

N°dordre: .....



UNIVERSITE DJILLALI LIABES DE SIDI BEL ABBES

FACULTE DES SCIENCES EXACTES

DEPARTEMENT D INFORMATIQUE

Domaine : LMD Mathematiques informatique

Specialite : Licence Academique

Parcours: Systemes informatiques  
(SI)

## MEMOIRE DE LICENCE

### SYSTÈME DE GESTION DES SIGNALEMENTS DES CITOYEN

Par

M<sup>r</sup> BENHELLAL MEHDI SALIM

M<sup>r</sup> BENSLIMANE MOHAMMED

Memoire soutenue le 11 Juin 2020 sous la direction de :

Dr. ENSEIGANT1 UDL SBA (President)

Dr. ENSEIGANT2 UDL SBA (Membre)

Dr. ENSEIGANT3 UDL SBA (Encadreur)

Annee Universitaire : 2019 - 2020

*À qui vous voulez...*

# REMERCIEMENTS

**L**A réalisation de ce mémoire a été possible grâce au concours de plusieurs personnes à qui on voudrais témoigner toute notre gratitude.

Un grand merci à Mr BOUCLI HACEN pour sa patience, sa disponibilité et surtout ses judicieux conseils

On remercie également toute l'équipe pédagogique de l'Université Djillali Liabès Sidi-Bel-Abbès

Enfin,nous aimerions remercier nos familles, amis pour leur soutien constant et leurs encouragements.

Lieu, le 24 mai 2020.

# TABLE DES MATIÈRES

TABLE DES MATIÈRES	iv
LISTE DES FIGURES	v
LISTE DES TABLEAUX	v
PRÉFACE	1
1 INTRODUCTION ET PROBLÉMATIQUE	2
1.1 INTRODUCTION . . . . .	3
1.2 PROBLÉMATIQUE . . . . .	3
1.3 OBJECTIFS . . . . .	3
CONCLUSION . . . . .	4
2 DÉVELOPPEMENT SOUS ANDROID	5
2.1 DESCRIPTION . . . . .	6
2.2 HISTORIQUE ET VERSIONS D'ANDROID . . . . .	6
2.3 FONCTIONNALITÉS D'ANDROID . . . . .	8
2.4 L'ARCHITECTURE D'ANDROID . . . . .	9
2.4.1 Application . . . . .	9
2.4.2 Framework et développement . . . . .	9
2.4.3 Bibliothèques . . . . .	10
2.4.4 Bibliothèques d'Android . . . . .	10
2.4.5 Android Runtime ( Moteur d'exécution ) . . . . .	11
2.4.6 Linux Kernel (Noyau Linux ) . . . . .	11
2.5 LA PHILOSOPHIE ET LES AVANTAGES D'ANDROID . . . . .	12
2.5.1 Open source . . . . .	12
2.5.2 Facile à développer . . . . .	12
2.5.3 Facile à vendre . . . . .	12
2.5.4 Google Play . . . . .	12
2.5.5 Flexible . . . . .	12
2.5.6 Complémentaire . . . . .	13
2.6 LES COMPOSANTS PRINCIPAUX D'UNE APPLICATION . . . . .	13
2.6.1 Activité ( Activity ) . . . . .	13
2.6.2 Services . . . . .	13
2.6.3 Recepteur de diffusion (Broadcast receiver) . . . . .	14
2.6.4 Fournisseurs de contenu (Content provider) . . . . .	14
2.6.5 Composants supplémentaires . . . . .	14
2.7 CYCLE DE VIE D'UNE APPLICATION ANDROID . . . . .	15
CONCLUSION . . . . .	16

CONCLUSION GÉNÉRALE	17
BIBLIOGRAPHIE	18

## LISTE DES FIGURES

2.1	Historique d'Android . . . . .	7
2.2	Les appareils d'Android de 2008 jusqu'à 2014 . . . . .	8
2.3	Les composant du système d'exploitation Android . . . . .	9
2.4	Logo Google Play . . . . .	12
2.5	Composants supplémentaires . . . . .	14
2.6	cycle de vie d'une Activité . . . . .	15

## LISTE DES TABLEAUX

# PRÉFACE, INTRODUCTION...

DANS les milieux industriels comme ...

L'objectif de cette thèse a été de ...

Nos contributions portent sur : ...

Le *premier chapitre* expose la problématique de la thèse.

Le *deuxième chapitre* présente en détail le modèle utilisé.

etc.

Cette thèse a fait l'objet de divers travaux écrits : ...

# INTRODUCTION ET PROBLÉMATIQUE

1

## SOMMAIRE

1.1 INTRODUCTION . . . . .	3
1.2 PROBLÉMATIQUE . . . . .	3
1.3 OBJECTIFS . . . . .	3
CONCLUSION . . . . .	4

## 1.1 INTRODUCTION

La commune est l'assise de la décentralisation, elle représente le cadre dans lequel s'exprime la volonté du peuple pour laquelle s'exerce le contrôle de l'action des pouvoirs publics, ainsi que le cadre de participation du citoyen à la gestion des affaires communales, consacrant à cet effet la démocratie participative. La participation des citoyens dans la gestion des collectivités locales est un aspect important de la vie publique.

## 1.2 PROBLÉMATIQUE

La communication continuelle entre les citoyens et la collectivité locale est nécessaire afin de résoudre leurs problèmes quotidiens. Ces problèmes peuvent être de plusieurs natures (fuite d'eau ou de gaz, nid de poule, détritus, problèmes d'éclairage public, feux de circulation défaillants, etc...) et nécessitent l'intervention et la coordination de plusieurs services de la Commune.

Ceci mène le citoyen à se poser plusieurs questions :

- Comment faire ? À qui dois-je m'adresser ?
- Quelle est l'autorité responsable de résoudre ce problème ?
- Quelles sont les procédures de la déclaration du problème ?

## 1.3 OBJECTIFS

L'objectif de la plateforme est de permettre à quiconque de signaler facilement un problème sans se soucier de l'autorité responsable de sa résolution. En outre cette plateforme permet aussi au citoyen de faire un suivi de la prise en charge du problème signalé. Afin de permettre au responsable, à l'équipe municipale, ou aux services techniques de la commune, d'intervenir plus rapidement lorsqu'une anomalie est rencontrée sur la voie publique ! La possibilité de faire remonter des informations géolocalisées en quelques secondes auprès de la municipalité représente un atout considérable dans la gestion d'une commune, favorise la démocratie participative et permet de mobiliser le côté citoyen des habitants qui peuvent signaler tout dysfonctionnement ou problème rencontré dans la ville : nid de poule, détritus, dégradation de mobilier urbain, problèmes d'éclairage, objets encombrants, etc. Depuis le site internet sur leur ordinateur, tablette, ou smartphone, ou directement depuis l'application mobile, les administrés peuvent signaler un problème en quelques clics et l'accompagner de photo et de l'adresse exacte de l'anomalie rencontrée dans la commune. Lorsque le signalement est envoyé depuis l'application smartphone, l'application citoyenne permet de géolocaliser directement le problème et récupère l'adresse exacte par GPS.

- L'utilisateur signale le problème rencontré dans sa ville.
- Le responsable et son équipe reçoivent le signalement.
- Le signalement est également transféré automatiquement à l'autorité concernée.
- Le signalement est pris en charge par les services techniques.
- Le statut de signalement est changé et l'utilisateur en est informé.



En plus de signaler au responsable le dysfonctionnement ou l'anomalie rencontrée très simplement, un système de personnalisation permet de faire atterrir directement le signalement au service technique concerné, c'est le responsable lui-même qui va définir les destinataires des signalements en fonction de la catégorie. Un système de changement de statut du signalement permet non seulement d'informer l'utilisateur de l'avancement du dossier, mais également aux services techniques en charge et au responsable, de savoir où en sont les interventions : une carte dynamique affiche des pointeurs localisant les signalements reçus et non traités afin de faciliter le travail des personnes en charge des réparations.

### CONCLUSION DU CHAPITRE

Notre plateforme permettra aux citoyens de facilement signaler les problèmes avec un gain considérable de temps .

# DÉVELOPPEMENT SOUS ANDROID 2

## SOMMAIRE

2.1	DESCRIPTION . . . . .	6
2.2	HISTORIQUE ET VERSIONS D'ANDROID . . . . .	6
2.3	FONCTIONNALITÉS D'ANDROID . . . . .	8
2.4	L'ARCHITECTURE D'ANDROID . . . . .	9
2.4.1	Application . . . . .	9
2.4.2	Framework et développement . . . . .	9
2.4.3	Bibliothèques . . . . .	10
2.4.4	Bibliothèques d'Android . . . . .	10
2.4.5	Android Runtime ( Moteur d'exécution ) . . . . .	11
2.4.6	Linux Kernel (Noyau Linux ) . . . . .	11
2.5	LA PHILOSOPHIE ET LES AVANTAGES D'ANDROID . . . . .	12
2.5.1	Open source . . . . .	12
2.5.2	Facile à développer . . . . .	12
2.5.3	Facile à vendre . . . . .	12
2.5.4	Google Play . . . . .	12
2.5.5	Flexible . . . . .	12
2.5.6	Complémentaire . . . . .	13
2.6	LES COMPOSANTS PRINCIPAUX D'UNE APPLICATION . . . . .	13
2.6.1	Activité ( Activity ) . . . . .	13
2.6.2	Services . . . . .	13
2.6.3	Recepteur de diffusion (Broadcast receiver) . . . . .	14
2.6.4	Fournisseurs de contenu (Content provider) . . . . .	14
2.6.5	Composants supplémentaires . . . . .	14
2.7	CYCLE DE VIE D'UNE APPLICATION ANDROID . . . . .	15
	CONCLUSION . . . . .	16

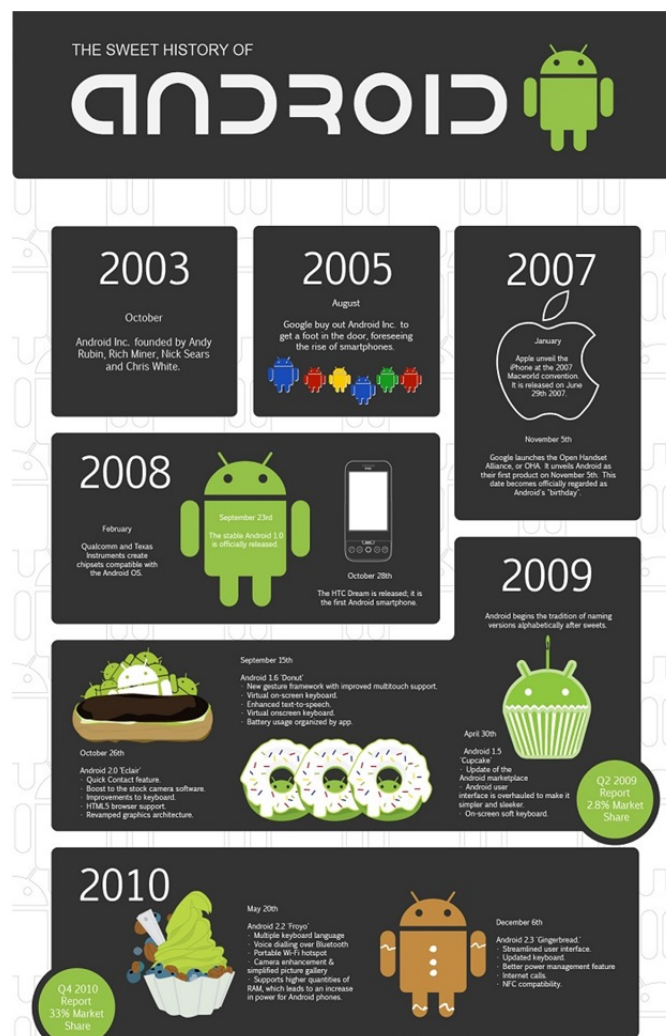
C E chapitre introductif de Latex  
Pas obligatoire!

## 2.1 DESCRIPTION

Android est un système d'exploitation mobile basé sur le noyau Linux et développé actuellement par Google. Le système a d'abord été conçu pour les Smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés et ordinateurs comme les télévisions (Android TV), les voitures (Android Auto) et les Smartwatch (Android Wear )... etc

## 2.2 HISTORIQUE ET VERSIONS D'ANDROID

Etant donné l'évolution rapide du système d'exploitation de Google, on a parfois tendance à oublier comment on en est arrivé là. Voici les événements importants du développement d'Android, de la création d'Android inc. par Andy Rubin, en passant par le rachat par Google en 2005, jusqu'à la récente arrivée de la version Lollipop :



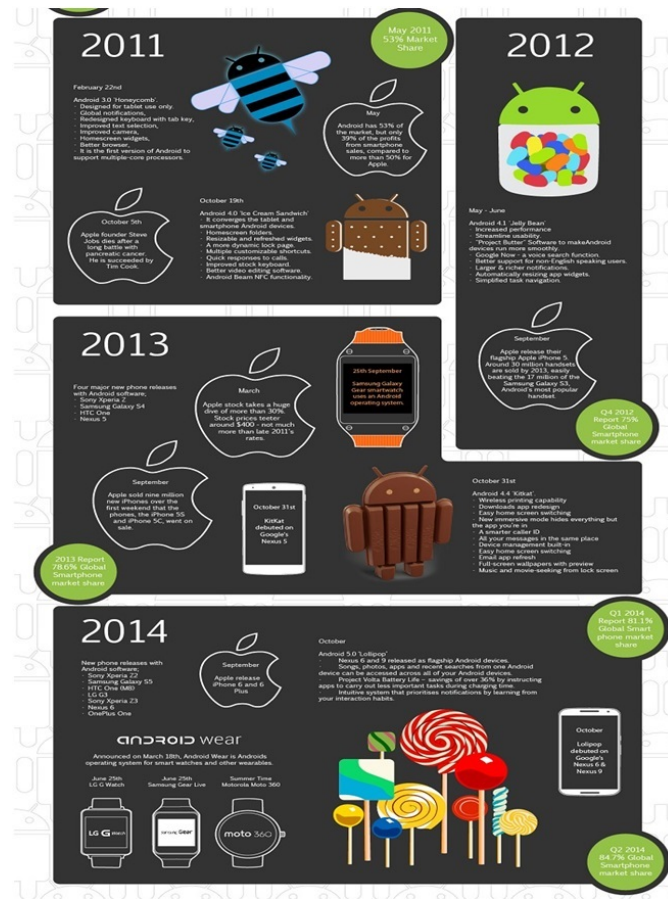


FIGURE 2.1 – Historique d'Android



FIGURE 2.2 – Les appareils d'Android de 2008 jusqu'à 2014

## 2.3 FONCTIONNALITÉS D'ANDROID

Android a été conçu pour intégrer au mieux les applications existantes de Google comme le service de courrier Gmail, l'agenda Google Calendar ou encore la cartographie Google Maps. Voici quelques fonctionnalités proposées par Android classées par version :

- Android version 1.5 (Cupcake )
  - Enregistrement et lecture des vidéos.
  - Mise en ligne directe des vidéos sur YouTube.
  - Mise en ligne directe des photos Picasa.
  - Prise en charge du Bluetooth A2DP.
  - Dossiers dynamiques et widgets pour le home.
  - Copier, coller étendu aux pages web.
  - Nouvelle version du clavier virtuel.
- Android version 1.6 (Donut )
  - L'application Galerie permet d'effacer plusieurs photos à la fois.
  - Amélioration de l'Android Market.
  - Amélioration de la vitesse de la recherche vocale et intégration étendue à plus d'applications natives.
  - Prise en charge sur une seule application de la prise de photo et de l'enregistrement vidéo.
  - Possibilité de rechercher simultanément dans les favoris, les historiques, les contacts et sur Google depuis le home via le widget recherche.

- Moteur Text-to-speech.
- Prise en charge de plusieurs résolutions d'écran.
- Android version 2.0/2.1 (eclair )
  - Fonds d'écran animés.
  - New browser interface avec prise en charge du HTML5.
  - Prise en charge du protocole Microsoft Exchange.
  - New contact lists.
  - Prise en charge du Bluetooth 2.1.
  - Amélioration du clavier virtuel.
  - Prise en charge en natif du flash et du zoom numérique pour des appareils photos.
  - Amélioration du ratio blanc/noir sur les fonds.
  - Gestion multi-comptes Gmail et ajout de la synchronisation avec Facebook.
- Android version 2.2
  - Augmentation de la performance et de la vitesse.
  - Fonctionnalité de Hot spot Wifi.
  - Partage de contact sur Bluetooth.
  - Mise à jour automatique des applications.

## 2.4 L'ARCHITECTURE D'ANDROID

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

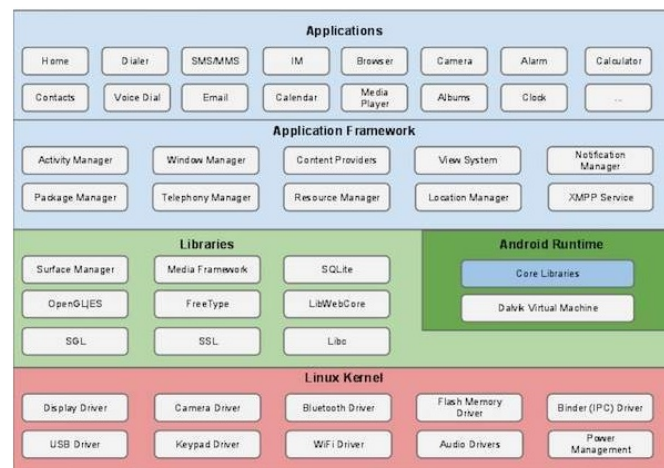


FIGURE 2.3 – Les composants du système d'exploitation Android

### 2.4.1 Application

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.

### 2.4.2 Framework et développement

En fournissant une plateforme de développement ouverte, Android offre aux développeurs la possibilité de créer des applications extrême-

ment 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.

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.

### 2.4.3 Bibliothèques

En plus du noyau Linux, il existe un ensemble de bibliothèques incluant le moteur de navigateur Web open source WebKit, la bibliothèque de bibliothèque bien connue, la base de données SQLite qui est un référentiel utile pour le stockage et le partage de données d'application, les bibliothèques à lire et à enregistrer de l'audio et de la vidéo SSL Les bibliothèques responsables de la sécurité sur Internet, etc.

### 2.4.4 Bibliothèques d'Android

Cette catégorie englobe les bibliothèques basées sur Java qui sont spécifiques au développement d'Android. Des exemples de bibliothèques dans cette catégorie comprennent les bibliothèques de cadres d'applications en plus de celles qui facilitent la construction d'interface utilisateur, le dessin graphique et l'accès à la base de données. Un résumé de certaines bibliothèques principales de base d'Android disponibles pour le développeur Android est le suivant

- 
- Android.app - Fournit l'accès au modèle d'application et constitue la pierre angulaire de toutes les applications Android.
- Android.content - Facilite l'accès au contenu, la publication et le messagerie entre les applications et les composants de l'application.

- `Android.database` - Utilisé pour accéder aux données publiées par les fournisseurs de contenu et comprend les classes de gestion de base de données SQLite.
- `Android.opengl` - Une interface Java à l'API de rendu graphique OpenGL ES 3D.
- `Android.os` - Fournit aux applications l'accès aux services standard du système d'exploitation, y compris les messages, les services système et la communication interprocessus.
- `Android.text` - Utilisé pour rendre et manipuler du texte sur l'affichage d'un périphérique.
- `Android.view` - Les éléments fondamentaux des interfaces utilisateur de l'application.
- `Android.widget` - Une collection riche de composants d'interface utilisateur pré-construits tels que des boutons, des étiquettes, des listes, des questionnaires de disposition, des boutons radio, etc.
- `Android.webkit` - Un ensemble de classes destinées à permettre l'intégration des fonctionnalités de navigation Web dans les applications.

Après avoir couvert les bibliothèques de base basées sur Java dans l'exécution de l'Android, il est maintenant temps d'attirer l'attention sur les bibliothèques basées sur C / C ++ contenues dans cette couche de la pile de logiciels Android.

#### **2.4.5 Android Runtime ( Moteur d'exécution )**

Il s'agit de la troisième section de l'architecture et disponible sur la deuxième couche en bas. Cette section fournit un composant clé appelé Dalvik Virtual Machine, qui est une sorte de machine virtuelle Java spécialement conçue et optimisée pour Android.

La VM Dalvik utilise des fonctionnalités principales de Linux comme la gestion de la mémoire et le multi-threading, intrinsèque dans la langue Java. La VM Dalvik permet à toute application Android de fonctionner dans son propre processus, avec sa propre instance de la machine virtuelle Dalvik.

Le runtime d'Android fournit également un ensemble de bibliothèques de base qui permettent aux développeurs d'applications Android d'écrire des applications Android en utilisant un langage de programmation Java standard.

#### **2.4.6 Linux Kernel (Noyau Linux )**

Au bas des couches, Linux-Linux 3.6 comporte environ 115 patches. Cela fournit un niveau d'abstraction entre le matériel de l'appareil et il contient tous les pilotes matériels essentiels comme la caméra, le clavier, l'affichage etc. De plus, le noyau gère tout ce dont Linux est vraiment bon, comme le réseautage et une vaste gamme de pilotes de périphériques, Qui supporte la douleur de l'interfaçage de matériel périphérique.



## 2.5 LA PHILOSOPHIE ET LES AVANTAGES D'ANDROID

### 2.5.1 Open source

Le contrat de licence pour Android respecte les principes de l'open source, c'est-à-dire que vous pouvez à tout moment télécharger les sources et les modifier selon vos goûts ! Bon, je ne vous le recommande vraiment pas, à moins que vous sachiez ce que vous faites... Notez au passage qu'Android utilise des bibliothèques open source puissantes, comme par exemple SQLite pour les bases de données et OpenGL pour la gestion d'images 2D et 3D.

### 2.5.2 Facile à développer

Toutes les API mises à disposition facilitent et accélèrent grandement le travail. Ces API sont très complètes et très faciles d'accès. De manière un peu caricaturale, on peut dire que vous pouvez envoyer un SMS en seulement deux lignes de code (concrètement, il y a un peu d'enrobage autour de ce code, mais pas tellement ).

### 2.5.3 Facile à vendre

Le Play Store (anciennement Android Market ) est une plateforme immense et très visitée, c'est donc une mine d'opportunités pour quiconque veut diffuser une application dessus.

### 2.5.4 Google Play

Google Play, est le magasin en ligne de Google. Celui-ci permet de télécharger des logiciels, des livres, des films ou de la musique, payants ou non. Il est aussi possible de les noter et de les commenter. En septembre 2011, il y avait plus de 520 000 applications sur Android Market, dont 65 % sont gratuites. (Voici le logo de Google Play )



FIGURE 2.4 – Logo Google Play

### 2.5.5 Flexible

Le système est extrêmement portable, il s'adapte à beaucoup de structures différentes. Les Smartphones, les tablettes, la présence ou l'absence de clavier ou de trackball, différents processeurs... On trouve même des

fours à micro-ondes qui fonctionnent à l'aide d'Android ! Non seulement c'est une immense chance d'avoir autant d'opportunités, mais en plus Android est construit de manière à faciliter le développement et la distribution en fonction des composants en présence dans le terminal (si votre application nécessite d'utiliser le Bluetooth, seuls les terminaux équipés de Bluetooth pourront la voir sur le Play Store) .

### 2.5.6 Complémentaire

L'architecture d'Android est inspirée par les applications composites, et encourage par ailleurs leur développement. Ces applications se trouvent essentiellement sur internet et leur principe est que vous pouvez combiner plusieurs composants totalement différents pour obtenir un résultat surpuissant. Par exemple, si on combine l'appareil photo avec le GPS, on peut poster les coordonnées GPS des photos prises.

## 2.6 LES COMPOSANTS PRINCIPAUX D'UNE APPLICATION

Les composants d'application sont les composants essentiels d'une application Android. Ces composants sont lâchement couples par le fichier manifeste d'application `AndroidManifest.xml` qui décrit chaque composant de l'application et comment ils interagissent. Il existe quatre composants principaux qui peuvent être utilisés dans une application Android :

### 2.6.1 Activité ( Activity )

Une activité représente un seul écran avec une interface utilisateur, dans unecourte Activité effectue des actions sur l'écran. Si une application comporte plus d'une activité, l'une d'entre elles doit être marquée comme l'activité présentée lors du lancement de l'application.

Une activité est implémentée sous la forme d'une sous-classe de classe d'activité comme suit :

```
Class public MainActivity extends Activity { }
```

### 2.6.2 Services

Un service est un composant qui s'exécute en arrière-plan pour effectuer des opérations à long terme. Par exemple, un service peut jouer de la musique en arrière-plan lorsque l'utilisateur se trouve dans une autre application, ou il peut récupérer des données sur le réseau sans bloquer l'interaction de l'utilisateur avec une activité. Un service est mis en?uvre comme sous-classe de la classe `Service` comme suit :

```
Class public MyService extends service { }
```

### 2.6.3 Récepteur de diffusion (Broadcast receiver)

Les récepteurs de diffusion répondent simplement aux messages diffusés provenant d'autres applications ou du système. Par exemple, les applications peuvent également déclencher des émissions pour permettre aux autres applications de savoir si certaines données ont été téléchargées sur le périphérique et qu'elles sont disponibles pour qu'elles puissent être utilisées, donc c'est un récepteur qui intercepte cette communication et lancera une action appropriée. Un récepteur de diffusion est implémenté sous la forme d'une sous-classe de la classe `BroadcastReceiver` et chaque message est un diffuseur comme objet `Intent`.

```
public class MyReceiver extends BroadcastReceiver {

    public void OnReceive ( contexte , intention ) { } }
```

### 2.6.4 Fournisseurs de contenu (Content provider)

Un fournisseur de contenu fournit des données d'une application à d'autres sur demande. De telles requêtes sont traitées par les méthodes de la classe `ContentResolver`. Les données peuvent être stockées dans le système de fichiers, la base de données ou ailleurs. Un fournisseur de contenu est implémenté comme une sous-classe de la classe `ContentProvider` et doit implémenter un ensemble standard d'API permettant à d'autres applications d'effectuer des transactions.

```
public class MyContentProvider extends ContentProvider {

    public void onCreate ( ) { } }
```

### 2.6.5 Composants supplémentaires

Composants et description
<b>Fragments</b> Représente une partie de l'interface utilisateur dans une activité.
<b>Vues</b> Les éléments UI qui sont dessinés à l'écran, y compris les boutons, les listes des formulaires, etc.
<b>Dispositions</b> Afficher les hiérarchies qui contrôlent le format d'écran et l'apparence des vues.
<b>Intents</b> Les messages se combinent ensemble.
<b>Ressources</b> Les éléments externes, tels que les chaînes, les constantes et les images étirables.

FIGURE 2.5 – Composants supplémentaires

## 2.7 CYCLE DE VIE D'UNE APPLICATION ANDROID

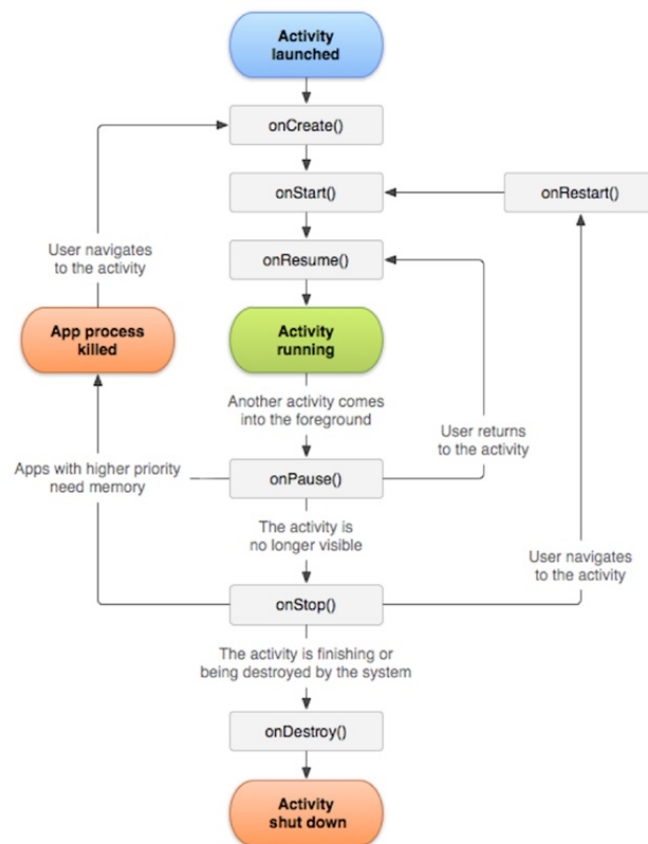


FIGURE 2.6 – cycle de vie d'une Activité

- Lorsque l'activité est lancée, le système Android appelle les méthodes `onCreate`, `onStart` et `onResume`.
- Lorsque l'activité s'arrête, le système Android appelle les méthodes `onPause`, `onStop` et `onDestroy`.
- Lorsque l'activité n'est plus au premier plan, mais qu'elle est tout de même affichée, `onPause` est appelée. Lorsque l'activité retourne au premier plan, `onResume` est appelée.
- Lorsque l'activité n'est plus affichée, `onResume` et `onStop` sont appelées. Lorsque l'activité est de nouveau affichée, `onRestart`, `onStart` et `onResume` sont appelées.
- Lorsque le système Android décide de tuer votre application, il appelle la méthode `onSaveInstanceState` qui vous permet de sauvegarder l'état de votre activité. Cet état sera passé en paramètre à la méthode `onCreate`, afin que vous puissiez restaurer l'état de l'activité.
- Le système Android s'occupe lui-même de sauvegarder et restaurer l'état des vues des layouts, à condition que ces vues aient chacune un ID unique dans le layout.
- Le changement de configuration comme le changement d'orientation de l'écran provoque un redémarrage de l'activité, vous devez donc sauvegarder et restaurer l'état dans ce cas.

## CONCLUSION DU CHAPITRE

Ceci est la conclusion. Personnellement[[Commowick et Malandain, 2007](#)], je n'aime pas que la conclusion soit numéroté, mais je veux qu'elle apparaisse dans la table des matiere, d'ou la commande addcontentsline.

# CONCLUSION GÉNÉRALE

Enfin : la conclusion générale!!!

Au cours de ce mémoire, nous avons développé un modèle ...

1. **Modélisation**
2. **Inférence statistique**

## PERSPECTIVES

Dans la continuité directe de notre travail de thèse, nous pouvons ...

# BIBLIOGRAPHIE

Olivier Commowick et Grégoire Malandain. Efficient selection of the most similar image in a database for critical structures segmentation. Dans *Proceedings of the 10th Int. Conf. on Medical Image Computing and Computer-Assisted Intervention - MICCAI 2007, Part II*, volume 4792 de LNCS, pages 203–210. Springer Verlag, 2007.

**Titre** Système de gestion des signalements des citoyen

**Resume** Le resume en francais ( $\approx$  1000 caracteres)

**Mots-cles** Les mots-cles en francais

**Title** Le titre en anglais

**Abstract** Le resume en anglais ( $\approx$  1000 caracteres)

**Keywords** Les mots-cles en anglais