéo. Il soutient aussi la capacité de télécharger, de jouer les chansons et de regarder le TV en ligne. Grâce à ses caractéristiques, dans le domaine loisir, il est le plus bon. La messagerie électronique du Windows Mobile n'est pas meilleure que BlackBerry mais il a la perspective du succès. Au lieu d'un modèle push de la messagerie électronique comme BlackBerry, Windows Mobile fait appel à un genre de « fast-pull », qui récupère les données rapidement au lieu de l'avoir poussé à l'appareil. De plus, il est moins sécurisé que BlackBerry. En termes d'applications, il a plus d'applications tierces que Symbian, BlackBerry mais moins d'applications tierces que Palm OS. Il utilise la même collection de l'APIs avec Windows, il est facile de développer une application pour le développeur de Microsoft, d'exécuter pour les « applications de Microsoft »



Figure 2.6 Windows mobile

mais il n'est pas bon pour les autres applications, les développeurs. Dans la nouvelle version (comme version 6), Windows Mobile a amélioré la capacité de connexion, de fonctionnement ensemble avec les autres appareils, et de compatibilité avec Office. Il a intégré plusieurs applications pour communication comme Windows Live, hotMail, Messenger, etc.

Maintenant, dans le monde il y a plusieurs appareils qui marchent sur Windows Mobile comme quelques modèles du Palm Treo pro, Motorola, Samsung, AT&T, T-Mobile, etc. Mais Nokia qui est le leader mondial du combiné téléphonique n'utilise pas Windows Mobile comme le SE pour les appareils.

En conclusion, grâce à la popularité de Microsoft, Windows Mobile est supporté par beaucoup de sociétés. Alors, il y a variété d'applications pour Windows Mobile, et il marche bien pour les applications de Windows. [17]

❖ iOS

L'Ios, [17] anciennement iPhone OS, est le système d'exploitation mobile développé par Apple pour l'iPhone, l'iPod touch et l'iPad. Il est dérivé de Mac OS X dont il partage les fondations (le kernel hybride XNU basé sur le micro-noyau Mach, les services Unix et Cocoa, etc.). iOS comporte quatre couches d'abstraction, similaires à celles de Mac OS X : une couche « Core OS », une couche « Core Services », une couche « Media » et une couche « Cocoa ». Le système d'exploitation

occupe moins d'un demi-gigaoctet (Go) de la capacité mémoire totale de l'appareil.



Figure 2.7 iPhone

Ce système d'exploitation n'avait aucun nom officiel avant la publication du kit de développement iPhone (SDK) le 6 mars 2008. Jusqu'à cette date, Apple se contentait de mentionner que « l'iPhone tourne sous OS X », une référence ambiguë au système

d'exploitation source d'iOS, Mac OS X. Ce n'est qu'à cette occasion que Scott Forstall présenta l'architecture interne du système d'exploitation, et dévoila alors le nom d'iPhone OS. Ce nom a été changé le 7 juin 2010 pour iOS.

Le kit de développement en question, disponible pour Mac OS X, propose les outils nécessaires à la création d'une application pouvant tourner sous iOS. Si son téléchargement et son utilisation sont gratuits, la publication de telles applications requiert d'adhérer au programme des développeurs Apple, pour la somme de 99 \$ par an.

Il n'en demeure pas moins que cette offre peut s'avérer intéressante pour bon nombre de développeurs, étant donnée la taille du marché créé par iOS.



Figure 2.8 La collections iPhone

En effet, Apple a annoncé, lors d'un événement musical le 9 septembre 2009, avoir vendu 50 millions d'iPhones et d'iPods Touch. À titre informatif, les 40 millions d'appareils sous iOS n'avaient été dépassés que trois mois plus tôt, le 8 juin 2009. De surcroît, le portail App Store, dédié à l'exposition de toutes les applications développées pour ce système d'exploitation, est souvent présenté comme un modèle économique couronné de succès : avec un catalogue de 300 000 applications, qui ont fait l'objet de 10 milliards de téléchargements, l'App Store s'est imposé en 18 mois comme une référence parmi les kiosques d'applications mobiles.

2. Android

2.2.1 Description

Android [15] est un système d'exploitation open-source pour smartphones, PDA et autres terminaux mobiles, conçu par Android, une start-up rachetée par Google en juillet 2005. Il existe d'autres types d'appareils possédant ce système d'exploitation tels que les téléviseurs et les tablettes. Afin de promouvoir ce nouveau système d'exploitation ouvert, Google a su fédérer autour de lui un consortium d'une trentaine d'entreprises : l'Open Handset Alliance (OHA) créée officiellement le 5 novembre 2007. Toutes ces entreprises interviennent, plus ou moins directement, dans le marché de la téléphonie mobile.

Le but de cette alliance est de mettre en place des normes ouvertes dans le domaine de la téléphonie mobile. Ce qui veut dire que les développeurs d'application Android pourront accéder aux fonctionnalités du cœur de téléphone via une API très fournie.

Android aura comme principaux concurrents Apple avec l'iPhone, Microsoft et son Windows Mobile et Nokia avec Symbian mais également des solutions libres telles que LIMO ou OpenMoko.

2.2.2 La naissance d'Android

Si je vous dis « Android », une grande majorité d'entre vous me diront « Google ». Vous avez raison, à l'heure actuelle, mais l'histoire commence bien avant, En effet, tout a débuté avec une société américaine du nom d'Android justement, fondée en 2003.

Celle-ci a été ensuite rachetée par Google deux ans plus tard (en 2005). L'objectif premier était de développer un système d'exploitation qui permettrait à l'utilisateur d'interagir avec ce dernier.

2.2.3 Historique d'Android [11]

Nom de code	Numéro de version	Date de sortie initiale	Niveau API
(No codename)	1.0	September 23, 2008	1
(Internally known as "Petit Four") ^[4]	1.1	February 9, 2009	2
Cupcake	1.5	April 27, 2009	3
Donut	1.6	September 15, 2009	4
Eclair	2.0–2.1	October 26, 2009	5–7
Froyo	2.2–2.2.3	May 20, 2010	8
Gingerbread	2.3–2.3.7	Décembres 6, 2010	9–10
Honeycomb	3.0–3.2.6	February 22, 2011	11–13
Ice Cream Sandwich	4.0–4.0.4	October 18, 2011	14–15

Jelly Bean	4.1–4.3.1	July 9, 2012	16–18
KitKat	4.4–4.4.4	October 31, 2013	19
Lollipop	5.0–5.1.1	November 12, 2014	21–22
Marshmallow^f	6.0–6.0.1	October 5, 2015	23
Nougat	7.0–7.1.2	August 22, 2016	24–25

Tableau 2.1 : Historique d'Android

2.2.4 Les atouts qui font le succès du système :

- **Le système est évolutif**

il est donc facilement portable d'un appareil à un autre : sur nos smartphones, nos tablettes mais aussi sur d'autres appareils électroniques du quotidien et même nos véhicules.

Par ailleurs, son fonctionnement même assure la possibilité de combiner des fonctionnalités, la preuve en est avec les services Google, ou la combinaison de l'appareil photo avec la géolocalisation par exemple, pour définir des lieux associés à vos clichés

- **Le développement est accessible**

C'est une autre conséquence du système open-source, la facilité de développement. Plusieurs APIs sont fournies en vue d'accélérer le développement. [12]

2.2.5 Android face à la concurrence :

Pour la première fois, elles se sont mises d'accord, et de cet accord est né l'Open Handset Alliance, au mois de novembre de la même année(2007). Concrètement, elle regroupait pas moins de 35 entreprises dont Google qui avaient suggéré le développement d'un système d'exploitation open-source, pour révolutionner le marché du mobile en proposant quelque chose

de nouveau, et balayer la concurrence (Windows Mobile à l'époque et iOS surtout).

À l'heure actuelle, le projet Android est un grand succès avec près de 84,7% de parts de marché sur le secteur des nouveaux smartphones vendus .[13]

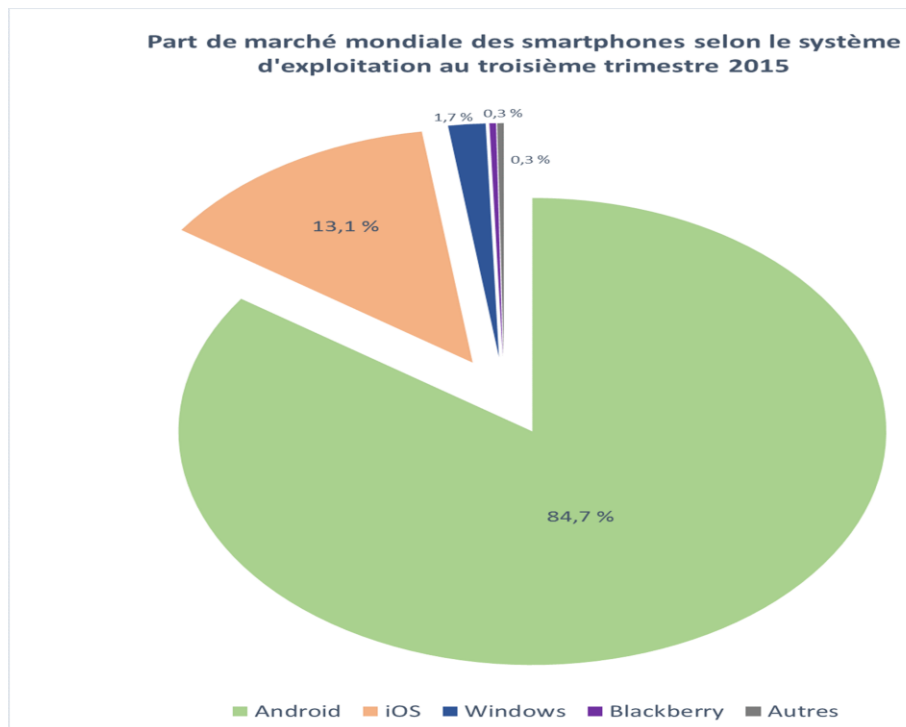


Figure 2.10 concurrence d'Android [13]

2.2.6 Architecture Android :

le diagramme suivant illustre les composants principaux du système d'exploitation Android. Chaque section sera décrite dans ce qui suit :

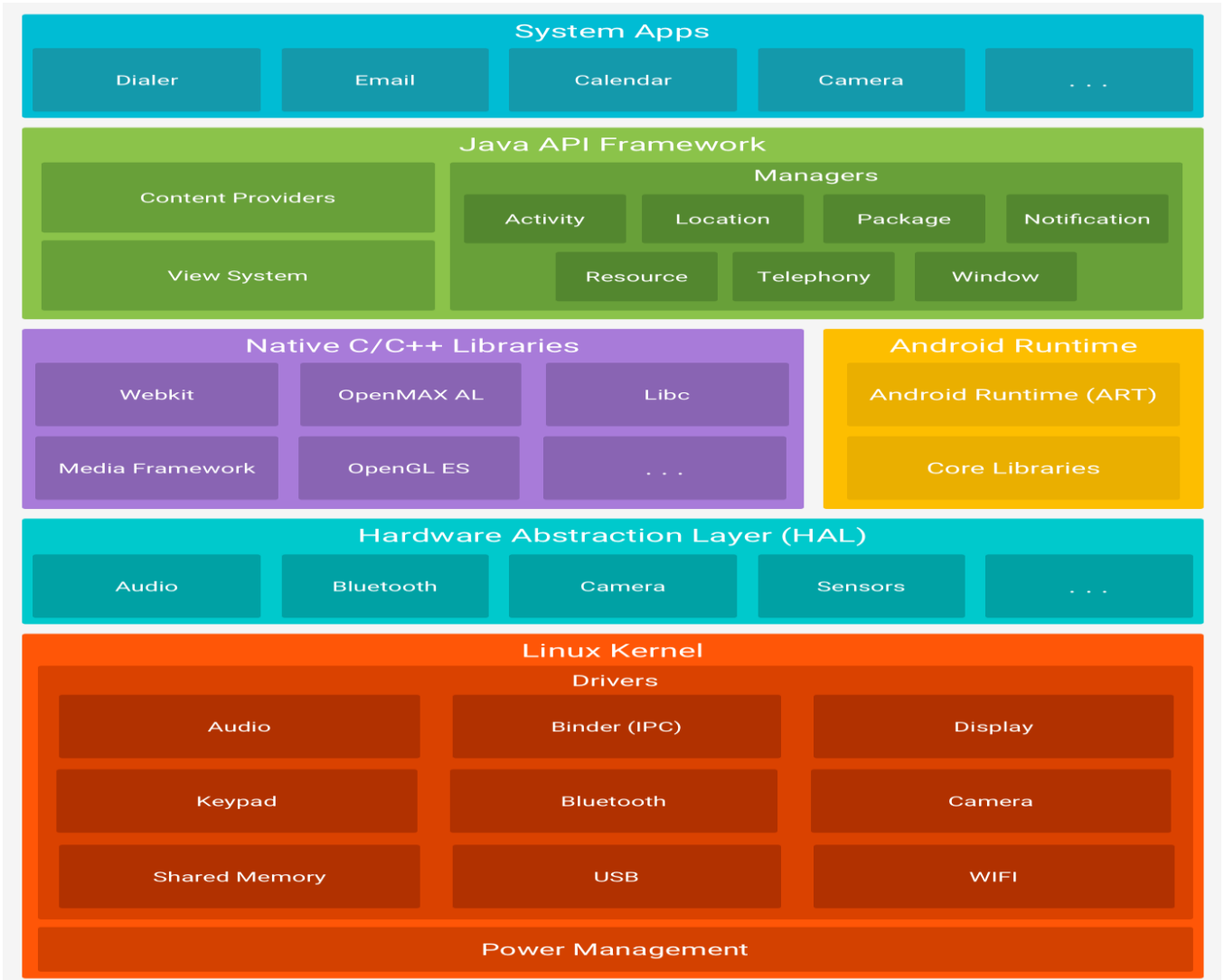


Figure 2.11 Architecture Android

Au-dessus de cette couche, on retrouve les librairies C/C++ utilisées par un certain nombre de composants du système Android.

Au-dessus des librairies, on retrouve l'Android Runtime. Cette couche contient les librairies cœurs du Framework ainsi que la machine virtuelle exécutant les applications.

Au-dessus de la couche "Android Runtime" et des bibliothèques cœurs, on retrouve le Framework permettant au développeur de créer des applications. Enfin au-dessus du Framework, il y a les applications.

2.6.1 Applications



Figure 2.12 Applications

Android est fourni avec un ensemble d'applications dont un client email, une application SMS, un calendrier, un service de cartographie, un navigateur... toutes écrites en JAVA.

Framework de développement :

En fournissant une plateforme de développement ouverte, Android offre aux développeurs la possibilité de créer des applications extrêmement riches et innovantes. Les développeurs sont libres de profiter du matériel périphérique et informations sur la localisation d'accès, exécuter des services d'arrière-plan, définir des alarmes, ajouter des notifications à la barre d'état, etc.



Figure 2.13 Framework de développement[14]

Les développeurs ont un accès complet au même framework API utilisé par les applications de base. L'architecture d'application est conçue pour simplifier la réutilisation des composants, n'importe quelle application peut publier ses capacités et n'importe quelle

autre application peut alors faire usage de ces capacités (soumis à des contraintes de sécurité appliquées par le framework). Ce même mécanisme permet aux composants d’être remplacés par l’utilisateur.

Toutes les applications sous-jacentes forment un ensemble de services et de systèmes, y compris:

- Un jeu extensible de vues qui peuvent être utilisées pour construire une application.
- Des fournisseurs de contenu qui permettent aux applications d’accéder aux données
- d’autres applications (telles que les Contacts), ou de partager leurs propres données
- Un gestionnaire de ressources.
- Un gestionnaire de notification qui permet à toutes les demandes d’afficher des alertes personnalisées dans la barre d’état.
- Un gestionnaire d’activité qui gère le cycle de vie des applications et propose une navigation commune. [14][15]

-Bibliothèques :

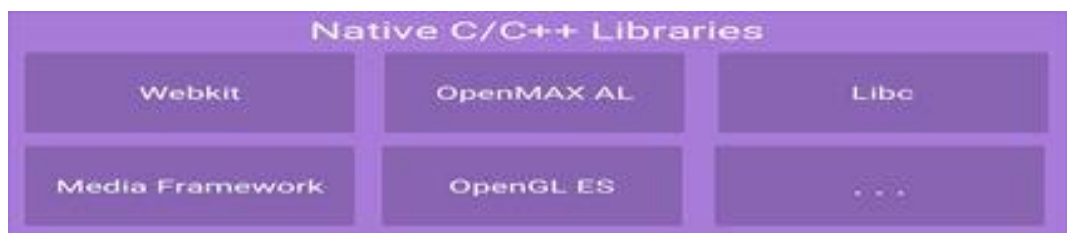


Figure 2.14 Bibliothèques [14]

Android dispose d’un ensemble de bibliothèques C / C++ utilisées par les différents composants du système Android. Elles sont offertes aux développeurs à travers le framework Android. En voici quelques-unes:

Système de bibliothèque C – une mise en œuvre dérivée de BSD de la bibliothèque C standard du système (libc), destinés aux systèmes embarqués basés sur Linux.

Comme cela a été dit précédemment, Android ne supporte pas la glibc, donc les ingénieurs d'Android ont développé une librairie C (libc) nommé Bionic libc . Elle est optimisée pour les appareils mobiles et a été développée spécialement pour Android.

Les ingénieurs d'Android ont décidé de développer une libc propre à la plateforme Android car ils avaient besoin d'une libc légère (la libc sera chargée dans chaque processus) et rapide (les appareils mobiles ne disposent pas de CPU puissant).

La Bionic libc a été écrit pour supporter les CPU ARM, bien que le support x86 soit présent. Il n'y pas de support pour les autres architectures CPU telles que PowerPC ou MIPS. Néanmoins, pour le marché des appareils mobiles, seulement l'architecture ARM est importante.

Cette libc est sous licence BSD. Elle reprend une grande partie du code des glibc issue d'OpenBSD, FreeBSD et NetBSD.

Ces caractéristiques importantes :

- Elle pèse environ 200Ko, soit la moitié de la glibc
- L'implémentation des pthreads (POSIX thread) a été complètement réécrite pour supporter les threads de la machine virtuelle Dalvik. De ce fait, la Bionic libc ne supporte pas les threads POSIX
- Les exceptions C++ et les "wide char" ne sont pas supportés
- Médiathèques – basée sur PacketVideo de OpenCore; les librairies permettant la lecture et l'enregistrement audio et vidéo, ainsi que la gestion des fichiers image, y compris MPEG4, H.264, MP3, AAC, AMR, JPG et PNG.
- SGL – le moteur graphique 2D.
- Bibliothèques 3D – une implémentation basée sur OpenGL ES 1.0 API; les bibliothèques utilisent l'accélération 3D matérielle (si disponible).
- FreeType – bitmap et vectoriel de rendu de police.
- SQLite – un moteur de base de données relationnelles puissant et léger, disponible pour toutes les applications. [14][15]

-Android Runtime :**Figure 2.15** Android Runtime [14]

Android inclut un ensemble de bibliothèques de base offrant la plupart des fonctionnalités disponibles dans les bibliothèques de base du langage de programmation Java.

Chaque application Android s'exécute dans son propre processus, avec sa propre instance de la

machine virtuelle Dalvik. Dalvik a été écrit pour que le dispositif puisse faire tourner plusieurs machines virtuelles de manière efficace. La machine virtuelle Dalvik exécute des fichiers dans l'exécutable Dalvik (. DEX), un format optimisé pour ne pas encombrer la mémoire. La machine virtuelle est la base de registres et fonctionne grâce aux classes compilées par un compilateur Java et transformées dans le format DEX.

La machine virtuelle Dalvik s'appuie sur le noyau Linux pour les fonctionnalités de base telles que le filetage et la gestion de la mémoire de bas niveau. [14][15]

-Linux Kernel**Figure 2.16** Linux Kernel [14]

Android est basé sur un kernel linux 2.6 mais ce n'est pas linux. Il ne possède pas de système de fenêtrage natif (X window system). La glibc n'étant pas supportée, Android utilise une libc customisée appelée Bionic libc.

Enfin, Android utilise un kernel avec différents patches pour la gestion de l'alimentation, le partage mémoire, etc. permettant une meilleure gestion de ces caractéristiques pour les appareils mobiles.

Android n'est pas linux mais il est basé sur un kernel linux. Pourquoi sur un kernel linux ?

Le kernel linux a un système de gestion mémoire et de processus reconnu pour sa stabilité et ses performances. Le model de sécurité utilisé par linux, basé sur un système de permission, est connu pour être robuste et performant. Il n'a pas changé depuis les années 70

Le kernel linux fournit un système de driver permettant une abstraction avec le matériel. Il permet également le partage de bibliothèques entre différents processus, le chargement et le déchargement de modules à chaud. [14][15]

3. Wifi Direct :

Introduction :

Wi-Fi direct est une nouvelle technologie définie par l'alliance Wi-Fi visant à améliorer la communication directe entre périphériques et périphériques sans nécessiter de point d'accès sans fil.

Wi-Fi directement s'appuie sur le mode d'infrastructure IEEE 802.11 réussi et permet aux périphériques de négocier qui prendront en charge les fonctionnalités AP-like.

La connectivité directe du périphérique à l'appareil était déjà possible dans la norme IEEE 802.11 d'origine, mais elle présente plusieurs inconvénients tels que le manque d'économie d'énergie ou les fonctionnalités QoS étendues.

L'appareil Wi-Fi à l'espace de communication de l'appareil est 802.11z, également connu sous le nom de Tunneled Direct Link Setup (TDLS), qui permet une communication directe avec le périphérique, mais nécessite que la station soit associée au même AP [17].

2.3.1 Présentation technique :

Dans un réseau Wi-Fi typique, le client scanne et associe aux réseaux sans fil disponibles, créés et annoncés par les points d'accès (AP).

Wi-Fi Direct est que ces rôles sont spécifiés comme étant dynamiques et, par conséquent, un périphérique Wi-Fi Direct doit implémenter à la fois le rôle d'un client et le rôle d'un AP. [18]

Ces rôles sont donc des rôles logiques qui pourraient même être exécutés simultanément par le même dispositif, ce type d'opération s'appelle mode Concurrent.

2.3.2 Architecture :

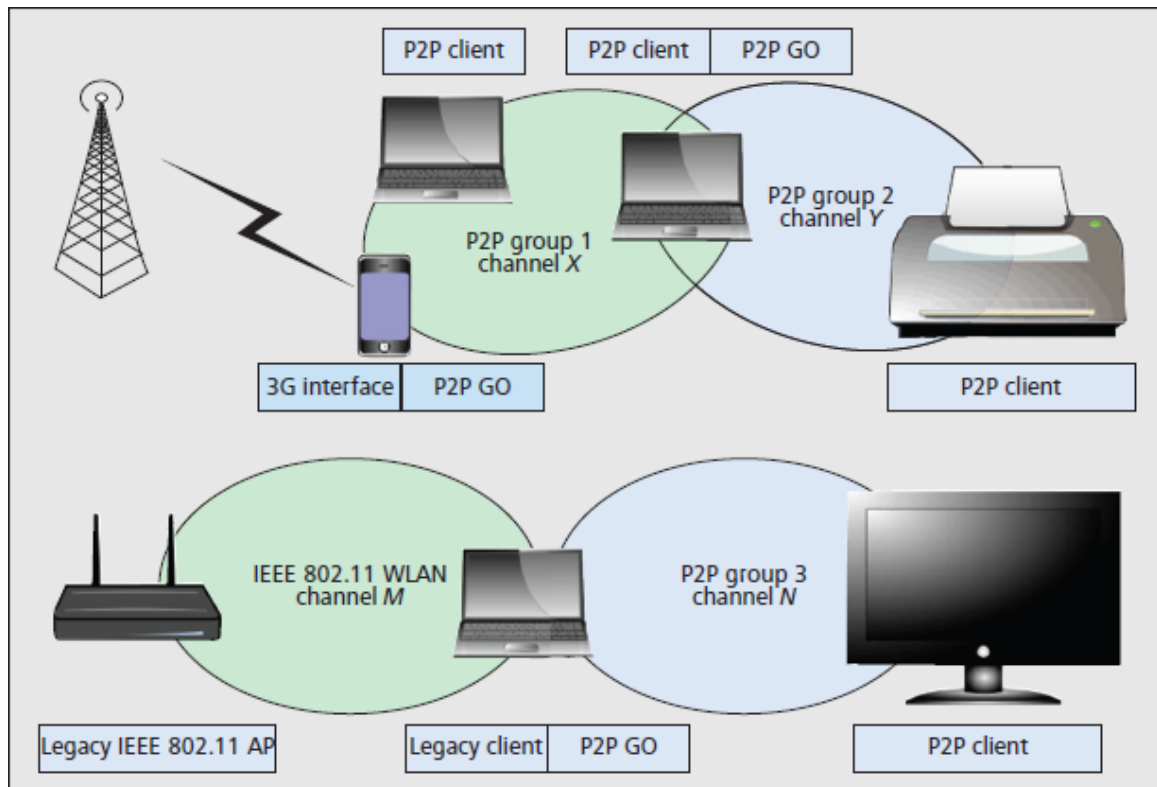
Le périphérique Wi-Fi direct communique en établissant un groupe P2P.

Le périphérique implémentant une fonctionnalité AP dans le groupe P2P est appelé le propriétaire du groupe P2P (P2P GO), et l'appareil agissant en tant que client est connu sous le nom de clients P2P.

Une fois que le groupe P2P est établi, d'autres clients P2P peuvent rejoindre le groupe comme dans un réseau Wi-Fi traditionnel. [18]

Lorsque l'appareil agit à la fois comme client P2P et comme P2P GO, l'appareil alternera généralement entre les deux rôles en partageant le temps par l'interface Wi-Fi. (Exemple: ordinateur portable 2 dans la figure supérieure)

Comme un AP traditionnel, un P2P GO s'annonce à travers des balises et doit supporter l'économie d'énergie pour ses clients associés.



Wi-Fi direct supported topologies and use cases.

Figure 2.17 Architecture de Wifi-Direct

Seul le P2P GO est autorisé à connecter de manière croisée les périphériques de son groupe P2P à un réseau externe (Exemple: mobile dans la figure supérieure)

Cette connexion doit être effectuée au niveau du réseau, généralement implémentée à l'aide de la traduction d'adresses réseau (NAT).

Wi-Fi direct n'autorise pas le transfert du rôle de P2P GO dans le groupe.

Si P2P GO quitte le groupe P2P, le groupe est décomposé et doit être rétabli.

2.3.3 Sécurité

Les périphériques Wi-Fi Direct sont nécessaires pour implémenter Wi-Fi Protected Setup (WPS) pour prendre en charge une connexion sécurisée avec une intervention minimale de l'utilisateur.

WPS permet d'établir une connexion sécurisée en introduisant un code PIN dans le client P2P ou en appuyant sur un bouton dans les deux périphériques P2P.

À la suite de la terminologie WPS, le P2P GO est requis pour implémenter un registre interne et le client P2P est requis pour implémenter un inscrit.

Le fonctionnement de WPS se compose de deux parties. Dans la première partie, le registre interne est chargé de générer et d'émettre les informations d'identification du réseau, c'est-à-dire les clés de sécurité, à l'Enrolle

Dans la deuxième partie, l'Enrollee (P2P Client) se désassocie et se reconnecte en utilisant ses nouvelles références d'authentification. [19]

2.3.5 Avantages :

Certains des avantages que les consommateurs verront à partir des appareils Wi-Fi Direct.

Mobilité et portabilité: les périphériques Wi-Fi Direct-certified se connectent à tout moment, n'importe où.

Utilitaire immédiat: les utilisateurs ont la possibilité de créer des connexions directes avec le tout premier périphérique certifié Wi-Fi Direct qu'ils apportent à la maison. Par exemple, un nouvel ordinateur portable certifié pour Wi-Fi Direct peut créer des connexions directes avec les périphériques Wi-Fi hérités existants dans la maison de l'utilisateur.

Facilité d'utilisation: les périphériques Wi-Fi Direct disposent de fonctionnalités permettant aux utilisateurs d'identifier les périphériques et les services disponibles avant d'établir une connexion.

Connexions sécurisées simples: Wi-Fi Protected Setup simplifie la création de connexions sécurisées entre périphériques. Dans la plupart des cas, les utilisateurs pourront se connecter en appuyant sur un bouton. [19]

Conclusion

Dans ce chapitre j'ai fait une étude globale sur les systèmes d'exploitation mobile avec un bref historique sur le système Android et leur fonctionnalité et dans la dernière partie j'ai présenté le Wifi direct (son définition, présentation technique, architecture, sécurité, et avantages).

CHAPITRE 3

CONCEPTION DU SYSTEME

Introduction

Ce chapitre est consacré à l'étude conceptuelle de notre projet. Nous allons d'abord énumérer les objectifs visés par notre projet, décrire en détail son fonctionnement et présenter les différents diagrammes permettant sa modélisation. Nous avons choisis de modéliser avec le formalisme UML (Unified Modeling Language) qui offre une flexibilité marquante qui s'exprime par l'utilisation des diagrammes.

3.1 ETUDE DE L'EXISTANT

Eléments de l'existant : Share-it

SHAREit est une application gratuite pour transférer des fichiers d'un périphérique compatible Wi-Fi à un autre similaire. Les utilisateurs peuvent l'utiliser pour transférer des fichiers, y compris des photos, des vidéos, de la musique, des contacts, des applications et tout autre fichier. C'est une application gratuite de Lenovo qui permet aux périphériques Windows, Windows Phone, Android et iOS de transférer des fichiers directement par des connexions Wi-Fi ad hoc. Il est actuellement disponible en 39 langues, dont l'anglais, le français, l'espagnol, le russe, l'arabe et le chinois. [20] [21]

Propriétés de l'application:

- ❖ L'application offre une vitesse plus rapide que Bluetooth et même NFC.
- ❖ Totalement sécurisée.
- ❖ la seule meilleure méthode pour le transfert de fichiers était soit par Bluetooth, soit par USB.

Eléments de l'existant : P2P Vidéo Call

Application Android qui permet faire des appels vidéo avec le wifi direct P2P

- ❖ Appel vidéo par Wi-Fi Direct (Wi-Fi P2P) à moins de 100 m
- ❖ 100% d'appel vidéo privé, la communication entre pairs ne se dirige pas via un serveur quelconque

- ❖ Aucun besoin d'Internet
- ❖ Aucune inscription de compte requise, capable de faire des appels vidéo vers des personnes à proximité
- ❖ Appel vidéo gratuit, 100% sûr d'utiliser Wi-Fi Direct / Bluetooth, sans frais de données.

3.2 Critique les applications précèdent :

- ❖ Share-It n'a pas la fonction de chat, soit des messages texte ou vidéo.
- ❖ P2P Vidéo Call n'a pas la fonction pour partager des fichiers et envoyer des messages.

Solution :

3.3 Présentation du projet

Le projet EASY-IT, consiste à développer une application intégrée dans un Smartphone tactile de type Android, qui permet à faire une conversation multimédia, partage des fichiers, envoie des messages instantané, et faire des appels vidéo entre deux Smartphone, qui utilise le Wifi direct (P2P).

3.4 Domaine couvert par le projet

Le projet s'insère dans le domaine des technologies de l'information et de la communication. Les Smartphones possèdent une fonctionnalité intéressante, qui est la possibilité faciliter les affaires d'études à l'université entre les étudiants et leur enseignants.

• Il peut toucher aux :

- Les étudiants
- Les enseignants.
- Les directeurs des bureaux

3.5 Les limites de l'application :

- ✓ Envoie des messages courts ou partager des fichiers par les enseignements au étudiants

- ✓ D'un système de sécurité .. de sorte que grâce à cette application, qui fonctionne de la même manière que des caméras de surveillance.
- ✓ Permettre aux étudiants de poursuivre un forum scientifique de près quand qu'il y aurait surpopulation dans les tribunes.

3.6 Architecture fonctionnelle :

Elles apparaîtront dans des fenêtres différentes et partageront toutes, la même interface générale de l'application.

- **Discover :**

Elle sera présentée comme une bouton, Lorsque l'utilisateur appuie sur le bouton pour détecter le matériel qui l'entoure.

- **ChatRoom :**

Fenêtre où la conversation se déroulant

- **Bouton d'émetteur :**

Cette fonctionnalité Lorsque vous avez fini d'écrire le message, appuyez sur le bouton de l'émetteur pour envoyer le message

- **Bouton d'Envoyer les fichiers:**

Choisissez tout type de fichiers à envoyer au destinataire.

UML

3.7 Définition

L'UML, acronyme pour (Unified Modeling Language) qu'on peut traduire par Langage de Modélisation Objet Unifié, est un outil universel qui permet de structurer un projet et de le matérialiser graphiquement sous forme de diagrammes compréhensibles par les non informaticiens. Aucune connaissance de langage informatique n'est prérequis. Cette modélisation permet dans un second temps de développer le code informatique, le plus souvent à l'aide d'un langage orienté objet (C++, Java par exemple). La description de projets en UML est une étape nécessaire qui permet de gagner beaucoup de temps dans le développement d'une application car la mise au point du code en est moins fastidieuse et le risque d'erreurs de conception ou de réalisation est plus limité. Bien que conçue pour la gestion de projets de grande envergure, l'utilisation de cette méthodologie est bénéfique même pour les projets les plus modestes. [22]

3.8 L'Histoire d'UML

La modélisation d'un projet correspond à sa mise en forme avant codage. C'est l'une des premières étapes d'un projet pris globalement global ou pour chaque itération du cycle de vie [Analyse > Conception > Implémentation > Testes]. Elle joue le même rôle que les maquettes d'architecture et répond aux mêmes besoins :

En 1994, Jim Rumbaugh (OMT : Object Modeling Technique) et GradyBooch (BOOCH) décidèrent de proposer une méthode unique en unifiant leurs travaux, ce qui donna naissance à la méthode unifiée (Unified method).

Ainsi, les deux variantes des méthodes BOOCH et OMT appelées respectivement Booch'93 et OMT-2, se sont rapprochées de façon considérable à tel point que les variations subsistantes sont devenues minimales.

Une année plus tard, ils ont été accompagnés par Ivar Jacobson, le créateur des cas d'utilisation (use cases) pour créer l'UML. La méthode unifiée se transforma donc en UML (Unified Modeling Language For Object Oriented Development).

La version 1.1 soumise à l'OMG (object management group) en septembre 1997 fut acceptée à l'unanimité et devint de ce fait un standard. Des transformations continues furent effectuées pour supprimer les incohérences, apporter des améliorations et ajouter de nouveaux concepts.

En effet, la version 1.2 a apporté un remaniement de la forme par rapport à la version 1.1 mais aucun ajout d'ordre technique ne fut introduit.

La version 1.3 a apporté de nombreux changements, que ce soit en terme de correction ou d'ajout. Ainsi, il y a eu modification des associations entre cas d'utilisation, simplification des stéréotypes, changement d'éléments des graphes d'activités et des automates d'états...

La version 2.0 a apporté des améliorations tout en ajoutant le dixième diagramme appelé diagramme d'architecture et l'ajout de la notion de profils. [22]

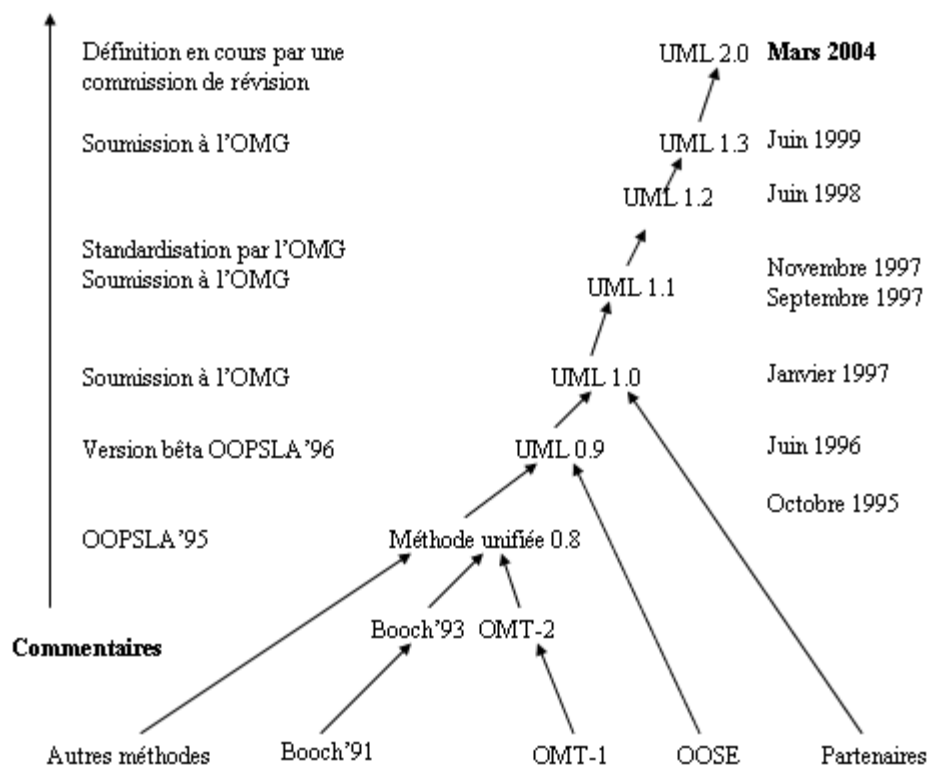


Figure 3.1 Historique d'UML.

3.9 les diagrammes

Les 13 diagrammes UML sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie.

Diagrammes structurels ou statiques

- Diagramme de classes
- Diagramme d'objets
- Diagramme de composants
- Diagramme de déploiement
- Diagramme des paquetages
- Diagramme de structure composite

Diagrammes comportementaux

- Diagramme des cas d'utilisation
- Diagramme états-transitions
- Diagramme d'activité

Diagrammes d'interaction ou dynamiques

- Diagramme de séquence
- Diagramme de communication
- Diagramme global d'interaction
- Diagramme de temps [23]

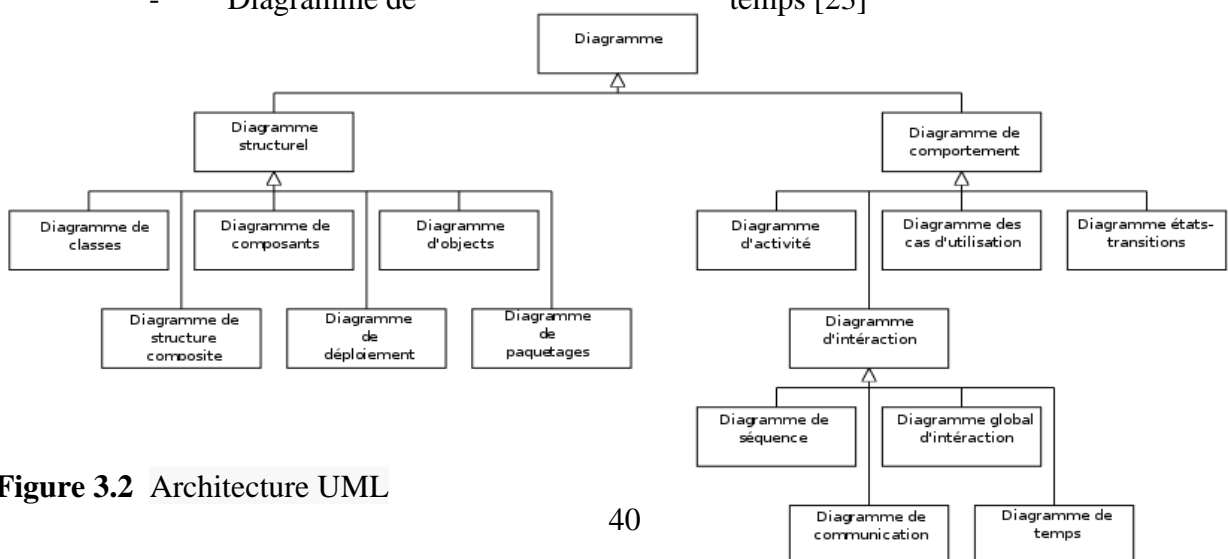


Figure 3.2 Architecture UML

3.10 Les diagrammes de l'Application :

- Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. C'est le premier diagramme du modèle UML, celui où s'assure la relation entre l'utilisateur et les objets que le système met en œuvre. [22]

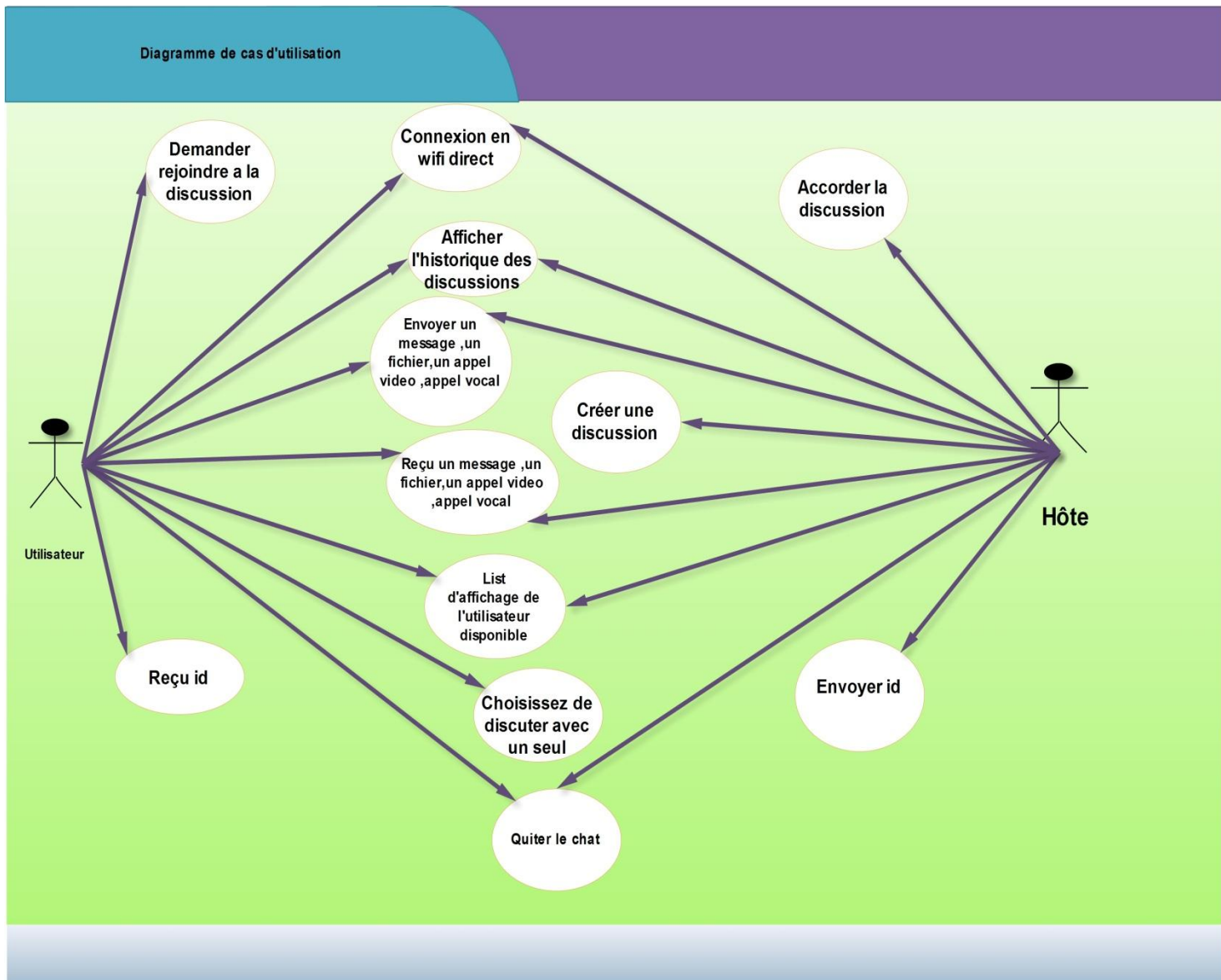


Figure 3.3 Le diagramme de cas d'utilisation

- Diagramme de séquence

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre. On peut représenter les mêmes opérations par un diagramme de communication graphe dont les nœuds sont des objets et les arcs (numérotés selon la chronologie) les échanges entre objets. En fait, diagramme de séquence et diagramme de communication sont deux vues différentes mais logiquement équivalentes (on peut construire l'une à partir de l'autre) d'une même chronologie. Ce sont des diagrammes d'interaction. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.[22]

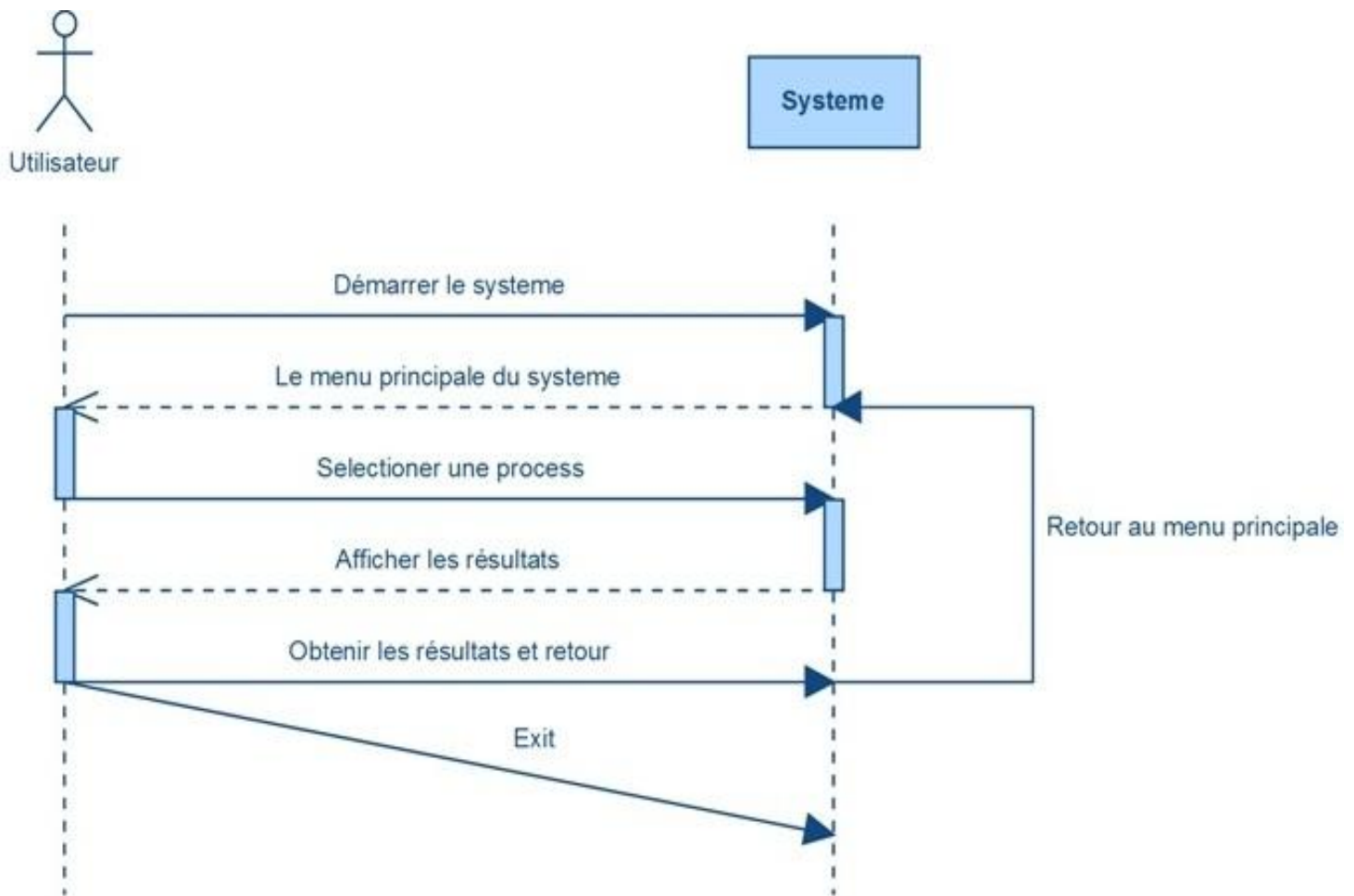


Figure 3.4 Le diagramme de Séquence de l'application

Description de diagramme de séquence de L'application:

Titre :		L'application Easy-it
Acteurs :		Utilisateur
Description générale		L'application Easy-it, contient Cinq boutons,
Description détaillée :		
Pré-condition (événement déclencheur)		La fenêtre courante est la page d'accueil de l'application Easy-it
Actions :		
N°	Activité acteur	Réponse système
1	Cliquez sur l'image ci-dessus pour accéder au réglage et à la connexion de votre appareil à d'autres par wifi- direct	Le réglage des réseaux wifi est apparu pour choisissez dans le wifi direct pour pour connecter avec le hôte .
2	Appuie sur le champ «Choisissez votre nom de chat»	Le champ Prêt à remplir
3	Appuie sur le bouton Démarrer le client de discussion	Le champ de conversation va affichera. et tapez votre message ici.
4	Appuie sur le bouton « Envoyer des message »	Votre message à envoyer vers le récepteur.
5	Appuie sur le bouton « Envoie des fichiers »	Une nouvelle fenêtre s'ouvre, elle affiche un champ pour sélectionné Les fichiers que vous voulez envoyer
6	Appuie sur le bouton «appelle vidéo »	Fenêtre de conversation visuelle apparaît
Post-condition (état des objets et produits à la fin du processus)		La fenêtre courante est l'interface d'accueil de l'application Easy-it.

Tableau 3.1 Description de diagramme séquence

-Le diagramme de Séquence de l'envoi un message.

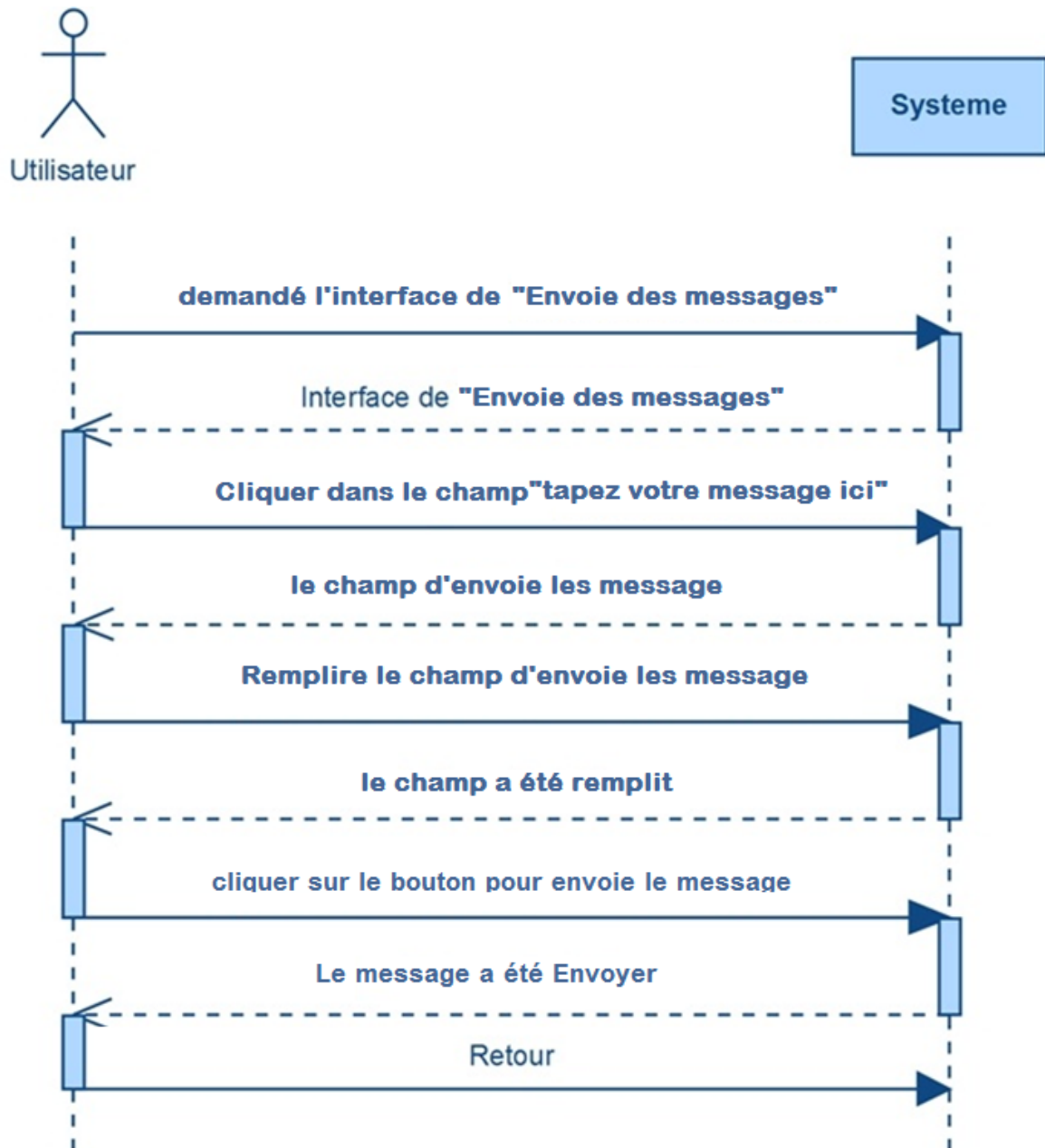


Figure 3.5 Le diagramme de Séquence de l'envoi un message.

-Le diagramme de Séquence de l'envoi un fichier (Audio, vidéo, photo, PDF)

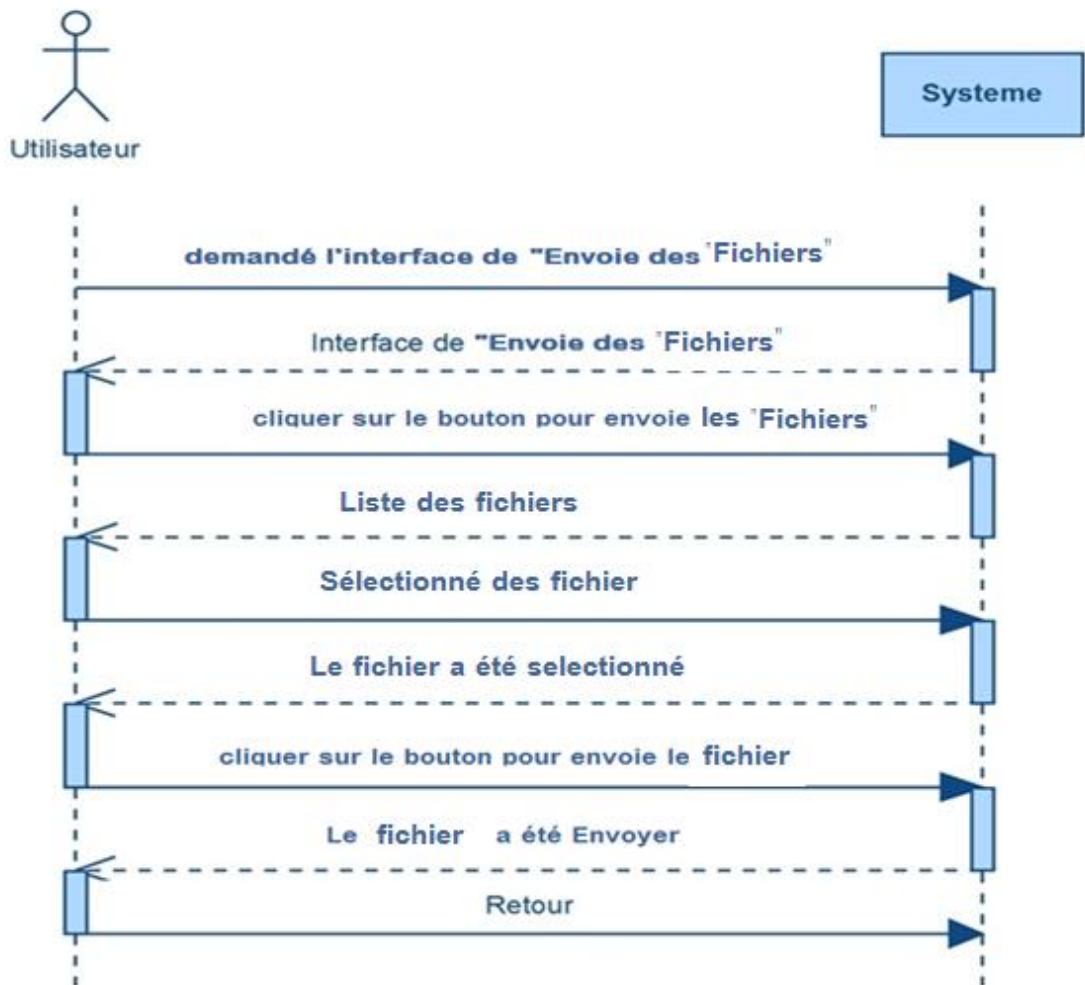
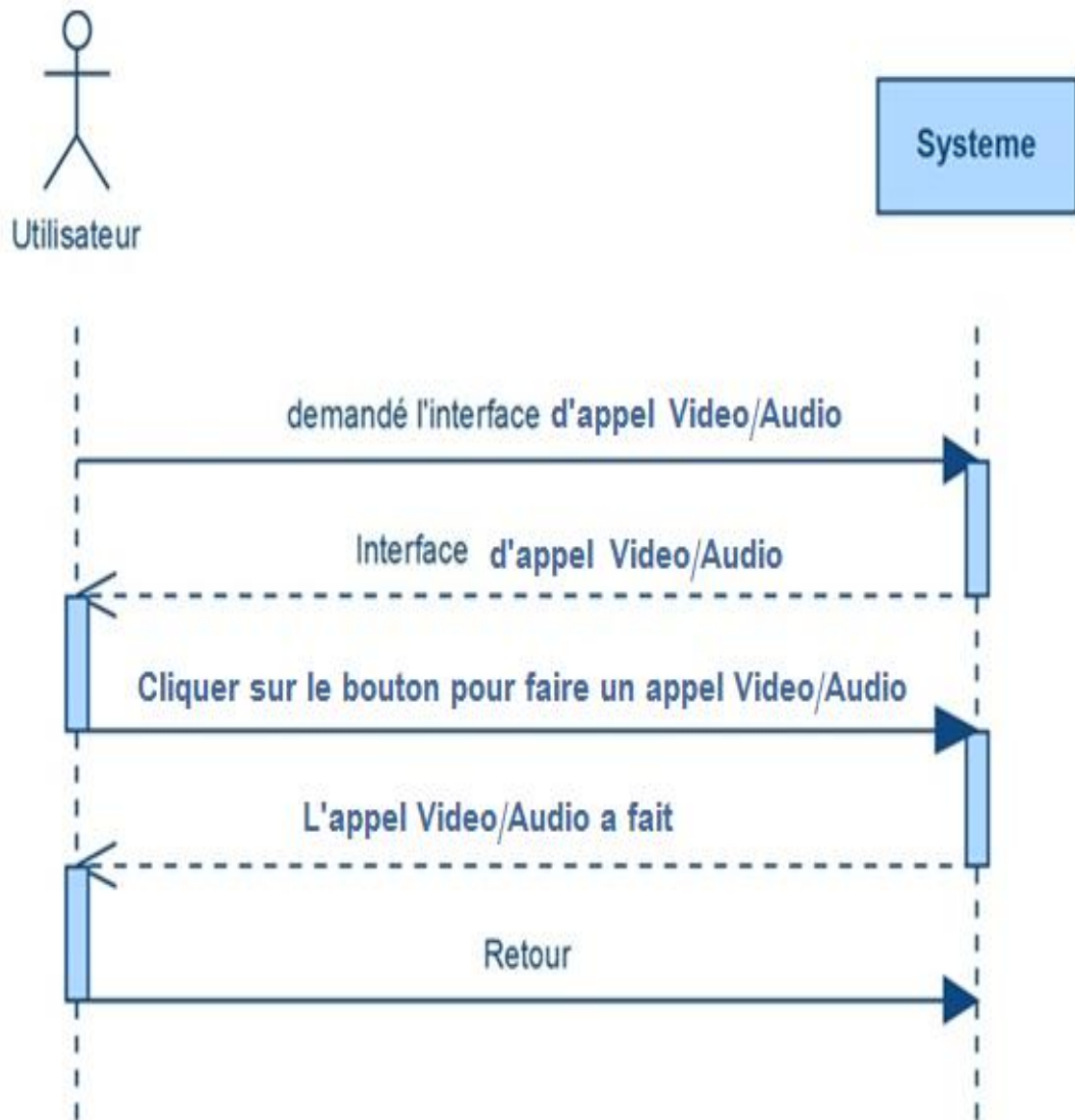


Figure 3.6 Le diagramme de Séquence de l'envoi un fichier (Audio, vidéo, photo, PDF)

Description de diagramme de séquence Envoie des messages / Envoie des fichiers :

Titre :	Envoie des messages / Envoie des fichiers	
Acteurs :	Utilisateur/ System	
Description générale	Ecrire ici les message qui vous voulait envoyer même chose pour le fichiers	
Description détaillée :		
Pré-condition (événement déclencheur)	La fenêtre courante est la page d'accueil de l'application Easy-It	
Actions :		
N°	Activité acteur	Réponse système
1	Appuie sur le bouton Démarrer le client de discussion	Le champ de conversation va affichera.
2	Remplir le champ de conversation	tapez votre message ici. Votre message à envoyer vers le récepteur.
3	Appuie sur le bouton « Envoyer des message »	Votre message a été Envoyer
4	Appuie sur le bouton « Sélectionné des fichiers»	Une nouvelle fenêtre s'ouvre, elle affiche un champ pour sélectionner Les fichiers qui vous voulez envoyer
5	Appuie sur le bouton « Envoyer des fichiers»	Votre fichier a été Envoyer

Tableau 3.2 Description de diagramme de séquence Envoi Message/Fichier

-Le diagramme de Séquence de l'appel Vidéo/Audio**Figure 3.7** Le diagramme de Séquence de l'appel Vidéo/Audio

Description de diagramme de séquence Appel Vidéo/Audio

Titre :		Appel Vidéo/Audio
Acteurs :		Utilisateur / System
Objectif du cas d'utilisation, description générale		Fait un appel audio ou une appel Audio qui utilise le Wifi-direct .
Description détaillée :		
Précondition (événement déclencheur)		La fenêtre courante est l'interface d'accueil de l'application Easy-It
Actions :		
N°	Activité acteur	Réponse système
1	Cliquez sur l'image ci-dessus pour accéder au réglage et à la connexion de votre appareil à d'autres par wifi- direct	Le réglage des réseaux wifi est apparu pour choisissez dans le wifi direct pour pour connecter avec le hôte
2	Appuie sur le bouton Démarrer le client de discussion	Le champ de conversation va affichera. .
3	Appuie sur le bouton Pour faire un appel vidéo/audio qui situé on haute	Une nouvelle fenêtre s'ouvre, elle affiche l'autre utilisateur
4	appuie sur le bouton rouge qui situé en bas	Pour Finir l'appel. apres la retour dans le champ de conversation textuel .
Post-condition (état des objets et produits à la fin du processus)		La fenêtre courante est l'interface d'accueil de l'application Easy-It.

Tableau 3.3. Description de diagramme de séquence Appel Vidéo/Audio

- Définition du diagramme de classes [22]

Le diagramme de classes est généralement considéré comme le plus important dans un développement orienté objet. Il représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens, que ceux-ci représentent un emboîtement conceptuel (héritage) ou une relation organique (agrégation)

Le diagramme met en évidence d'éventuelles relations entre ces classes.

Le diagramme de classes comporte 7 concepts :

- classe
- attribut
- méthodes
- identifiant
- relation
- opération
- généralisation / spécialisation

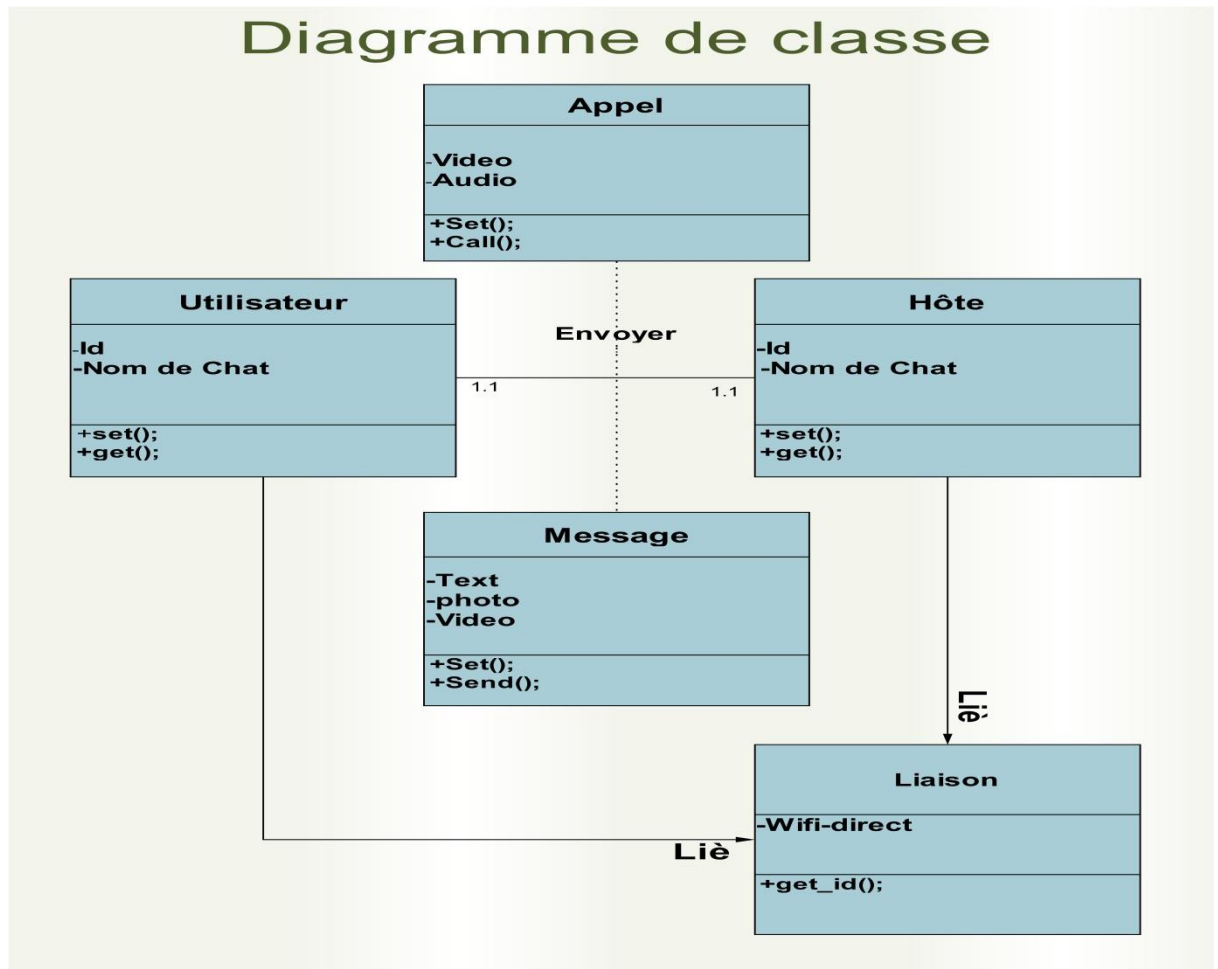


Figure 3.8 diagramme de classe

-Description de diagramme de classe

Cette figure représente le diagramme de classe de l'application, qui contient trois classes : Utilisateur, Hôte et Liaison,. En remarque que la classe utilisateur est en relation avec la classe Hôte par l'envoi d'un Message et faire des appels vidéo au audio, et avec la classe Liaison pour lie les deux classes l'hôte et l'utilisateur .

- Définition du diagramme d'activité :

Un diagramme d'activité permet de modéliser le comportement du système, dont la séquence des actions et leurs conditions d'exécution. Les actions sont les unités de base du comportement du système. Un diagramme d'activités permet de grouper et de dissocier des actions.

Conclusion:

Dans ce chapitre, nous avons exposé les différents besoins auxquels doit répondre l'application à réaliser ainsi que les diagrammes du langage UML à savoir le diagramme de cas d'utilisation, le diagramme de classe , le diagramme de séquence et le diagramme d'activité . Dans le chapitre suivant nous allons aborder l'ensemble des choix techniques et logiciels qui nous ont aidés dans la réalisation de ce projet.

CHAPITRE 4

LA REALISATION

Introduction

Dans ce chapitre consacré à la réalisation et la mise en œuvre de notre application Easy-it, je vais présenter les outils de développement adoptés; l'environnement utilisé c'est Android studio, enfin je montre les principales interfaces et fenêtres de l'application.

3.1 Les outils utilisés

3.1.1 Android studio

Android Studio est l'environnement de développement intégré officiel (IDE) pour la plateforme Android.

Il a été annoncé le 16 mai 2013 à la conférence Google I / O.

Android Studio était en phase d'aperçu d'accès précoce à partir de la version 0.1 en mai 2013, puis a entré en phase bêta à partir de la version 0.8 qui a été diffusée en juin 2014 ,la première version stable

a été publiée en décembre 2014, à partir de la version 1.0. Basé sur le logiciel IntelliJ IDEA de JetBrains, Android Studio est spécialement conçu pour le développement d'Android. Il est disponible pour téléchargement sur Windows, MacOS et Linux, et remplacé les outils de développement Android (ADT) d'Eclipse en tant que premier IDE de Google pour le développement d'applications Android natives

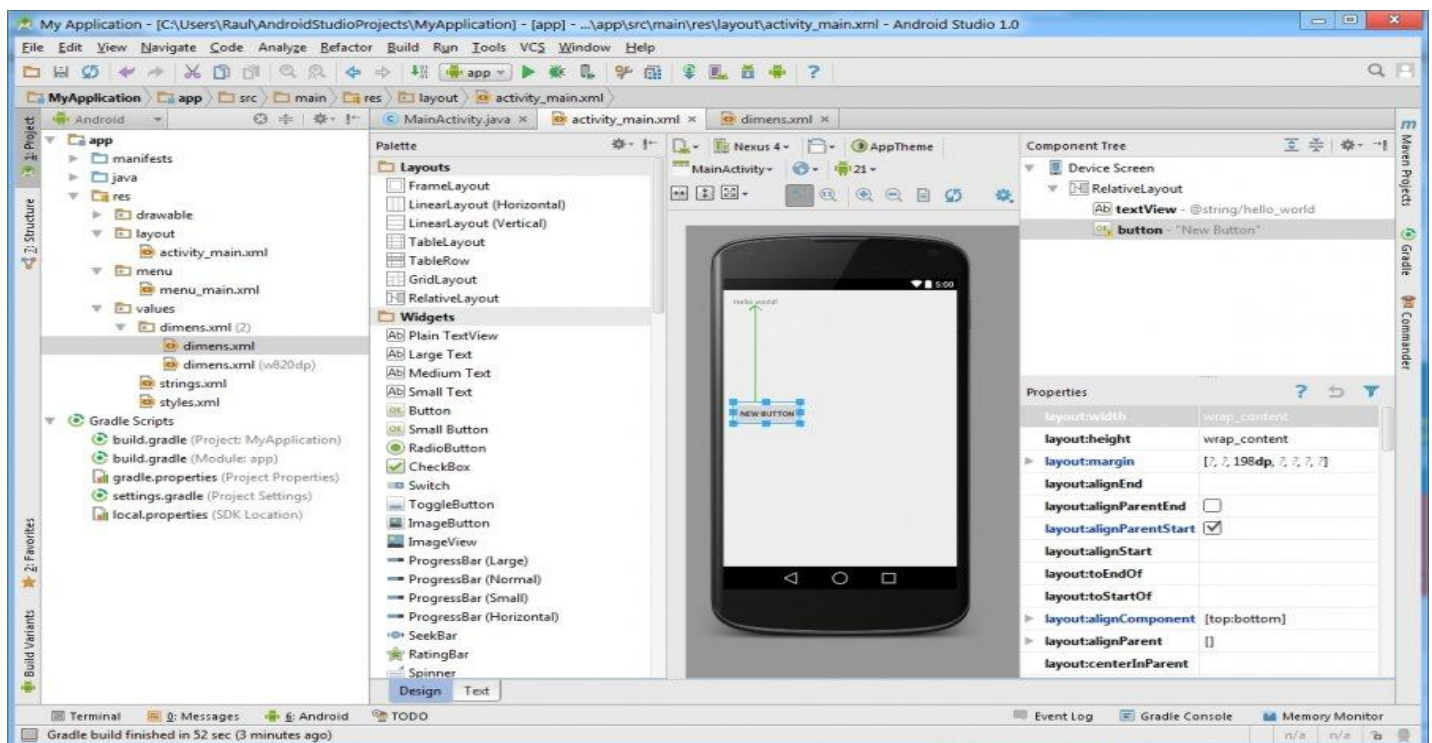


Figure 4.1 Android studio

3.1.2 Le plugin ADT

Pour développer Android, j'ai à installer le plugin Android qui rajoutera à Android studio les fonctionnalités spécialisées dans le développement sous Android.

3.1.3 Software Development Kit (SDK)

C'est un kit de développement basé sur le langage Java. Il s'agit des outils que Google fournit pour interagir avec Android.

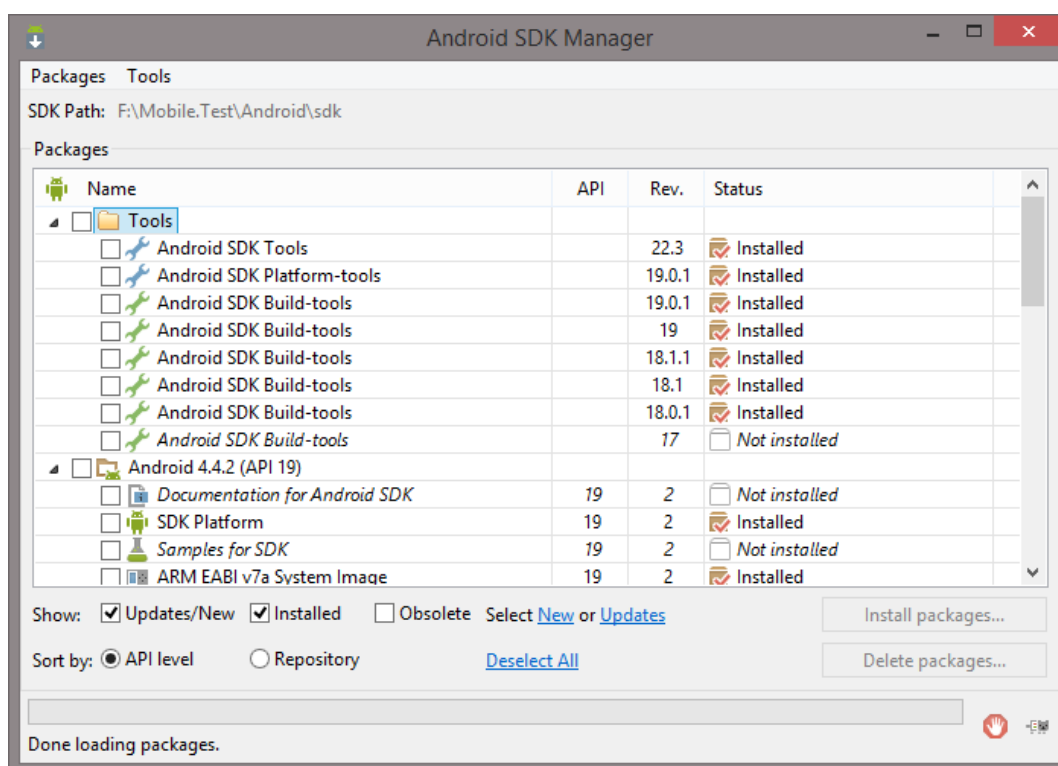


Figure 4.2 SDK

3.1.4 L'émulateur :

Le SDK Android comprend un émulateur de périphérique mobile virtuel qui fonctionne sur l'ordinateur. L'émulateur vous permet de prototyper, développer et tester des applications Android sans l'aide d'un dispositif physique.

3.1.5 Les langages de programmation

- **JAVA**

Java est le langage le plus utilisé dans le développement mobile. L'un de ses plus grands avantages est que les logiciels créés avec ce langage peuvent être facilement installés et exécutés sur différents systèmes d'exploitation, que ce soit Windows, Mac OS, Linux ou autre. Avec un petit coup de main de Google, qui vous fournit l'environnement de développement Android Studio, vous pourrez créer des applications mobiles bien complexes.

Enfin, nous rappelons l'Android studio qui base sur le Java, ce dernier étant un langage de programmation orienté objet utilisable sur divers systèmes d'exploitation, est un langage assez robuste, portable et à hautes performances.

- **XML**

Lors du développement de vos applications avec Android Studio, vous tomberez très probablement sur des bouts de code écrits en XML. Ce langage de balisage est utilisé pour gérer l'affichage des contenus sur l'écran. Il n'est pas indispensable pour créer une application Android, mais il facilite le développement en permettant de séparer l'affichage des algorithmes. Avec XML, on gagne du temps et on simplifie le code de l'application, ce qui permet d'éviter des erreurs.

3.2 Captures d'écrans :

- Première interface :



Figure 4.3 Première interface de l'application

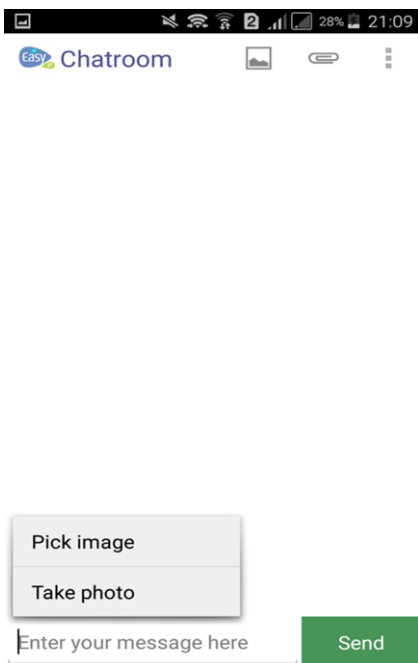


Figure 4.5 L'Envoie des Photos

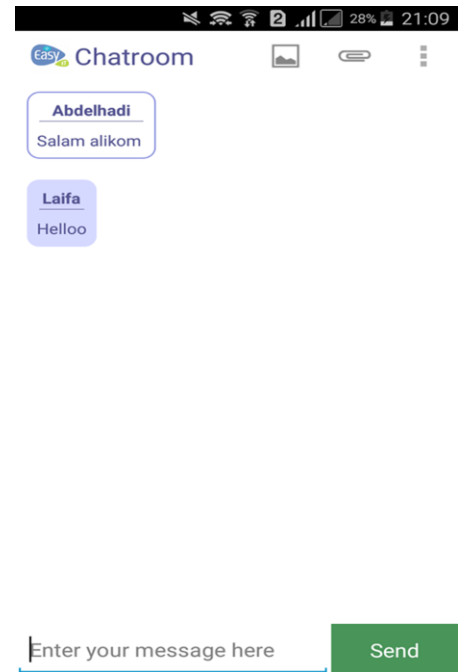


Figure 4.4 L'Envoie des messages



- **L'envoi du Message vidéo :**

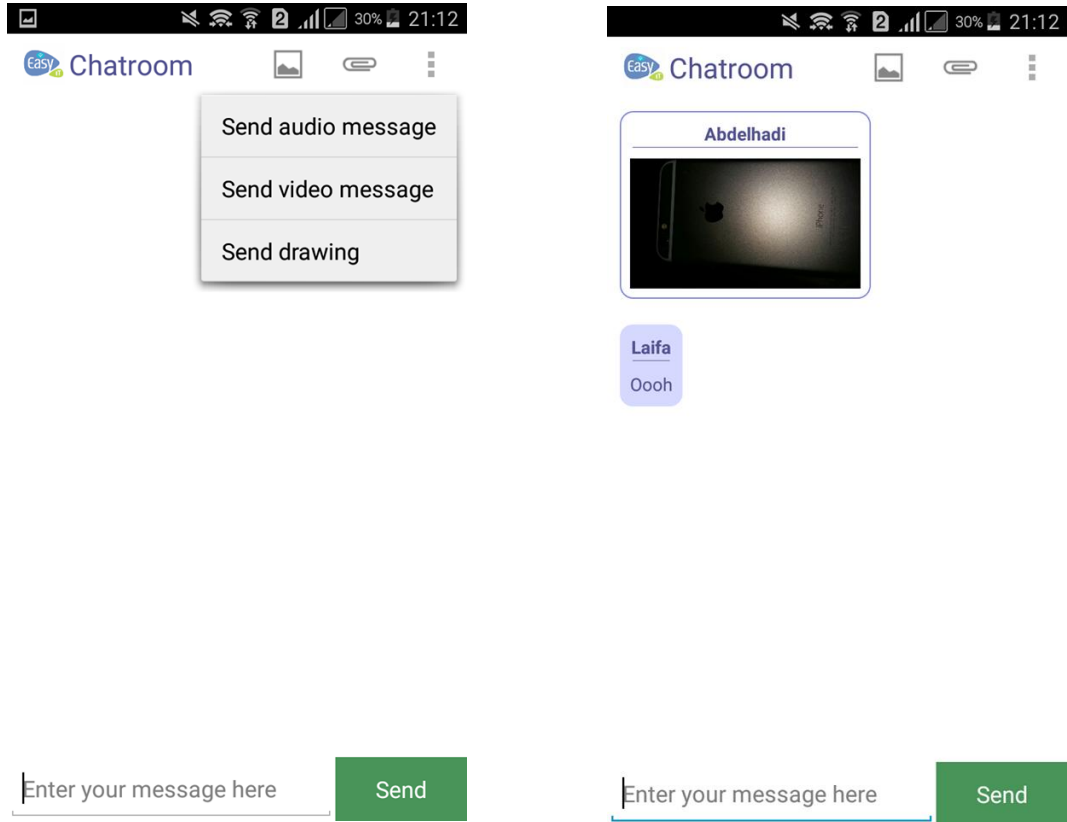


Figure 4.6 Message vidéo

- **L'envoi des messages Audio :**

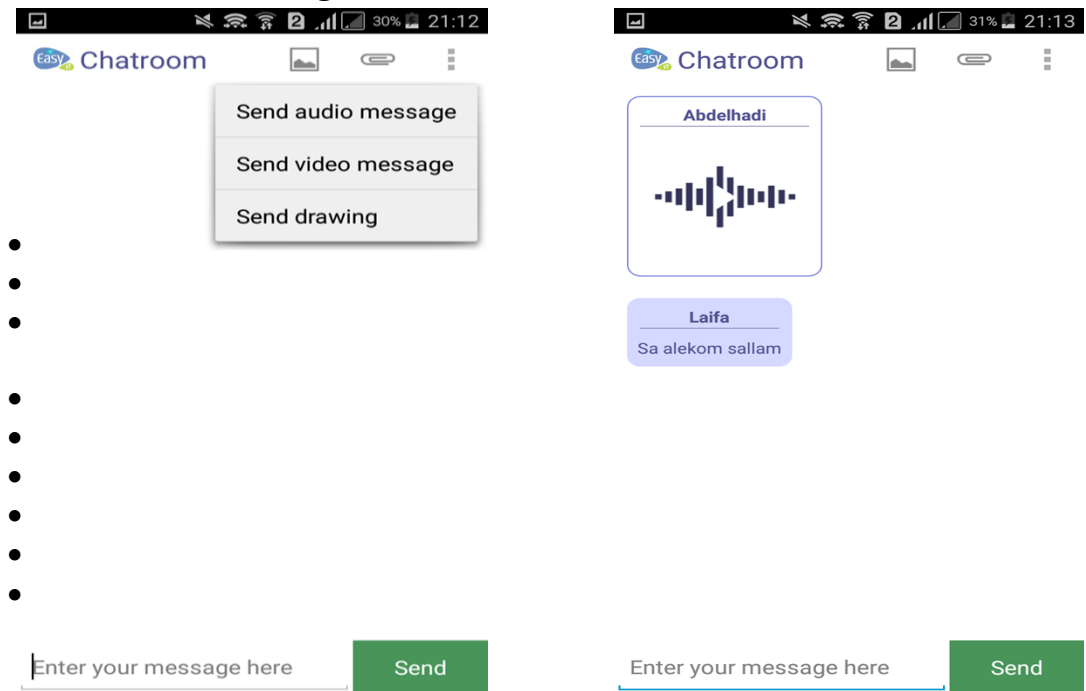


Figure 4.7 Message Audio

- **L'envoi des fichiers:**

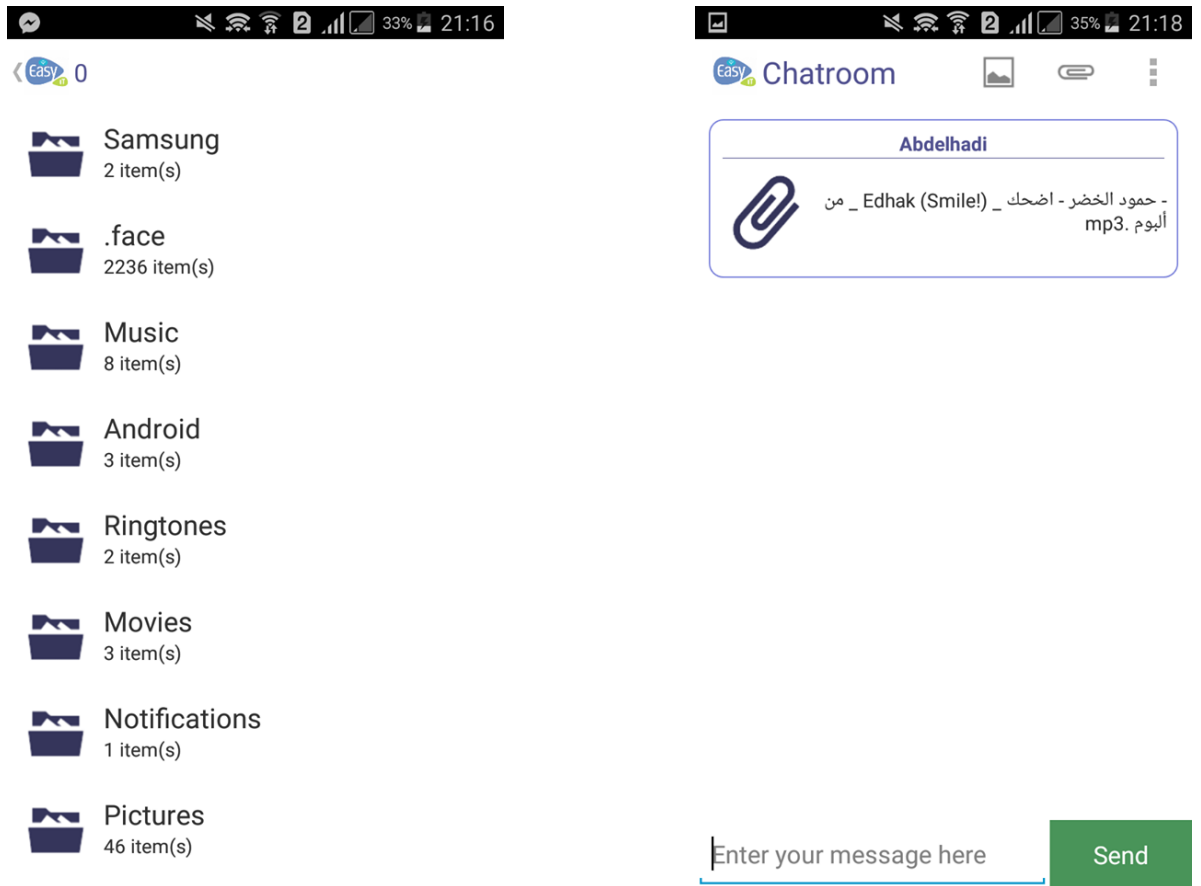


Figure 4.8 L'envoi des fichiers

Conclusion

Dans ce chapitre, nous avons présenté les différents outils et langages de programmation utilisés et on a montré les principales interfaces et fenêtres de l'application.

Conclusion générale :

En vue de croissance de Smartphone au niveau mondial ces derniers années, la téléphonie mobile est sans doute le secteur le plus dynamique, le plus rentable et le plus innovant de tout l'industrie de télécommunication que fait partie de nos quotidiens, et que plupart de ces Smartphone ont comme système exploitation l'Android .Nous étions menées a explorer ce nouveau système d'exploitation pour mobiles et à faire une applications dans le cadre de notre Projet Applications Smartphone Programmation sur Android.

Mon travail est organisé en quatre chapitres, dans la première, présent la ville intelligente Puis dans la deuxième chapitre généralités sur les Smartphone, Android et

Wifi-Direct (devisée en trois partie),ensuite je fais l'étude du domaine comme une troisième chapitre qui contient deux partie l'un pour l'étude préalable(étude de l'existant) et l'autre pour l'analyse et conception général du système. Enfin l'explication des outils utilise dans le développement de l'application avec des captures d'écran comme une quatrième chapitre.

J'ai utilisé quelque outils dans mon application comme : Android Studio, SDK, Langage de programmation comme Java et XML.

Mon vision future dans ce projet :

- ✓ Création une site web pour cette application. Page Facebook, Twitter, Instagram.
- ✓ J'envisage d'ajouter cette application dans le Google Play.

En conclusion<<Easy-It>> est une application sous Android pour les smartphone destine aux les étudiants, les enseignants pour faciliter les affaires d'études à l'université entre les étudiants et leur enseignants.

BIBLIOGRAPHIE :

Ouvrage :

[1] Joëlle Simard LA VILLE INTELLIGENTE COMME VECTEUR POUR LE DÉVELOPPEMENT DURABLE Juillet 2015

[2] Rudolf Giffinger, *Smart cities – Ranking of European medium-sized cities*, Centre of Regional Science, 2007

Site web :

[3] http://www.libelium.com/smart_parking/ (Gascón, David; Asín, Alicia;Smart Sensor Parking Platform enables city motorists save time and fuel) site Web.

Ouvrage :

[4] : Des villes intelligentes pourquoi et comment ? Marie-Andrée Doran Directrice, Institut Technologies de l'information et Sociétés (ITIS) JIQ 2012 ||19 novembre 2012.

Site web :

[5] : Saskia Sassen, Talking back to your intelligent city <http://archive.wikiwix.com/cache/?url=http%3A%2F%2Fwhatmatters.mckinseydigital.com%2Fcities%2Ftalking-back-to-your-intelligent-city>

Article :

[6] Parlement européen Mapping Smart Cities in the EU January 2014 (pdf)

Site web :

[07] <http://auto-net.fr/petite-histoire-du-smartphone/> .

[08] http://www.cefaceconumerique.com/upload/Ressources/Fiche_Smartphone_Extrait.pdf

Article :

[9] Nguyen Tien Thinh, Système d'exploitation pour les mobiles, Licence, Hanoi, Juillet 2009.

[10] Marko Gargenta, Learning Android, O'REILLY Media, United States of America, March 2011.

Site web :

[11] <http://www.android.com/history>

[12] <https://pgsmart.wordpress.com/2014/05/09/les-atouts-qui-font-le-succes-du-systeme/>

[13] <http://www.phonandroid.com/toute-l-histoire-et-la-chronologie-d-android-dossier.html#concurrency>

Ouvrage :

[14] Khaoula MRABET & Nessrine TRABELSI, Application de messagerie simple sur Android, Licence, TELECOM Sud paris, 2011.

Articles :

[15] Marko Gargenta, Learning Android, O'REILLY Media, United States of America, March 2011.

Site web :

[16] <https://developer.android.com/guide/platform/index.html>

Ouvrage :

[17] IEEE 802.11-2013 Standard, Device-To-Device communication with Wi-Fi direct: Overview and experimentation, 2007.

[18] Wi-Fi Alliance, P2P Technical Group, Wi-Fi Peer-to-Peer (P2P) Technical Specification v1.0, December 2009.

[19] Wi-Fi Alliance, Wi-Fi Protected Setup Specification v1.0h, Dec. 2006.

[20] Biersdorfer, J. D. (24 November 2014). "AirDrop Alternatives for Windows and Android". New York Times (New York, New York). Retrieved 15 April 2015.

[21] Jump up Ahaskar, Abhijit (2016-04-07). "Lenovo SHAREit: File sharing further improved". <http://www.livemint.com/>. Retrieved 2017-04-05.

Site web :

[22] OMG, http://www.omg.org/gettingstarted/what_is_uml.htm

Partie pratique :

<https://github.com>

<https://stackoverflow.com>

abdullaheid.net

GLOSSAIRE

Terme	Description
Android	système d'exploitation open source pour Smartphones.
Android Studio	Une plateforme libre pour développer des applications mobiles et de bureau en utilisant des technologies web.
Smartphones	Téléphones évolués, qui possèdent des fonctions similaires à celles des assistants personnels. Certains peuvent lire des vidéos, des MP3 et se voir ajouter des programmes spécifiques.
Fenêtre	Elément d'interface graphique d'un programme.
ADT	Abstract Data Type, plugin Android pour Eclipse.
API	Application Programming Interface : interface de programmation, bibliothèque.
AVD	Android Virtual Device, terminal Android virtuel et configurable.
PDA	Personal Digital Assistant, appareil numérique portable.
SDK	Software Development Kit, kit de développement permettant la création des applications de type défini.
UML	Unified Modeling Language, langage de modélisation graphique dans le monde du génie logiciel.
XML	Extensible Markup Language, langage informatique de balisage générique.

ملخص:

هذا التطبيق يسمح للطلبة والاساتذة من التواصل الأني ببساطة وبطريقة آمنة من خلال المحادثة الكتابية او المرئية والصوتية ، كما يمكن ايضا ارسال وتبادل الملفات والصور والفيديو الى المستعمل الآخر .

الكلمات المفتاحية:

Wifi-Direct, Android, Smartphone, P2P, Smart City .

Abstract:

This application allows students and teachers to communicate in real time simply and securely through a textual conversation or video and audio, and can also send and Share files, images and videos to the other user.

Keywords:

Wifi-Direct, Android, Smartphone, P2P, Smart City.

Résumé:

Cette application permet aux étudiants et professeurs de communication en temps réel simplement et de façon sécuritaire à travers une conversation textuelle ou vidéo et audio, et peut également envoyer et partager des fichiers, des images et des vidéos à l'autre utilisateur.

Mots-clés:

Wifi-Direct, Android, Smartphone, P2P, Smart city.