

Formation Android

Épisode IV et V



WRONG DROID

Xavier MARIN



@XavMarin

<https://github.com/Giwi>

CTO chez @qaobee

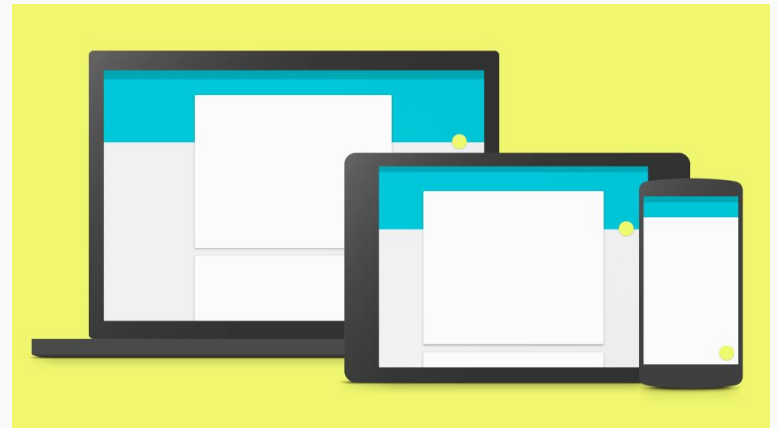
Objectifs de l'atelier

Maitriser et être autonome sur les éléments suivants:

- Découvrir le Material Design
- Composants graphiques avancés

Le Material Design

- Introduit avec Android L preview
- Ensemble de composants, APIs et guidelines
- Plus seulement sur Android

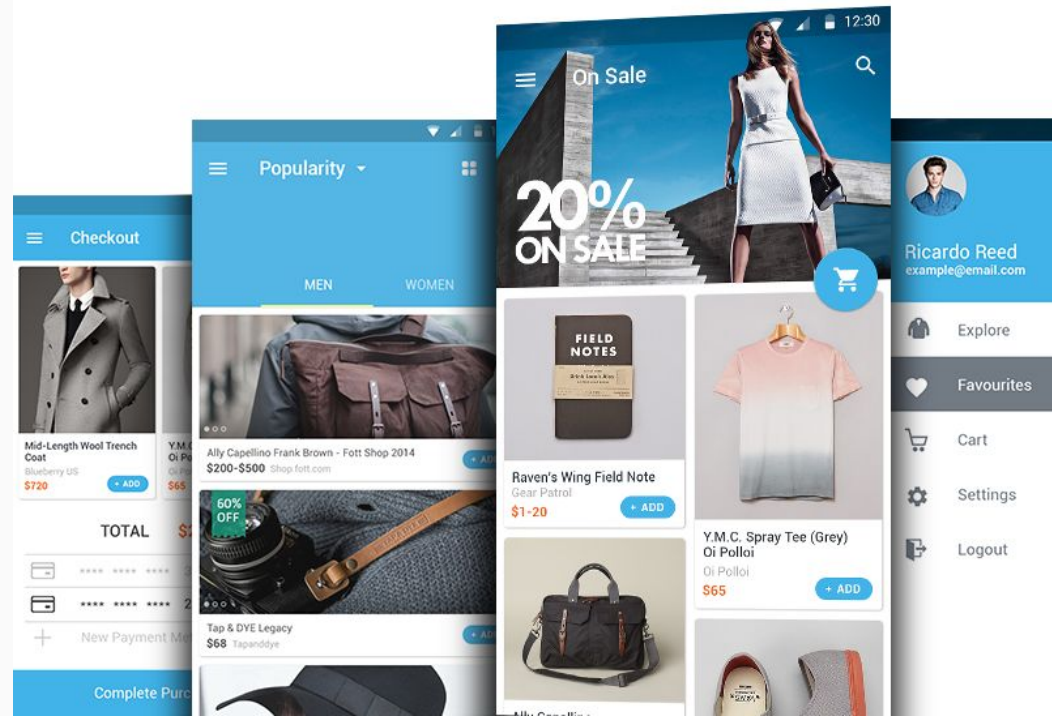


<http://developer.android.com/training/material/index.html>
<http://www.materialup.com/>

Le Material Design

Objectifs :

- Synthétiser les principes d'un bon design
- Fournir une expérience unifiée



Le Material Design

Les composants :

<http://www.google.com/design/spec/components/>

Bottom sheets

Buttons

Buttons: Floating Action Buttons

Cards

Chips

Data tables

Dialogs

Dividers

Grid lists

Lists

Lists: Controls

Menus

Pickers

Progress & activity

Selection controls

Sliders

Snack bars & toasts

Subheaders

Steppers

Le Material Design

Oui, mais c'est pour Android Lollipop ...

Le Material Design

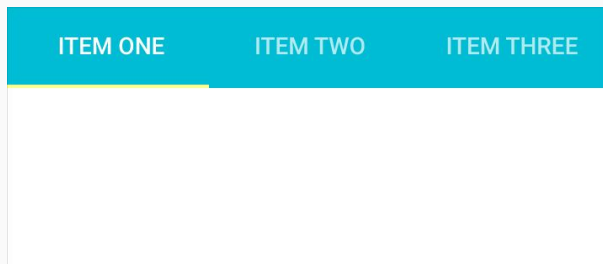
et non !!!

Android Design support Library

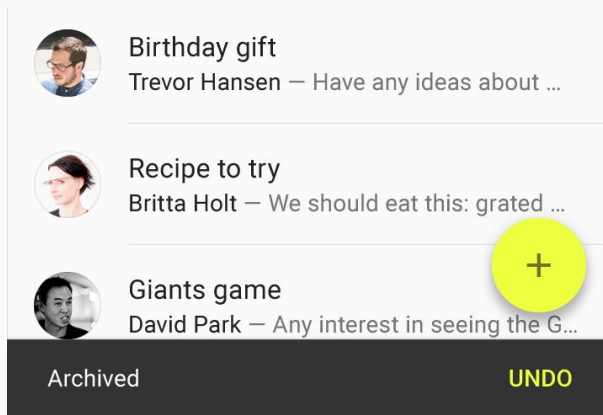
Merci Google

Le Material Design: les composants

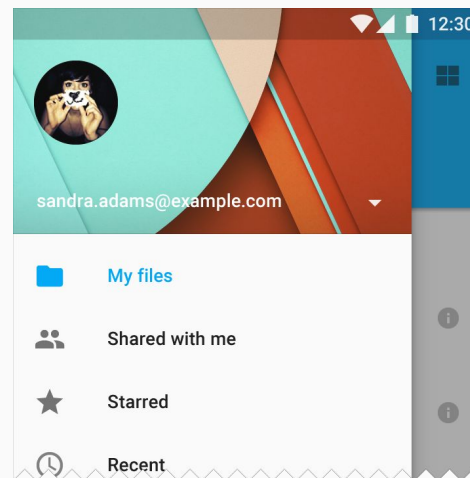
Tabs



Snackbar



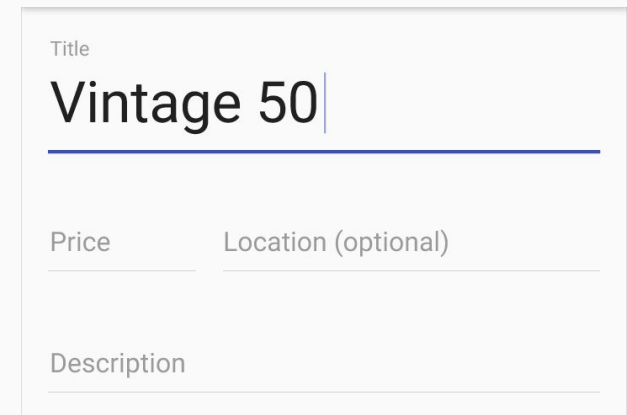
Navigation Drawer



Floating Action Button

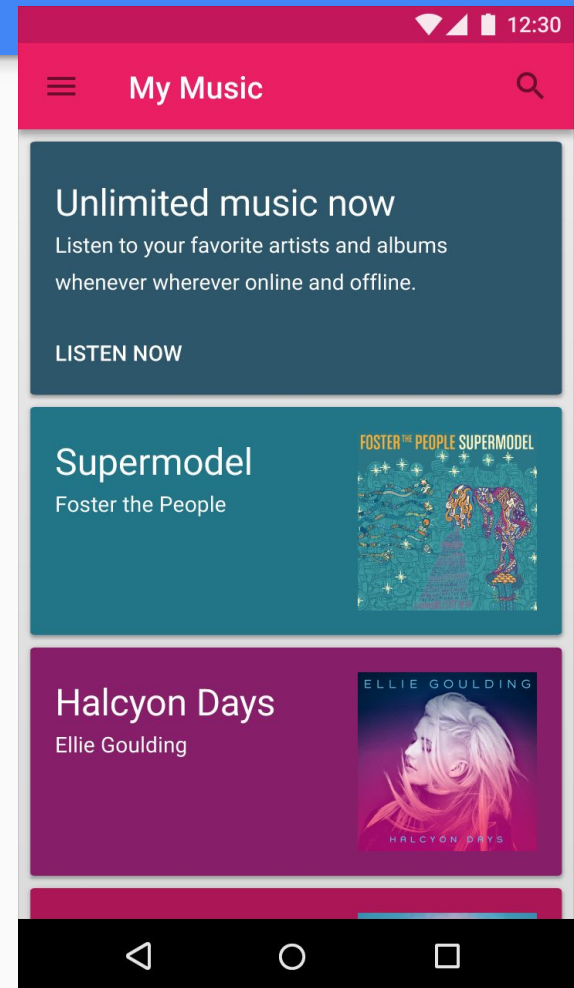


Floating label



Le Material Design

- Et aussi Android Support v7
- Cards & RecyclerView



Le Material Design

Comment l'utiliser?

```
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    compile 'com.android.support:design:22.2.0'  
    compile 'com.android.support:appcompat-v7:22.0.0'  
    compile 'com.android.support:support-v4:22.+'  
    compile 'com.android.support:cardview-v7:+'  
    compile 'com.android.support:recyclerview-v7:+'  
}
```

Le Material Design

Le style

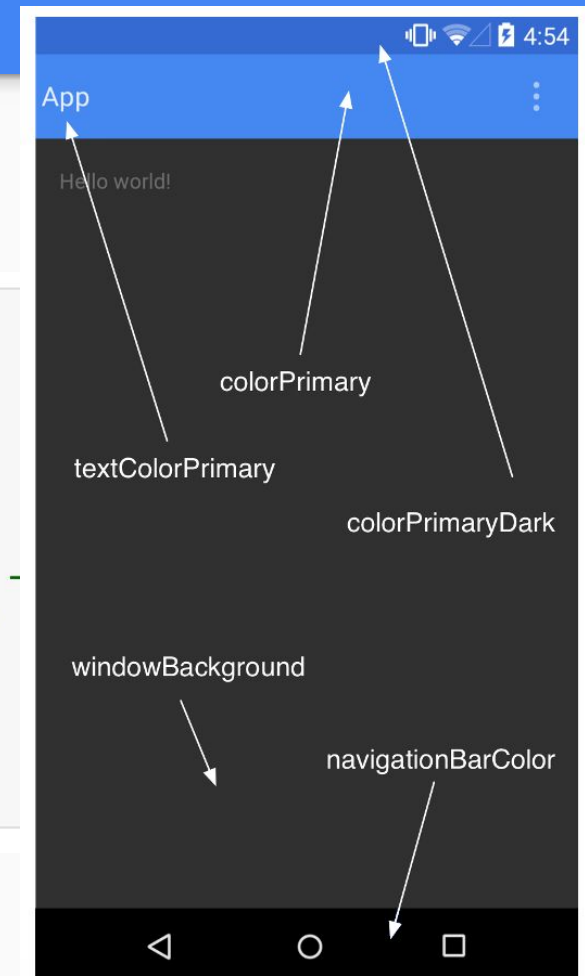
Le theme Theme.Material est disponible mais non compatible avant API21

```
<style name="Base.Theme.TchatApp" parent="Theme.AppCompat.Light.NoActionBar">
    <item name="colorPrimary">@color/colorPrimary</item>
    <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
    <item name="colorAccent">@color/colorAccent</item>
    <item name="android:windowBackground">@color/background</item>
</style>
```

Le Material Design

Les couleurs

```
<resources>
  <!-- inherit from the material theme -->
  <style name="AppTheme" parent="android:Theme.Material">
    <!-- Main theme colors -->
    <!-- your app branding color for the app bar -->
    <item name="android:colorPrimary">@color/primary</item>
    <!-- darker variant for the status bar and contextual app bars ->
    <item name="android:colorPrimaryDark">@color/primary_dark</item>
    <!-- theme UI controls like checkboxes and text fields -->
    <item name="android:colorAccent">@color/accent</item>
  </style>
</resources>
```



Le Material Design

Et sur notre appli TwitterLike, ça ressemblerait à quoi?



A word cloud featuring various verbs related to design, renovation, and reuse. The words are arranged in a roughly rectangular shape, with 'REIMAGINE' being the largest and most prominent word in the center. Other words include 'RECONSTRUCT', 'REMAKE', 'RECYCLE', 'REJUVENATE', 'REVAMP', 'REDESIGN', 'RENOVATE', 'RETHINK', 'REUSE', 'RECREATE', 'REFURBISH', 'RESTORE', 'RENEW', 'REMODEL', 'REDO', and 'REVISE'. Each word is in a different color and font size, creating a vibrant and dynamic visual effect.

RECONSTRUCT REMAKE
RECYCLE REJUVENATE
REVAMP REDESIGN RENOVATE
REIMAGINE
RETHINK REUSE RECREATE
REFURBISH RESTORE RENEW
REMODEL REDO REVISE

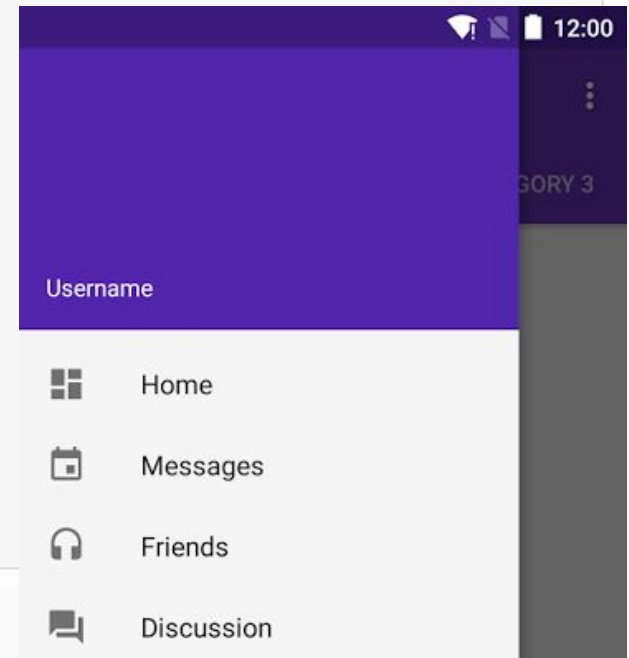
DrawerLayout et NavigationView

Activité principale

```
<android.support.v4.widget.DrawerLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true">

    <!-- your content layout -->

    <android.support.design.widget.NavigationView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        app:headerLayout="@layout/drawer_header"
        app:menu="@menu/drawer" />
</android.support.v4.widget.DrawerLayout>
```



DrawerLayout et NavigationView

Menu

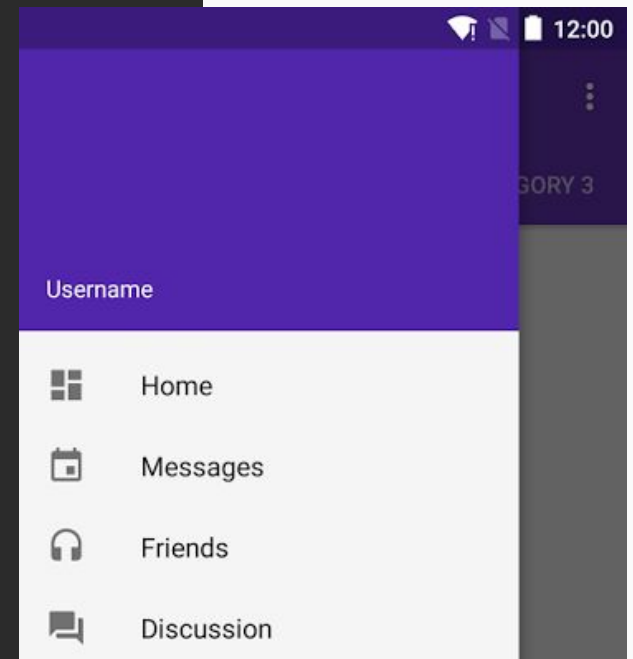
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <item android:id="@+id/tchat_disconnect"
        android:icon="@android:drawable/ic_menu_delete"
        android:title="@string/disconnect"
        android:showAsAction="always"/>

    <item android:id="@+id/tchat_users"
        android:icon="@android:drawable/ic_menu_delete"
        android:title="@string/users"
        android:showAsAction="always"/>

    <item android:id="@+id/tchat_tchat"
        android:title="refresh"
        android:icon="@android:drawable/ic_menu_rotate"
        android:showAsAction="always"/>

</menu>
```



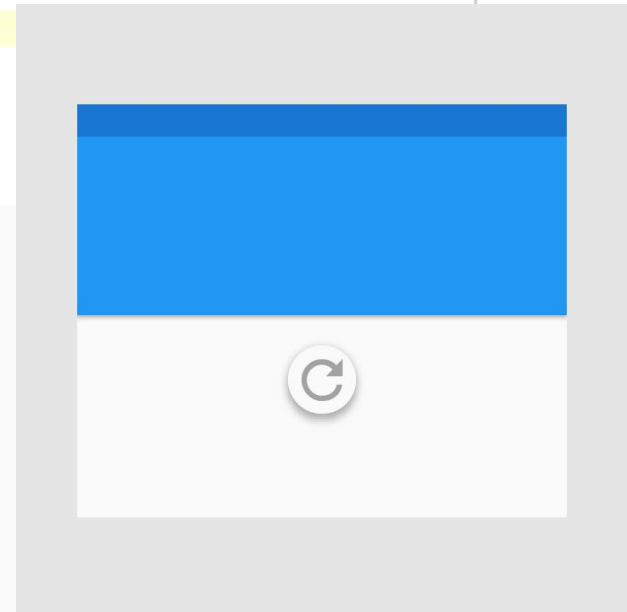
DrawerLayout et NavigationView

```
/**
 * setup drawer.
 * @param navigationView
 */
private void setupNavigationView(NavigationView navigationView) {
    navigationView.setNavigationItemSelectedListener(
        new NavigationView.OnNavigationItemSelectedListener() {
            @Override
            public boolean onNavigationItemSelected(MenuItem menuItem) {
                if(menuItem.getItemId() == R.id.tchat_disconnect){
                    token = null;
                    TchatActivity.this.finish();
                } if(menuItem.getItemId() == R.id.tchat_users){
                    Intent i = new Intent(TchatActivity.this, UsersActivity.class);
                    i.putExtra(Constants.INTENT_TOKEN, token);
                    startActivity(i);
                } else {
                    menuItem.setChecked(true);
                    mDrawerLayout.closeDrawers();
                }
                return true;
            }
        });
}
```

SwipeRefreshLayout

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v4.widget.SwipeRefreshLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/users_swiperefresh"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:fitsSystemWindows="true" />
</android.support.v4.widget.SwipeRefreshLayout>
```



SwipeRefreshLayout

```
private void setupRefreshLayout() {  
    swipeLayout.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {  
        @Override  
        public void onRefresh() {  
            loading();  
        }  
    });  
    swipeLayout.setColorSchemeResources(R.color.colorAccent, R.color.colorPrimaryDark, R.color.colorPrimary);  
    /**  
     * this is to avoid error on double scroll on listview/swipeRefreshLayout  
     */  
    list.setOnScrollListener(new AbsListView.OnScrollListener() {  
        @Override  
        public void onScrollStateChanged(AbsListView view, int scrollState) {  
        }  
        @Override  
        public void onScroll(AbsListView view, int firstVisibleItem, int visibleItemCount, int totalItemCount) {  
            boolean enable = false;  
            if (list != null && list.getChildCount() > 0) {  
                // check if the first item of the list is visible  
                boolean firstItemVisible = list.getFirstVisiblePosition() == 0;  
                // check if the top of the first item is visible  
                boolean topOfFirstItemVisible = list.getChildAt(0).getTop() == 0;  
                // enabling or disabling the refresh layout  
                enable = firstItemVisible && topOfFirstItemVisible;  
            }  
            swipeLayout.setEnabled(enable);  
        }  
    });  
}
```

Bon, on code?



A faire

- `NavigationDrawer`
- `SwipeRefreshLayout`



A votre tour !

<https://cesi-giwisoft.rhcloud.com/apidoc/>

