

الجمهورية الجزائرية الديمقراطية  
REpubLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
جامعة أبي بكر بلقايد – تلمسان  
Université Aboubakr Belkaïd – Tlemcen –  
Faculté de TECHNOLOGIE



## MEMOIRE

Présenté pour l'obtention du **diplôme de MASTER**

En : Télécommunications

Spécialité : Réseaux et Télécommunications

Par : HAMADOUCHE Mohammed Walid

Thème

# Développement d'une application Android VoIP client

Soutenu publiquement, le 04/07/2021, devant le jury composé de :

M. MERZOUGUI Rachid	PR	Univ. Tlemcen	Président
M. MOUSSAOUI Djillali	MCB	Univ. Tlemcen	Examineur
M. HADJILA Mourad	MCA	Univ. Tlemcen	Encadrant
M. BIBI TRIKI Chakib	Ing	ICOSNET	Co-Encadrant

## *Remerciements*

Je tiens à remercier en premier Dieu tout puissant de m'avoir donné la volonté et m'avoir armé de force pour percer et aboutir dans mes études.

Je remercie en deuxième lieu, mon encadreur Mr.BIBI TRIKI Chakib pour toutes les orientations et les conseils ainsi que mon co-encadreur Dr.HADJILA Mourad pour son soutien permanent.

Je remercie aussi mes chers parents pour tous leurs sacrifices et qu'ils ont été toujours auprès de moi pour me soutenir et m'encourager toute au long de mes études.

Je remercie mes frères Aymen et Yacine et ma sœur Hidayet qui ont été toujours présents et d'avoir cru en moi .

Je veux adresser aussi tous mes remerciements aux personnes avec lesquelles j'ai pu échanger et qu'ils m'ont aidé pour la rédaction de ce mémoire spécialement mon ami SARI Aymen qui était avec moi jusqu'au bout.

## ***Dédicaces***

*Je dédie ce projet de fin d'études aux personnes les plus chers à mon cœur :*

***Mes très chers parents Mourad et Nadja***

*Pour toutes vos patiences.*

*Pour vos sacrifices durant toutes ces années.*

*Pour avoir été toujours à mes côtés et m'avoir poussé le plus loin possible.*

*Pour toute l'affection et le soutien moral que m'avez offert.*

*Que dieu le tout puissant vous accorde la santé, le bonheur, et une longue vie*

***Ma sœur Hidayet***

***Mes frères Aymen et Yacine***

*Pour avoir toujours cru en moi, aidé, encouragé sans cesse.*

*En leur souhaitant à tous de la réussite dans leur vie.*

*J'ai de la chance de vous avoir à mes côtés.*

***En particulier mes très chers amis Arslane, Riyad, Fayçal, Madjid, Marwan, Aymen, Khalil, Adil, Mohammed et toute ma famille, tantes, oncles, cousines, cousins ainsi que mes camarades de promotion***

*Pour la bonne ambiance qu'ils m'ont offerte. Je les remercie pour leur présence indispensable dans ma vie.*

*Merci pour vos conseils, orientations et écoute.*

## **Résumé**

La voix sur IP ou VoIP est un terme utilisé en téléphonie IP qui consiste à numériser la voix et à l'encapsuler dans des paquets numériques discrets comme n'importe quelle donnée échangée sur le réseau IP. Avec le déploiement du haut et très haut débit, la téléphonie classique devienne caduque. C'est une solution souple et évolutive, ce qui a pousser les entreprises moderne de migrer vers elle. Dans ce mémoire, on s'est basé sur le développement d'une application VoIP en acheminement la communication avec un serveur SIP sous logiciel Android Studio.

**Mot clés :** VoIP , SIP , IP , Application , Android.

## **Abstract**

Voice over IP or VoIP is a term used in IP telephony that digitizes voice and encapsulates it in discrete digital packets like any data exchanged over the IP network. With the deployment of high and very high speed broadband, traditional telephony is becoming obsolete. It is a flexible and scalable solution, which has pushed modern businesses to migrate to it. In this thesis, we were based on the development of a VoIP application by routing the communication with a SIP server under Android Studio software.

**Keywords :** VoIP , SIP , IP , Application , Android.

## **Table des matières**

Remerciement .....	I
Dédicace .....	II
Résumé.....	III
Table des matières .....	IV
Sommaire .....	IV
Liste des figures .....	VII
Liste des abréviations.....	VIII

## **Sommaire**

Introduction générale.....	1
----------------------------	---

### CHAPITRE I : Généralités sur la VoIP

I.1	Introduction à la VoIP .....	2
I.2	Qu'est-ce que la VoIP ? .....	2
I.3	Principes de fonctionnement de la Voix sur IP .....	3
I.3.1	Acquisition du signal .....	3
I.3.2	Numérisation.....	3
I.3.2.1	Echantillonnage du signal sonore .....	3
I.3.2.2	Quantification .....	3
I.3.3	Compression.....	4

I.3.4	Habillage des en-têtes .....	4
I.3.5	Emission et transport.....	4
I.3.6	Réception .....	4
I.3.7	Conversion Numérique Analogique.....	4
I.3.8	Restitution.....	4
I.4	Les modèles réseaux.....	5
I.4.1	Modèle de référence OSI .....	5
I.4.1.1	La couche application.....	6
I.4.1.2	La couche présentation .....	6
I.4.1.3	La couche session.....	7
I.4.1.4	La couche transport .....	7
I.4.1.5	La couche réseau .....	7
I.4.1.6	La couche liaison des données .....	7
I.4.1.7	La couche physique.....	8
I.4.2	Le modèle TCP/IP.....	8
I.4.2.1	Le protocole TCP .....	9
I.4.2.2	Le protocole IP .....	9
I.5	IP version 4.....	11
I.6	IP version 6.....	13
I.6.1	Pourquoi utiliser IPv6 au lieu d'IPv4 ? .....	14
I.6.2	Où sont passé IPv1, IPv2 et IPv3 ? .....	14
I.6.3	Qu'est-ce qui s'est arrivé à IP version 5 ? .....	15
I.7	L'architecture de la VoIP .....	15
I.8	Les principaux protocoles de la VoIP .....	16
I.8.1	Le protocole H323 .....	16
I.8.1.1	Introduction .....	16
I.8.1.2	Fonctionnement.....	17
I.8.2	Le protocole SIP .....	19
I.8.2.1	Introduction .....	19
I.8.2.2	Fonctionnement.....	20
I.9	Les protocoles de transport .....	23
I.9.1	Le protocole RTP.....	23
I.9.1.1	Description générale de RTP.....	23
I.9.1.2	Les fonctions de RTP .....	23

I.9.1.3	Avantages et inconvénients.....	24
I.9.2	Le protocole RTCP.....	25
I.9.2.1	Description générale de RTCP .....	25
I.9.2.2	Point fort et limite du protocole RTCP .....	25
I.10	Conclusion.....	25

## CHAPITRE II : Notre application client sur la VoIP

II.1	Pourquoi une application mobile ? .....	26
II.2	Les systèmes d'exploitation mobiles.....	26
II.3	Qu'est ce qu'un Android ? .....	27
II.4	Architecture Android .....	28
II.4.1	La couche Applications.....	28
II.4.2	La couche Application FrameWork.....	29
II.4.3	La couche d'exécution Android .....	30
II.4.4	Libraries.....	31
II.4.5	La couche noyaux linux .....	31
II.5	Installation et configuration du logiciel .....	32
II.5.1	Installation.....	32
II.5.2	Configuration.....	34
II.6	Notre application.....	37
II.6.1	Principe .....	37
II.6.2	L'interface de l'application.....	38
II.6.3	La configuration du client.....	38
II.7	Résultats .....	39
II.8	Conclusion.....	42
	Conclusion générale .....	43
	Bibliographie .....	44

## Liste des figures

### *Chapitre I*

Figure I.1 : Le modèle OSI .....	5
Figure I.2 : Modèle TCP/IP .....	8
Figure I.3 : Entête IPv4 .....	11
Figure I.4 : Entête IPv6 .....	13
Figure I.5 : Architecture générale de la voix sur IP .....	16

### *Chapitre II*

Figure II.1 : Evolution des parts de marché des OS pour téléphones de 2009 à 2017 dans le monde ..	27
Figure II.2 : Architecture Android .....	28
Figure II.3 : Début d'installation .....	32
Figure II.4 : Installation du Android studio et SDK Android .....	33
Figure II.5 : Confirmation de l'installation .....	33
Figure II.6 : Android Studio .....	34
Figure II.7 : Les plateformes SDK .....	35
Figure II.8 : Les outils SDK .....	35
Figure II.9 : Les sites de mis à jour .....	36
Figure II.10 : Principe de fonctionnement .....	37
Figure II.11 : Interface de l'application .....	38
Figure II.12 : Configuration du client .....	39
Figure II.13 : Appel Sortant .....	40
Figure II.14 : Appel Entrant .....	40
Figure II.15 : Durée d'appel .....	41
Figure II.16 : Etat du serveur durant l'appel .....	41



## Liste des abréviations

<i>PSTN</i>	Public Switched Telephone System
<i>VoIP</i>	Voice over Internet Protocol
<i>IP</i>	Internet Protocol
<i>IoT</i>	Internet of Things
<i>PABX</i>	Private Automatic Branch Exchange
<i>ToIP</i>	Telephony over Internet Protocol
<i>FoIP</i>	Fax over Internet Protocol
<i>SIP</i>	Session Initiation Protocol
<i>DAC</i>	Digital Analog Converter
<i>ADC</i>	Analog Digital Converter
<i>DSP</i>	Digital Signal Processing
<i>OSI</i>	Open Systems Interconnection
<i>HTTP</i>	Hypertext Transfer Protocol
<i>HTTPS</i>	Hypertext Transfer Protocol Secure
<i>SMTP</i>	Simple Mail Transfer Protocol
<i>FTP</i>	File Transfer Protocol
<i>POP</i>	Post Office Protocol
<i>IMAP</i>	Internet Message Access Protocol
<i>SSH</i>	Secure Shell
<i>DNS</i>	Domain Name System
<i>TCP</i>	Transmission Control Protocol
<i>WWW</i>	World Wide Web
<i>UDP</i>	User Datagram Protocol
<i>ITU</i>	International Communication Union
<i>ISDN</i>	Integrated Services Digital Network

<b><i>QoS</i></b>	Quality of Service
<b><i>PBX</i></b>	Private Branch eXchange
<b><i>LAN</i></b>	Local Area Network
<b><i>RTP</i></b>	Real time Transport Protocol
<b><i>RTCP</i></b>	Real-time Transport Control Protocol
<b><i>JDK</i></b>	Java Development Kit
<b><i>SDK</i></b>	Software Development Kit

*Introduction*

*Générale*

## Introduction générale

Le système téléphonique commuté public (PSTN, Public Switched Telephone System) est connu par la plupart d'entre nous. C'est un système qui nous permet de communiquer avec des personnes dans le monde entier en composant une série de chiffres.

La VoIP est une méthode alternative, qui peut fonctionner en acheminant les signaux vocaux numérisés sur les réseaux IP, comme les réseaux Intranet d'entreprise ou, dans certains cas, l'Internet public. On remarque que, depuis plus de 100 ans, le système PSTN n'a pas fait de grand changement malgré les différentes modifications et améliorations apportées, telles que la tonalité de composition et l'identification de l'appelant. Cependant, pour l'utilisateur, il s'agit toujours de composer une série de chiffres appartenant à la personne avec qui il veut établir un contact. Néanmoins, pour parvenir à cet aboutissement, le fonctionnement qui est derrière tout ça a énormément changé durant les dernières années.

La VoIP est une technologie, mais ce n'est pas vraiment la dernière technologie. Dans les années 90, les premiers logiciels de cette technologie étaient disponibles, bien que les documents et brevets sur ce sujet remontent à des décennies. Le principe de base est simple: il s'agit essentiellement de la même technologie que celle utilisée pour diffuser la musique sur Internet. Le son est capté par le microphone et numérisé par la carte son. Utilisez ensuite le codec (qui est une abréviation de « CODER / DECODER ») audio pour compresser le son numérisé. Ce dernier supprime les données inutiles tout en conservant la lisibilité du son pour rendre le flux suffisamment compact pour être envoyé sur le réseau en temps réel. Le son est encodé à la fin de la transmission, envoyé sur le réseau, puis décodé à la réception, puis diffusé via des haut-parleurs ou des écouteurs.

N'oublions pas que les réseaux de télécommunications tels que la 4G, 5G mais aussi les réseaux de masse telle que l'IoT (Internet of Things), se basent sur le concept IP.

Ce mémoire a pour objectif de mettre au point une application client de la voix sur IP (VoIP), ainsi, il sera divisé en deux chapitres: le premier est une introduction au réseau IP, ou nous allons aborder le concept de base de la VoIP et le deuxième chapitre sera concentré sur le développement de notre application (client IP sous Android).

# CHAPITRE

# I

Généralités sur la VoIP

## **I.1 Introduction à la VoIP**

Pendant de nombreuses années, la transmission de la voix et des données doit être effectuée séparément au sein de l'entreprise. Pour la transmission de la voix, il fallait utiliser le réseau téléphonique, et pour les données, le réseau informatique doit être utilisé. En effet, l'entreprise est confrontée à plusieurs problèmes comme l'augmentation du budget et des dépenses télécoms, la perte de la bande passante allouée à la communication qui ne peut être établie, l'échec de l'intégration du service de transmission de données dans PABX n'est que pour n'en nommer que quelques-uns. En tenant compte de ces limitations, la VoIP (Voice over IP) a été conçue comme le sujet de ce projet. Comme toute innovation technologique qui se respecte, la VoIP doit non seulement simplifier le travail, mais aussi faire des économies à l'entreprise qui dépense beaucoup d'argent en communications téléphoniques, mais en revanche, le prix des communications ToIP (téléphonie sur IP) est dérisoire en comparaison. En particulier, plus l'interlocuteur est éloigné, plus la différence de prix est intéressante. De plus, les téléphones IP utilisent jusqu'à dix fois moins de bande passante que les téléphones traditionnels. Cela a suscité un grand intérêt pour la voix sur les réseaux privés. Il semble qu'après avoir examiné certains doutes sur la qualité de service, l'entreprise est désormais convaincue que les solutions proposées sur le marché ont une maturité technologique plus élevée, qu'il s'agisse d'une entreprise mono-site ou multi-sites, l'enquête montre que l'entreprise migre actuellement vers le phénomène des systèmes de téléphonie IP.

La technologie VoIP envisagée à l'origine était une technologie propriétaire, il y a eu donc une grande différence entre elles. Cependant, un système qui doit relier les gens et les systèmes nécessite un certain degré de normalisation. C'est pourquoi il existe des protocoles standards (tels que H323 ou SIP).

## **I.2 Qu'est-ce que la VoIP ?**

La Voix sur IP (ou VoIP) est une technologie qui transmet la voix sur le réseau internet au lieu du réseau téléphonique traditionnel. Elle consiste à coder la voix (signal analogique) en signaux numériques, puis à transmettre ces informations dans des paquets de données séparés vers une application qui sera responsable de la conversion inverse.

Par conséquent, au lieu d'utiliser un réseau informatique et un réseau téléphonique commuté (RTC), l'entreprise peut fusionner tout le contenu en un seul réseau à l'aide de la VoIP. Cela est dû au fait que la téléphonie devient « DATA ».

Bien que VoIP signifie voix sur IP, il ne se limite pas à la voix, mais inclut également tous les canaux de communication modernes. Ainsi, par exemple, cette technologie est utilisée pour diffuser des appels vidéo, des fax ou des messages instantanés. Nous avons également parlé de FoIP, qui est utilisé pour le fax IP et la messagerie IP et de ToIP (téléphonie IP). Quand on parle de VoIP, on parle d'une technologie ou d'un groupe de technologies.

### **I.3 Principes de fonctionnement de la Voix sur IP [1]**

Lorsque vous souhaitez communiquer avec un autre, une connexion est établie entre les deux terminaux. Ensuite, l'utilisateur peut émettre un son (signal analogique) à travers le microphone, puis la machine le numérise et le compresse (en synthétisant le signal), et envoie les données encapsulées dans le paquet de données au récepteur, qui effectuera l'opération inverse. Le format du message source. Par conséquent, la transmission de la voix sur un réseau IP nécessite tout ou partie des étapes suivantes:

#### **I.3.1 Acquisition du signal**

Cette étape consiste naturellement à capter la voix d'un micro, qu'il s'agisse de celui d'un téléphone ou d'un micro casque.

#### **I.3.2 Numérisation**

La voix passe alors dans un CAN (Convertisseur Analogique Numérique) qui réalise deux tâches distinctes à savoir :

##### ***I.3.2.1 Echantillonnage du signal sonore***

C'est un prélèvement périodique de ce signal. Il s'agit d'enregistrer à des intervalles très rapprochés la valeur d'un signal afin de pouvoir disposer d'un enregistrement proche de la valeur réelle de ce signal.

##### ***I.3.2.2 Quantification***

Elle consiste à affecter une valeur numérique en binaire à chaque échantillon. Plus les échantillons sont codés sur un nombre de bits importants, meilleure sera la qualité.

### **I.3.3 Compression**

Le signal une fois numérisé peut être traité par un DSP (Digital Signal Processor) qui va le compresser afin d'avoir la bande passante nécessaire pour le transmettre. L'avantage de la compression est de réduire la bande passante nécessaire pour transmettre le signal.

### **I.3.4 Habillage des en-têtes**

Les données doivent encore être enrichies en informations avant d'être converties en paquets de données à expédier sur le réseau.

### **I.3.5 Emission et transport**

Les paquets sont acheminés depuis le point d'émission pour atteindre le point de réception sans qu'un chemin précis soit réservé pour leur transport.

### **I.3.6 Réception**

Lorsque les paquets arrivent à destination, il est essentiel de les placer dans le bon ordre et assez rapidement car faute de quoi une dégradation de la voix se fera ressentir.

### **I.3.7 Conversion Numérique Analogique**

La conversion numérique analogique se fait à l'aide d'un CNA, c'est la réciproque de l'étape 2.

### **I.3.8 Restitution**

Dès lors, la voix peut être retranscrite par un haut-parleur du casque, du combiné téléphonique ou de l'ordinateur.



## I.4 Les modèles réseaux

### I.4.1 Modèle de référence OSI [2]

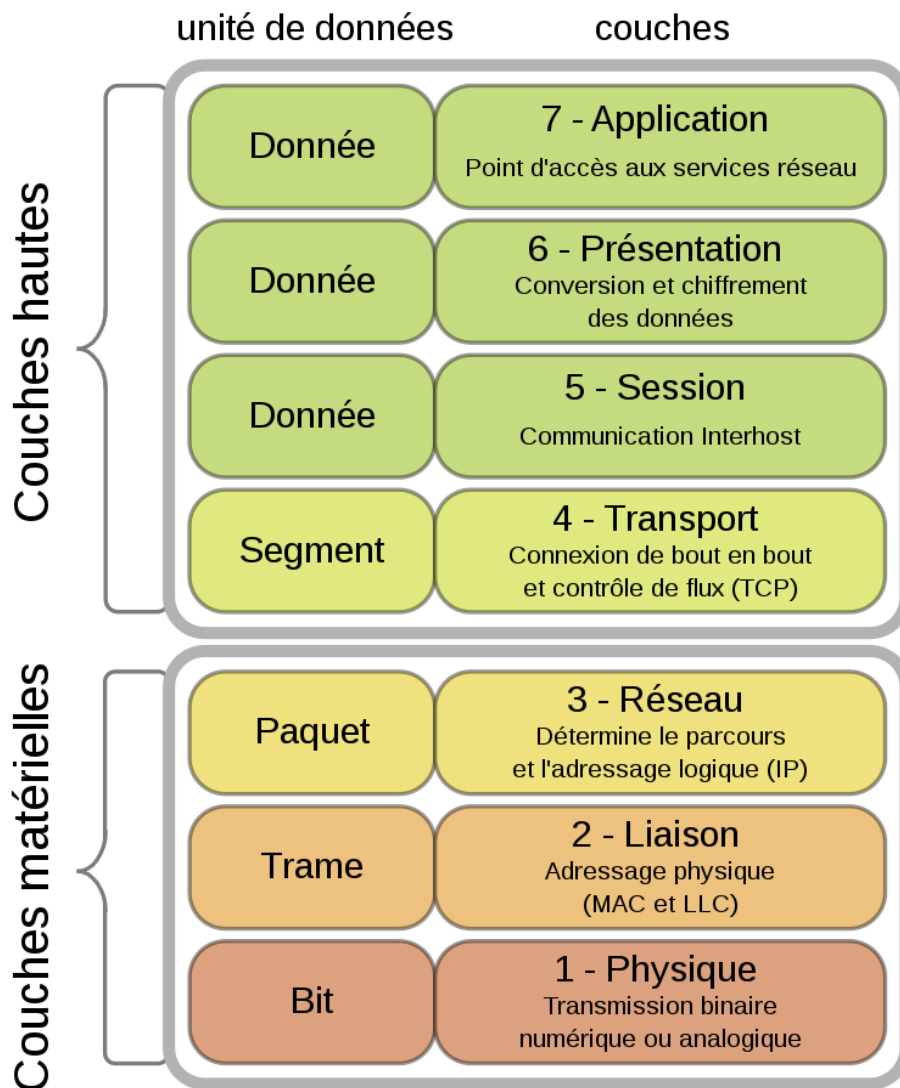


Figure I.1 : Le modèle OSI [3]

Le modèle OSI (Open Systems Interconnection) est un modèle de référence (théorique) normé en 1984 ayant pour but de définir des normes de communication entre différents systèmes informatiques.

Le modèle fournit un système de communication composé de 7 couches différentes. L'idée de cette représentation est de décomposer à nouveau la communication entre les deux appareils en différentes phases bien définies, afin que nous puissions ensuite étendre

indépendamment les composants de chaque couche sans avoir à modifier l'ensemble du processus de communication le plus rapidement possible.

Les 7 couches définies par ce modèle sont les suivantes :

#### **I.4.1.1 La couche application**

La couche application est la couche la plus proche de l'utilisateur. La majorité des protocoles utilisés par les utilisateurs se situent dans cette couche (HTTP, SMTP, FTP, etc.).

Cette couche interagit avec les applications logicielles qui implémentent des composants de communication. La couche application est le point d'accès aux services réseaux.

La couche application a généralement pour fonction d'identifier les interlocuteurs, de déterminer si les ressources sont disponibles et de synchroniser les communications.

On peut notamment utiliser les protocoles suivants (en se limitant au monde IP) :

- Les protocoles FTP (IETF), NFS (Sun Microsystems) et AFS, SMB/CIFS (Microsoft) pour le transfert de fichiers.
- Les protocoles Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), Internet Message Access Protocol (IMAP) pour les services de messagerie.
- Les protocoles Telnet, rlogin, Secure Shell (SSH) pour les sessions distantes.
- Le protocole HTTP pour le transfert de ressources hypermédia comme les documents HTML.
- Des protocoles d'exploitation et de gestion comme Domain Name System (DNS) pour la résolution d'adresse, Simple Network Management Protocol pour la supervision.

#### **I.4.1.2 La couche présentation**

La couche présentation est chargée du formatage des données de la couche applicative afin qu'elles puissent être envoyées à travers le réseau puis être lues à nouveau par les applications.

#### **I.4.1.3 La couche session**

La couche session contrôle les connexions entre les ordinateurs. Cette couche permet l'ouverture et la fermeture de session et gère la synchronisation des échanges ainsi que les transactions.

#### **I.4.1.4 La couche transport**

La couche transport fournit les moyens concrets pour transférer des données de taille variable d'une source vers une destination en conservant la qualité du service.

L'enjeu de la couche de transport est de réceptionner les données qui viennent des couches supérieures, de les découper et de la faire transiter jusqu'à la couche réseau.

Cette couche est la première à communiquer directement avec la machine de destination : elle gère les communications de bout en bout ("end to end") entre processus (programmes en cours d'exécution).

#### **I.4.1.5 La couche réseau**

La couche réseau fournit les moyens concrets pour transférer des données de taille variable (appelés "paquets") entre différents réseaux de nœuds.

On va notamment effectuer le routage et l'adressage des paquets dans cette couche, c'est-à-dire qu'on va définir la route que vont emprunter les paquets pour aller d'un point de départ à un point d'arrivée (d'un interlocuteur à l'autre).

#### **I.4.1.6 La couche liaison des données**

La couche liaison des données gère les communications entre deux machines directement connectées entre elles.

On va dans cette couche découper des données brutes en frames de tailles variables puis les envoyer de manière séquentielle.

On va également dans cette couche détecter et pouvoir corriger les erreurs pouvant survenir dans la couche physique, définir le protocole pour établir et mettre fin à une

connexion entre deux périphériques connectés physiquement et définir le protocole de contrôle de flux (régulation du trafic) entre eux.

#### I.4.1.7 La couche physique

La couche physique est chargée de la transmission des signaux entre les interlocuteurs. Son service est limité à l'émission et la réception d'un bit ou d'un train de bits continu.

La couche physique est, comme son nom l'indique, la couche dans laquelle sont définis les protocoles du monde physique (les différents câbles de transmission).

#### I.4.2 Le modèle TCP/IP

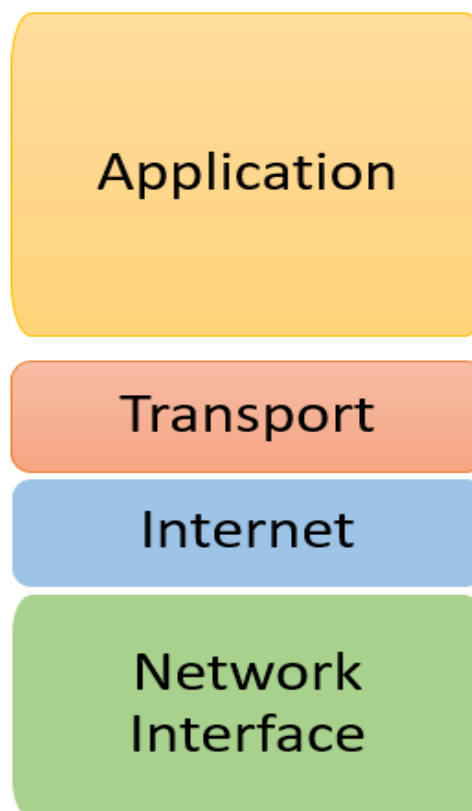


Figure I.2 : Modèle TCP/IP [4]

Le modèle TCP/IP (également connu sous le nom de « modèle Internet »), qui a débuté en 1976 a été stabilisé avant la sortie du modèle OSI en 1984.

Pour référence, TCP/IP est un modèle dérivé d'ARPANET, et son but était de maintenir la communication à tout prix en cas d'attaque nucléaire. Ce protocole est un réseau basé sur le routage de paquets à travers une couche appelée Internet.

Le modèle TCP/IP est une approche réaliste ou pratique du modèle réseau et le modèle OSI est un modèle idéalisé ou théorique. En conséquence, c'est le modèle TCP/IP qui est utilisé comme modèle de réseau de référence pour Internet.

Le nom du modèle TCP/IP vient de ces deux protocoles principaux : TCP (Transmission Control Protocol) et IP (Internet Protocol).

#### **I.4.2.1 Le protocole TCP**

Le protocole de contrôle de transmission est un protocole de transport fiable en mode de connexion. Ce mode consiste à établir une session de communication entre deux parties qui souhaitent échanger des données. Cette session a un début, une fin et une vérification (vérification d'erreurs) généralement un identifiant et un mot de passe.

Ce protocole est utilisé pour environ 95% de tout le trafic internet et les applications qui l'utilisent le plus sont SMTP/POP3/IMAP (la messagerie), HTTP/HTTPS (World Wide Web), FTP (transfert de fichiers).

Une session TCP fonctionne en trois phases :

- l'établissement de la connexion.
- les transferts de données.
- la fin de la connexion.

#### **I.4.2.2 Le protocole IP**

Le protocole internet est une série de protocoles de communication de réseau informatique conçus pour internet.

Le protocole IP fait partie de la suite de protocoles internet et fournit un service d'adressage unique pour tous les terminaux connectés. Il permet aussi de découper l'information à transmettre en paquets, les adresser et les livrer indépendamment et de recomposer le message initial à l'arrivée (commutation de paquets).

On trouve plusieurs versions du protocole IP dont seulement deux sont utilisées (IPv4 et IPv6).

Les 4 couches définies par ce modèle sont les suivantes : [2]

#### **a- La couche application**

Le modèle TCP/IP regroupe les trois couches de session, présentation et application du modèle OSI dans une seule couche application. En effet, d'un point de vue pratique, cela ne fait souvent pas beaucoup de sens de séparer ces couches.

Cette couche contient tous les protocoles de haut niveau : FTP pour le transfert de fichiers, SMTP pour les mails, HTTP pour le WWW, DNS pour les noms de domaine.

#### **b- La couche transport**

La couche transport assure la communication logique entre processus. Cette couche détermine comment les données doivent être envoyées : de manière fiable ou pas.

Concrètement, on va pouvoir choisir entre deux protocoles dans la couche transport : TCP (Transmission Control Protocol) et UDP (User Datagram Protocol).

TCP est un protocole de transfert fiable orienté connexion. Ce protocole contrôle et s'assure qu'il n'y ait ni perte ni corruption de données. Il est donc en charge des erreurs. TCP est le protocole le plus utilisé sur le Web aujourd'hui.

UDP est un protocole de transfert non fiable et qui ne nécessite pas de connexion préalable. Ce protocole est particulièrement utilisé pour les échanges où la perte de quelques données n'est pas grave (appel vidéo, jeu en ligne, etc.) car il est plus rapide que TCP.

#### **c- La couche Internet**

Le but principal de la couche Internet est d'assurer la communication logique entre hôte, c'est-à-dire de transmettre coûte que coûte les paquets d'un hôte à un autre et de faire en sorte qu'ils arrivent à destination. Le protocole principal de cette couche est IP (Internet

Protocol ou Protocole Internet). Les paquets peuvent prendre différentes routes pour arriver à destination et arriver dans un ordre différent de l'ordre dans lequel ils ont été envoyés.

Dans son fonctionnement, la couche Internet n'est pas responsable d'une transmission fiable. Elle ne fournit qu'un service peu fiable et une livraison optimale (via le routage et l'adressage). Étant donné que la livraison de paquets entre divers réseaux est une opération intrinsèquement peu fiable et sujette aux pannes, la charge de la fiabilité a été placée avec les points d'extrémité d'un chemin de communication, c'est-à-dire les hôtes, plutôt que sur le réseau. Ce sera aux protocoles de plus haut niveau d'assurer la fiabilité du service.

#### d- La couche accès réseau

La couche accès réseau du modèle TCP/IP regroupe les couches : physique et de liaison des données du modèle OSI. Cette couche définit comment envoyer des paquets IP à travers le réseau (via des protocoles comme Ethernet ou Wireless entre autres).

### I.5 IP version 4

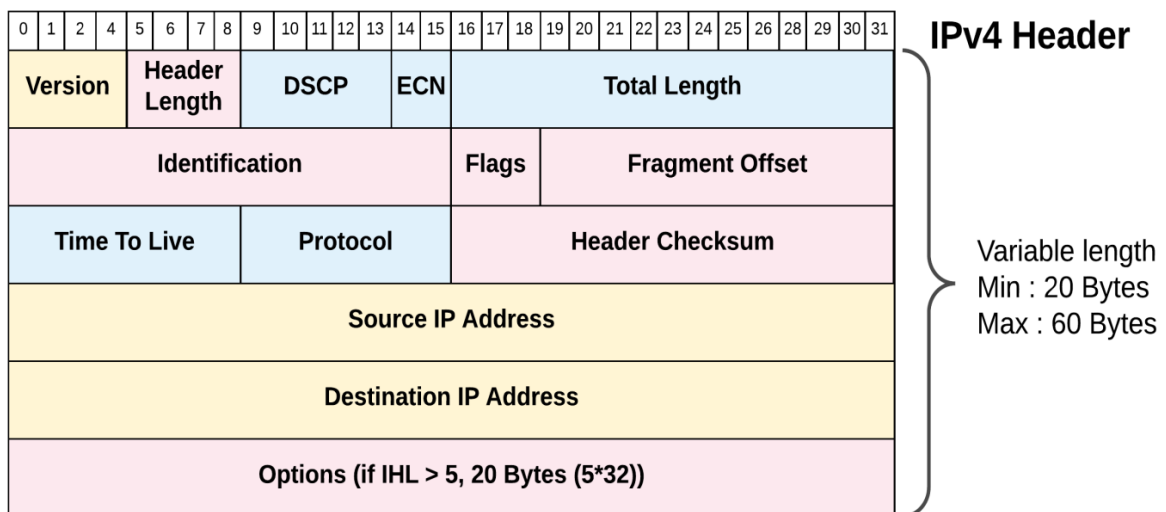


Figure I.3 : Entête IPv4 [5]

- **Le champ Version (*Version*) (4 bits) :** indique simplement la version du protocole. Il peut y avoir plusieurs versions de protocole IP. On utilise la version IPv4, mais la version IPv6 est déjà très présente et qui est l'avenir d'internet.

On voit ce champs en premier parce qu'il est important de savoir la version utilisée pour savoir les informations qu'on va trouver car selon la version le contenu de l'en-tête change.

- **Le champ longueur de l'en-tête (IHL : Header Length) (4 bits) :** il sert à indiquer la longueur de l'en-tête que sera toujours 20 octets.
- **Le champ type de service (TOS : Type Of Service) (8 bits) :** permet de marquer un paquet comme étant plus important qu'un autre. Mais cela a été très peu utilisé.
- **Le champ longueur totale (TPL : Total Length) (16 bits) :** décrit la longueur du paquet. Ça comprend l'en-tête et les données, qui sont représentées ici par le champ DATA.
- **Le Fragment Identification (ID : Identification) (16 bits) :** c'est un identifiant unique qui permet de savoir d'où proviennent les fragments. Par exemple, si un datagramme a un ID « X », tous les fragments qui le suit auront un ID « X ».
- **Le champ Indicateur (Flag) (3 bits) :** définit divers indicateurs de contrôle qui concerne la fragmentation.
- **Le champ Décalage de fragment (Fragment Offset) (13 bits) :** est lui aussi lié à la fragmentation. Il indique l'endroit où le fragment a été fait.
- **Le champ durée de vie (TTL : Time To Live) (8 bits) :** il comprend le nombre de routeurs que le paquet peut encore traverser avant d'être détruit. Ça permet d'éviter qu'un paquet ne tourne indéfiniment dans un réseau, par exemple à cause d'un problème de routage. On peut le comparer comme à une date de péremption.
- **Le champ Protocole (Protocol) (8 bits) :** indique le protocole qui est utilisé pour les données du paquet soit TCP ou UDP dans la majorité des cas.
- **Le champ somme de contrôle (Checksum) (16 bits) :** permet de contrôler l'intégrité de l'entête, s'il estime que le paquet a été modifié sur sa route (si jamais un 1 ou un 0 a changé, la somme de contrôle ne sera pas correcte), alors il sera détruit.
- **L'Adresse source (Source Address) (32 bits) :** c'est l'adresse IP de la machine qui a émis le paquet.
- **L'Adresse de destination (Destination Address) (32 bits) :** c'est celle de la machine à qui est destiné le paquet.
- **Le champ Option (Option) :** comprend divers paramètres facultatifs, qui sont très rarement utilisés.
- **Le champ Données (Data) :** correspond aux données du paquet [6].



## I.6 IP version 6

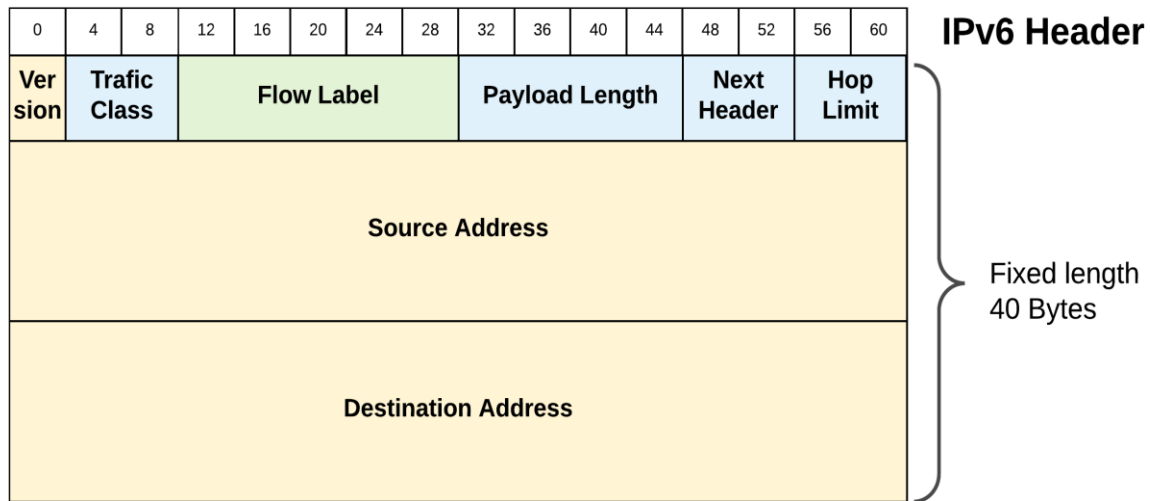


Figure I.4 : Entête IPv6 [5]

- **Champ Version (Version) (4 bits)** : comme on a dit au paravent, ce champ indique la version utilisée, ici IPv6.
- **Classe de trafic (Traffic Class) (8 bits)** : identiquement au champ TOS d'IPv4, il sert à distinguer la classe ou la priorité du paquet.
- **Etiquette de flot (Flow Label) (20 bits)** : permet le marquage d'un flux pour un traitement différencié dans le réseau et le numéro de port.
- **Longueur du contenu (Payload length) (16 bits)** : contrairement à la longueur d'IPv4 le champ longueur de l'entête IPv6 indique le nombre d'octet des données qui suivent cette entête parce que ses options sont considérés comme étant une donnée.
- **Entête suivante (Next Header) (8 bits)** : identifie le type de data ou de l'option qui se trouve derrière l'entête IPv6, c'est les mêmes valeurs du champ « Protocole » d'IPv4.
- **Limite de sauts (Hop Limit) (8 bits)** : il indique le nombre des routeurs maximum et décrémente de 1 par chaque routeur qui traverse, le paquet est détruit si ce champ atteint 0. Il est semblable au champ « TTL » d'IPv4.
- **L'adresse source (Source Address) (128 bits)** : c'est l'adresse IP de la machine qui a émis le paquet.
- **L'adresse destination (Destination Address) (128 bits)** : c'est celle de la machine à qui est destiné le paquet.

### **I.6.1 Pourquoi utiliser IPv6 au lieu d'IPv4 ?**

La limite théorique d'IPv4 est de 4,3 milliards d'adresses, à sa sortie, c'était plus que suffisant. Mais avec le développement et la mondialisation d'Internet, les adresses sont devenues insuffisantes, en particulier à l'ère actuelle des téléphones intelligents et l'IoT. Alors que, IPv6 en théorie, donne un peu près 340 décillions d'adresses et en plus de ça, il a corrigé plusieurs lacunes d'IPv4 dont la principale est la sécurité.

Dans les années 90, internet a commencé à manquer d'adresses IPv4. Mais, à chaque fois des ingénieurs ont trouvé une solution au problème, mais ils pourraient bientôt transformer l'objectif en une solution durable. IPv6 a été développé pour résoudre définitivement ces problèmes de capacité. Lorsqu'IPv4 ne peut plus supporter la charge, IPv6 doit être utilisé.

Maintenant, IPv6 coexiste avec IPv4 parce qu'il est déployé lentement, au fur et à mesure qu'IPv4 est retiré.

### **I.6.2 Où sont passé IPv1, IPv2 et IPv3 ?**

TCP version 1 a été conçue en 1973. Ceci a été documenté par RFC 675. TCP version 2 a été documentée en mars 1977. En août 1977, Jon Postel s'est rendu compte qu'ils allaient dans la mauvaise direction avec le protocole. «Nous nous trompons dans la conception de nos protocoles Internet en violant le principe de la superposition. Plus précisément, nous essayons d'utiliser TCP pour faire deux choses: servir de protocole de bout en bout au niveau de l'hôte et servir de protocole de routage et d'emballage Internet. Ces deux éléments doivent être fournis de manière modulaire et en couches. Je suggère qu'un nouveau protocole inter-réseau distinct soit nécessaire et que TCP soit utilisé strictement comme protocole de bout en bout de niveau hôte ».

À ce stade, TCP et IP ont été séparés, les deux étant versionnés numéro 3 au printemps 1978. La stabilité a été ajoutée dans la quatrième révision et c'est ainsi que nous sommes arrivés à IPv4 [7].

### I.6.3 Qu'est-ce qui s'est arrivé à IP version 5 ?

IPv5 n'est jamais devenu un protocole officiel. Il y a de nombreuses années, ce que l'on appelle IPv5 a commencé sous un autre nom: Internet Stream Protocol, ou simplement ST. Il a été développé comme un moyen de diffuser des données vidéo et vocales, et c'était expérimental. Il n'a jamais été transféré à un usage public parce qu'il a utilisé l'adressage d'IPv4, qui est finalement devenu un problème.

IPv4 est limité dans le nombre d'adresses disponibles et en 2011, les derniers blocs d'adresses IPv4 restants ont été alloués. IPv5 aurait souffert de la même limitation. Ainsi, il a été abandonné avant de devenir un standard, et le monde est passé à IPv6.

## I.7 L'architecture de la VoIP [8]

La VoIP étant une nouvelle technologie de communication, elle n'a pas encore de standard unique. En effet, chaque constructeur apporte ses normes et ses fonctionnalités à ses solutions. Les trois principaux protocoles sont H.323, SIP et MGCP/MEGACO. Il existe donc plusieurs approches pour offrir des services de téléphonie et de visiophonie sur des réseaux IP. Certaines placent l'intelligence dans le réseau alors que d'autres préfèrent une approche égale à égale avec l'intelligence répartie à la périphérie. Chacune ayant ses avantages et ses inconvénients. La figure I.5 décrit, de façon générale, la topologie d'un réseau de téléphonie IP. Elle comprend toujours des terminaux, un serveur de communication et une passerelle vers les autres réseaux. Chaque norme a ensuite ses propres caractéristiques pour garantir une plus ou moins grande qualité de service. L'intelligence du réseau est aussi déportée soit sur les terminaux, soit sur les passerelles/ contrôleur de commutation, appelées Gatekeeper. On retrouve les éléments communs suivants:

- **Le routeur** : permet d'aiguiller les données et le routage des paquets entre deux réseaux. Certains routeurs permettent de simuler un Gatekeeper grâce à l'ajout de cartes spécialisées supportant les protocoles VoIP.
- **La passerelle** : permet d'interfacer le réseau commuté et le réseau IP.
- **Le PABX** : est le commutateur du réseau téléphonique classique. Il permet de faire le lien entre la passerelle ou le routeur, et le réseau téléphonique commuté (RTC). Toutefois, si tout le réseau devient IP, ce matériel devient obsolète.
- **Les Terminaux** : sont généralement de type logiciel (software phone) ou matériel (hardphone), le softphone est installé dans le PC de l'utilisateur. L'interface audio peut être un microphone et des haut-parleurs branchés sur la carte son, même si un casque

est recommandé. Pour une meilleure clarté, un téléphone USB ou Bluetooth peut être utilisé. Le hardphone est un téléphone IP qui utilise la technologie de la Voix sur IP pour permettre des appels téléphoniques sur un réseau IP tel que l'Internet au lieu de l'ordinaire système PSTN. Les appels peuvent parcourir par le réseau internet comme par un réseau privé. Un terminal utilise des protocoles comme le SIP (Session Initiation Protocol) ou l'un des protocoles propriétaire tel que celui utilisé par Skype.

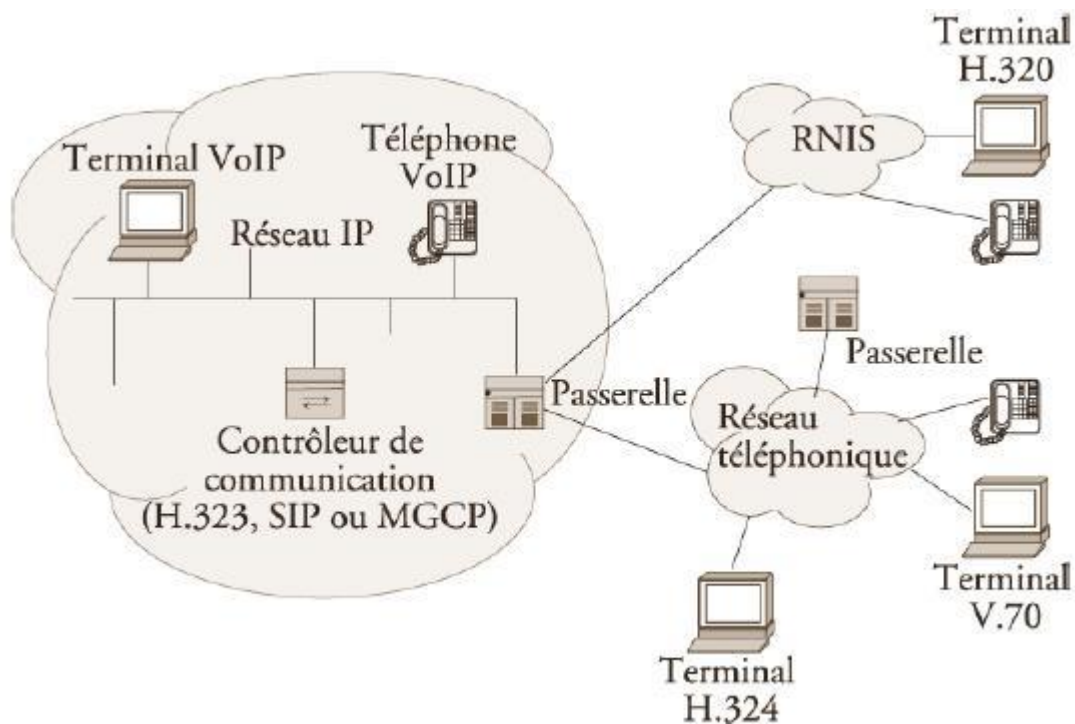


Figure I.5 : Architecture générale de la voix sur IP [9]

## I.8 Les principaux protocoles de la VoIP [10]

### I.8.1 Le protocole H323

#### I.8.1.1 Introduction

Avec le développement du multimédia sur les réseaux, il est devenu nécessaire de créer des protocoles qui supportent ces nouvelles fonctionnalités, telles que la visioconférence : l'envoi de son et de vidéo avec un souci de données temps réel. Le protocole H.323 est l'un d'eux. Il permet de faire de la visioconférence sur des réseaux IP.

H.323 est un protocole de communication englobant un ensemble de normes utilisées pour l'envoi de données audio et vidéo sur Internet. Il existe depuis 1996 et a été initié par l'ITU (International Communication Union), un groupe international de téléphonie qui développe des standards de communication. Concrètement, il est utilisé dans des programmes tels que Microsoft Netmeeting, ou encore dans des équipements tels que les routeurs Cisco. Il existe un projet Open H.323 qui développe un client H.323 en logiciel libre afin que les utilisateurs et les petites entreprises puissent avoir accès à ce protocole sans avoir à déboursé beaucoup d'argent.

### I.8.1.2 Fonctionnement

Le protocole H.323 est utilisé pour l'interactivité en temps réel, notamment la visioconférence (signalisation, enregistrement, contrôle d'admission, transport et encodage). C'est le leader du marché pour la téléphonie IP. Il s'inspire du protocole H.320 qui proposait une solution pour la visioconférence sur un réseau numérique à intégration de service (RNIS ou ISDN en anglais), comme par exemple le service Numéris proposé par France Telecom. Le protocole H.323 est une adaptation de H.320 pour les réseaux IP. A l'heure actuelle, la visioconférence sur liaison RNIS est toujours la technique la plus déployée. Elle existe depuis 1990. Les réseaux utilisés sont à commutation de circuits. Ils permettent ainsi de garantir une Qualité de Service (QoS) aux utilisateurs (pas de risque de coupure du son ou de l'image). Aujourd'hui, c'est encore un avantage indiscutable. Par contre, comme pour le téléphone, la facturation est en fonction du débit utilisé, du temps de communication et de la distance entre les appels.

H.323 définit plusieurs éléments de réseaux :

- **Les terminaux :** dans un contexte de téléphonie sur IP, deux types de terminaux H.323 sont aujourd'hui disponibles. Un poste téléphonique IP raccordé directement au réseau Ethernet de l'entreprise. Un PC multimédia sur lequel est installé une application compatible H.323.
- **Les passerelles (GW: Gateway) :** elles assurent l'interconnexion entre un réseau IP et le réseau téléphonique, ce dernier pouvant être soit le réseau téléphonique public, soit un PABX d'entreprise. Elles assurent la correspondance de la signalisation et des signaux de contrôle et la cohésion entre les médias. Pour ce faire, elles implémentent les fonctions suivantes de transcodage audio (compression, décompression), de modulation, démodulation (pour les fax), de suppression

d'échos, de suppression des silences et de contrôle d'appels. Les passerelles sont le plus souvent basées sur des serveurs informatiques standards (Windows NT, Linux) équipés d'interfaces particulières pour la téléphonie (interfaces analogiques, accès de base ou accès primaire RNIS, interface E1, etc.) et d'interfaces réseau, par exemple de type Ethernet. La fonctionnalité de passerelle peut toutefois être intégrée directement dans le routeur ainsi que dans les PBX eux-mêmes.

- **Les portiers (GK: Gatekeeper) :** ils sont des éléments optionnels dans une solution H.323. Ils ont pour rôle de réaliser la traduction d'adresse (numéro de téléphone – adresse IP) et la gestion des autorisations. Cette dernière permet de donner ou non la permission d'effectuer un appel, de limiter la bande passante si besoin et de gérer le trafic sur le LAN. Les « gardes-barrière » permettent également de gérer les téléphones classiques et la signalisation permettant de router les appels afin d'offrir des services supplémentaires. Ils peuvent enfin offrir des services d'annuaires.
- **Les unités de contrôle multipoint (MCU, Multipoint Control Unit) :** Référence au protocole T.120 qui permet aux clients de se connecter aux sessions de conférence de données. Les unités de contrôle multipoint peuvent communiquer entre elles pour échanger des informations de conférence.

Dans un contexte de téléphonie sur IP, la signalisation a pour objectif de réaliser les fonctions suivantes :

- **Recherche et traduction d'adresses :** sur la base du numéro de téléphone du destinataire, il s'agit de trouver son adresse IP (appel téléphone, PC) ou l'adresse IP de la passerelle desservant le destinataire. Cette fonction est prise en charge par le Gatekeeper. Elle est effectuée soit localement soit par requête vers un annuaire centralisé.
- **Contrôle d'appel :** l'équipement terminal («endpoint» = terminal H.323 ou passerelle) situé à l'origine de l'appel établit une connexion avec l'équipement de destination et échange avec lui les informations nécessaires à l'établissement de l'appel. Dans le cas d'une passerelle, cette fonction implique également de supporter la signalisation propre à l'équipement téléphonique à laquelle elle est raccordée (signalisation analogique, Q.931, etc.) et de traduire cette signalisation dans le format défini dans H.323. Le contrôle d'appel est pris en charge soit par les équipements terminaux soit par le Gatekeeper.

- Dans ce cas, tous les messages de signalisation sont routés via le Gatekeeper, ce dernier jouant alors un rôle similaire à celui d'un PBX.
- Services supplémentaires : déviation, transfert d'appel, conférence, etc.

Trois protocoles de signalisation sont spécifiés dans le cadre de H.323 à savoir :

- **RAS (Registration, Admission and Status)** : ce protocole est utilisé pour communiquer avec un Gatekeeper. Il sert notamment aux équipements terminaux pour découvrir l'existence d'un Gatekeeper et s'enregistrer auprès de ce dernier ainsi que pour les demandes de traduction d'adresses. La signalisation RAS utilise des messages H.225.0 6 transmis sur un protocole de transport non fiable (UDP, par exemple).
- **Q.931** : H.323 utilise une version simplifiée de la signalisation RNIS Q.931 pour l'établissement et le contrôle d'appels téléphoniques sur IP. Cette version simplifiée est également spécifiée dans la norme H.225.0.
- **H.245** : ce protocole est utilisé pour l'échange de capacités entre deux équipements terminaux. Par exemple, il est utilisé par ces derniers pour s'accorder sur le type de codec à activer. Il peut également servir à mesurer le retard aller-retour (Round Trip Delay) d'une communication.

Une communication H.323 se déroule en cinq phases :

- Établissement d'appel.
- Échange de capacité et réservation éventuelle de la bande passante à travers le protocole RSVP (Ressource reSerVation Protocol).
- Établissement de la communication audio-visuelle.
- Invocation éventuelle de services en phase d'appel (par exemple, transfert d'appel, changement de bande passante, etc.).
- Libération de l'appel.

## I.8.2 Le protocole SIP

### I.8.2.1 Introduction

Le protocole SIP (Session Initiation Protocole) a été initié par le groupe MMUSIC (Multiparty Multimedia Session Control) et désormais repris et maintenu par le groupe SIP de l'IETF donnant la RFC 3261 rendant obsolète la RFC 2543. SIP est un protocole de

signalisation appartenant à la couche application du modèle OSI. Son rôle est d'ouvrir, modifier et libérer les sessions. L'ouverture de ces sessions permet de réaliser de l'audio ou vidéoconférence, de l'enseignement à distance, de la voix (téléphonie) et de la diffusion multimédia sur IP essentiellement. Un utilisateur peut se connecter avec les utilisateurs d'une session déjà ouverte. Pour ouvrir une session, un utilisateur émet une invitation transportant un descripteur de session permettant aux utilisateurs souhaitant communiquer de s'accorder sur la compatibilité de leur média, SIP permet donc de relier des stations mobiles en transmettant ou redirigeant les requêtes vers la position courante de la station appelée. Enfin, SIP possède l'avantage de ne pas être attaché à un médium particulier et est sensé être indépendant du protocole de transport des couches basses.

### I.8.2.2 Fonctionnement

SIP intervient aux différentes phases de l'appel :

- Localisation du terminal correspondant,
- Analyse du profil et des ressources du destinataire,
- Négociation du type de média (voix, vidéo, données, etc...) et des paramètres de communication,
- Disponibilité du correspondant, détermine si le poste appelé souhaite communiquer, et autorise l'appelant à le contacter.
- Etablissement et suivi de l'appel, avertit les parties appelant et appelé de la demande d'ouverture de session, gestion du transfert et de la fermeture des appels.
- Gestion de fonctions évoluées : cryptage, retour d'erreurs, ...

Avec SIP, les utilisateurs qui ouvrent une session peuvent communiquer en mode point à point, en mode diffusif ou dans un mode combinant ceux-ci. SIP permet donc l'ouverture de sessions en mode :

- Point-à-point – Communication entre 2 machines, on parle d'unicast.
- Diffusif – Plusieurs utilisateurs en multicast, via une unité de contrôle M.C.U (Multipoint Control Unit)
- Combinatoire – Plusieurs utilisateurs pleinement interconnectés en multicast via un réseau à maillage complet de connexions.

Voici les différents éléments intervenant dans l'ouverture de session :



- Suivant nature des échanges, choix des protocoles les mieux adaptés (RSVP, RTP, RTCP, SAP, SDP).
- Détermination du nombre de sessions, comme par exemple, pour véhiculer de la vidéo, 2 sessions doivent être ouvertes (l'une pour l'image et l'autre pour la vidéo).
- Chaque utilisateur et sa machine sont identifiés par une adresse que l'on nomme Url SIP et qui se présente comme une Url Mailto.
- Requête Uri permettant de localiser le proxy server auquel est rattachée la machine de l'appelé.
- Requête SIP, une fois le client (machine appelante) connecté à un serveur SIP distant, il peut lui adresser une ou plusieurs requêtes SIP et recevoir une ou plusieurs réponses de ce serveur. Les réponses contiennent certains champs identiques à ceux des requêtes, tels que : Call-ID, Cseq, To et From.

Les échanges entre un terminal appelant et un terminal appelé se font par l'intermédiaire de requêtes :

- Invite – Cette requête indique que l'application (ou utilisateur) correspondante à l'Url SIP spécifié est invité à participer à une session. Le corps du message décrit cette session (par ex : média supportés par l'appelant). En cas de réponse favorable, l'invité doit spécifier les médias qu'il supporte.
- Ack – Cette requête permet de confirmer que le terminal appelant a bien reçu une réponse définitive à une requête Invite.
- Options – Un proxy server en mesure de contacter l'UAS (terminal) appelé, doit répondre à une requête Options en précisant ses capacités à contacter le même terminal.
- Bye – Cette requête est utilisée par le terminal de l'appelé a fin de signaler qu'il souhaite mettre un terme à la session.
- Cancel – Cette requête est envoyée par un terminal ou un proxy server à fin d'annuler une requête non validée par une réponse finale comme, par exemple, si une machine ayant été invitée à participer à une session, et ayant acceptée l'invitation ne reçoit pas de requête Ack, alors elle émet une requête Cancel.
- Register – cette méthode est utilisée par le client pour enregistrer l'adresse listée dans l'URL TO par le serveur auquel il est relié.

Une réponse à une requête est caractérisée, par un code et un motif, appelés code d'état et raison phrase respectivement. Un code d'état est un entier codé sur 3 bits indiquant un résultat à l'issue de la réception d'une requête. Ce résultat est précisé par une phrase, textbased (UTF-8), expliquant le motif du refus ou de l'acceptation de la requête. Le code d'état est donc destiné à l'automate gérant l'établissement des sessions SIP et les motifs aux programmeurs. Il existe 6 classes de réponses et donc de codes d'état, représentées par le premier bit :

- 1xx = Information – La requête a été reçue et continue à être traitée.
- 2xx = Succès – L'action a été reçue avec succès, comprise et acceptée.
- 3xx = Redirection – Une autre action doit être menée afin de valider la requête.
- 4xx = Erreur du client – La requête contient une syntaxe erronée ou ne peut pas être traitée par ce serveur.
- 5xx = Erreur du serveur – Le serveur n'a pas réussi à traiter une requête apparemment correcte.
- 6xx = Echec général – La requête ne peut être traitée par aucun serveur.

Dans un système SIP, on trouve deux types de composantes, les users agents (UAS, UAC) et un réseau de serveurs :

- L'UAS (User Agent Server) – Il représente l'agent de la partie appelée. C'est une application de type serveur qui contacte l'utilisateur lorsqu'une requête SIP est reçue. Et elle renvoie une réponse au nom de l'utilisateur.
- L'U.A.C (User Agent Client) – Il représente l'agent de la partie appelante. C'est une application de type client qui initie les requêtes.
- Le relais mandataire ou PS (Proxy Server), auquel est relié un terminal fixe ou mobile, agit à la fois comme un client et comme un serveur. Un tel serveur peut interpréter et modifier les messages qu'il reçoit avant de les retransmettre.
- Le RS (Redirect Server) – Il réalise simplement une association (mapping) d'adresses vers une ou plusieurs nouvelles adresses. (lorsqu'un client appelle un terminal mobile – redirection vers le PS le plus proche – ou en mode multicast – le message émis est redirigé vers toutes les sorties auxquelles sont reliés les destinataires). Notons qu'un Redirect Server est consulté par l'Uac comme un simple serveur et ne peut émettre de requêtes contrairement au Ps.

- Le LS (Location Server) – Il fournit la position courante des utilisateurs dont la communication traverse les Rs et PS auxquels il est rattaché. Cette fonction est assurée par le service de localisation.
- Le RG (Registrar) – C'est un serveur qui accepte les requêtes Register et offre également un service de localisation comme le LS. Chaque PS ou RS est généralement relié à un Registrar.

## **I.9 Les protocoles de transport [11]**

### **I.9.1 Le protocole RTP**

#### **I.9.1.1 Description générale de RTP**

RTP (Real time Transport Protocol) est un protocole qui a été développé par l'IETF afin de faciliter le transport temps réel de bout en bout des flots données audio et vidéo sur les réseaux IP, c'est à dire sur les réseaux de paquets.

RTP est un protocole qui se situe au niveau de l'application et qui utilise les protocoles sous-jacents de transport TCP ou UDP.

Mais l'utilisation de RTP se fait généralement au-dessus d'UDP ce qui permet d'atteindre plus facilement le temps réel.

Les applications temps réels comme la parole numérique ou la visioconférence constitue un véritable problème pour Internet.

Une application temps réel définit une présence d'une certaine qualité de service (QoS) que RTP ne garantit pas du fait qu'il fonctionne au niveau applicatif, de plus RTP est un protocole qui se trouve dans un environnement multipoint, donc on peut dire que RTP possède à sa charge, la gestion du temps réel, mais aussi l'administration de la session multipoint.

#### **I.9.1.2 Les fonctions de RTP**

Le protocole RTP, Real Time Transport Protocol, standardisé en 1996, a pour but d'organiser les paquets à l'entrée du réseau et de les contrôler à la sortie.

Ceci de façon à reformer les flux avec ses caractéristiques de départ. RTP est un protocole de bout en bout, volontairement incomplet et malléable pour s'adapter aux besoins des applications. Il sera intégré dans le noyau de l'application. RTP laisse la responsabilité du contrôle aux équipements d'extrémité. RTP est un protocole adapté aux applications présentant des propriétés temps réel. Il permet ainsi de :

- Mettre en place un séquençement des paquets par une numérotation et ce afin de permettre ainsi la détection des paquets perdus. Ceci est un point primordial dans la reconstitution des données. Mais, il faut savoir quand même que la perte d'un paquet n'est pas un gros problème si les paquets ne sont pas perdus en trop grands nombres. Cependant, il est très important de savoir quel est le paquet qui a été perdu afin de pouvoir pallier à cette perte.

Et ce par le remplacement par un paquet qui se compose d'une synthèse des paquets précédant et suivant.

- Identifier le contenu des données pour leurs associer un transport sécurisé.
- Reconstituer la base de temps des flux (horodatage des paquets : possibilité de resynchronisation des flux par le récepteur).
- L'identification de la source c'est à dire l'identification de l'expéditeur du paquet. Dans un multicast, l'identité de la source doit être connue et déterminée.
- Transporter les applications audio et vidéo dans des trames (avec des dimensions qui sont dépendantes des codecs qui effectuent la numérisation). Ces trames sont incluses dans des paquets afin d'être transportées et doivent de ce fait être récupérées facilement au moment de la phase de dépaquetisation afin que l'application soit décodée correctement. En revanche, ce protocole présente quelques limites. En effet, il ne procure pas de :
  - Réserve de ressources sur le réseau.
  - Fiabilité des échanges : puisqu'il n'assure pas la retransmission automatique et régulation automatique du débit.
  - Garantie dans le délai de livraison et dans la continuité du flux temps réel.

### **I.9.1.3 Avantages et inconvénients**

Le protocole RTP permet de reconstituer la base de temps des différents flux multimédia (audio, vidéo, etc.), de détecter les pertes de paquets, et d'identifier le contenu des paquets pour leur transmission sécurisée. Par contre, il ne permet pas de réserver des ressources dans le réseau ou d'apporter une fiabilité dans le réseau. Ainsi il ne garantit pas le délai de livraison.

## **I.9.2 Le protocole RTCP**

### **I.9.2.1 Description générale de RTCP**

Le protocole RTCP (Real-time Transport Control Protocol) est fondé sur la transmission périodique de paquets de contrôle à tous les participants d'une session. C'est le protocole UDP qui permet le multiplexage des paquets de données RTP et des paquets de contrôle RTCP. Le protocole RTP utilise le protocole RTCP, qui transporte les informations supplémentaires suivantes pour la gestion de la session :

- Les récepteurs utilisent RTCP pour renvoyer vers les émetteurs un rapport sur la QoS. Ces rapports comprennent le nombre de paquets perdus, le paramètre indiquant la variance d'une et le délai aller-retour. Ces informations permettent à la source de s'adapter, par exemple, de modifier le niveau de compression pour maintenir une QoS.
- Une synchronisation supplémentaire entre les médias. Les applications multimédias sont souvent transportées par des flots distincts. Par exemple, la voix, l'image ou même des applications numérisées sur plusieurs niveaux hiérarchiques peuvent voir les flots gérés suivre des chemins différents.
- L'identification car en effet, les paquets RTCP contiennent des informations d'adresses, comme l'adresse d'un message électronique, un numéro de téléphone ou le nom d'un participant à une conférence téléphonique.
- Le contrôle de la session, car RTCP permet aux participants d'indiquer leur départ d'une conférence téléphonique (paquet Bye de RTCP) ou simplement de fournir une indication sur leur comportement. Le protocole RTCP demande aux participants de la session d'envoyer périodiquement les informations citées ci-dessus. La périodicité est calculée en fonction du nombre de participants de l'application.

### **I.9.2.2 Point fort et limite du protocole RTCP**

Le protocole de RTCP est adapté pour la transmission de données temps réel. Il permet d'effectuer un contrôle permanent sur une session et ces participants. Par contre, il fonctionne en stratégie bout à bout. Et il ne peut pas contrôler l'élément principal de la communication.

## **I.10 Conclusion**

Dans ce chapitre, nous avons vu la VoIP et son principe de fonctionnement, et qu'elle est la solution la plus rentable pour mener des conversations. Actuellement, il est clair que

cette technologie continuera d'évoluer que ce soit en termes d'intégration, de fiabilité ou de coût. Nous avons vu les deux modèles réseaux OSI qui un modèle de référence et le TCP/IP.

Nous avons vu aussi que la VoIP a plusieurs standards avec leurs propres caractéristiques. Et dans le prochain chapitre, on va débiter notre application Android coté client qui va être testé avec un serveur SIP pour avoir à la fin une application VoIP de bout en bout.

# CHAPITRE

# II

Notre application client de la VoIP

## II.1 Pourquoi une application mobile ?

La population mondiale continue d'augmenter et le nombre d'abonnés actifs aux téléphones augmente également. Selon les dernières données du TeleGeography Institute, à la fin de 2009, il y avait 4.61 milliards d'abonnés mobiles actifs, contre 6.97 milliards il y a cinq ans et presque 8 milliards à la fin de 2019.

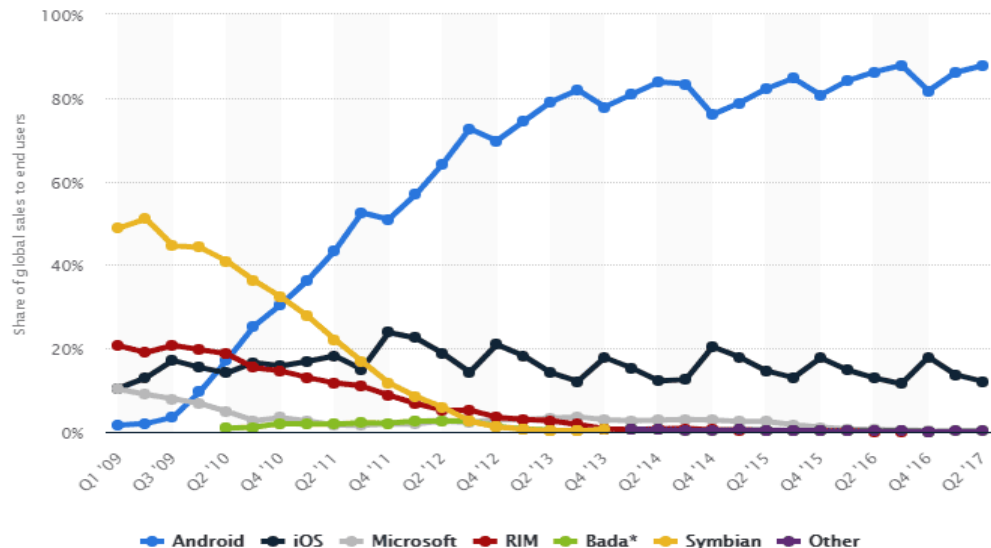
L'institut prévoit que d'ici la fin de 2026, le nombre d'abonnés atteindra les neuf milliards et que le taux de croissance annuel sera 1.9% pour les 7 prochaines années.

## II.2 Les systèmes d'exploitation mobiles

On trouve que seulement 6 entreprises représentent 99% du marché mondial des systèmes d'exploitation :

- **ANDROID** : c'est le système d'exploitation de Google, on le trouve dans la majorité des smartphones et tablettes.
- **IOS** : le système d'exploitation d'Apple, on le trouve dans les Iphone et les Ipad.
- **WINDOWS** : c'est le système d'exploitation de la même entreprise américaine du système d'exploitation des ordinateurs.
- **BLACKBERRY OS** : développé par BlackBerry qu'on trouve dans les smartphones BlackBerry.
- **SYMBIAN OS** : il est parmi les premiers systèmes d'exploitation. C'était l'OS des téléphones Nokia et Motorola.
- **TIZEN OS** : c'est l'ancien système d'exploitation des propriétaires de téléphones Samsung.





© Statista 2017

**Figure II.1 : Evolution des parts de marché des OS pour téléphones de 2009 à 2017 dans le monde [12]**

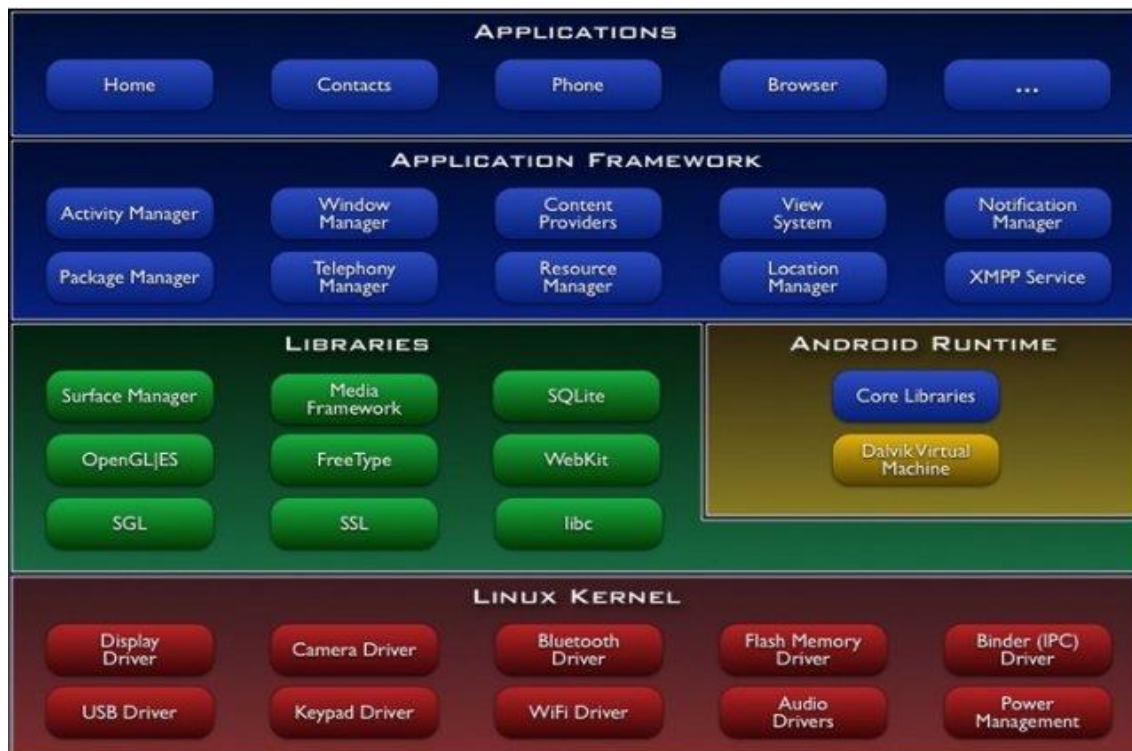
### II.3 Qu'est ce qu'un Android ?

Android est un système d'exploitation pour mobile comme « Windows » ou « Linux », basé sur Kernel Linux et un grand nombre de constructeurs le propose sur leurs appareils telles que Samsung, Motorola, LG, Acer, Sony et bien plus. C'est un système d'exploitation open source qui utilise le noyau Linux. Ce logiciel connaît une forte croissance sur le marché mondial des OS mobiles et le développement de ces applications est en Java ou Kotlin.

En plus, c'est le système d'exploitation qui couvre la majorité des smartphones aujourd'hui. La motivation qui nous a laissé choisir Android :

- Un grand nombre de constructeurs proposent Android sur leurs appareils mobiles.
- Plates forme gratuite.
- Open source.
- Personnalisable à l'infini.
- La communauté Android est la plus forte.

## II.4 Architecture Android [13]



**Figure II.2 : Architecture Android [14]**

On sait qu'Android est basé sur Kernel Linux 2.6.xx, au dessus de cette couche, on trouve le « Hardware Abstraction Layer » qui sépare la plateforme logique du matériel et au dessus, on a « les librairies » C/C++ que certain nombre de composants du système l'utilisent, après il y a « l'Android Runtime » qui contient la machine virtuelle et les librairies cœurs du framework. Il nous reste les deux dernières couches :

« Application framework » qui se trouve au dessus d'Android Runtime et Librairies, c'est cette couche qui permet au développeur de créer des applications. Et la dernière couche « Applications » où on trouve les applications.

Nous allons commencer par la partie la plus proche des utilisateurs d'une manière descendante :

### II.4.1 La couche Applications

C'est un ensemble de programme écrit généralement en Java et on a aussi le langage Kotlin qui est apparu en 2017 et devenu le langage de programmation officiel et recommandé par Google, cette couche contient l'application home, contacts, browser et nos applications.

Une application Android peut être composée des éléments suivants :

- ✓ **Activité « Activity »** : une interface graphique « une vue » (équivalent à une fenêtre ou boîte de dialogue en java).
- ✓ **Service** : un programme qui s'exécute en arrière plan.
- ✓ **Fournisseurs de contenus « Provider »** : un mécanisme (une classe) permettant le partage d'informations entre applications.
- ✓ **Widgets** : une vue accrochée au bureau d'Android.
- ✓ **Notifications** : permet de notifier l'utilisateur de la survenue d'événements.
- ✓ **Intention « Intent »** : un mécanisme (une classe) peut être utilisé pour transiter des données entre deux activités.
- ✓ **Récepteurs d'Intents «BroadcastReceiver»**.

#### II.4.2 La couche Application Framework

Cette couche représente un ensemble d'outils (programme système au niveau du smart phone, outil de développement au niveau du PC, ensemble de class Java, etc...) qui se limitent à la gestion des différentes parties de l'application (activity, ressource, content, package, Telephony, ...) de la couche application.

On donne un exemple des types de gestionnaire :

##### a) Activity manager vs Window Manager :

Pour comprendre le sens du nom gestion, prenons l'exemple de Window Manager et Activity Manager :

- **Le gestionnaire des fenêtres** : est un « programme système » responsable de la partie affichage/modification des pixels et de l'organisation de l'écran. Il permet d'allouer la surface, et décider où les applications vont être placées et superposées.

- **Le gestionnaire d'activités** : est un « programme système » responsable de la gestion des différents états d'une activité : (démarrage, exécution, pause, stop, destruction).

Ex : sauvegarde/restauration du contexte

##### b) Gestionnaire de téléphone

Cette fois, la gestion signifie qu'il existe une classe Java développée par la société Android qui s'appelle `TelephonyManager`. Cette classe contient des méthodes qui permettent de gérer les appels téléphoniques :

- récupérer l'état de l'appel : sonné, occupé, raccroché, etc.
- récupérer le numéro de série de la carte SIM.
- Récupérer la puissance du signal.
- Etc.

### c) Gestionnaire des ressources

En Android, il faut organiser les ressources du projet dans un ensemble de dossiers et sous dossiers bien déterminé.

Après avoir placé et organisé les fichiers, il faut donner un identificateur (ID) pour chaque ressource (fichier).

## II.4.3 La couche d'exécution Android

Le composant principal dans le runtime android est la machine virtuelle `dalvik`. L'environnement d'exécution android a été conçu spécifiquement pour Android pour répondre aux besoins de l'exécution dans un environnement embarqué où on a batterie limitée, mémoire limitée, processeur limité. La machine virtuelle `dalvik` exécute quelque chose appelé fichiers DEX, point D-E-X. Et ce sont des codecs d'octets qui sont les résultats de la conversion, à buildtime, `.class` et `.jar`. Donc ces fichiers, lorsqu'ils sont convertis en `.dex`, deviennent un byte code beaucoup plus efficace qui peut fonctionner très bien sur les petits processeurs. Ils utilisent la mémoire très efficacement, les structures de données sont conçues pour être partagées entre les processus chaque fois que possible, et il utilise un interpréteur de code octet hautement optimisé par processeur. Le résultat final est qu'il est possible d'avoir plusieurs instances de la machine virtuelle `dalvik` fonctionnant sur l'appareil en même temps, une dans chacun de plusieurs processus. Le niveau suivant est celui des bibliothèques de base. C'est en bleu, ce qui signifie qu'il est écrit dans le langage de programmation java. Et la bibliothèque de base contient toutes les classes de collection, tous les utilitaires et outils qu'on veut utiliser.

#### II.4.4 Libraries

Tout ce qui est ici en vert est écrit en C et C++. C'est à ce niveau où beaucoup de la puissance de base de la plate-forme Android vient. On va décrire certains de ces éléments. On va commencer par le gestionnaire de surface. Le responsable de la surface est responsable de la composition des différentes surfaces de dessin sur l'écran. Ainsi, c'est le gestionnaire de surface qui est responsable de prendre différentes fenêtres qui sont détenues par différentes applications qui s'exécutent dans différents processus et tous dessiner à différents moments et de s'assurer que les pixels finissent sur l'écran quand ils sont censés. Ci-dessous, nous avons deux boîtes : OpenGL ES et SGL, et ces deux constituent le noyau de nos bibliothèques graphiques. OpenGL ES est une bibliothèque 3D. Nous avons une implémentation logicielle qui peut être accélérée par le matériel si l'appareil a une puce 3D dessus. Les graphiques SGL sont pour les graphiques 2D, sur laquelle la plupart de nos dessins applicatifs sont basés. Une chose intéressante à propos de la plate-forme graphique Android est que vous pouvez combiner des graphiques 3D et 2D dans la même application. Passons maintenant au cadre médiatique. Le cadre média a été fourni par PacketVideo, l'un des membres de l'Alliance du combiné ouvert, et qui contient tous les codecs qui composent le noyau de l'expérience médiatique. Alors là-dedans, on trouve MPEG-4, H.264, MP3, AAC, tous les codecs audio et vidéo dont on a besoin pour construire une expérience multimédia riche. On utilise FreeType pour rendre nos polices. Nous avons une implémentation de SQLite. Nous l'utilisons comme noyau de la plupart de notre stockage de données. Nous avons WebKit qui est le moteur de navigateur open source. C'est ce qu'on utilise comme noyau de notre navigateur. C'est le même navigateur qui alimente Safari à partir d'Apple. Et on a travaillé avec ce moteur pour le rendre bien sur les petits écrans et sur les appareils mobiles.

#### II.4.5 La couche noyaux linux

Le noyau est l'élément du système d'exploitation qui permet d'exploiter le matériel de votre smart phone.

- ❖ Par exemple, le driver WiFi permet de contrôler l'antenne WiFi de votre smart phone.
- ❖ Quand Android veut activer le WiFi, on peut imaginer qu'il utilise la fonction « `ActiverWiFi()` ».
- ❖ Cette fonction est implémentée par le constructeur de la carte WiFi.

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

## II.5 Installation et configuration du logiciel

### II.5.1 Installation

- Il faut tout d'abord installer le JDK (Java Development Kit), avoir aussi 2 Go de mémoire RAM au minimum et un espace disque plus de 1.5 Go.
- Au niveau du processeur, l'émulation ne peut se faire que sur un cœur du processeur.
- Maintenant, on passe au téléchargement d'un fichier contenant un ensemble d'outils indispensable pour développer nos applications Android (Android Studio et le SDK Android).
- Une fois le fichier téléchargé, on lance l'installation.

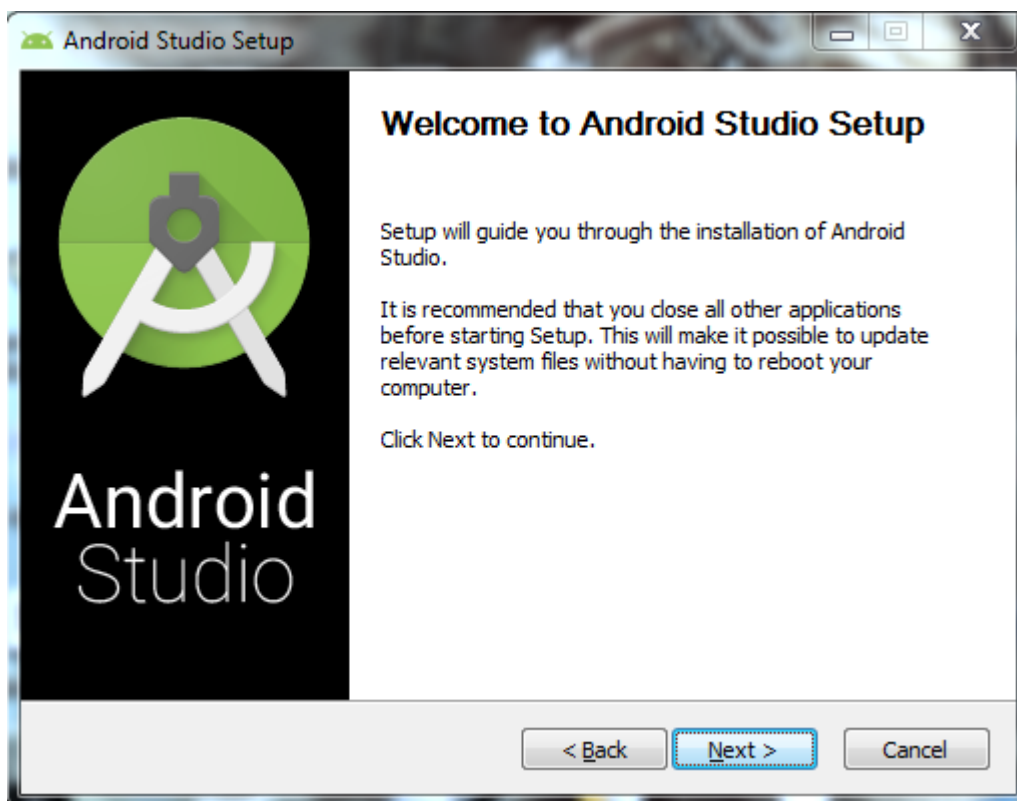
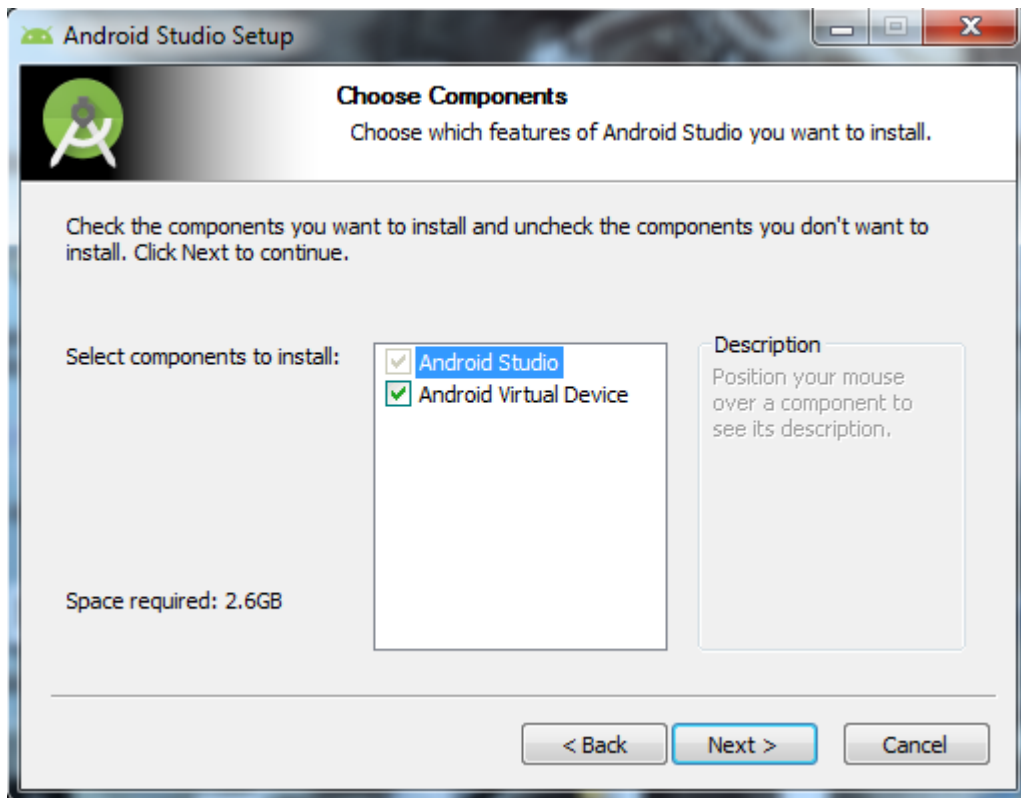
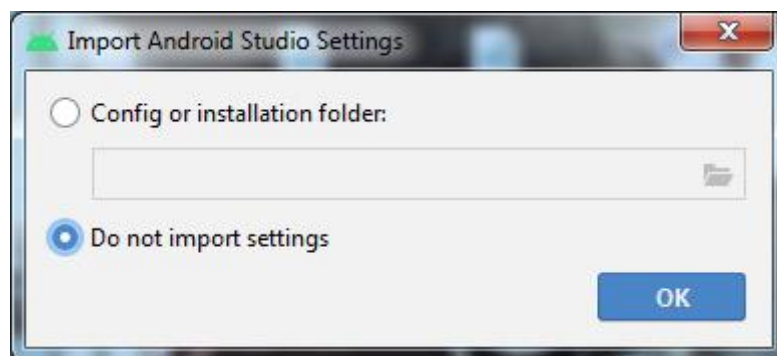


Figure II.3 : Début d'installation



**Figure II.4 : Installation du Android studio et SDK Android**

- On sélectionne les composants à installer (Android Virtuel Device pour lancer l'émulateur), puis on appuie sur « Next ».

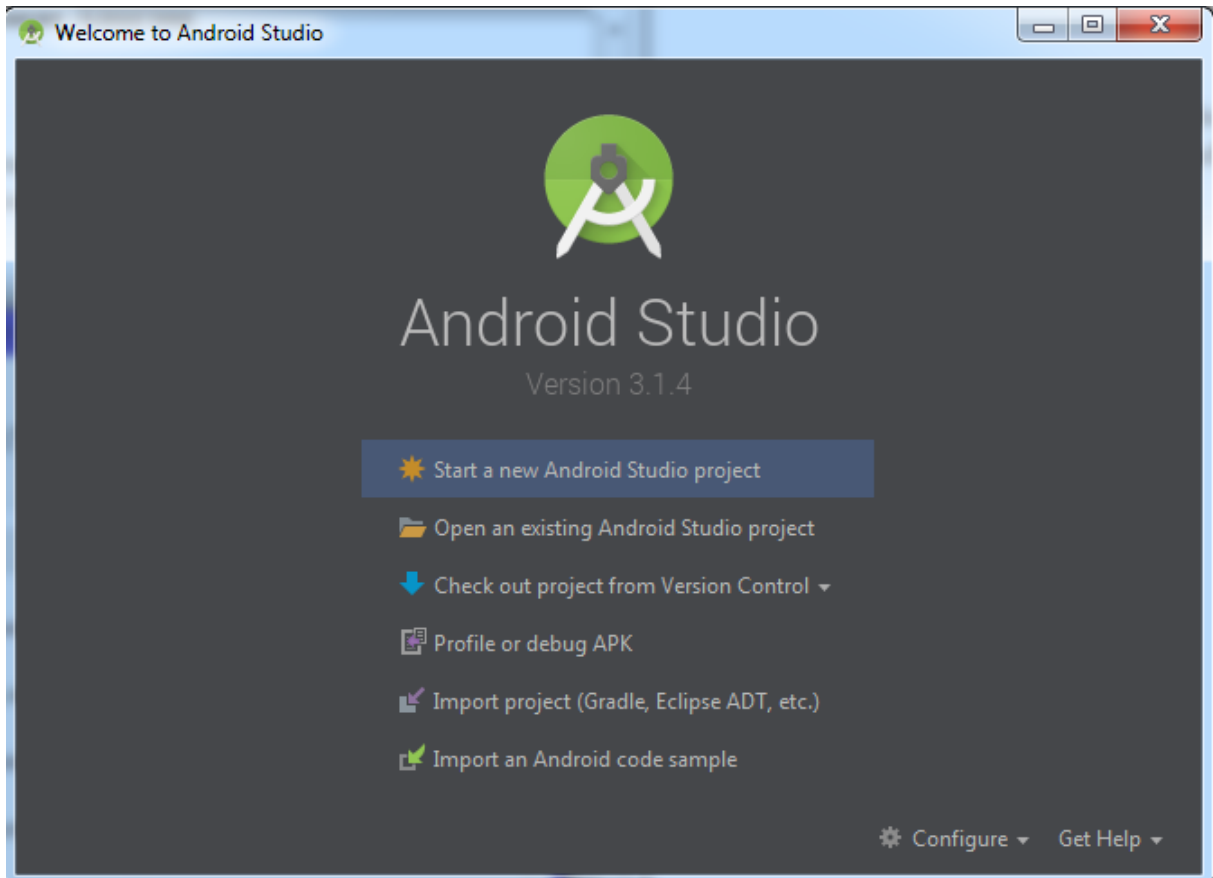


**Figure II.5 : Confirmation de l'installation**

- Au premier lancement, une boîte de dialogue va s'afficher demandant s'il existe déjà une version d'Android Studio installée précédemment. Sinon, on sélectionne la deuxième option.

## II.5.2 Configuration

Maintenant que l'Android Studio est installé sur notre PC, on passe à l'étape de configuration pour le bon fonctionnement de logiciel.



**Figure II.6 : Android Studio**

- Au niveau de la configuration, on clique sur Configure puis SDK Manager et on vérifie que le SDK a été installé.



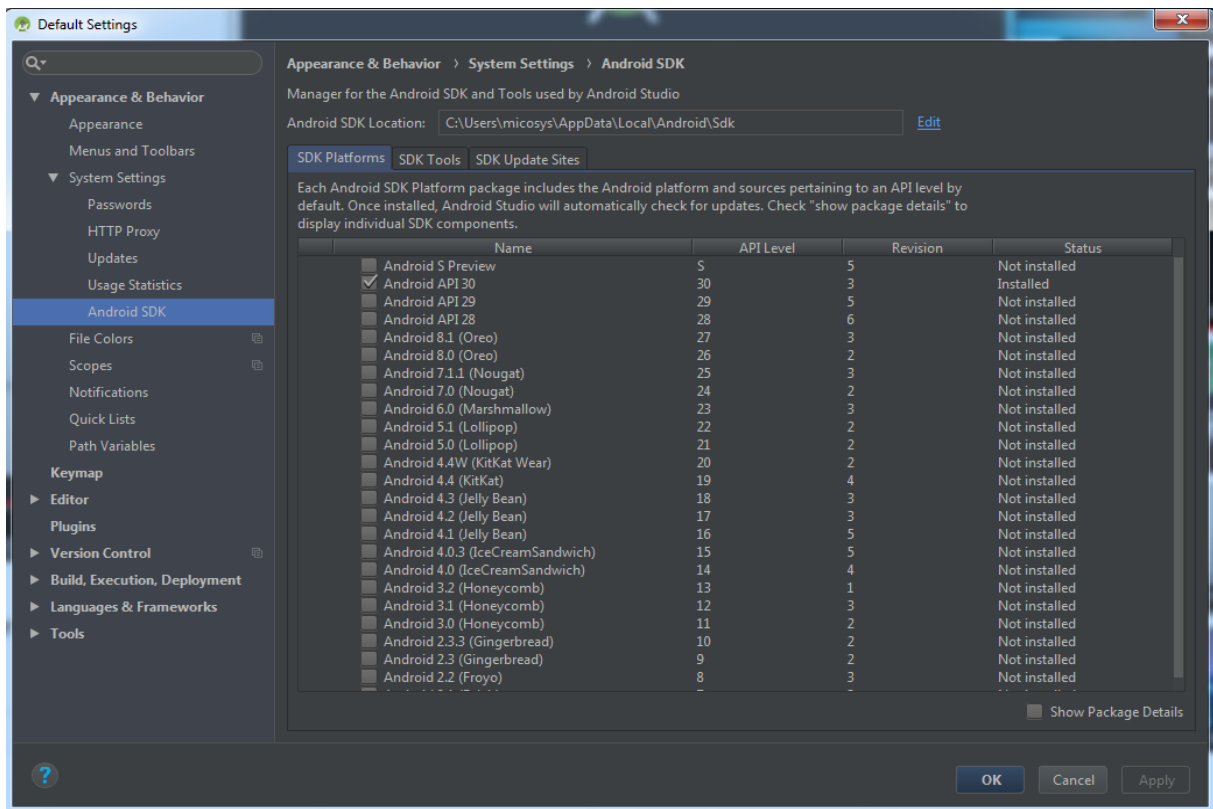


Figure II.7 : Les plateformes SDK

- On choisit les plateformes qu'on veut cibler dans SDK platforms.

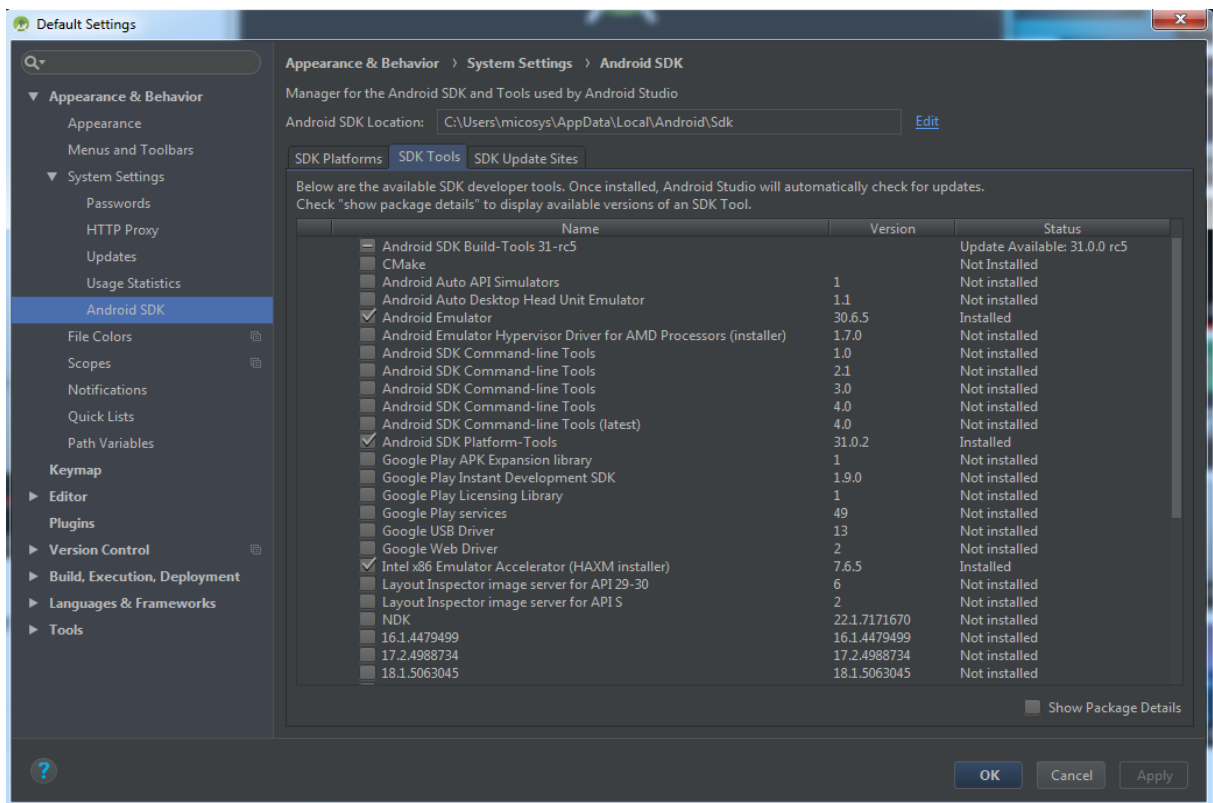


Figure II.8 : Les outils SDK

- Au niveau du SDK Tools, on a téléchargé les outils nécessaires à la construction de notre application, puis l'émulateur, puis les outils nécessaires à la communication avec l'émulateur ou le téléphone et on va utiliser des images x86. Donc, il nous faudra installer l'accélérateur pour l'émulateur sinon, il sera trop long.

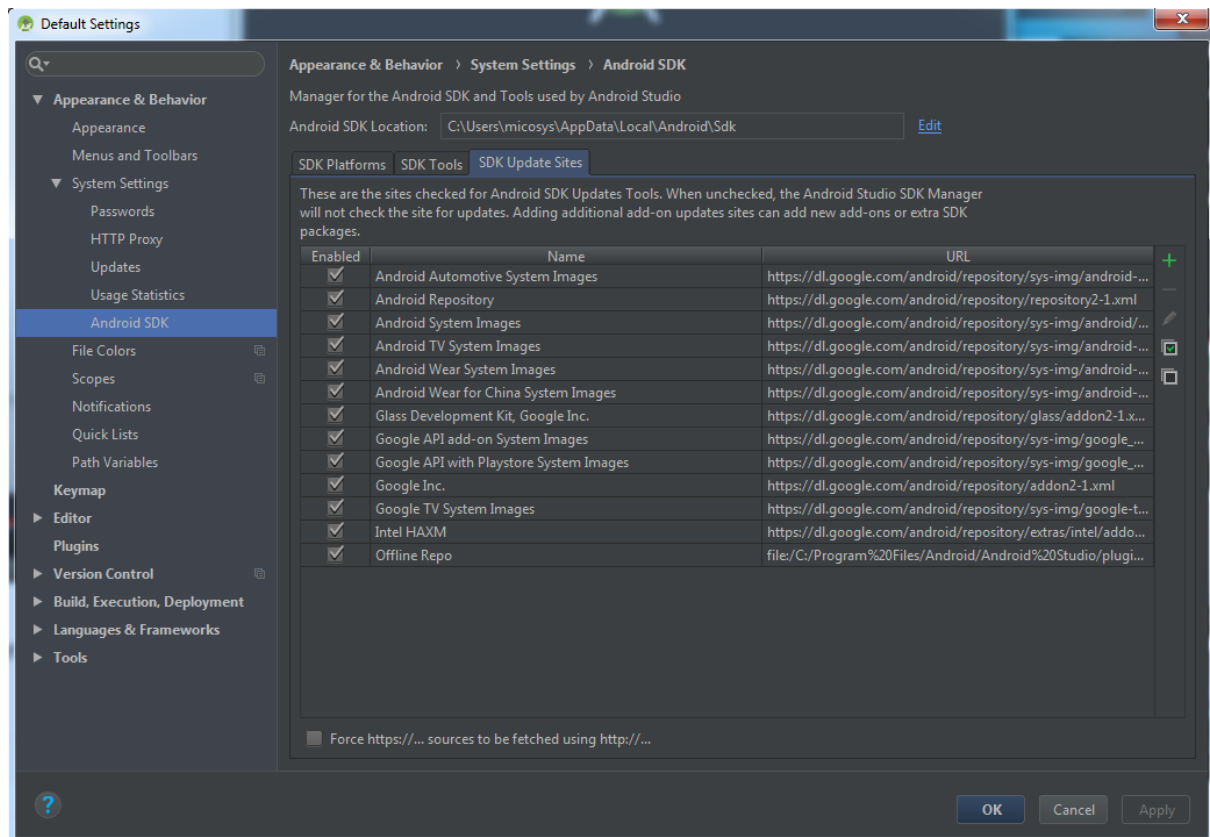


Figure II.9 : Les sites de mis à jour

- Au niveau des sites de mis à jour, par défaut on ne change rien.

## II.6 Notre application

### II.6.1 Principe

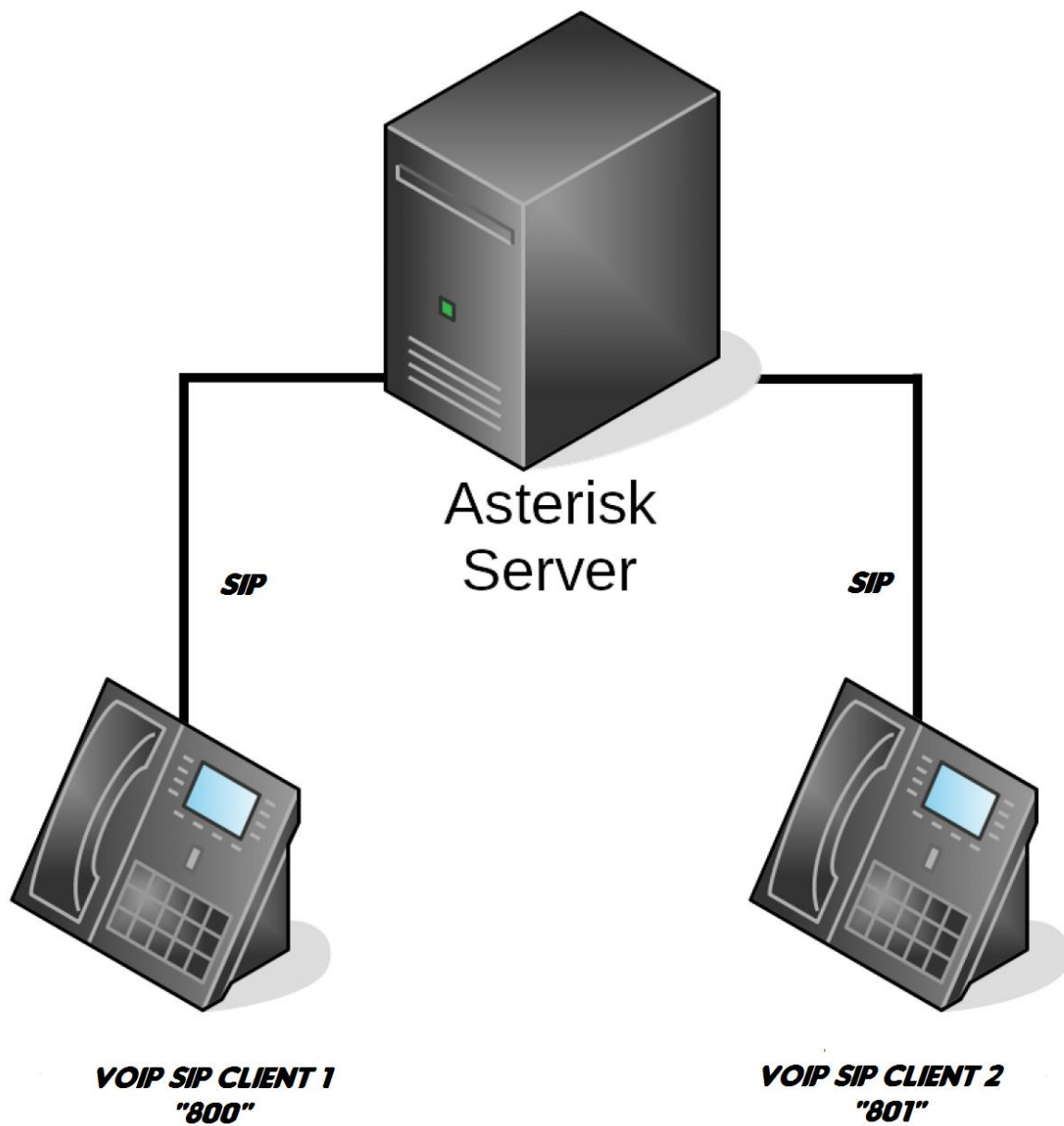


Figure II.10 : Principe de fonctionnement

Le principe de fonctionnement est qu'un client 1 par exemple peut communiquer facilement avec un autre client 2 via le serveur Asterisk.

## II.6.2 L'interface de l'application

On a développé une interface très simple où on fait la configuration de notre application et la numérotation vers le serveur.

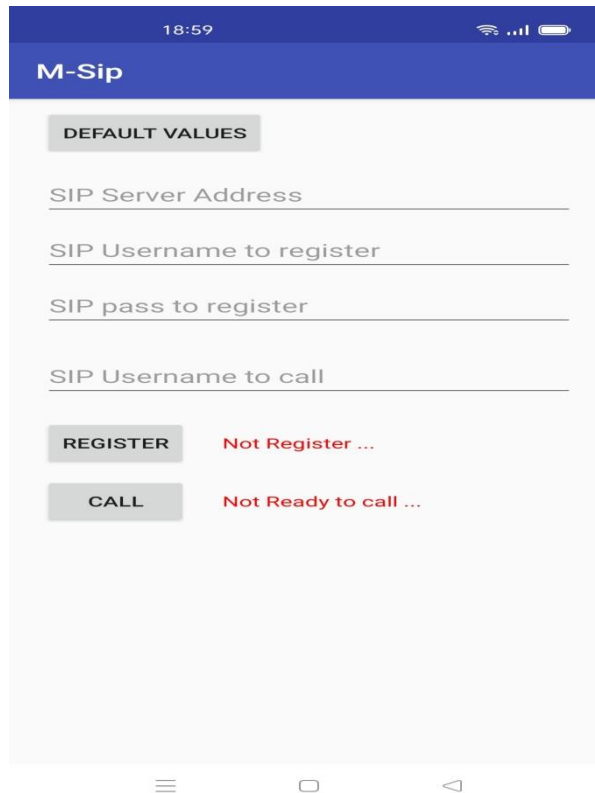
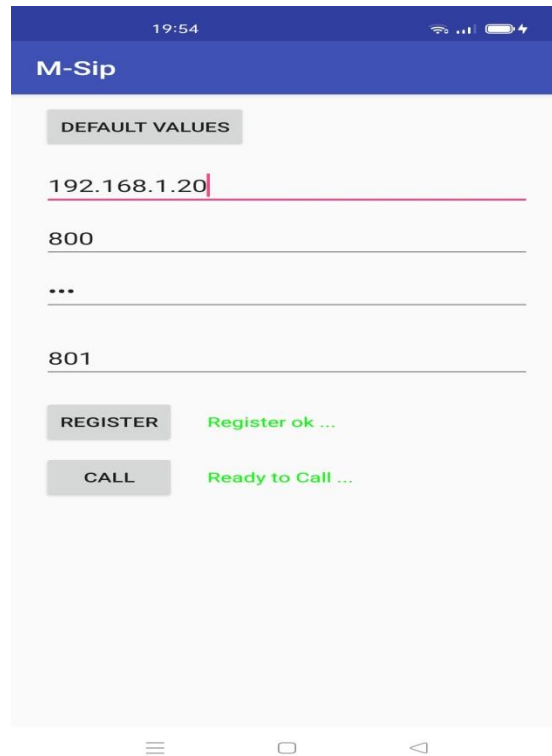


Figure II.11 : Interface de l'application

## II.6.3 La configuration du client

Comme on peut voir dans la figure précédente, pour que notre application fonctionne, il faut remplir certains paramètres qu'on trouve dans notre interface.

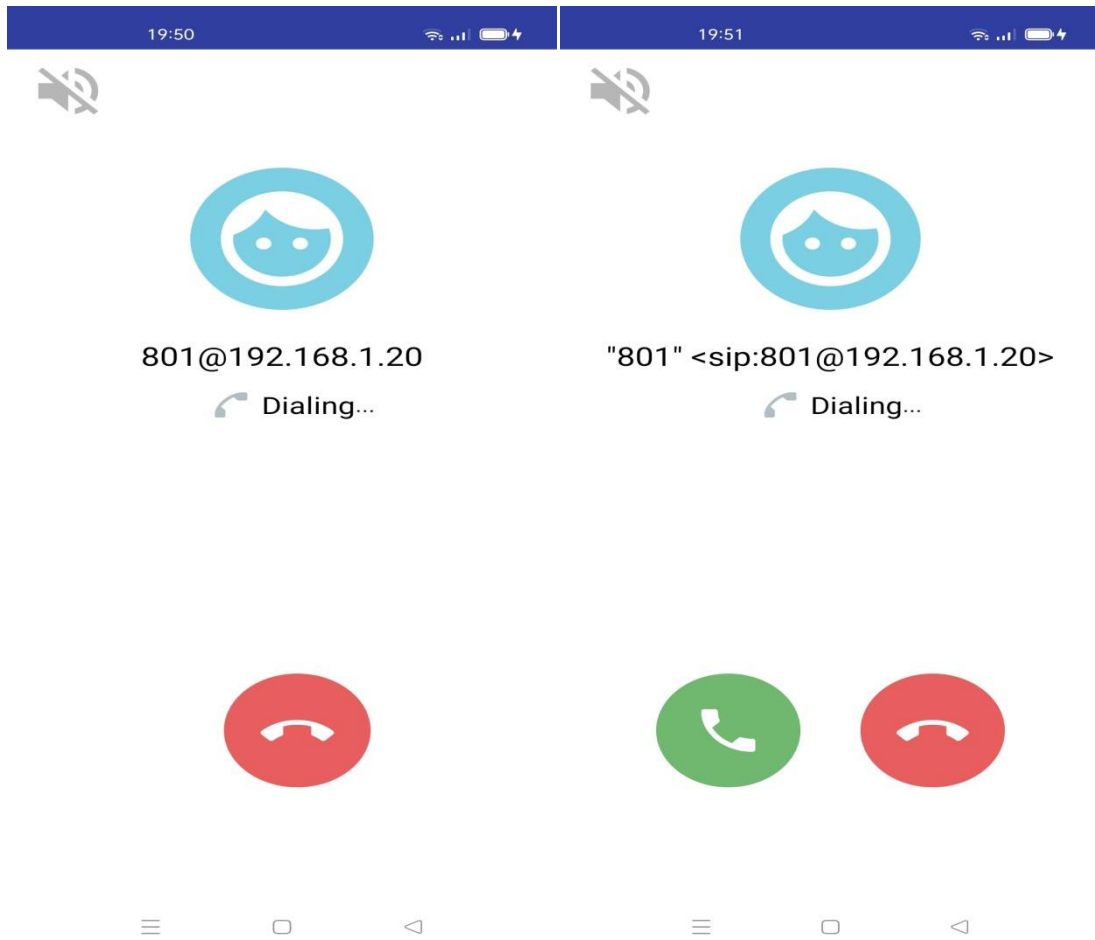
- SIP Server Address : L'adresse IP de notre serveur.
- SIP Username to register : Le nom du client.
- SIP pass to register : Le mot de passe pour accéder au serveur.
- SIP Username to call : Le numéro du client qu'on veut appeler.



**Figure II.12 : Configuration du client**

## **II.7 Résultats**

Pour voir si tout fonctionne bien, on a fait un appel entrant et un appel sortant avec le serveur Asterisk.



**Figure II.13 : Appel Sortant**

**Figure II.14 : Appel Entrant**

- On remarque que les deux appels (entrant et sortant) fonctionnent bien.
- On n'a pas voulu changer les ID pour savoir comment fonctionne la communication avec le serveur.

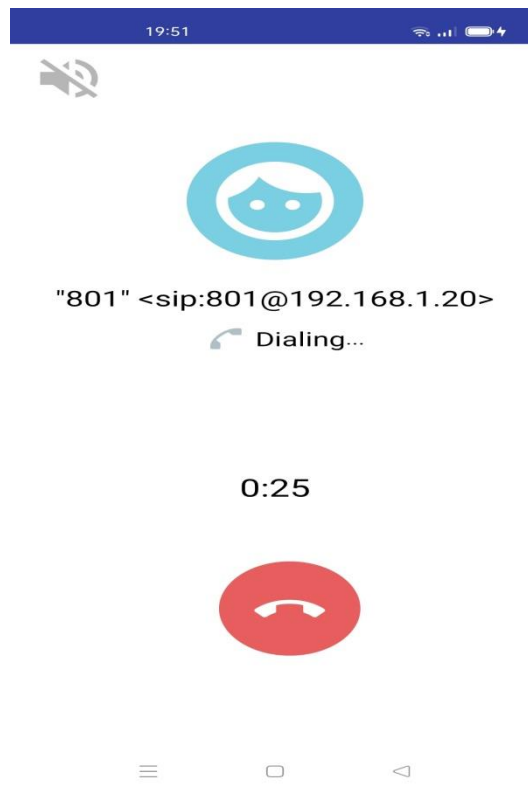


Figure II.15 : Durée d'appel

```

Connected to Asterisk 16.18.0 currently running on arsrlyad-VirtualBox (pid = 971)
arsrlyad-VirtualBox*CLI> sip show peers
Name/username      Host                Dyn Forcerport Comedia   ACL Port   Status   Description
800/800            192.168.1.21       D Yes      Yes       43157     OK (52 ms)
801/801            192.168.1.5        D Yes      Yes       60246     OK (42 ms)

2 sip peers [Monitored: 2 online, 0 offline Unmonitored: 0 online, 0 offline]
== Using SIP VIDEO CoS mark 6
== Using SIP RTP CoS mark 5
> 0x7f29ec021500 -- Strict RTP learning after remote address set to: 192.168.1.21:4010
-- Executing [801@CLIENT_IN:1] Dial("SIP/800-00000010", "SIP/801,20") in new stack
== Using SIP VIDEO CoS mark 6
== Using SIP RTP CoS mark 5
-- Called SIP/801
-- SIP/801-00000011 is ringing
-- SIP/801-00000011 is ringing
> 0x7f29e40017c0 -- Strict RTP learning after remote address set to: 192.168.1.5:4012
-- SIP/801-00000011 answered SIP/800-00000010
-- Channel SIP/801-00000011 joined 'simple_bridge' basic-bridge <2ab8e442-9909-4c3f-9d37-da2103bba71d>
-- Channel SIP/800-00000010 joined 'simple_bridge' basic-bridge <2ab8e442-9909-4c3f-9d37-da2103bba71d>
> Bridge 2ab8e442-9909-4c3f-9d37-da2103bba71d: switching from simple_bridge technology to native_rtp
> Remotely bridged 'SIP/800-00000010' and 'SIP/801-00000011' - media will flow directly between them
> 0x7f29e40017c0 -- Strict RTP switching to RTP target address 192.168.1.5:4012 as source
> 0x7f29ec021500 -- Strict RTP switching to RTP target address 192.168.1.21:4010 as source
> 0x7f29ec021500 -- Strict RTP learning complete - Locking on source address 192.168.1.21:4010
-- Channel SIP/800-00000010 left 'native_rtp' basic-bridge <2ab8e442-9909-4c3f-9d37-da2103bba71d>
== Spawn extension (CLIENT_IN, 801, 1) exited non-zero on 'SIP/800-00000010'
-- Channel SIP/801-00000011 left 'native_rtp' basic-bridge <2ab8e442-9909-4c3f-9d37-da2103bba71d>
arsrlyad-VirtualBox*CLI>

```

Figure II.16 : Etat du serveur durant l'appel

- On a fait un test de 25 secondes pour s'assurer que tout marche bien.

## **II.8 Conclusion**

Dans ce chapitre, on a vu la motivation qui nous a poussé à développer une application mobile sous le système d'exploitation Android avec le logiciel studio Android, et nous avons fait un test pour vérifier le bon fonctionnement de notre application.



# **Conclusion**

## **Générale**

## Conclusion générale

Dans ce mémoire, nous avons vu que le protocole IP est devenu un standard unique permettant l'interopérabilité des réseaux mondialisés. D'aucuns pensent d'ailleurs que l'intégration de la voix sur IP n'est qu'une étape vers le tout IP.

Nous avons vu aussi que la VoIP est une technologie émergente et que toute entreprise qui veut être compétitive et moderne aujourd'hui se tournera vers elle vu les avantages qu'elle présente. C'est la solution la plus rentable pour effectuer des conversations. Même si elle n'a pas encore de standard unique, chaque standard possède ses propres caractéristiques pour garantir une bonne qualité de service. Et malgré que la normalisation n'ait pas atteint la maturité suffisante pour sa généralisation au niveau des réseaux IP, il n'est pas dangereux de miser sur ces standards vu qu'ils ont été acceptés par l'ensemble de la communauté de la téléphonie. Cette technologie vise principalement à améliorer le cadre de travail des employés de l'entreprise en libérant l'utilisateur du lieu d'implantation du poste téléphonique.

En Algérie, cette technologie n'est pas encore très bien développée vu l'absence des fournisseurs de VoIP. Cependant, il est possible de déployer quelques applications de cette technologie au sein des entreprises multi-sites ce qui nous permettra de migrer les communications du réseau RTC vers le réseau IP. C'est la raison qui nous a poussé à développer notre application vu que le marché de la VoIP bien que jeune encore se développe à une vitesse fulgurante.

---

## Bibliographie

- [1]..... DÉON, Sébastien. *VoIP et ToIP Asterisk: La téléphonie sur IP (conception, installation, configuration, déploiement...)*. Editions ENI, 2007 (Dernière consultation le 11/05/2021)
- [2]..... HE, D.D. et ZHAN, S. Design and realization of VoIP based on android OS. *Modern Electronics Technique*, 2011, vol. 34, no 6, p. 28-31. (Dernière consultation le 11/05/2021)
- [3]..... A. Sinaeepourfard and H. Mohamed Hussain, “Comparison of VoIP and PSTN services by statistical analysis,” in Proceedings - 2011 IEEE Student Conference on Research and Development, SCORED 2011, 2011, pp. 459–461. (Dernière consultation le 24/06/2021)
- [4]..... Zahid Anwar, William Yurcik, Ralph E. Johnson, Munawar Hafiz and Roy H. Campbell, (2006) “Multiple Design Patterns for Voice over IP (VoIP) Security”, Performance, Computing, and Communications Conference (IPCCC), pp 485-492. (Dernière consultation le 24/06/2021)
- [5].....<https://cisco.goffinet.org/ccna/ipv4/en-tetes-ipv4-et-ipv6/> (Dernière consultation le 24/06/2021)
- [6]..... BIBI TRIKI Chakib, BERRAHIL Zeyneb. Développement d’une application client de la voix sur IP VOIP. 2017. Thèse de master. Université Abou Bakr Belkaid. (Dernière consultation le 11/05/2021)
- [7].....<https://www.alertlogic.com/blog/where-is-ipv1-2-3-and-5/> (Dernière consultation le 11/05/2021)
- [8].....BOUZAIDA, Rebha. Etude et Mise en place d’une Solution VOIP Sécurisée. 2011. Thèse de doctorat. Université Virtuelle de Tunis. (Dernière consultation le 11/05/2021)
- [9] [10].... <https://www.frameip.com/voip/> (Dernière consultation le 11/05/2021)

- [11].....WEI, X., SELLAL, K., et BOUSLIMANI, Y. Security implementation for a VoIP server. In : *2012 International Conference on Computer Science and Service System*. IEEE, 2012. p. 983-985 (Dernière consultation le 11/05/2021)
- [12].....<https://www.leptidigital.fr/wp-content/uploads/2017/12/parts-de-marche-os-mobile-2009-2017.png> (Dernière consultation le 24/06/2021)
- [13]..... Mousaab Bada. Cours : DEVELOPPEMENT DES APPLICATIONS MOBILE. Université de Batna 2. (Dernière consultation le 03/07/2021)
- [14].....<http://www.androidphones.fr/local/cache-vignettes/L640xH460/system-architecture-f99db.jpg> (Dernière consultation le 24/06/2021)