




ActionBar Activity

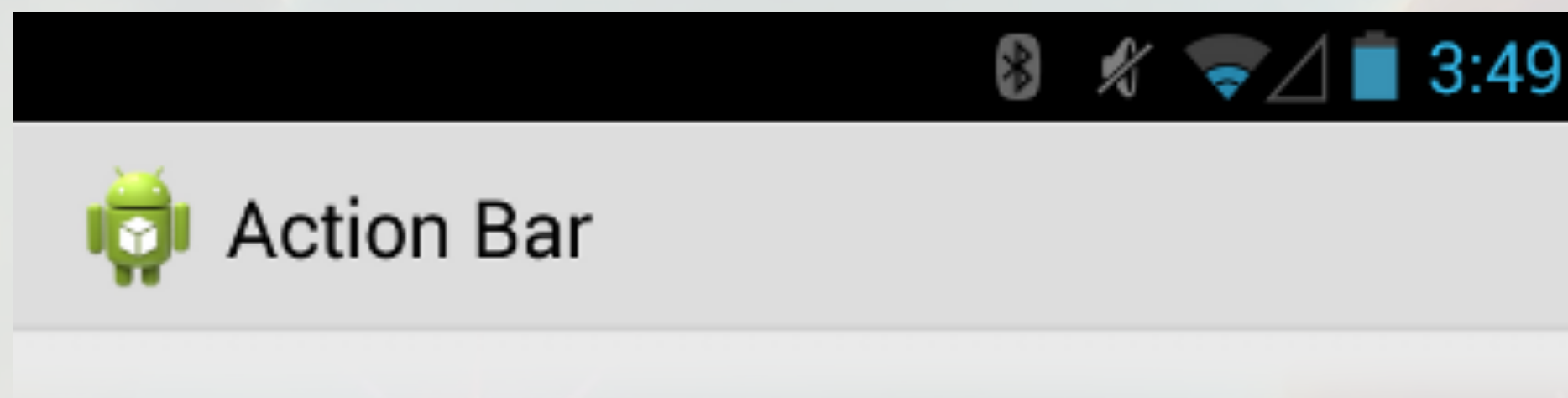
Renault@lrde.epita.fr




Description

Not a "classical" activity

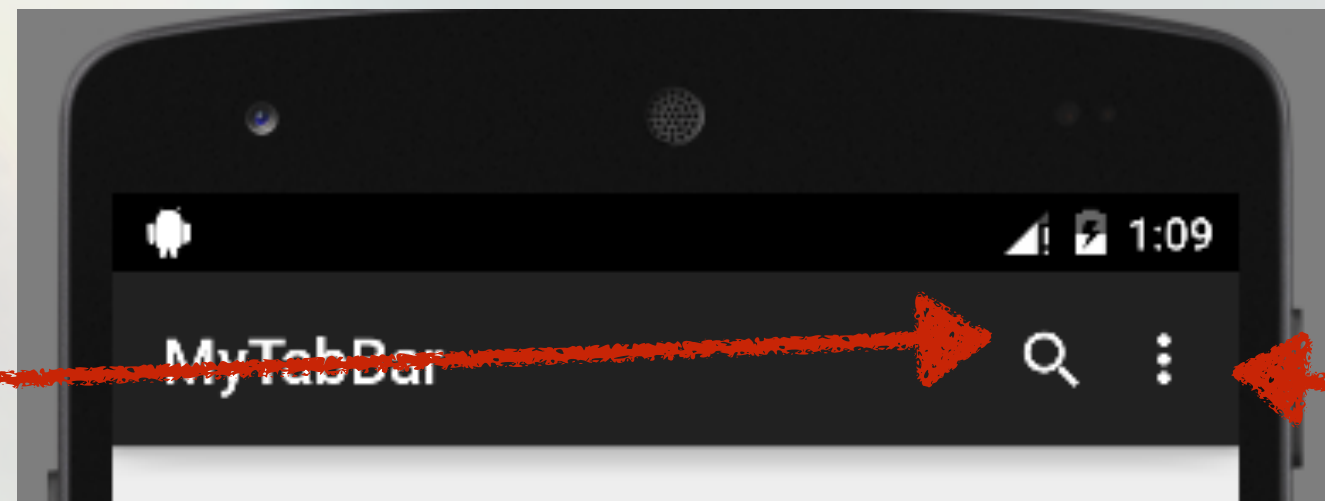
-  No full screen mode
-  Some informations stay visible
 - ▶ **Time**
 - ▶ **Battery**
 - ▶ **Connectivity**
-  Presents contextual informations: title, icon, ...



-  Easily configurable
 - ▶ **Add buttons**
 - ▶ **Add menus**

Adding Buttons

A button



Option Menu

Adding a button is simple

Just modify res/menu/menu_main.xml

```
<item android:id="@+id/action_settings_promote"  
      android:title="@string/action_settings"  
      android:icon="@drawable/abc_ic_search_api_mtrl_alpha"  
      android:orderInCategory="100"  
      app:showAsAction="ifRoom" />
```

- Order them by importance
- Buttons may be displayed only if rooms are available otherwise they are displayed with their names in the Option Menu

Order Buttons



Automatic layout for buttons according to two criteria



android:orderInCategory:

- ▶ **priority between the different elements**
- ▶ **smallest weights are low in the hierarchy**



app:showsAsAction:

- ▶ **main positioning options**
- ▶ **always: the button must appears in the Action Bar**
- ▶ **never: the button will never appears**
- ▶ **ifroom: button may appears**
 - **depends on the other button**
 - **depends on device's screen size**



Guidelines: do not force a button to be in the ActionBar

OptionsMenu



Inflate it when the user wants to inspect it

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action
    // bar if it is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}
```



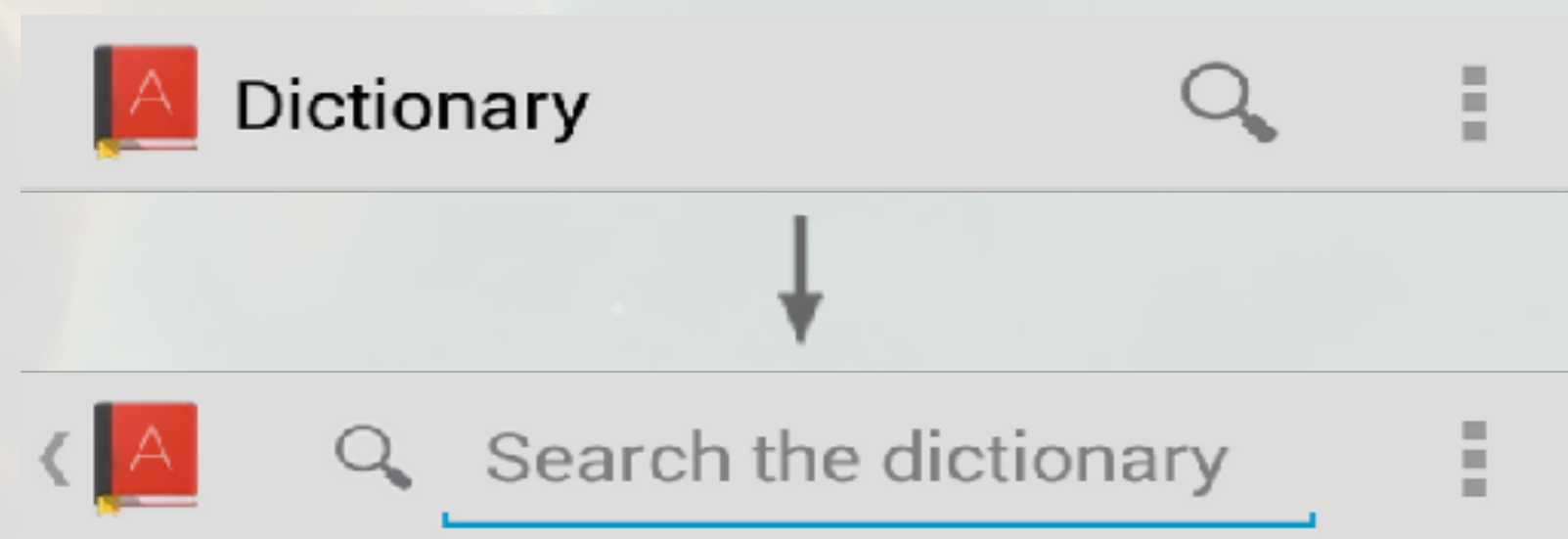
Get clicks on items

```
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here.
    int id = item.getItemId();
    if (id == R.id.action_settings)
        return true;
    return super.onOptionsItemSelected(item);
}
```


Sophisticated Buttons

ActionsViews

-  View is changing when we click the button



-  Use a SearchView

```
<item android:id="@+id/action_search"  
    android:title="@string/action_search"  
    android:icon="@drawable/ic_action_search"  
    app:showAsAction="ifRoom|collapseActionView"  
    app:actionViewClass="android.support.v7.widget.SearchView"  
>
```

Move a button, if necessary, i.e. if a button requires space

How to manipulate an ActionView

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_main, menu);
    SearchView searchView =
        (SearchView)menu.findItem(R.id.action_search).getActionView();
    searchView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            // Click when view is expanded ...
        }
    });
    searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String s) {
            // Submit Text ...
            return false;
        }
        @Override
        public boolean onQueryTextChange(String s) {
            // Detect that the text has changed ..
            return false;
        }
    });
    return true;
}
```


Even more complex buttons

221



Action Providers

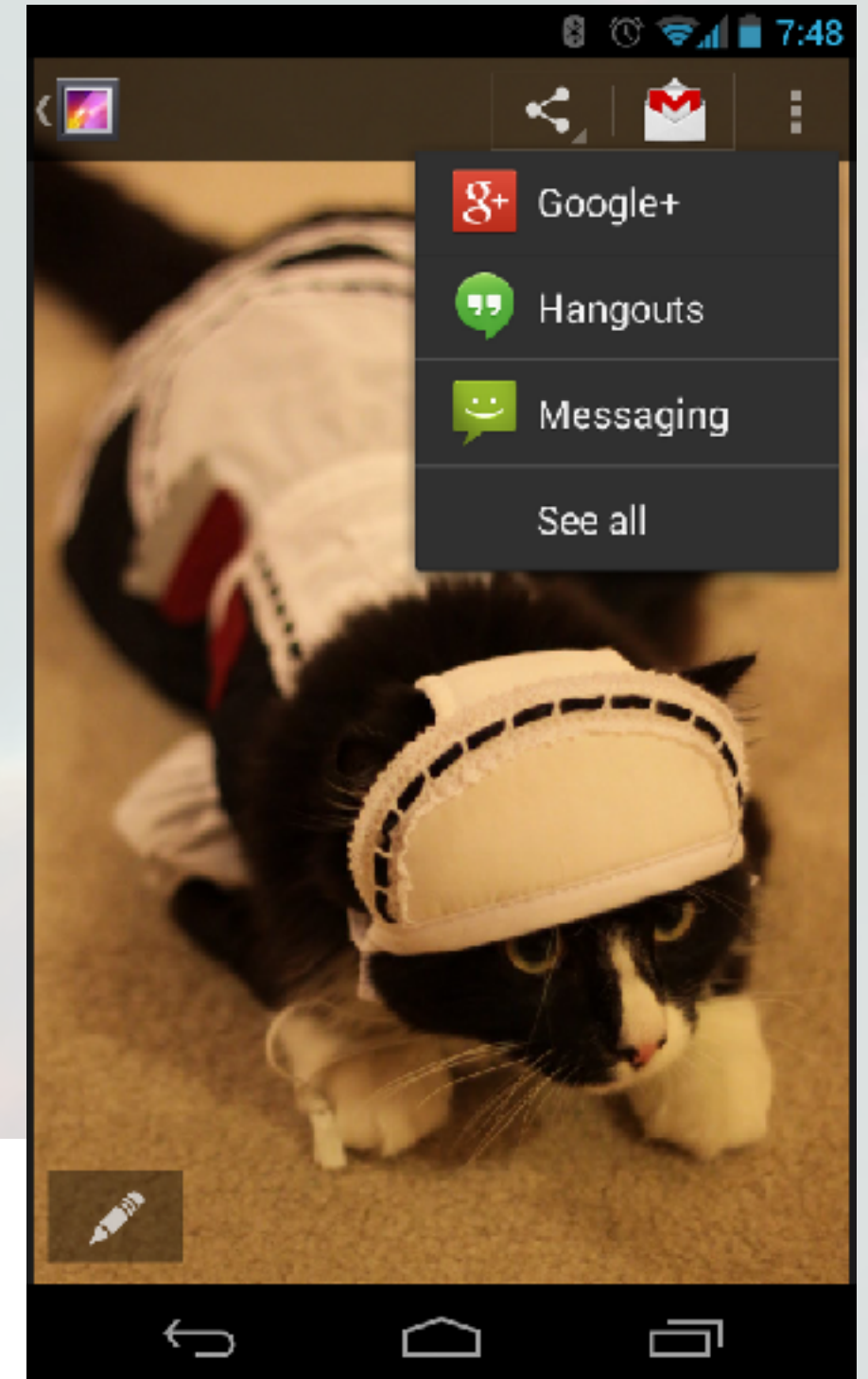
- Close to ActionViews
- Can have their own layout
- Overlap all current actions
- Trigger other Activities



Add the new menu

```
<item
```

```
    android:id="@+id/action_share"  
    android:title="Share"  
    app:showAsAction="ifRoom|collapseActionView"  
    app:actionProviderClass=  
        "android.support.v7.widget.ShareActionProvider" />
```



Manipulate ActionProvider

222



Build the intent you want to launch

```
private Intent getDefaultShareIntent() {
    Intent intent = new Intent(Intent.ACTION_SEND);
    intent.setType("text/plain");
    intent.putExtra(Intent.EXTRA_SUBJECT, "SUBJECT");
    intent.putExtra(Intent.EXTRA_TEXT, "Extra Text");
    return intent;
}
```



Launch it in the onCreateOptionsMenu

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuItem item = menu.findItem(R.id.menu_share);
    provider = (ShareActionProvider)
        MenuItemCompat.getActionProvider(item);
    provider.setShareIntent(getDefaultShareIntent());
    return super.onCreateOptionsMenu(menu);
}
```

Summary

223



ActionBarActivity helps to customize your Application

- Add menus
- Add ContentProvider



Integration of Fragments is possible

- Helps to create dynamic views
- Add scrolling effect or reusable components



Building its own ActionProvider

- Only specify the onCreateActionView method



Before API 21, stack bar action helps to build Bar with tabs



