

Le développement d'une application mobile sous Android



Les Modules

1. Découverte de l'environnement
2. Les fondamentaux
3. Les interactions
4. Les **Composants** d'interactions
5. La **Persistiance**
6. Les **Tâches Asynchrones**
7. Les services **Web Distants**
8. Les **Services et Broadcasts**
9. **Room Persistence Library**
10. Les **WebView**
11. (La Géolocalisation)
12. (Le Design)
13. (La Publication)

Le développement d'une application mobile sous Android

Module 1 - Découverte de l'environnement



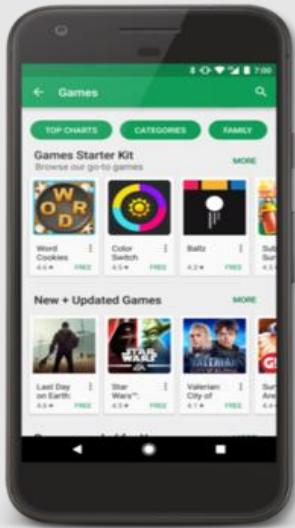
Objectifs

- Découvrir la plateforme Android
- Découvrir l'architecture Android
- Découvrir l'environnement de développement
- Créer un premier projet

Découverte de l'environnement

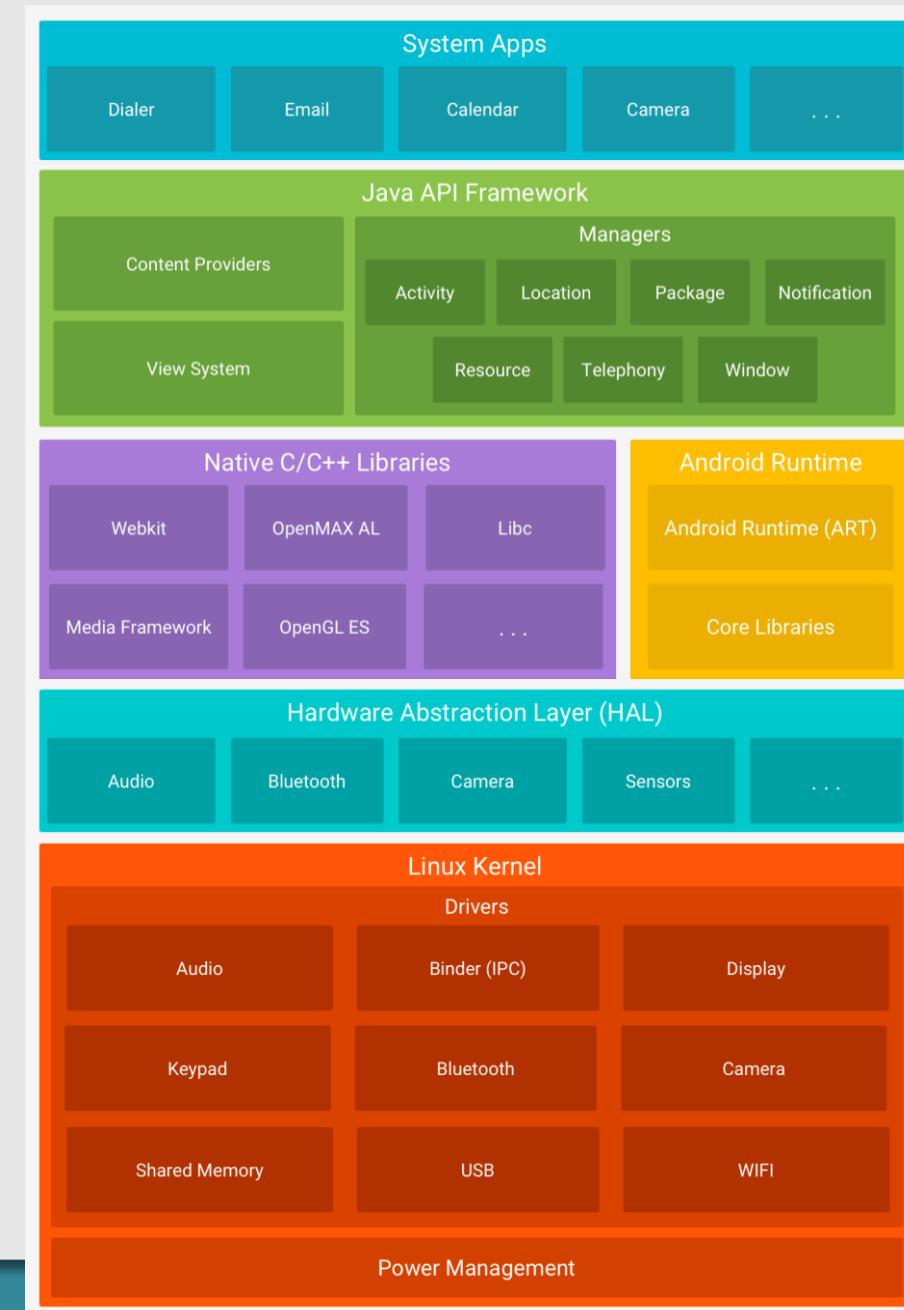
Présentation

Système d'exploitation



Découverte de l'environnement Présentation

Linux

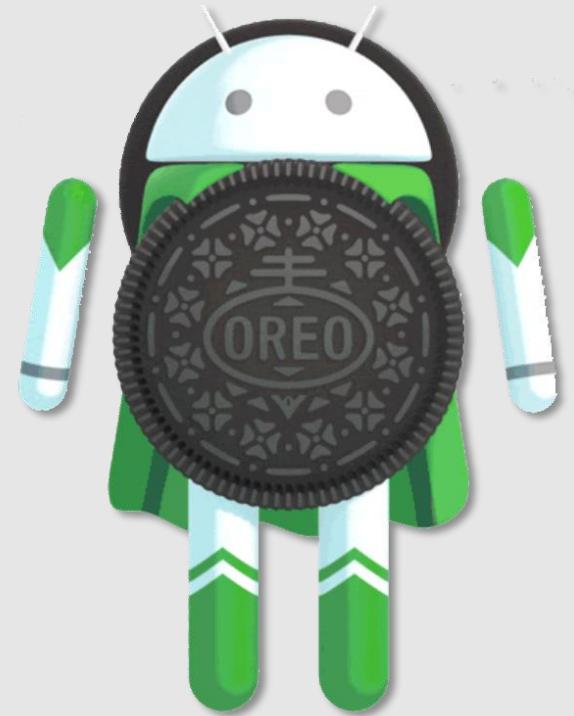


Découverte de l'environnement

Présentation

versions

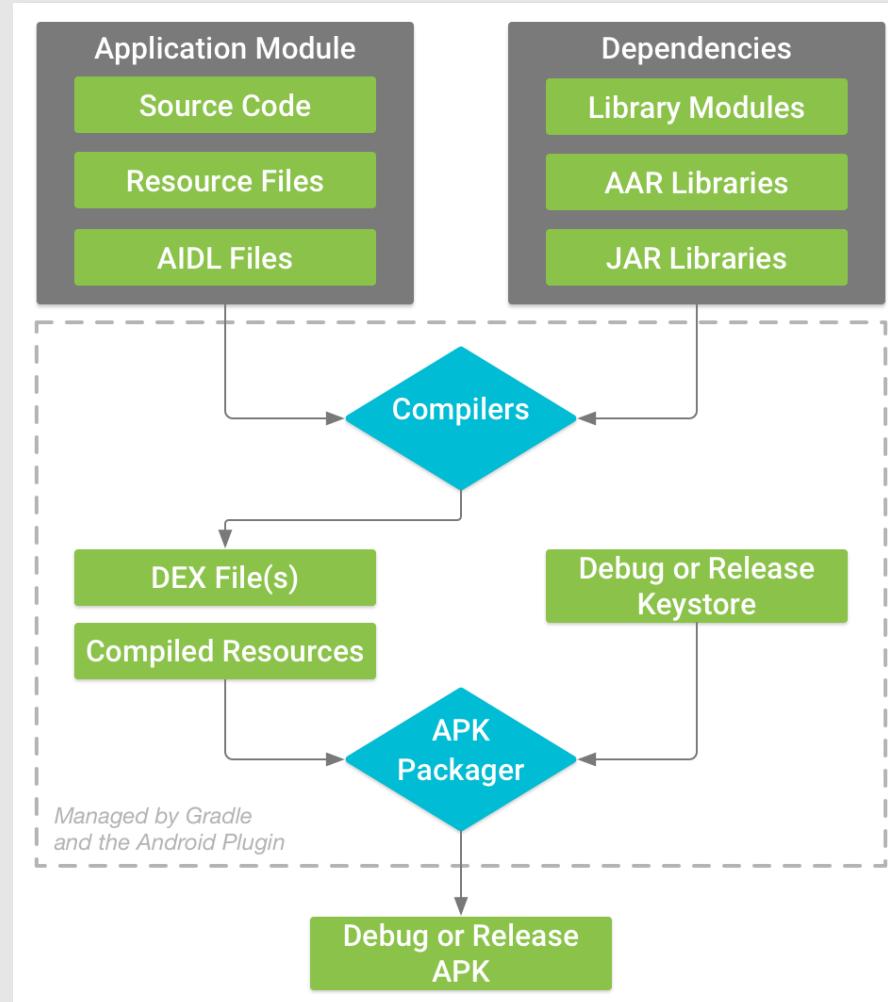
Version	API	Nom de code	Année
9.0	28	Pie	2019
8.1	27	Oreo	2017
8.0	26		
7.1	25	Nougat	2016
7.0	24		
6.0	23	Marshmallow	2015



Découverte de l'environnement

Présentation

Production
d'une application



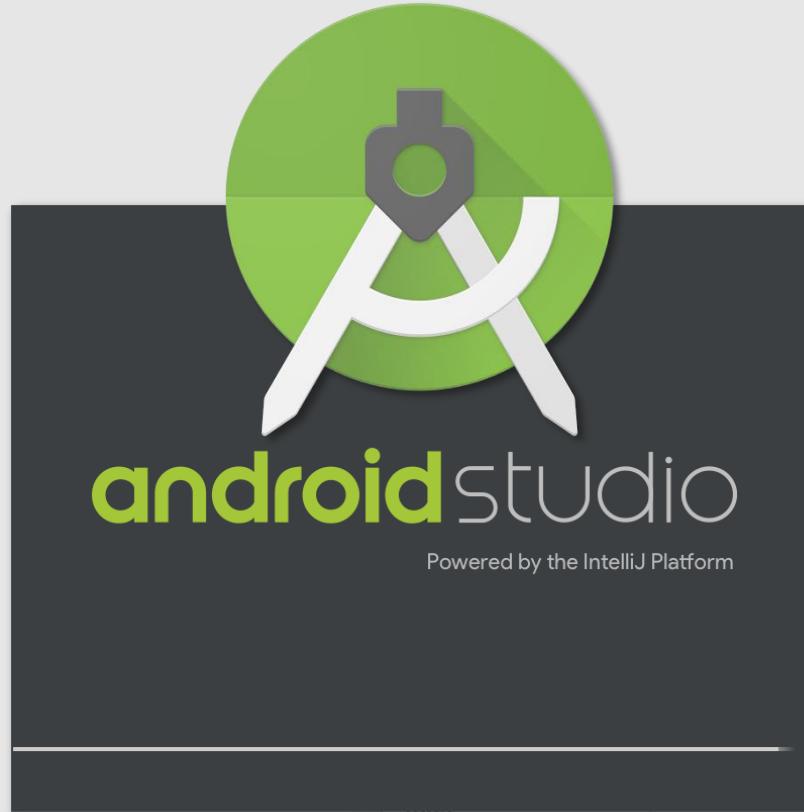
Découverte de l'environnement

Présentation

Environnement
de développement



SDK Android



Découverte de l'environnement

Environnement de développement

Démonstration



Découverte de l'environnement
Créer un projet

Démonstration



Découverte de l'environnement
Exécuter un projet

Démonstration



Conclusion

- Vous avez découvert la plateforme Android
- Vous avez découvert l'architecture Android
- Vous avez découvert l'environnement de développement
- Vous savez comment créer un projet
- Vous savez comment exécuter un projet

Découverte de l'environnement

Création d'un nouveau projet

TP



Le développement d'une application mobile sous Android

Module 2 - Les fondamentaux

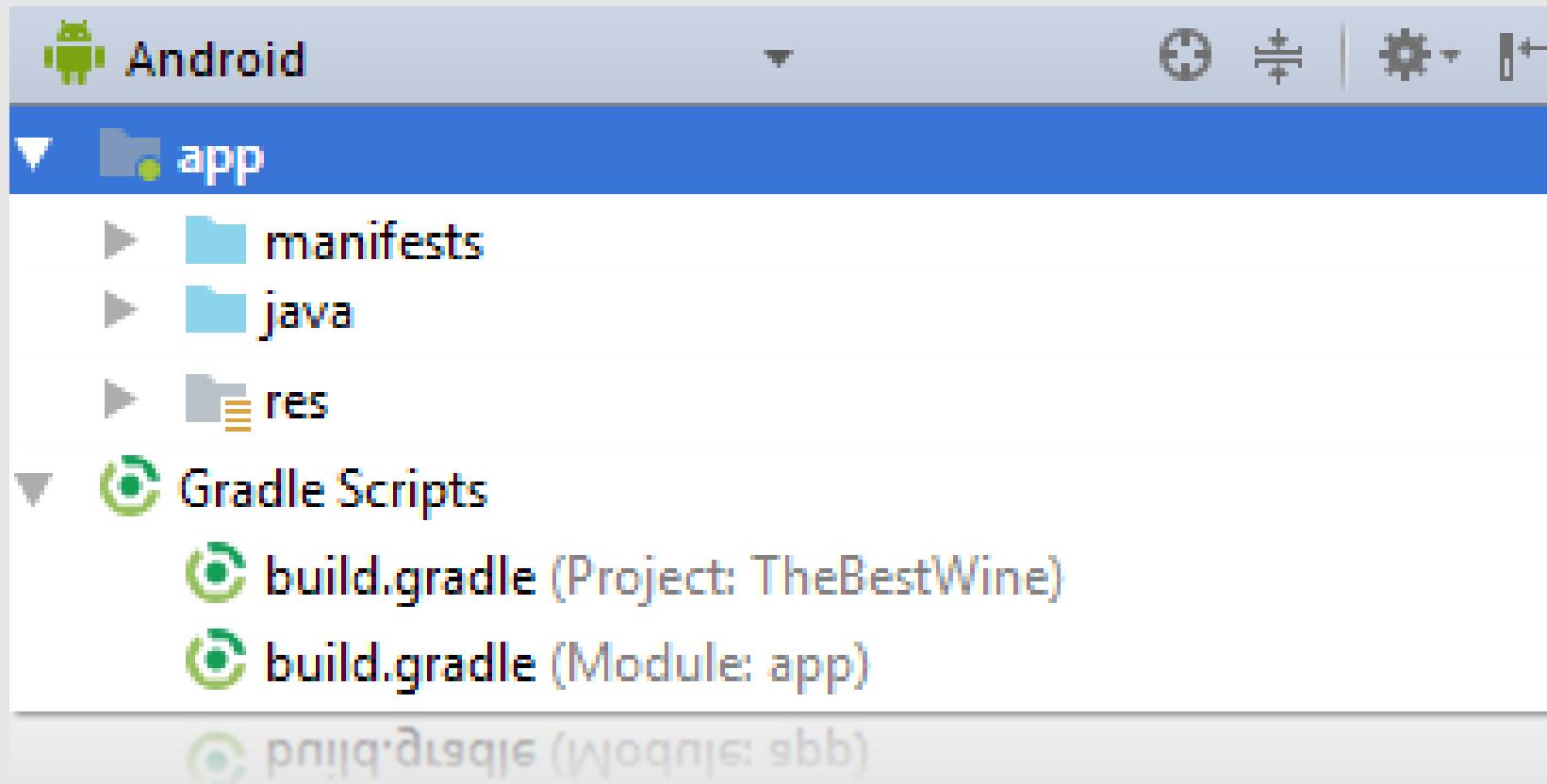


Objectifs

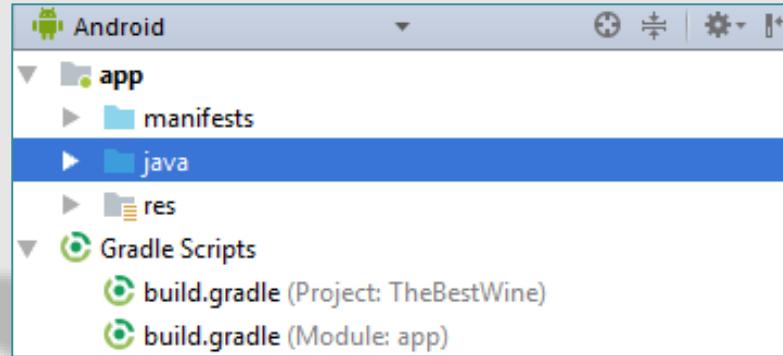
- Découvrir la structure d'une application
- Découvrir les principales composantes d'une application
- Savoir définir une interface graphique simple

Les fondamentaux

Structure d'un projet



Les fondamentaux Activity



- Une activité représente un écran de votre application
- Une application est composée d'une ou plusieurs activités
- Les activités sont les contrôleurs permettant d'interagir avec l'utilisateur
- Toute activité doit être inscrite dans le fichier `AndroidManifest.xml`

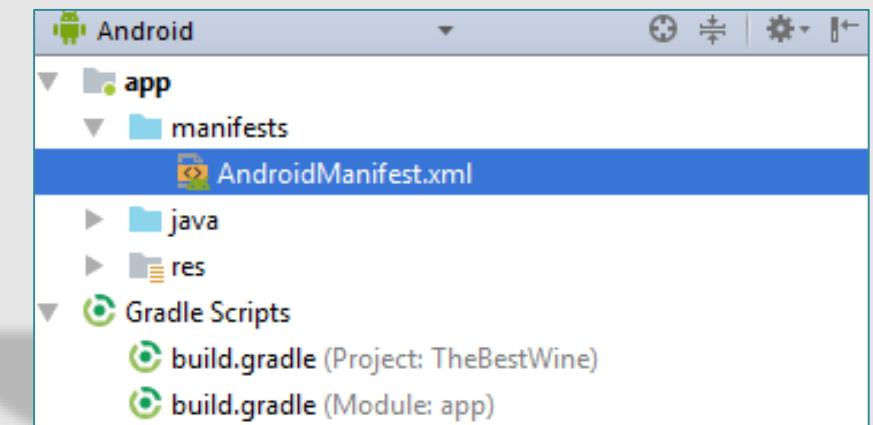
Les fondamentaux AndroidManifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="fr.eni.thebestwine">

    <uses-permission android:name="android.permission.CALL_PHONE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

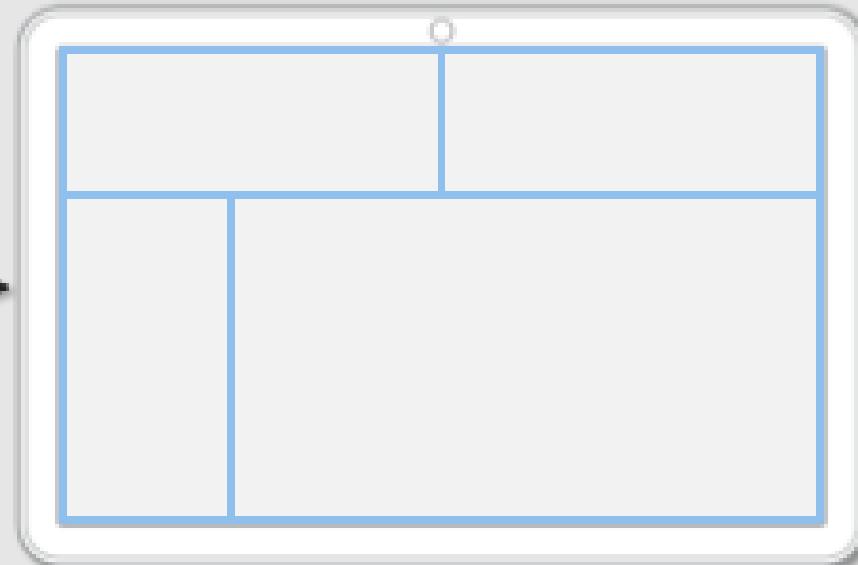
        <activity android:name=".MainActivity" />
        <activity android:name=".AjouterActivity" />
        <activity android:name=".ContactActivity" />
    </application>
</manifest>
```



Les fondamentaux

Les ressources

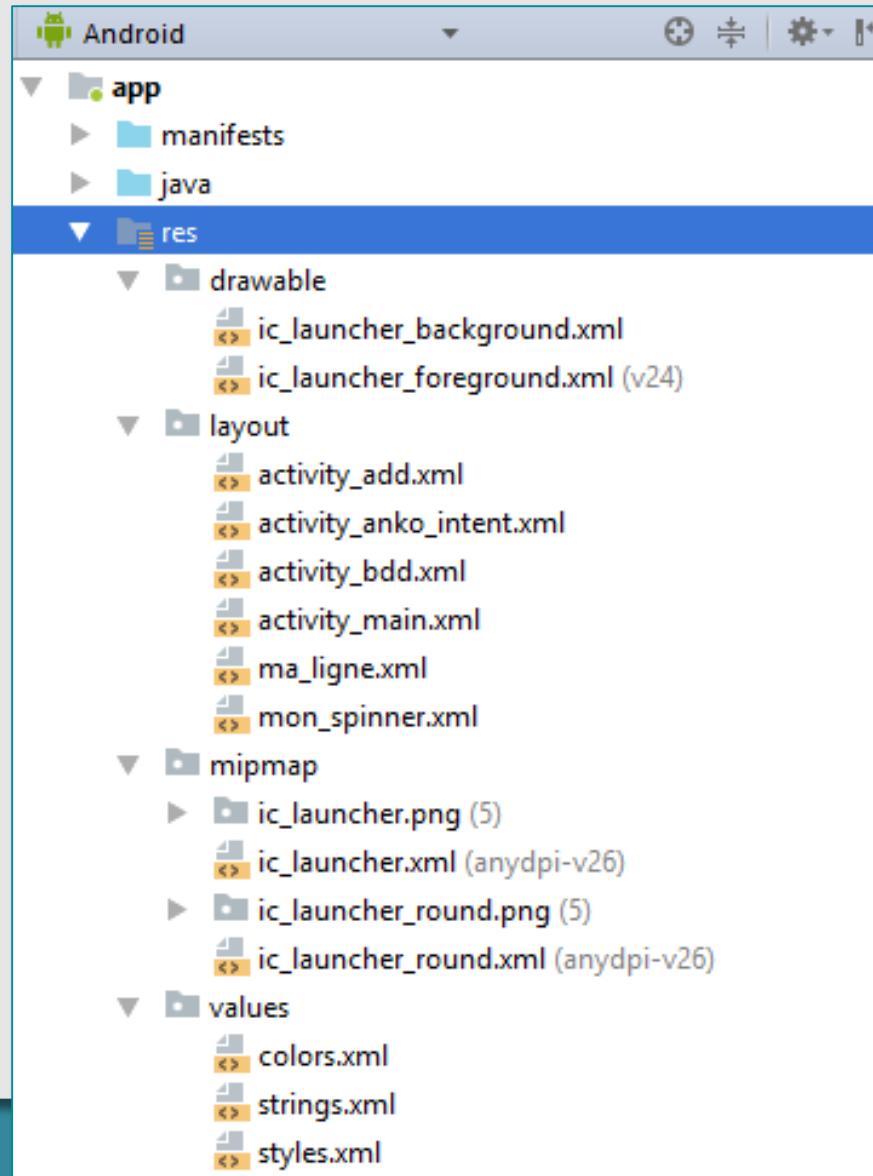
Présentation



Les fondamentaux

Les ressources

Ajouter des
ressources



Les fondamentaux

Ajouter des ressources

Démonstration



Les ressources

Accéder aux
ressources

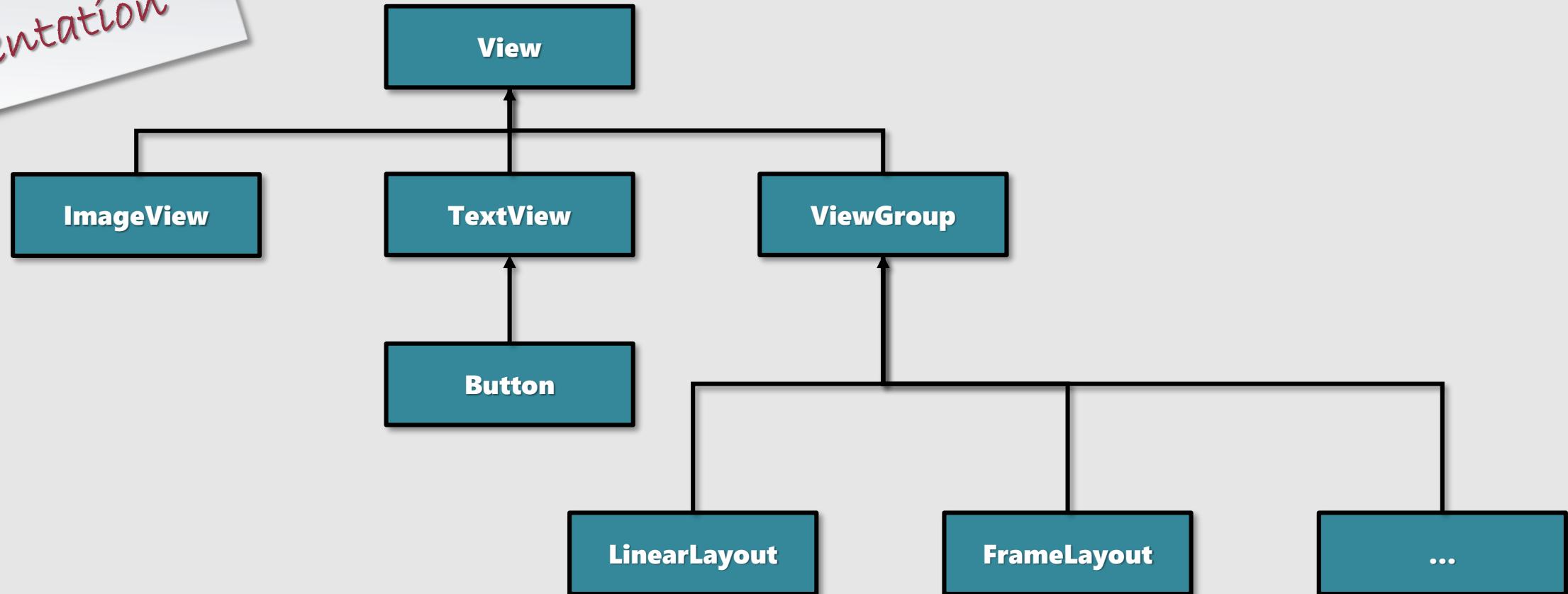


- Depuis la **classe R**, Android Studio propose deux moyens d'accéder aux ressources :
 - **Depuis le Code (.Class)**
 - `R.string.hello`
 - `R.layout.activity_main`
 - **Depuis la Description (XML)**
 - `@string/app_name`
 - ...>"@string/hello"<...

Les fondamentaux

Les interfaces graphiques

Présentation

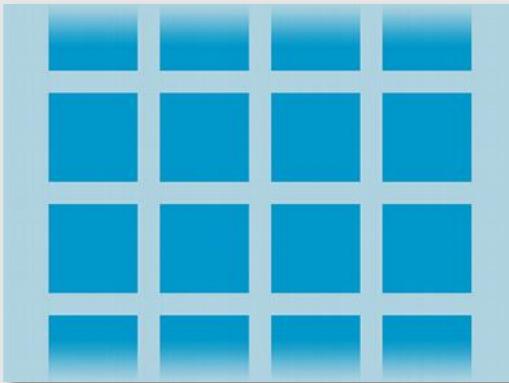


Les fondamentaux

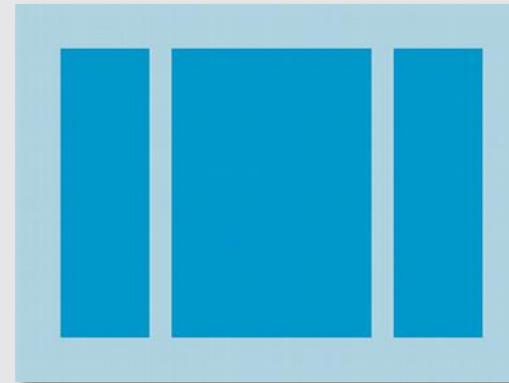
Les interfaces graphiques

Layout

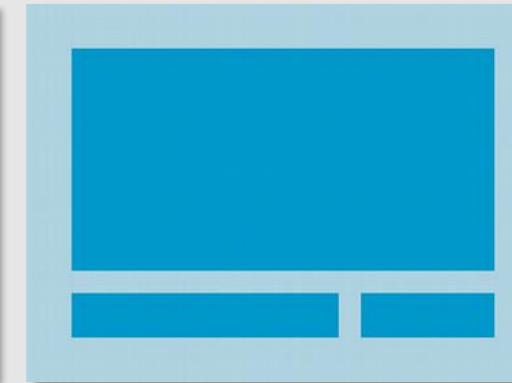
GridLayout



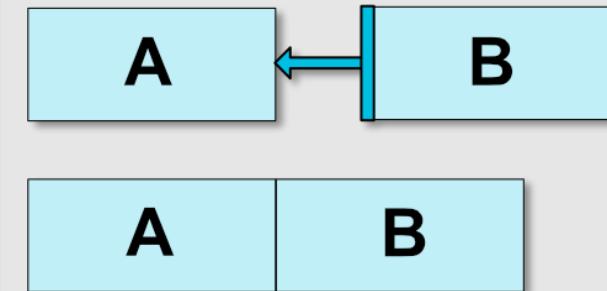
LinearLayout



RelativeLayout



ConstraintLayout



Les interfaces graphiques

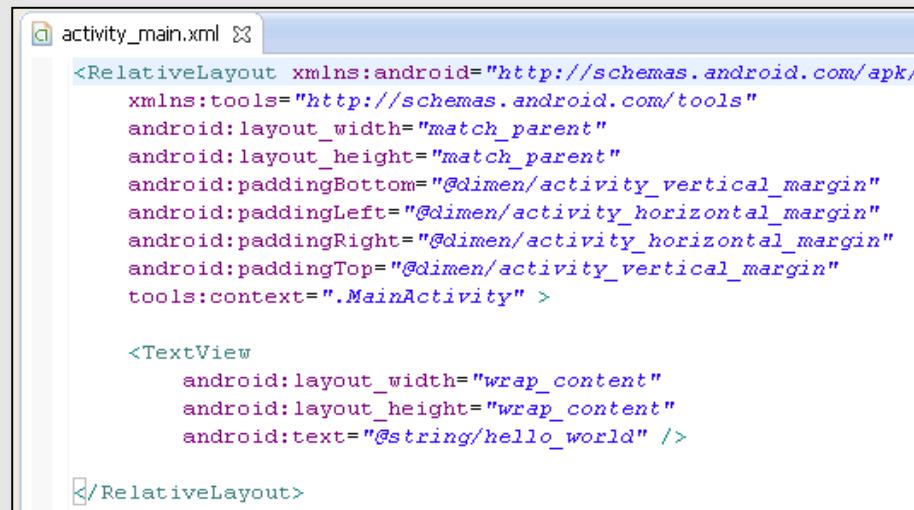
widget

- **TextView**
- **EditText**
- **Button / ToggleButton**
- **CheckBox**
- **RadioButton**
- **ProgressBar**
- **SearchView**
- **ImageButton**
- ***...et bien plus encore.***

Les fondamentaux

Les interfaces graphiques

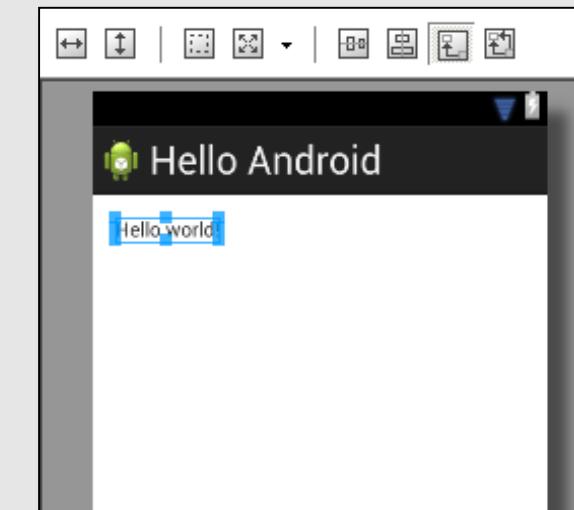
Définition



```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

</RelativeLayout>
```



Les fondamentaux

Définir une interface graphique simple

Démonstration



Les fondamentaux

Définir une seconde interface graphique simple

Démonstration



Les fondamentaux

Les interfaces graphiques

Dimensions

Dimension	Signification
dp	Density-independent Pixels
sp	Scale-independent Pixels
pt	Points
px	Pixels
mm	Millimeters (mm)
in	Inches ("")

Les fondamentaux

Les dimensions

Démonstration



Conclusion

- Vous connaissez la structure d'un projet Android
- Vous savez ajouter des ressources à un projet
- Vous savez définir une interface graphique simple

Les fondamentaux

Définir une IHM simple

TP



Le développement d'une application mobile sous Android

Module 3 - Les interactions



Objectifs

- Savoir utiliser les logs
- Comprendre comment fonctionne une activité
- Savoir interagir entre plusieurs activités
- Comprendre le concept des fragments
- Comprendre comment fonctionnent les permissions

Les interactions

Log

Présentation

ERROR

WARNING

DEBUG

INFO

VERBOSE

```
private static final String TAG = "MyActivity";  
Log.v(TAG, "index=" + i);
```

Les interactions

Logcat

Présentation

The screenshot shows the Android Studio Logcat window. At the top, there are dropdown menus for the device ('Emulator Nexus_5X_A'), package ('fr.eni.android.cycleoflife'), log level ('Debug'), search ('Q'), regex ('Regex') checked, and filter ('Life_Cycle'). The main area displays log entries:

- 04/05 15:15:27: Launching app
Client not ready yet..Waiting for process to come online
- Connected to process 7512 on device emulator-5554
2019-04-05 16:25:48.948 29107-10021/? E/android.media.AudioRecord: Error code -20 when initializing native AudioRecord object.
- I/zygote: Background concurrent copying GC freed 9983(2MB) AllocSpace objects, 0(0B) LOS objects, 56% free, 1160KB/2MB, paused 1.533ms total 104.005ms
- I/Life_Cycle: onCreate
- I/Life_Cycle: onStart
- I/Life_Cycle: onResume
- D/OpenGLRenderer: HWUI GL Pipeline
- D/: HostConnection::get() New Host Connection established 0xa5da4900, tid 7532
- I/zygote: android::hardware::configstore::V1_0::ISurfaceFlingerConfigs::hasWideColorDisplay retrieved: 0

At the bottom, there are tabs for Run, Logcat (which is selected), TODO, Terminal, Build, Profiler, and Event Log. Below the tabs is a toolbar with icons for Run, Stop, Logcat, TODO, Terminal, Build, Profiler, and Event Log.

Les interactions Logcat

Démonstration



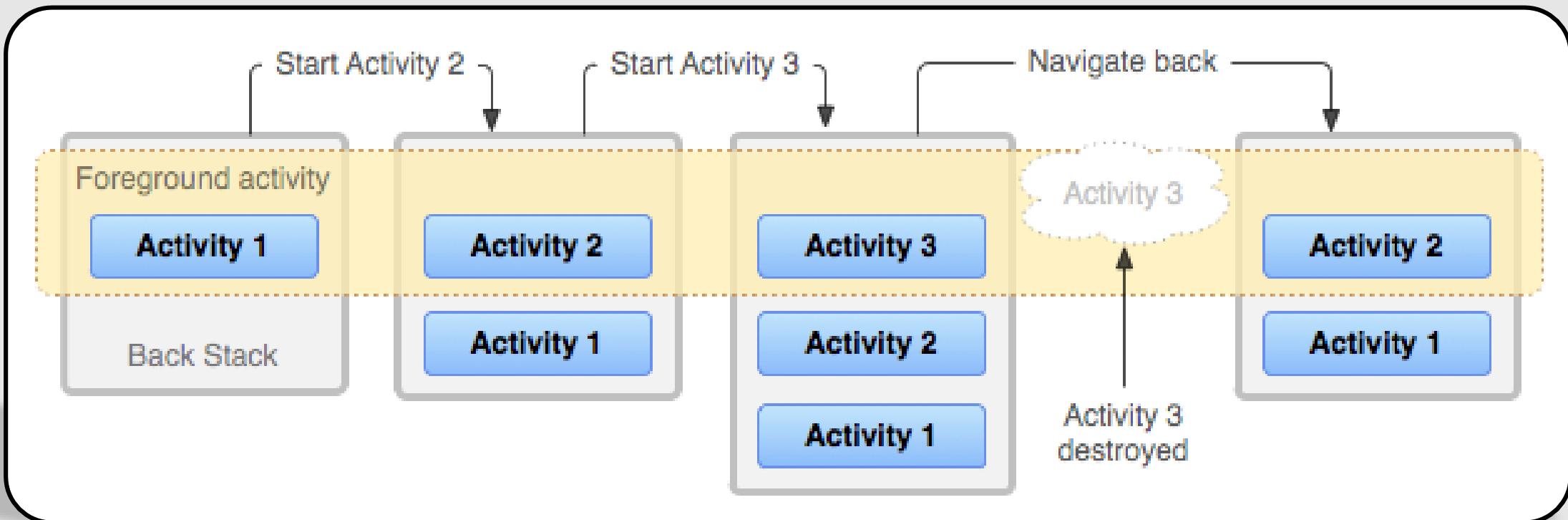
Activity

Présentation

- Une activité représente un écran de votre application
- Une application est composée d'une ou plusieurs activités
- Les activités sont les contrôleurs permettant d'interagir avec l'utilisateur
- Toute activité doit être inscrite dans le fichier `AndroidManifest.xml`

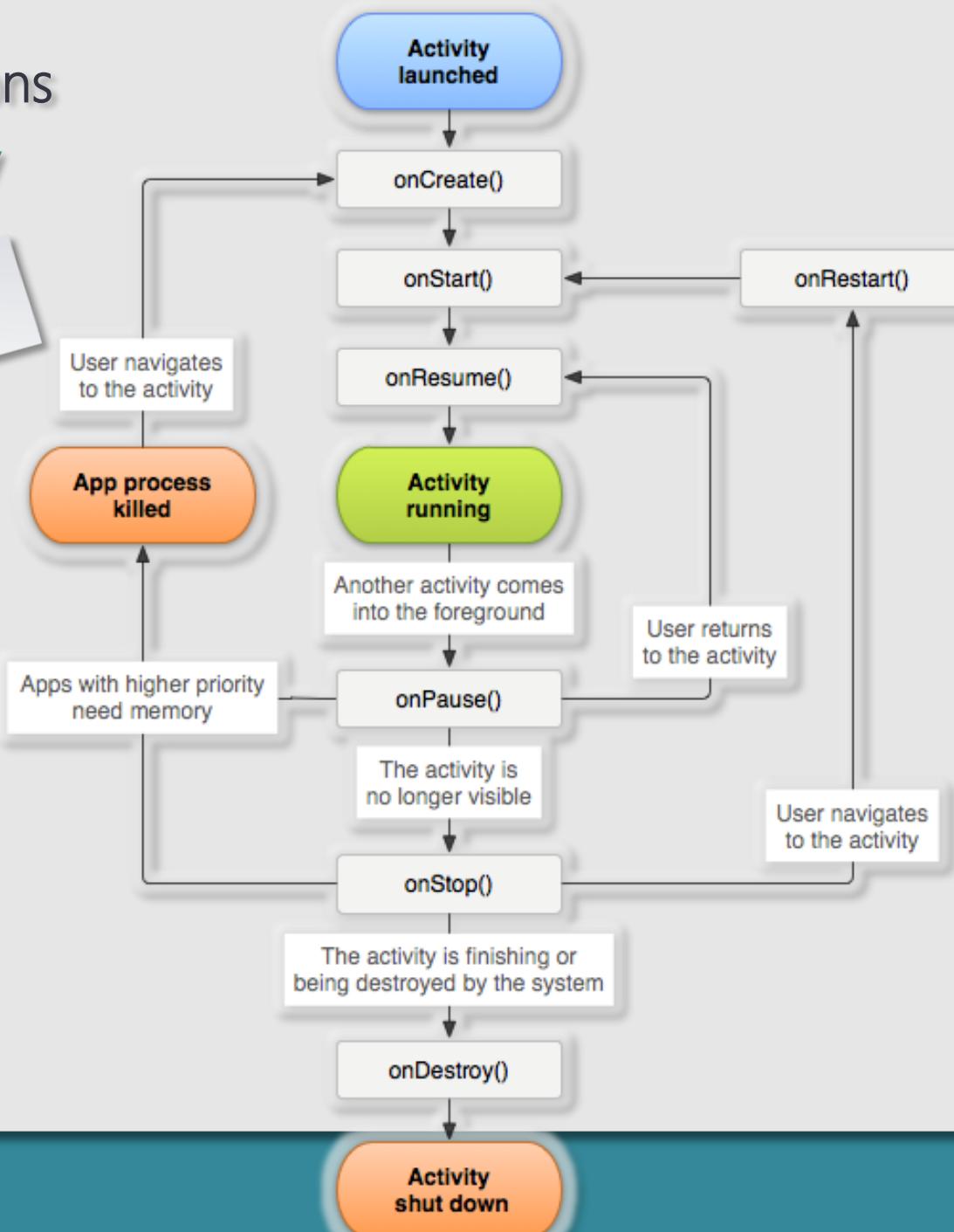
Les interactions Activity

Pile de gestion



Les interactions Activity

cycle de vie



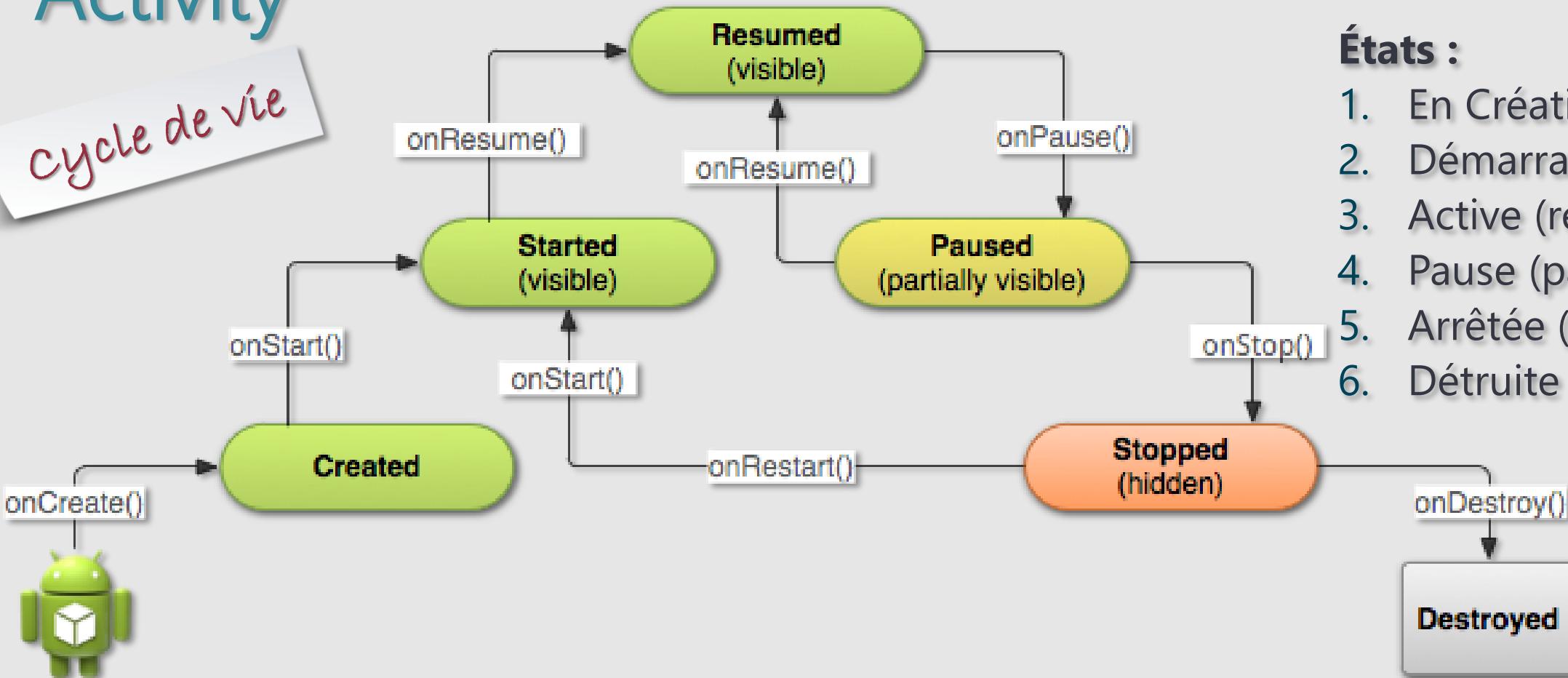
android.app.Activity

CallBacks :

- `onCreate()`
- `onStart()`
 - `onRestoreInstanceState()`
- `onResume()`
- `onPause()`
 - `onSaveInstanceState()`
- `onStop`
- `onRestart()`
- `onDestroy()`

Les interactions Activity

cycle de vie



`android.app.Activity`

États :

1. En Création (created)
2. Démarrage (started)
3. Active (resumed)
4. Pause (paused)
5. Arrêtée (stopped)
6. Détruite (destroyed)

Les interactions

Activity : Sauvegarde & Restauration

cycle de vie



Les interactions

Les activités

Démonstration



Les interactions Activity

Gestion d'évènements

Fichier XML

```
<Button  
    android:id="@+id	btn_add"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Add"  
    android:layout_weight="2"  
    android:background="@color/colorPrimary"  
    android:textColor="@color/white"  
    android:onClick="onClickAdd"  
/>
```

Fichier Class

```
public class MainActivity extends AppCompatActivity {  
  
    private static final String TAG = "MainActivity";  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
  
    public void onClickAdd(View view) {  
        Log.i(TAG, "Vous avez cliqué sur le bouton Add");  
    }  
}
```

Les interactions

Alertes : Toast, AlertDialog, Snackbar

Exemple :

- ```
AlertDialog alertDialog = new
AlertDialog.Builder(this).create();
alertDialog.setTitle("Confirmation");
alertDialog.setMessage("... AlertDialog ?");
...
alertDialog.show();
```
- ```
Toast.makeText(this, "Ceci est un Toast",  
Toast.LENGTH_SHORT).show();
```
- ```
Snackbar.make(view, "Voici une Snackbar",
Snackbar.LENGTH_SHORT).show();
```



Les interactions

# Gestion d'évènements

# Démonstration



Les interactions

# Gestion d'évènements

TP



Les interactions

# Création d'une nouvelle activité

## Démonstration

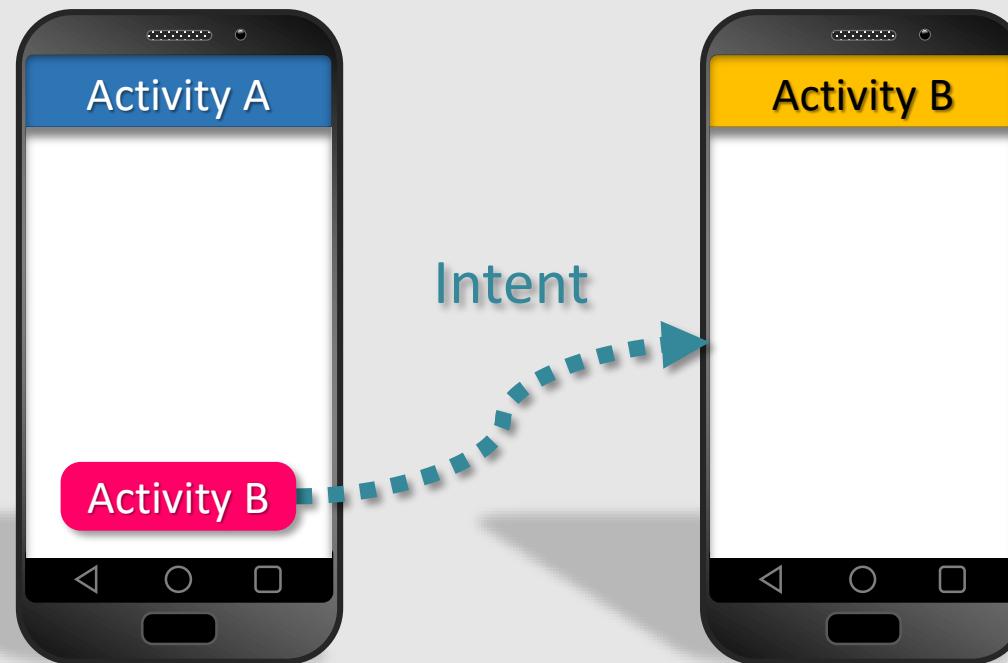


Présentation

- Permet de faire une demande d'action pour :
  - Démarrer une activité
  - Démarrer un service
  - Diffuser un message à d'autres applications

# Les interactions Intent

*Intent explicite*



Les interactions

# Utilisation d'un Intent explicite

## Démonstration



Les interactions

# Utilisation d'un intent explicite

TP



composition

- **action : Nom d'une action (lire, éditer...)**
- **data : URI qui référence les données à traiter et/ou type MIME**
- component : Nom du composant à démarrer
- category : Catégorie indiquant le type de composant devant gérer l'intention
- **extra : informations supplémentaires (données à transporter)**
- flag : indicateur définissant comment démarrer une nouvelle activité

Les interactions

# Utilisation d'un Intent explicite avec extras

## Démonstration



Les interactions

# Utilisation d'un intent explicite avec extras

TP



# Les interactions Intent avec retour

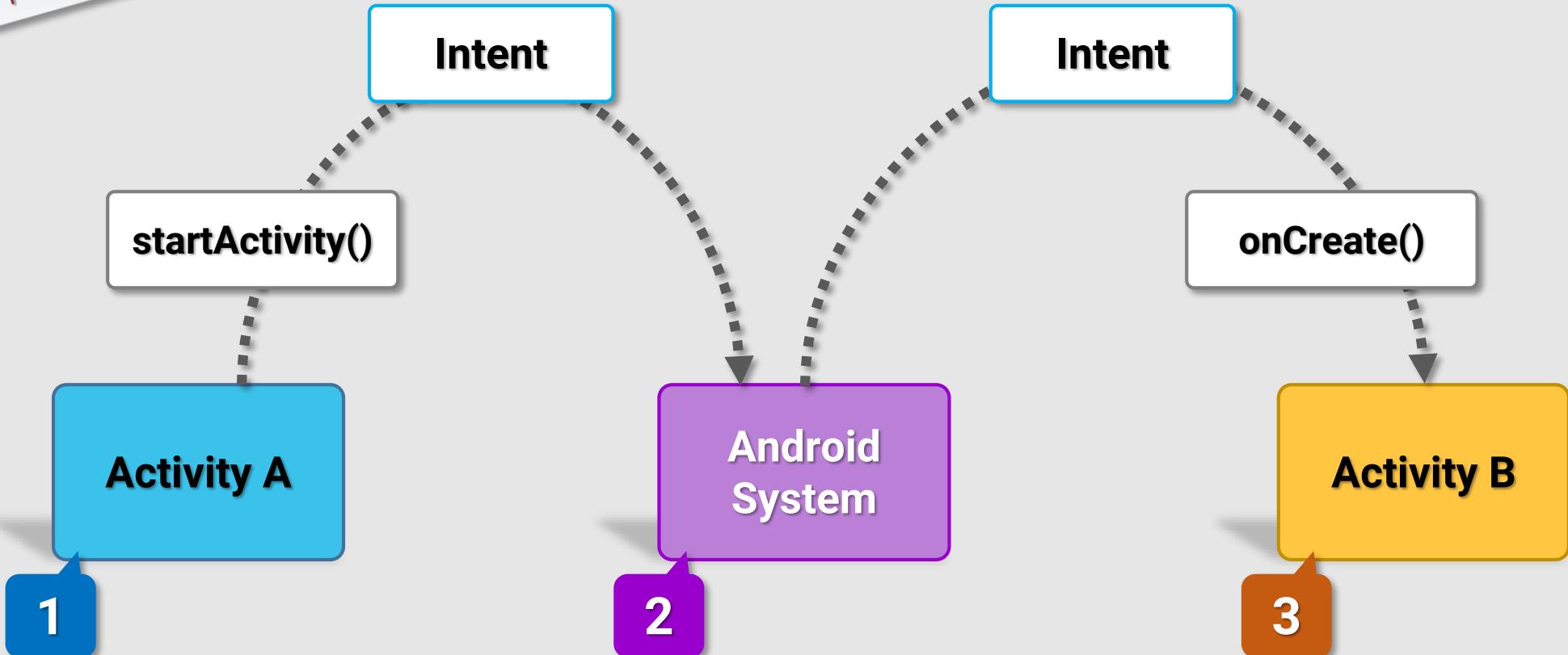
## Démonstration



# Les interactions

## Intent

Intent implicite



# Les interactions Intent Filter

intentManager

```
<activity android:name=".MainActivity">
 <intent-filter>
 <action android:name="android.intent.action.MAIN" />
 <category android:name="android.intent.category.LAUNCHER" />
 </intent-filter>
</activity>

<activity android:name=".LecteurPdfActivity" >
 <intent-filter>
 <action android:name="android.intent.action.VIEW" />
 <data android:mimeType="application/pdf"/>
 <category android:name="android.intent.category.DEFAULT" />
 </intent-filter>
</activity>

<activity android:name=".EditeurPdfActivity" >
 <intent-filter>
 <action android:name="android.intent.action.EDIT" />
 <data android:mimeType="application/pdf"/>
 <category android:name="android.intent.category.DEFAULT" />
 </intent-filter>
</activity>

<activity android:name=".PaintPngActivity">
 <intent-filter>
 <action android:name="android.intent.action.EDIT" />
 <data android:mimeType="image/png"/>
 <data android:mimeType="image/jpeg"/>
 <category android:name="android.intent.category.DEFAULT" />
 </intent-filter>
</activity>
```

Les interactions

# Utilisation d'un Intent implicite

## Démonstration



# Les interactions Permission

Présentation



# Les interactions Permission

Fonctionnement

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.android.app.myapp" >
 <uses-permission android:name="android.permission.RECEIVE_SMS" />
 <uses-permission android:name="android.permission.CAMERA" />
 <uses-permission android:name="android.permission.CALL_PHONE" />
 ...
 <application ... >
 ...
 </application>
</manifest>
```

Les interactions

# Gestion des permissions

## Démonstration

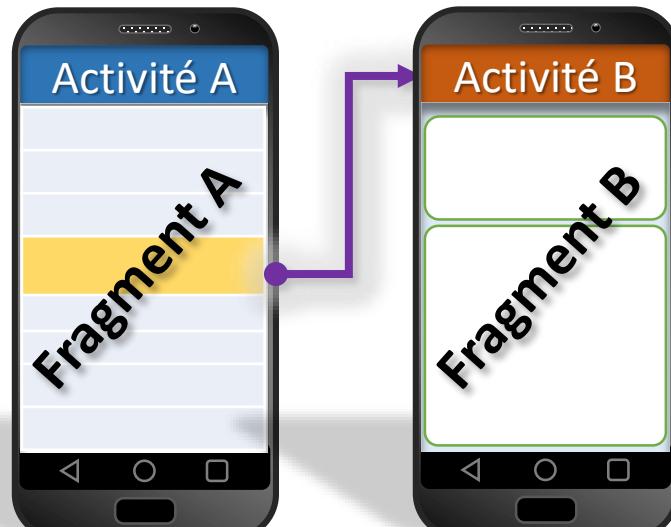


# Les interactions Fragment

Présentation

## SmartPhone

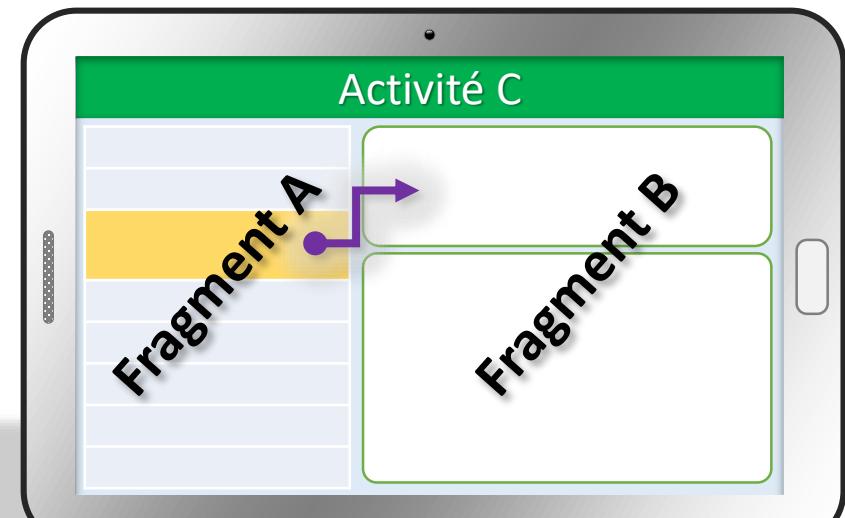
La sélection d'un élément dans le Fragment A exécute l'Activité B



L'Activité A contient un Fragment A, et l'Activité B contient un Fragment B.

## Tablette

La sélection d'un élément dans le Fragment A, met à jour le Fragment B

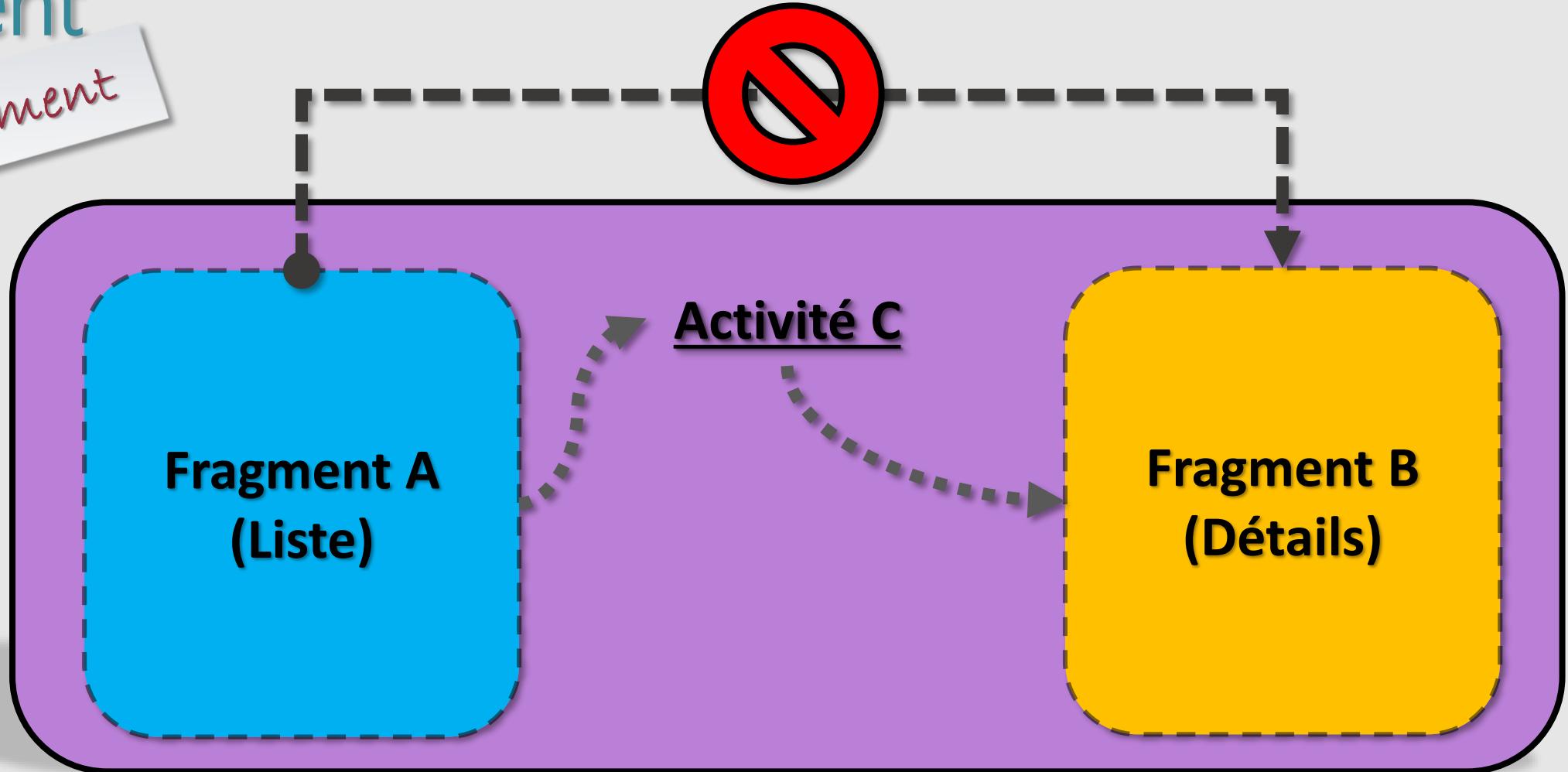


L'Activité C contient un Fragment A, et un Fragment B.

Les interactions

# Fragment

Fonctionnement



Les interactions

# Création de fragments

## Démonstration



# Conclusion

- Vous savez créer des activités
- Vous savez utiliser les logs
- Vous connaissez le concept des fragments
- Vous savez gérer des permissions

# Le développement d'une application mobile sous Android

**Module 4 - Les composants d'interactions**



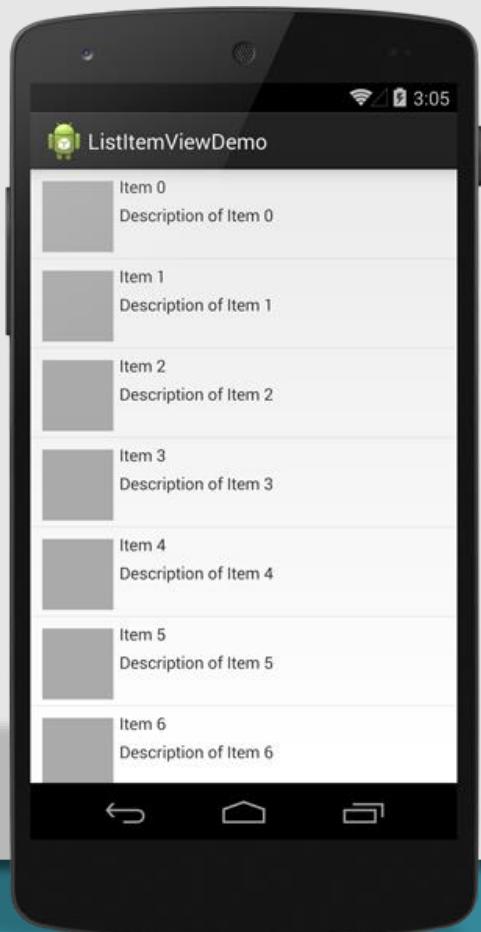
# Objectifs

- Savoir créer une liste
- Savoir créer un menu

Les composants d'interactions

# Liste : ListView

Présentation



- Mise en place d'une **ListView** en 4 étapes :
- 1. créer une **ListView** dans l'activité
- 2. créer un **Layout** de ligne
- 3. créer un **Adapter**
  - Héritage **ArrayAdapter<T>**
  - **Constructeur** : contexte, ressources, liste d'objet
  - Redéfinir **getView()**
- 4. relier l'**Adapter** à la **ListView**

Les composants d'interactions

# Création d'une liste simple

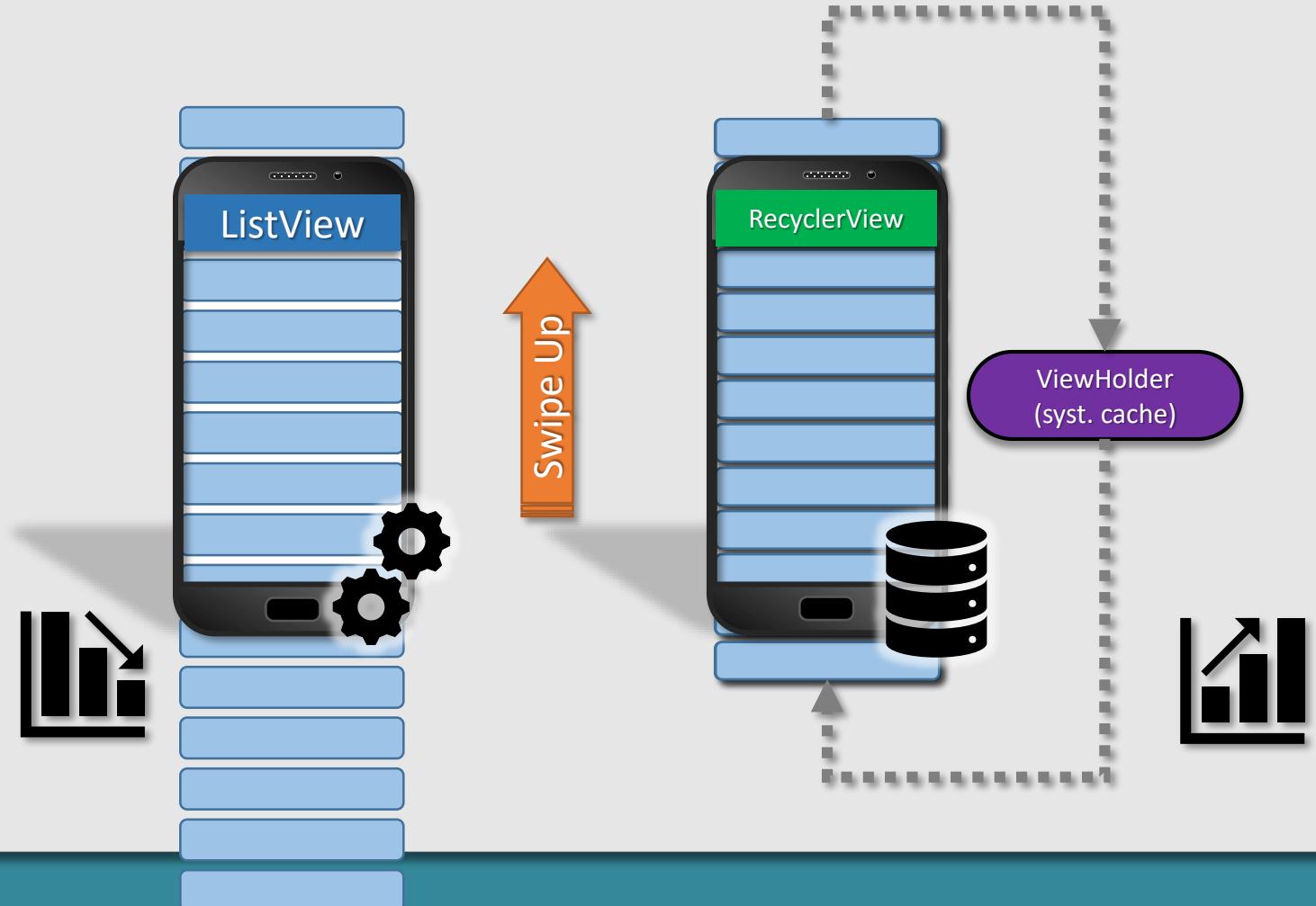
## Démonstration



# Les composants d'interactions

## Liste : RecyclerView

Présentation



Les composants d'interactions

# Création d'un RecyclerView

## Démonstration



Les composants d'interactions

# Créer un recycler view

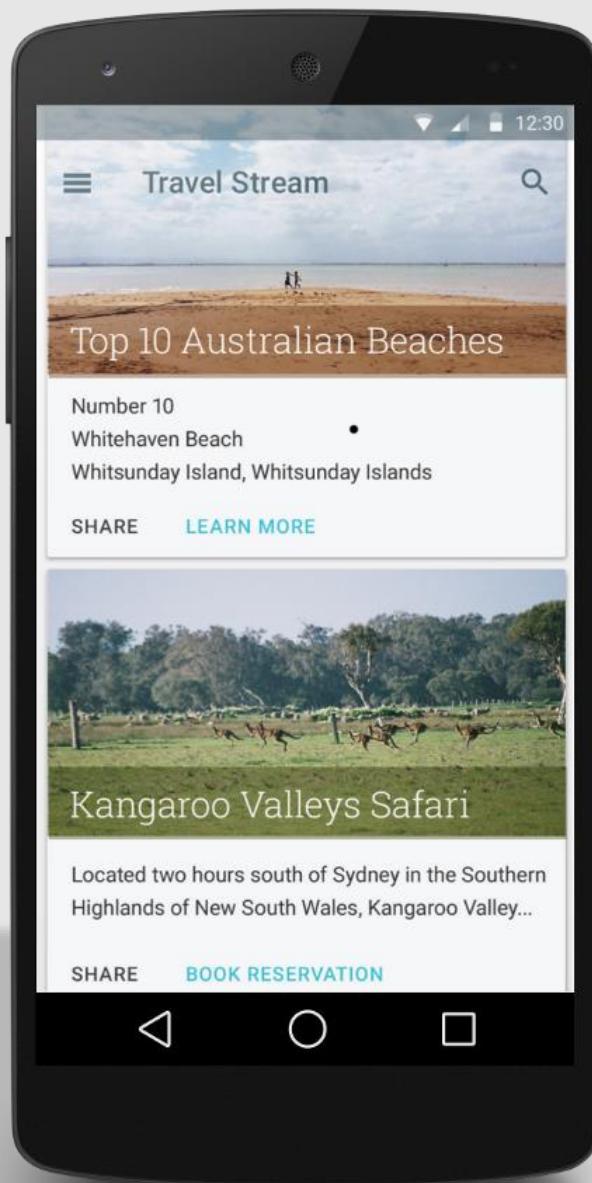
TP



# Les composants d'interactions

## CardView

Présentation



Les composants d'interactions  
Création d'une CardView

Démonstration



Les composants d'interactions

# Création d'une CardView

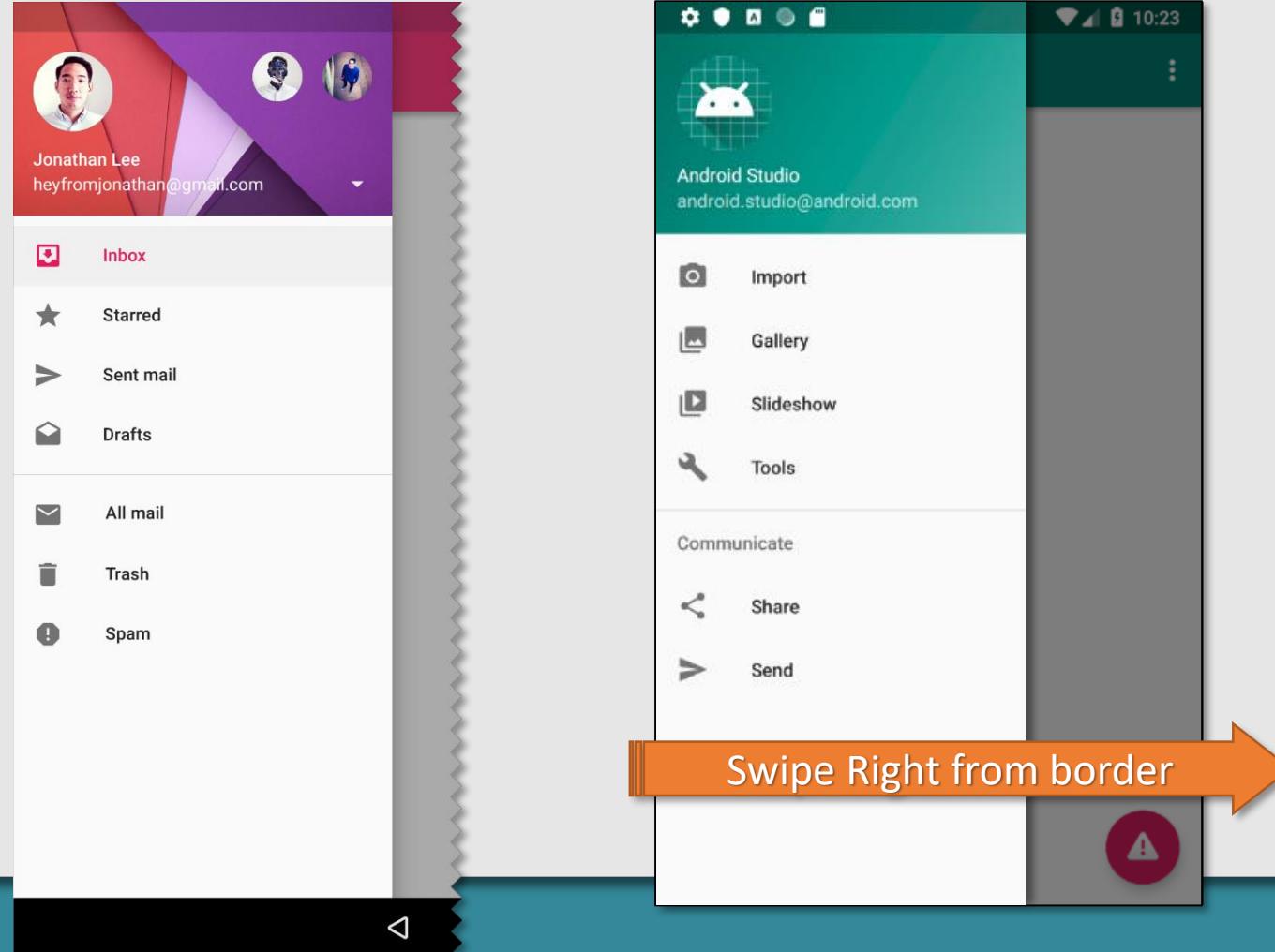
TP



# Les composants d'interactions

# NavigationDrawer

Présentation



Les composants d'interactions

# Utilisation d'un NavigationDrawer

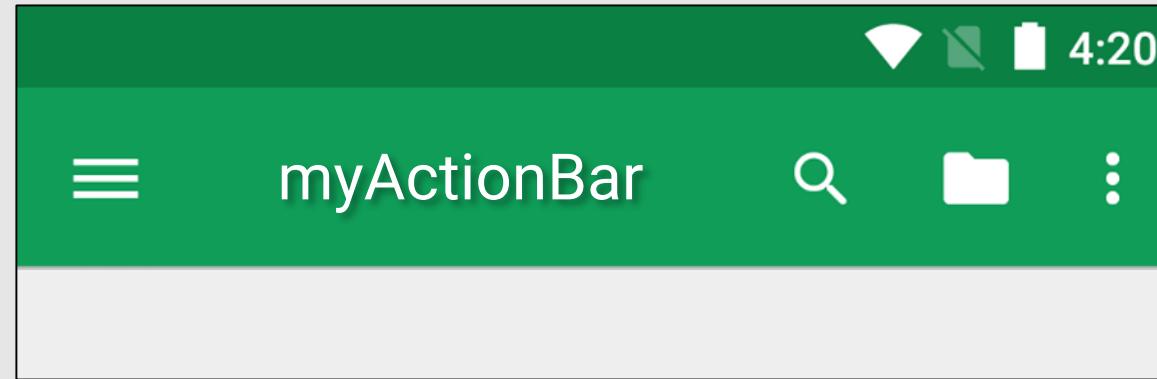
## Démonstration



Les composants d'interactions

# ActionBar

Présentation



# Les composants d'interactions

# Utilisation d'une ActionBar

## Démonstration



Les composants d'interactions

# ActionBar

TP



# Conclusion

- Vous savez créer une liste de manière optimisée
- Vous savez créer un menu

# Le développement d'une application mobile sous Android

**Module 5 - La persistance**

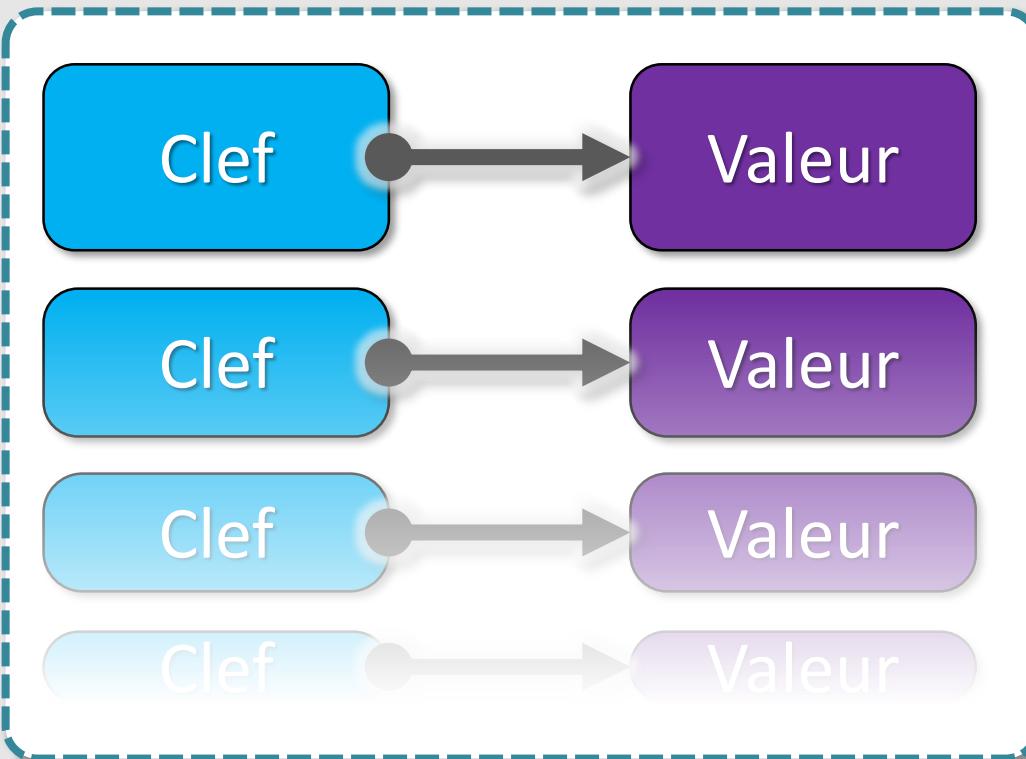


# Objectifs

- Connaître les différents mécanismes de persistance
- Savoir enregistrer des données de la meilleure façon possible

# La persistance SharedPreferences

Présentation



- `getPreferences`  
Préférences privées à une activité
- `getSharedPreferences`  
Préférences partagées entre toutes les activités

La persistance

# Utiliser les SharedPreferences

## Démonstration



La persistance

# Utiliser les SharedPreferences

TP



# La persistance Fichier brut

Présentation



- FileOutputStream

La persistance

Lire et écrire dans un fichier brut

# Démonstration



# La persistance Base de données

Présentation



- SQLiteOpenHelper
- SQLiteDatabase

La persistance

Base de Données : création et l'exploitation...

# Démonstration



La persistance

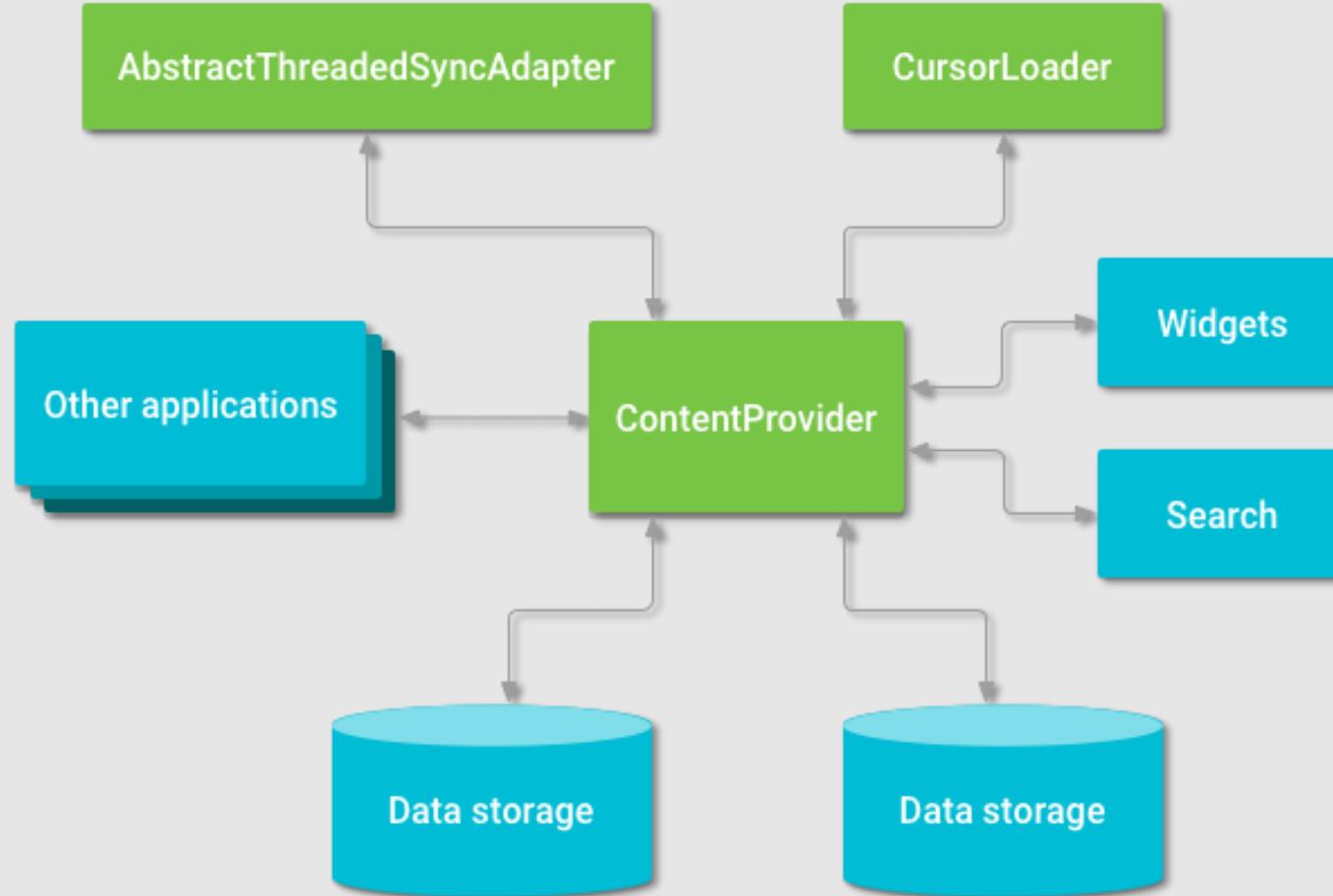
# Base de Données : création et l'exploitation...

TP



# La persistance ContentProvider

Présentation



La persistance

# Utiliser un ContentProvider

## Démonstration



La persistance

# Utilisation d'un Content Provider

TP



# Conclusion

- Vous savez utiliser les mécanismes Android de persistance

# Le développement d'une application mobile sous Android

**Module 6 - Les tâches asynchrones**



# Objectifs

- Savoir utiliser une ProgressBar
- Savoir utiliser des Threads
- Savoir utiliser des Handlers
- Savoir utiliser des AsyncTask

Les tâches asynchrones

# ProgressBar

Présentation

ProgressBar  
(Widget)

ProgressBar (Horizontal)



Les tâches asynchrones

# Utilisation d'une ProgressBar

## Démonstration



## Thread

Présentation

- Permet d'exécuter des tâches en parallèle
- Permet de ne pas bloquer l'IHM
- Possibilité de créer plusieurs threads

```
new Thread(new Runnable() {
 @Override
 public void run() {
 //Traitement parallèle...
 }
}).start();
```

N.B.: La méthode `.isAlive()` permet de savoir si le thread est encore en vie, afin d'éviter d'exécuter un thread plusieurs fois de suite, ce qui pourrait engendrer des exceptions.

# Les tâches asynchrones Utilisation d'un Thread

## Démonstration



Les tâches asynchrones

# Handler

Présentation

- Met à jour l'IHM
- Recevoir des messages de la part des threads secondaires
- Moyen de communication entre le thread principal et les threads secondaires

Les tâches asynchrones

# Utilisation de la classe Handler

## Démonstration



Les tâches asynchrones

# AsyncTask

Présentation

- Simplifier l'utilisation des Threads :
  - Une fonction pour l'initialisation
  - Une fonction pour le traitement
  - Une fonction pour la progression
  - Une fonction pour la finalisation

Les tâches asynchrones

# Utilisation de la classe AsyncTask

## Démonstration



## Conclusion

- Vous savez utiliser une ProgressBar
- Vous savez utiliser un Thread
- Vous savez utiliser un Handler
- Vous savez utiliser une AsyncTask

Les tâches asynchrones

# Utilisation des Tâches Asynchrones

TP



# Le développement d'une application mobile sous Android

**Module 7 - Les Services Web Distants**



# Objectifs

- Savoir exploiter des données au format JSON
- Communiquer avec un service web

Présentation

- **JSON : JavaScript Object Notation**
- Permet de stocker des données de manière structurée

*Format*

- Objet JSON : ensemble de couples nom/valeur non ordonnés
- Tableau JSON : ensemble de valeurs ordonnées
- Une valeur peut être :
  - Valeur atomique
  - Objet JSON
  - Tableau JSON

Les services web distants

# Exploitation d'un objet JSON

## Démonstration



Les services web distants

# Exploitation d'un tableau d'objets JSON

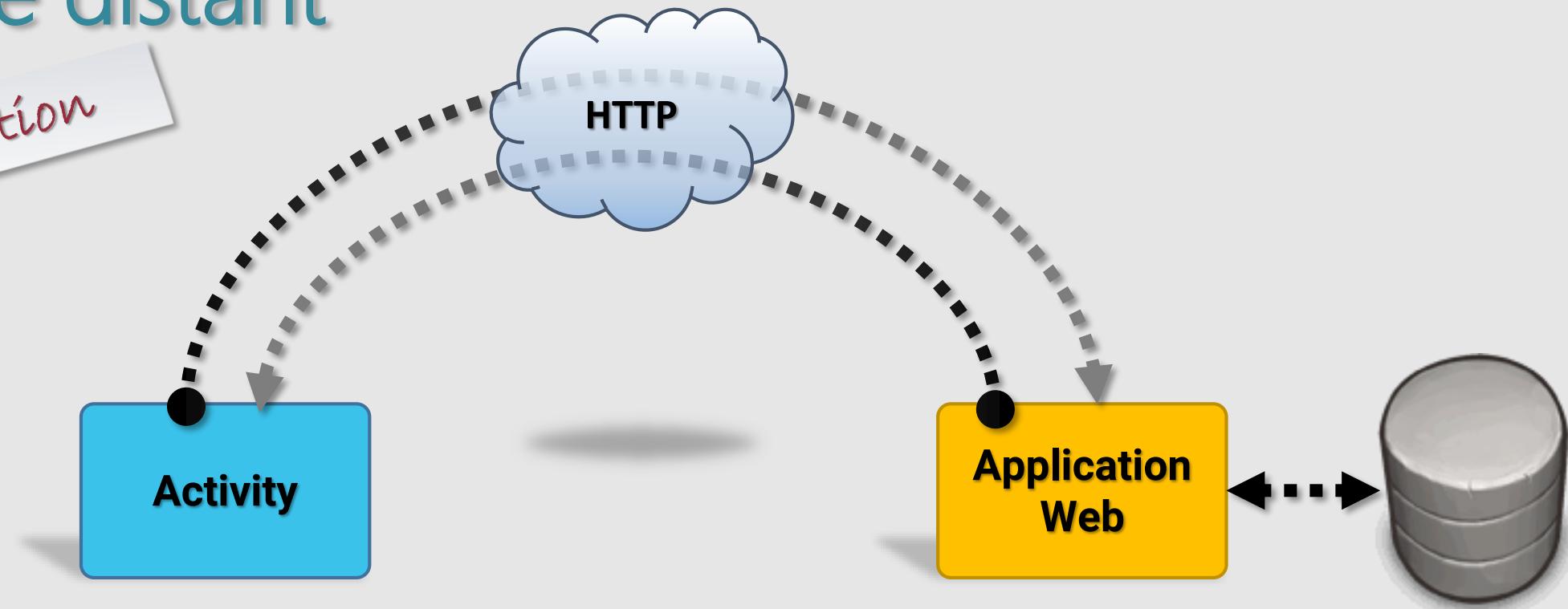
## Démonstration



Les services web distants

# Service distant

Présentation



*Android interdit l'accès à des sites distants depuis le Thread principal !*

Les services web distants

# Utilisation d'un service distant

## Démonstration



# Conclusion

- Vous savez exploiter des données au format JSON
- Vous savez communiquer avec un service web

# Le développement d'une application mobile sous Android

**Module 8 - Les services et broadcasts**



# Objectifs

- Savoir utiliser les Services
- Savoir utiliser les Intent Services
- Savoir utiliser les Broadcasts

# Les services (Service)

Présentation

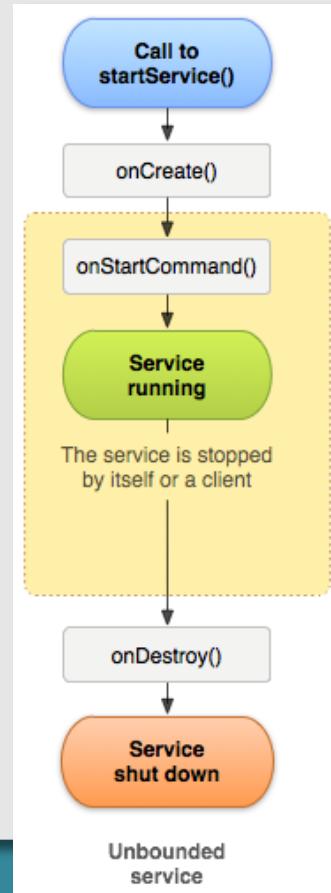
- Composant sans interface graphique fonctionnant en arrière-plan
- Objectif double :
  - Factorisation d'une fonctionnalité
  - Traitement long
- Utilisations possibles :
  - Utilisation avec communication unidirectionnelle
  - Utilisation avec communication bidirectionnelle

Les services et broadcasts

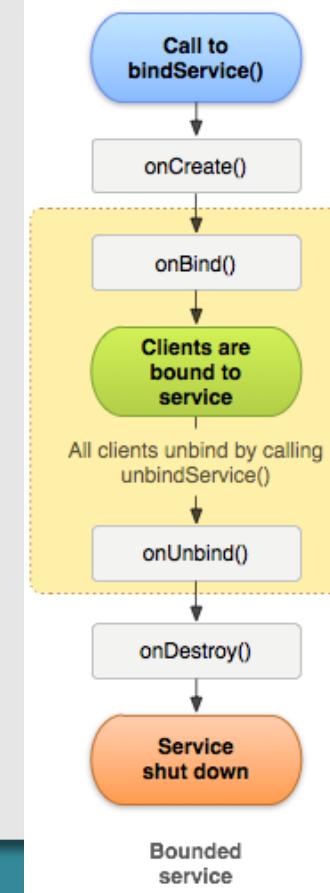
# Les modes de service (Bind)

cycle de vie

Mode Unidirectionnel



Mode Bidirectionnel



Les services et broadcasts

# Utilisation d'un Service

## Démonstration



# Les intent services (*IntentService*)

Présentation

- C'est un service particulier (la classe *IntentService* héritant de *Service*)
- Ses différents traitements sont exécutés dans un seul même *Thread* secondaire :
  - L'enchaînement des traitements sont réalisés avec un système de file d'attente (*spool*).

Les services et broadcasts

# Utilisation d'un Intent Service

## Démonstration



# Les broadcasts (BroadcastReceiver)

Présentation

- **Broadcasts :**
  - Évènements soumis à des abonnements.
- **BroadcastReceiver :**
  - Classe pour la Réception d'évènements
  - Exécute une seule action (***onReceive***),
  - Pas de cycle de vie particulier (création, exécution /inscription, destruction)
- **sendBroadcast :**
  - Envoyer un évènement (type *intent*).
  - Possibilité d'ajouter des informations supplémentaires (*putExtra*)

Les services et broadcasts

# Utilisation d'un Broadcast

## Démonstration



# Conclusion

- Vous savez créer des Services
- Vous savez utiliser les Intent Services
- Vous savez créer des Broadcasts



# Le développement d'une application mobile sous Android

**Module 9 - Room Persistence Library**



# Objectifs

- Comprendre le fonctionnement de Room
- Savoir mettre en place Room

Room Persistence Library

# Room

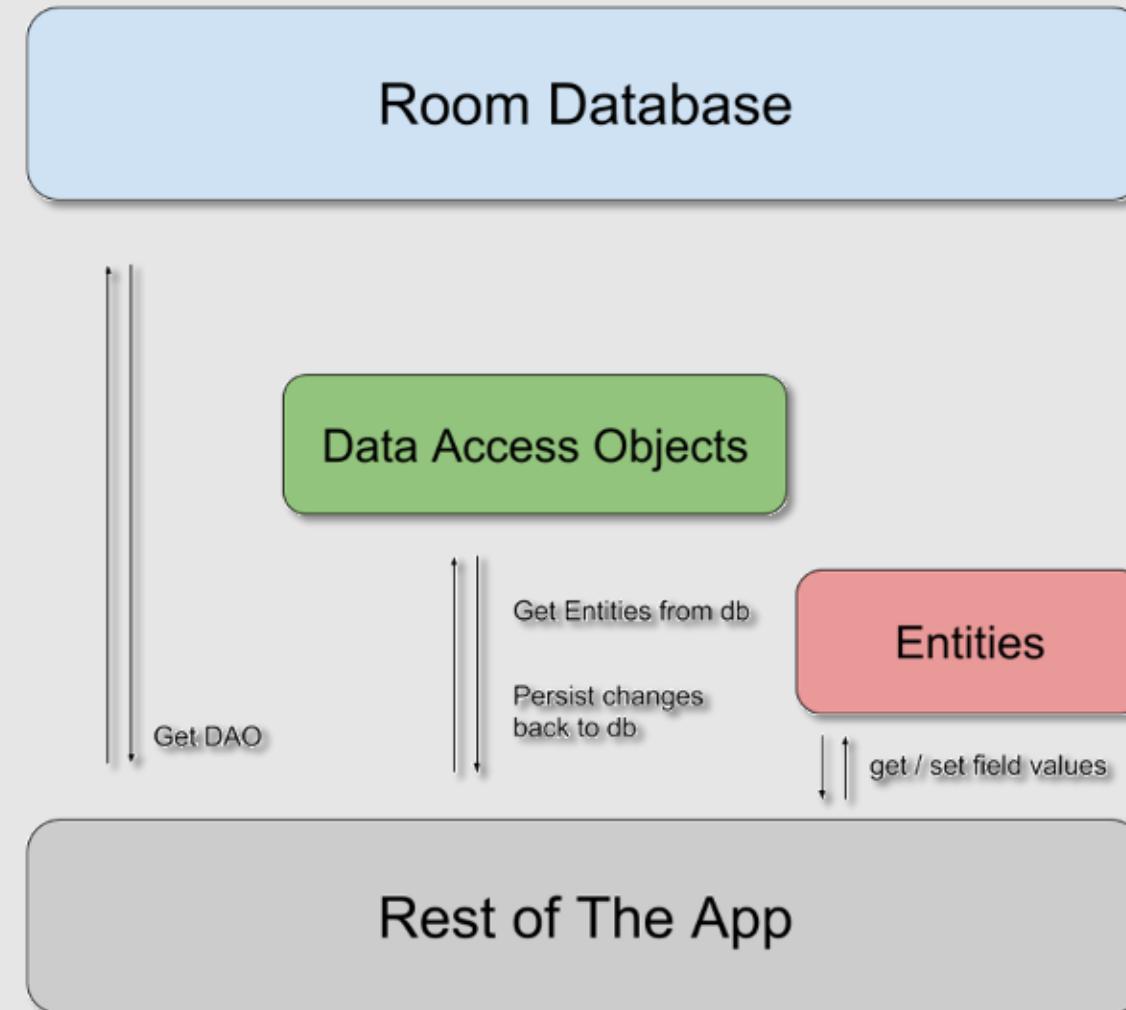
Présentation

- ORM officiel pour les applications Android

# Room Persistence Library

# Room

Fonctionnement



## Room

Mise en place

- Définir les classes entités avec des annotations
- Définir les classes DAO avec des annotations
- Définir la classe base de données avec des annotations
- Utiliser l'ORM

Room Persistence Library

# Mettre en place Room

## Démonstration



# Room

Avantages

- Facilité de développement
- Facilité de maintenance
- Meilleure productivité

N.B. : *Tous les appels aux méthodes Room (DataBase, DAO) doivent être réalisées depuis un Thread secondaire.*

## Conclusion

- Vous connaissez Room
- Vous connaissez les avantages à utiliser Room
- Vous savez mettre en place Room

# Le développement d'une application mobile sous Android

**Module 10 - Les WebView**



# Objectifs

- Savoir utiliser les WebView avec des pages web distantes
- Savoir utiliser les WebView avec des pages web locales

# WebView

Présentation

- Permet d'inclure des pages HTML :
  - page Web Distante
  - page Web Locale

Les WebView

Accès à une page web distante

# Démonstration



# Accès à une page Web Locale

Présentation

- Le code HTML se trouve dans l'application
- Possibilité d'utiliser du code natif (Java/Kotlin) avec JavaScript

*N.B. : La particularité se résume dans la possibilité d'utiliser du code natif depuis du code Javascript.*

Les WebView

Accès à une page web locale

# Démonstration



# Conclusion

- Vous savez accéder à des pages web distantes
- Vous savez accéder à des pages web locales



# Le développement d'une application mobile sous Android

**Module 11 - La géolocalisation**



# Objectifs

- Découvrir les bases avec Open Street Map (OSM)
- Mettre en place un premier projet simple utilisant OSM

# La géolocalisation **osmdroid**

Présentation

- Bibliothèque pour intégrer facilement Open Street Map dans une application
- <https://github.com/osmdroid/osmdroid/wiki>

# La géolocalisation osmdroid

Mise en place

- Télécharger la bibliothèque via Gradle
- Définir les permissions nécessaires
- Définir une MapView
- Utiliser osmdroid pour charger la MapView

La géolocalisation

Mettre en place OSM sur une application

# Démonstration



## Conclusion

- Vous savez mettre en place osmdroid
- Vous savez utiliser les fonctionnalités de base

# Le développement d'une application mobile sous Android

**Module 12 - Le design**



# Objectifs

- Modifier le style d'une application
- Créer un dégradé en fond d'écran
- Utiliser un Ripple Effect
- Créer une animation de rotation
- Créer une animation « diapositive »

# Le design Expérience utilisateur

Présentation



# Le design Expérience utilisateur

conseils

- Concentrez-vous sur l'objectif principal
- Mettez en relief le contenu attendu par les utilisateurs
- Pensez « de haut vers le bas »
- Donnez un chemin logique à suivre
- Faites en sorte que l'utilisation soit facile et naturelle
- Parlez naturellement à travers l'application
- Minimisez les efforts nécessaires pour les interactions
- Voyez le mobile comme un collaborateur (et pas seulement une machine)
- Soyez succinct
- Faites en sorte que toute recherche soit rapide et efficace
- Rédigez une description correcte de l'application,  
informez au maximum
- Utilisez les éléments graphiques de base dès que possible
- Contentez les utilisateurs avec de beaux visuels
- Prenez en charge le changement d'orientation du mobile
- Garantissez la prise en charge des gestuelles basiques et dédiées
- Demandez à sauvegarder uniquement lorsque nécessaire
- Optez pour un démarrage immédiat
- Soyez toujours prêt à ce que votre application soit stoppée
- Ne forcez pas l'application à quitter, c'est un choix de l'utilisateur

Le design

# Modifier le style d'une application

## Démonstration



Le design

# Créer un fond dégradé

## Démonstration



Le design  
Créer un Ripple Effect

Démonstration



Le design

# Créer une animation rotation

## Démonstration



Le design

# Créer une animation « diapositive »

## Démonstration



# Conclusion

- Vous savez personnaliser le style d'une application

# Le développement d'une application mobile sous Android

**Module 13 – La publication**



# Objectifs

- Connaitre les étapes permettant de publier une application



# La publication Google Play

Présentation

- Android Store Application



La publication

# Etapes pour publier

- Créer un compte développeur
- Exporter l'application
- Envoyer sur Google Play



# La publication Suivi et mise à jour

Google play | Developer Console PREVIEW

blu

All applications Reports Settings

Add new application

## ALL APPLICATIONS

APP NAME	PRICE	ACTIVE INSTALLS	AVG. RATING / TOTAL	ERRORS	LAST UPDATE	STATUS
The big blue	Free	12	★ 5.00 / 1	0	Aug 15, 2012	Unpublished
The Handy Developer Guide	Free	756	★ 5.00 / 2	6	Sep 26, 2012	Published
The big green	Free			—	—	Draft
The big red	\$2.00	136	—	14	Dec 3, 2010	Published
The big yellow	Free	3,672,387	★ 5.00 / 1	119	Jan 18, 2012	Unpublished
The little pink	Free	7,452,652	★ 5.00 / 1,986,412	8	Jun 14, 2012	Published
The little red	Free	2,412	★ 3.33 / 335	341	Dec 8, 2010	Published

Page 1 of 7

# Conclusion

- Vous savez comment publier une application

