

# Formation Android

Épisode I



# Xavier MARIN



@XavMarin

<https://github.com/Giwi>

CTO chez @qaobee

# Objectifs de la formation

Maîtriser et être autonome sur les éléments suivants:

- Activity
- Fragment
- AsyncTask
- View & Layout

Les bonus:

- Material Design
- Stockage de données
- Gestion des différents écrans



# Android: c'est quoi?



## APPLICATIONS

Home

Contacts

Phone

Browser

...

## APPLICATION FRAMEWORK

Activity  
Manager

Window  
Manager

Content  
Providers

View  
System

Notification  
Manager

Package  
Manager

Telephony  
Manager

Resource  
Manager

Location  
Manager

XMPP  
Service

## LIBRARIES

Surface  
Manager

Media  
Framework

SQLite

OpenGL|ES

FreeType

WebKit

SGX

SSL

libc

## ANDROID RUNTIME

Core  
Libraries

Dalvik Virtual  
Machine

## LINUX KERNEL

Display  
Driver

Camera  
Driver

Bluetooth  
Driver

Flash Memory  
Driver

Binder (IPC)  
Driver

USB  
Driver

Keypad  
Driver

WiFi  
Driver

Audio  
Drivers

Power  
Management

# Dalvik?

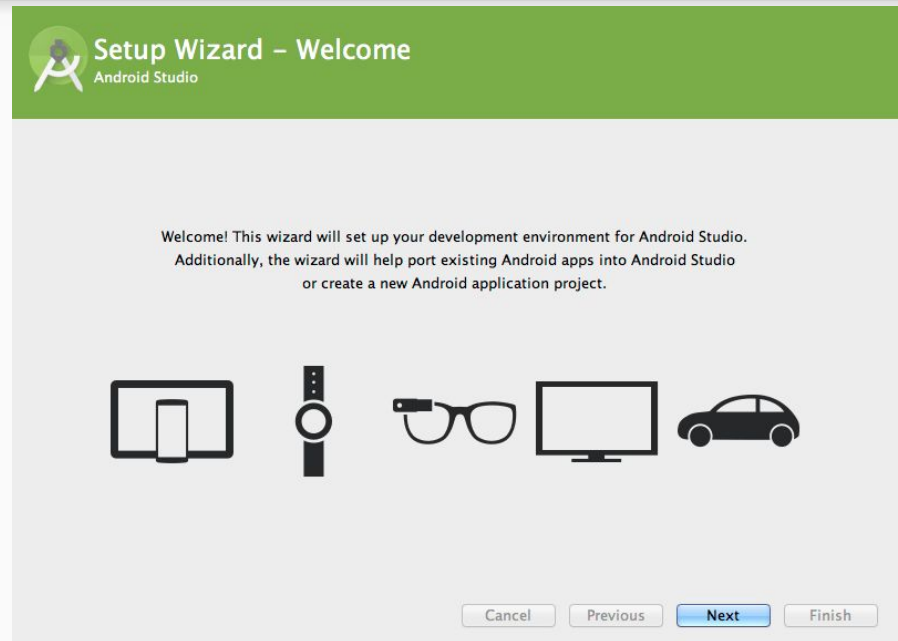
## Machine virtuelle

- != JVM
- Exécute des fichiers .dex
- Espace d'adressage 32 bits



# On développe pour Android

Pour quel matériel?



Téléphones



Tablettes



Wear



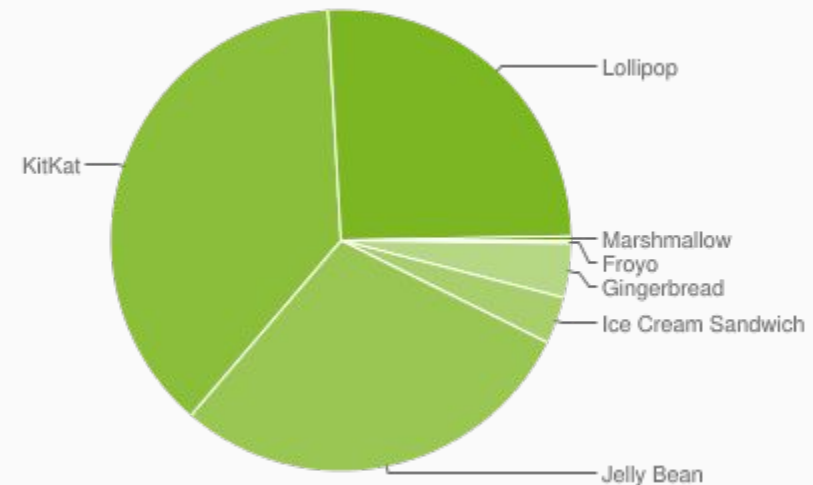
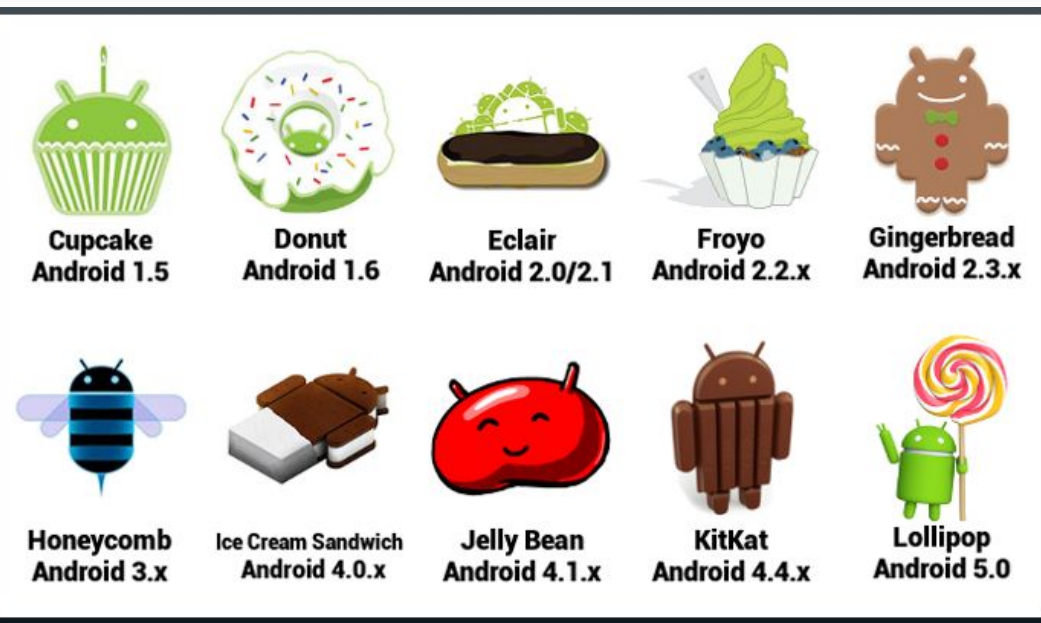
TV



Auto

# Vous avez déjà entendu parler de la fragmentation?

Novembre 2015





# Quels sont les outils de développements?



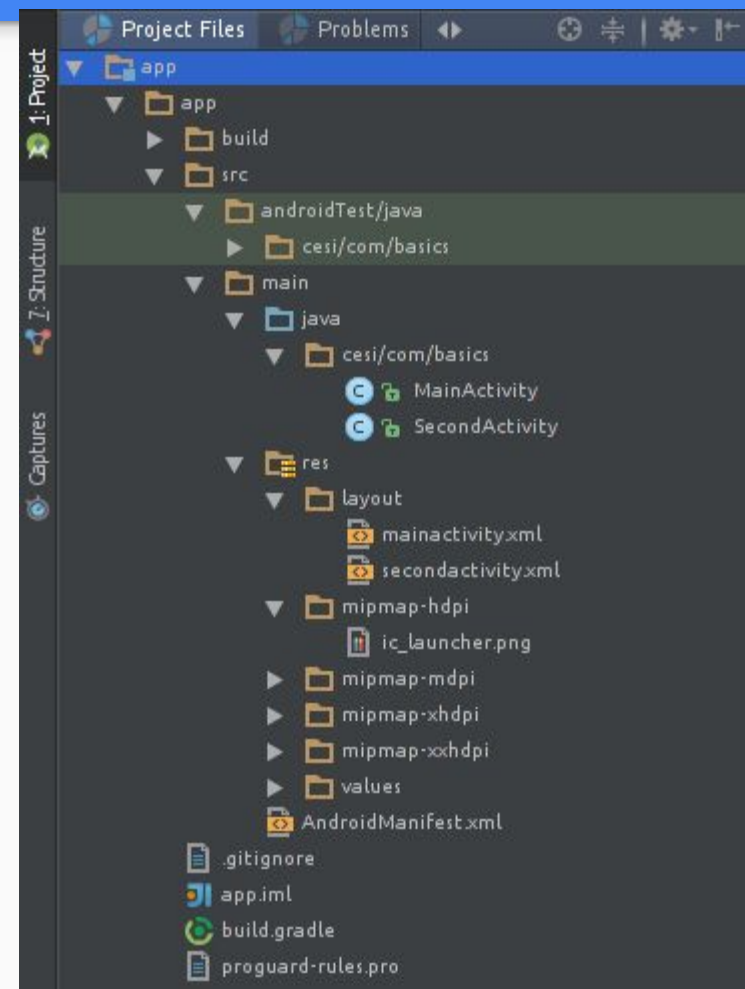
Powered by the IntelliJ Platform





# Les fondements des apps

- Génération d'un APK
- Les apps tournent dans une sandbox
  - User, VM et processus isolés par app
  - Principe du monde privilège
- Communication entre applications



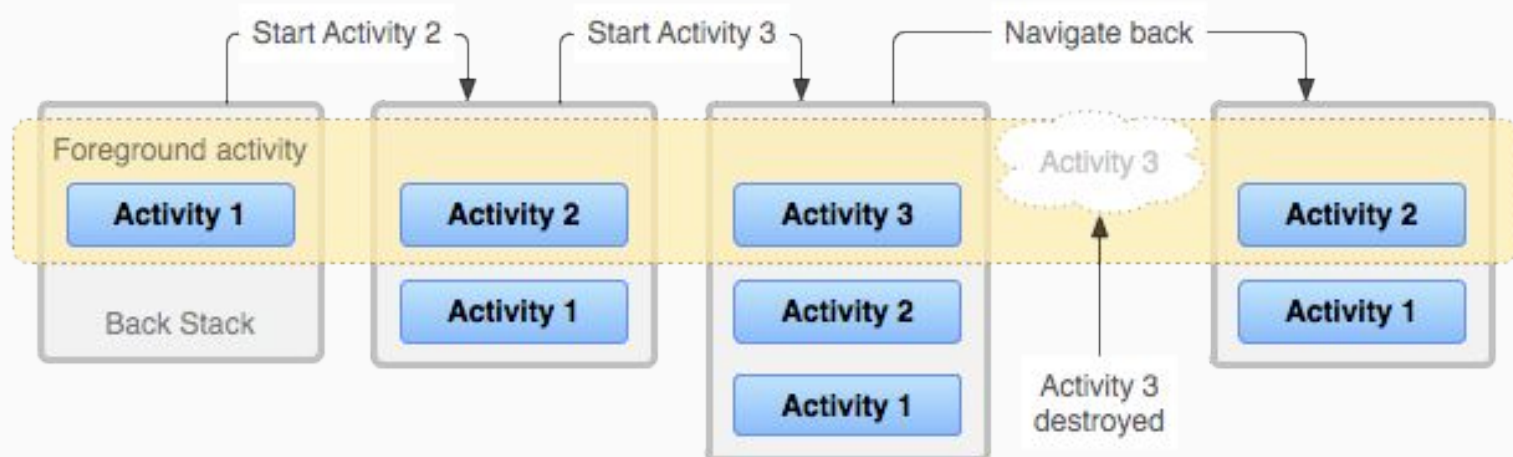
# Les fondements des apps

- Activities
- Services
- Intent
- Processes & Threads
- Permissions
- Manifest

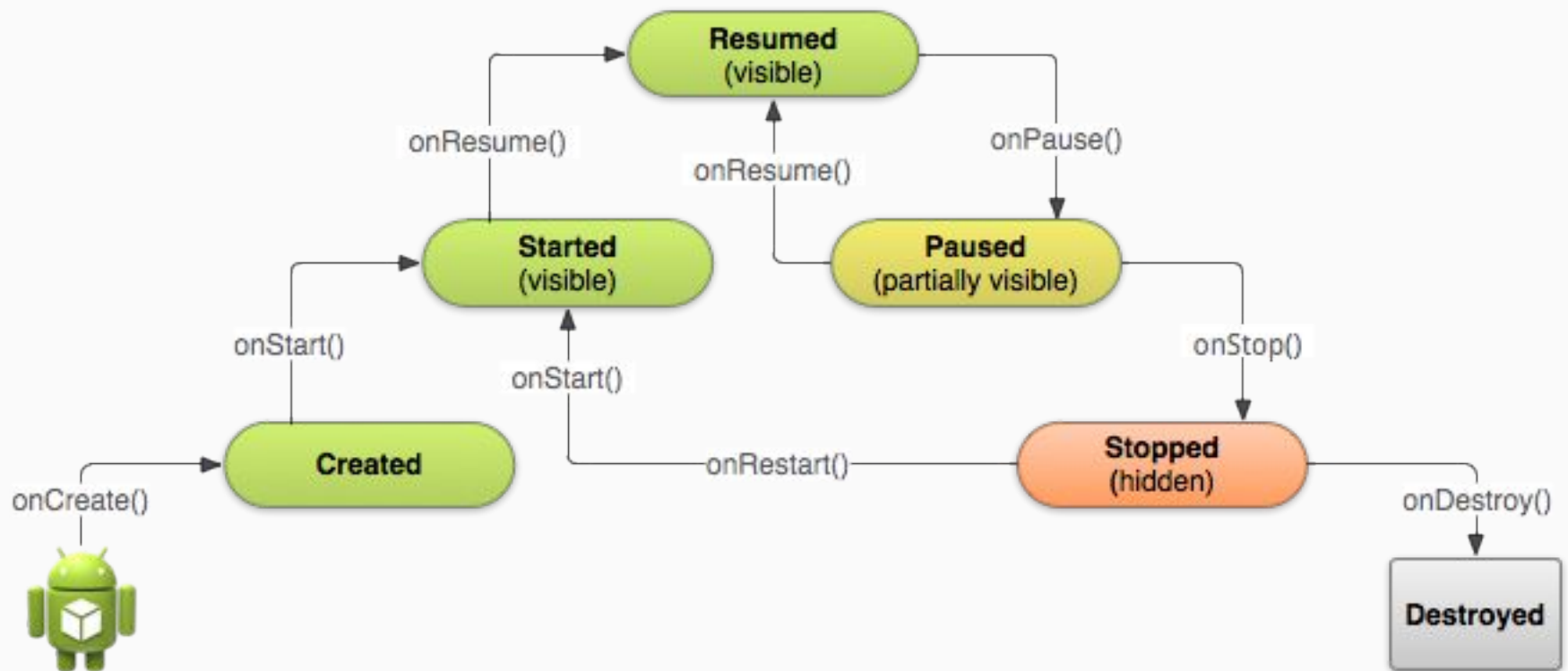


# Activity

- Assimilable à un écran
- Composée de :
  - Contrôleur : code Java
  - Héritage de Activity
- Vue : XML (possible en Java)
- Doit être déclarée dans le fichier manifest
- Une activité par écran
- Une seule activité à un instant T
- Il existe une pile d'activité



# Activity : Le cycle de vie



# Activity : Sauvegarde du contexte

- Pourquoi?
  - Rotation d'écran
  - Bouton back
  - etc ...
- OnSavedInstance
  - Défaut : sauvegarde tous les champs avec id
- Récupération du contexte dans le onCreate



Bon, on code?



# Atelier 0

2 activités

- 2 vues
- 1 Intent pour changer d'activité
  - en passant un String

Une dernière petite aide?

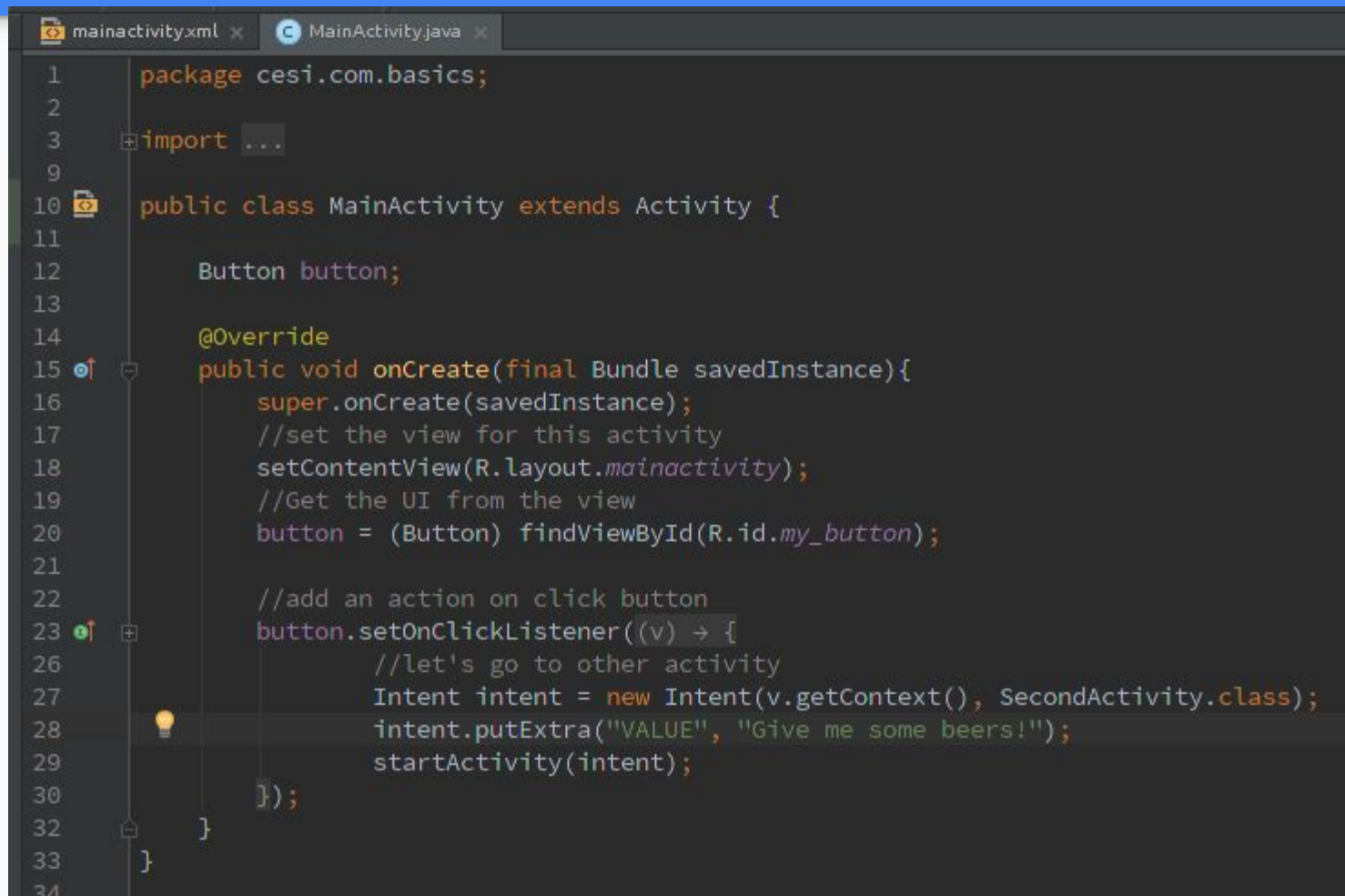


# Atelier 0 : aide

Vue : les layouts

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:gravity="center"
6     android:orientation="vertical">
7
8     <Button
9         android:id="@+id/my_button"
10        android:layout_width="wrap_content"
11        android:layout_height="wrap_content"
12        android:text="@string/text_btn" />
13
14 </LinearLayout>
```

# Atelier 0 : aide

A screenshot of an IDE window showing the MainActivity.java file. The window has two tabs: 'mainactivity.xml' and 'MainActivity.java'. The code is written in Java and includes package declarations, imports, and the MainActivity class which extends Activity. The onCreate method is overridden, and it contains logic to find a button by ID and set an onClick listener that starts a new activity named SecondActivity. The code is color-coded, and there are some UI elements like a lightbulb icon and a cursor visible in the editor.

```
1 package cesi.com.basics;
2
3 import ...
4
5
6
7
8
9
10 public class MainActivity extends Activity {
11
12     Button button;
13
14     @Override
15     public void onCreate(final Bundle savedInstanceState){
16         super.onCreate(savedInstanceState);
17         //set the view for this activity
18         setContentView(R.layout.mainactivity);
19         //Get the UI from the view
20         button = (Button) findViewById(R.id.my_button);
21
22         //add an action on click button
23         button.setOnClickListener((v) -> {
24             //let's go to other activity
25             Intent intent = new Intent(v.getContext(), SecondActivity.class);
26             intent.putExtra("VALUE", "Give me some beers!");
27             startActivity(intent);
28         });
29     }
30 }
31
32
33 }
```

# Atelier 0 : aide

## Le manifest : contient activities et permissions

mainactivity.xml x MainActivity.java x AndroidManifest.xml x

manifest application

```
1 <manifest xmlns:android="http://schemas.android.com/apk/res/android" package="cesi.com.basics">
2   <application android:allowBackup="true" android:label="@string/app_name"
3     android:icon="@mipmap/ic_launcher" android:theme="@style/AppTheme">
4     <activity
5       android:name="cesi.com.basics.MainActivity"
6       android:label="@string/app_name"
7       android:icon="@mipmap/ic_launcher">
8       <intent-filter>
9         <action android:name="android.intent.action.MAIN" />
10        <category android:name="android.intent.category.LAUNCHER" />
11      </intent-filter>
12    </activity>
13  </application>
14 </manifest>
```

A votre tour !

