

Android

III - Advanced interfaces

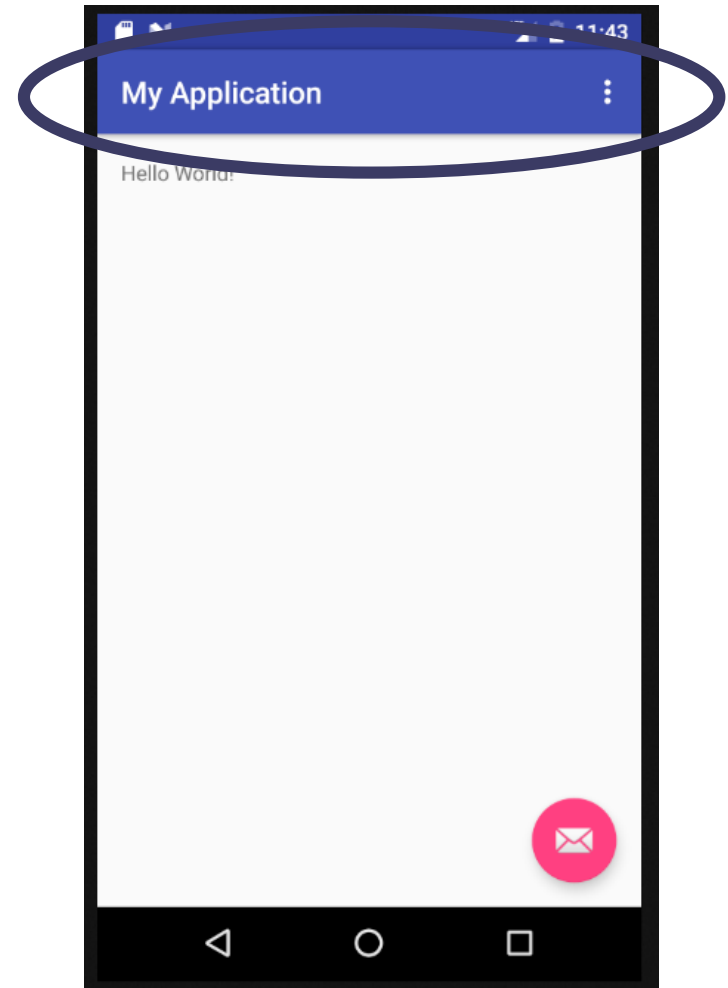


Stefan BORNHOFEN
EISTI



Toolbar

- Indicates application or activity name
- Options menu
- The system can adapt the toolbar appearance to different screen configurations



Basic Activity

Options Menu

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.myMenu, menu);
    return true;
}
```

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.item1: <your code here>; return true;
        case R.id.item2: <your code here>; return true;
        case R.id.item3: <your code here>; return true;
        case R.id.item4: <your code here>; return true;
        default: return super.onOptionsItemSelected(item);
    }
}
```

Options Menu

```
<?xml version="1.0" encoding="utf-8"?>
<menu
  xmlns:android="http://schemas.android.com/apk/res/android">
  <item
    android:id="@+id/first_menu_item"
    android:title="First Menu Item">
    <menu>
      <item
        android:id="@+id/first_submenu_item"
        android:title="First Submenu Item"></item>
      <item
        android:id="@+id/second_submenu_item"
        android:title="Second Submenu Item"></item>
    </menu>
  </item>
  <item
    android:id="@+id/second_menu_item"
    android:title="Second Menu Item"></item>
  <item
    android:id="@+id/third_menu_item"
    android:title="Third Menu Item"></item>
</menu>
```



Submenu

You can prominently present special menu items actions for better navigation

```
<item
  android:id="@+id/item1"
  android:title="Item">
  app:showAsAction="ifRoom"
</item>
```

Toasts and Snackbars

Short messages that automatically time out from the screen.

Toast

- can't perform actions or handle user input
- can't be dismissed by swiping

SnackBar (API Level 23+)

- can perform actions
- can be dismissed by swiping

```
Toast.makeText(getApplicationContext(),  
    "This is a toast.",  
    Toast.LENGTH_SHORT).show();
```

```
SnackBar.make(findViewById(android.R.id.content),  
    "This is a Snackbar",  
    SnackBar.LENGTH_LONG).show();
```

Dialogs

Small windows that prompt the user to make a decision or enter additional information.

```
private void areYouSure() {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setMessage("Are you sure?")
        .setCancelable(false)
        .setPositiveButton("Yes", new
            DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int id)
                { // do something
                } })
        .setNegativeButton("No", new
            DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int id)
                { dialog.cancel(); } });
    AlertDialog alert = builder.create();
    alert.show();
}
```

Button click: Runtime creation

```
Button b = (Button)findViewById(R.id.mybutton);
b.setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        // do something
    }
});
```

Long click

```
b = (Button)findViewById(R.id.mybutton);
b.setOnLongClickListener(new OnLongClickListener() {
    public boolean onLongClick(View v) {
        // do something
        return true;
    }
});
```



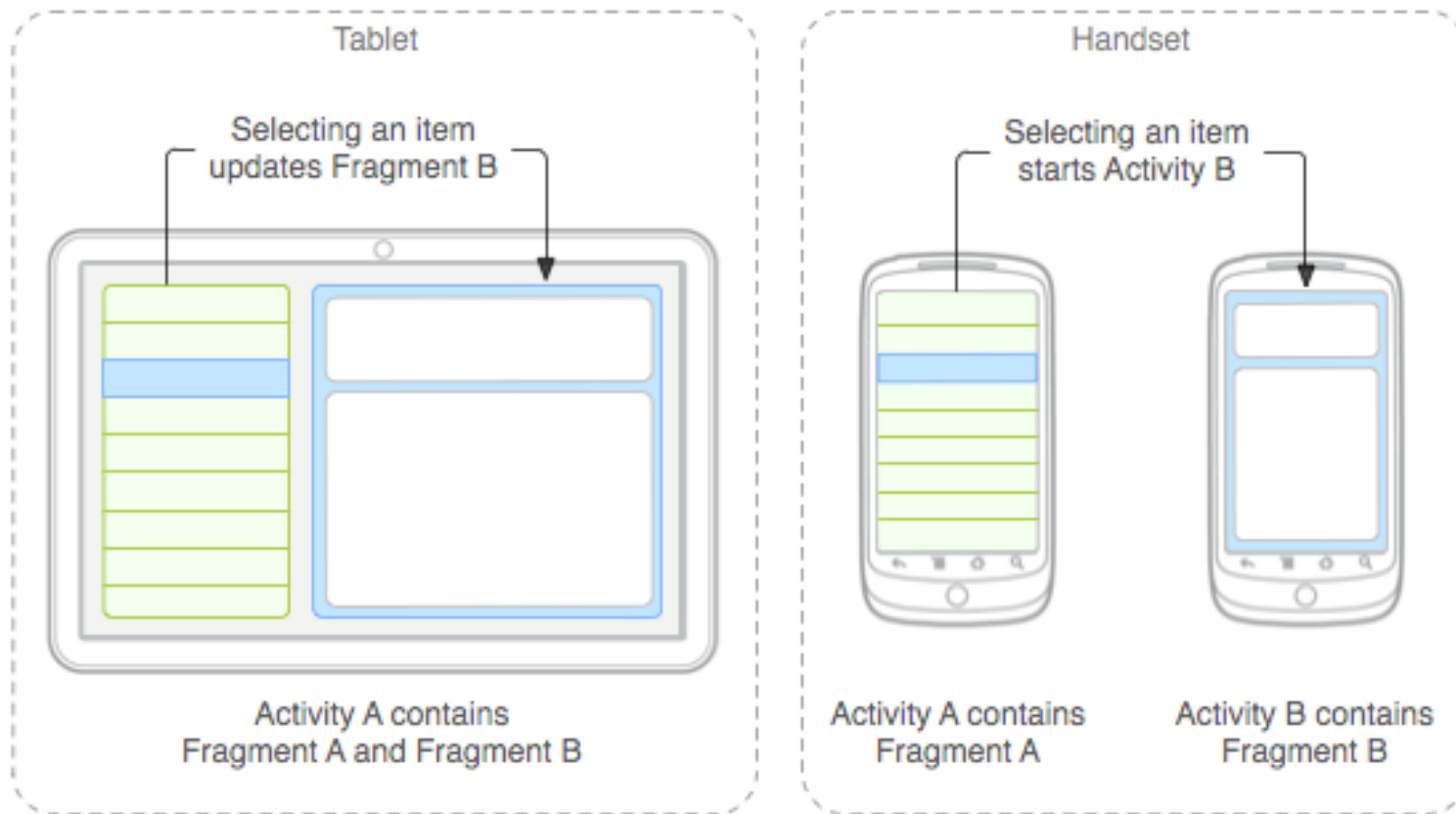
The click has been consumed,
don't handle onClick any more

Fragments

- A behavior or a portion of user interface in an Activity.
- You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities (like a “sub-activity”).
- You can dynamically add or remove fragments during runtime
- The fragment's lifecycle is directly affected by the host activity's lifecycle.

Fragments

Fragments in different layouts



Fragment Lifecycle

- Usually, you should implement at least the following lifecycle methods:

onCreate()

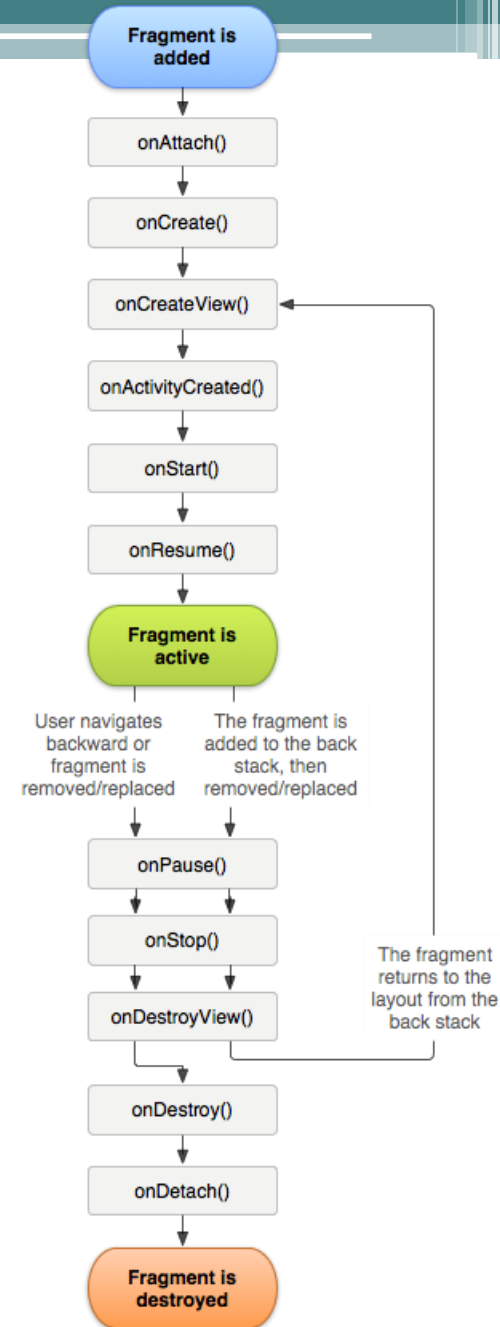
- The system calls this when creating the fragment. You should initialize essential components of the fragment.

onCreateView()

- The system calls this when it's time for the fragment to draw its user interface for the first time. To draw a UI for your fragment, you must return a View from this method that is the root of your fragment's layout. (null if the fragment does not provide a UI.)

onPause()

- The system calls this method as the first indication that the user is leaving the fragment. This is usually where you should commit any persistent changes.



Minimal Fragment Code

```
public class MyFragment extends Fragment {
    @Override
    public View onCreateView(
        LayoutInflater inflater,
        ViewGroup container,
        Bundle savedInstanceState)
    {
        // Inflate the layout for this fragment
        return inflater.inflate(
            R.layout.myfragment,
            container, false);
    }
}
```

Add Fragments to an Activity

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:orientation="horizontal"
  android:layout_width="match_parent"
  android:layout_height="match_parent">
  <fragment android:name="fr.eisti.android.MyFragment"
    android:id="@+id/fragment1"
    android:layout_weight="1"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    tools:layout="@layout/fragment1" />
  <fragment android:name="fr.eisti.android.OtherFragment"
    android:id="@+id/fragment2"
    android:layout_weight="2"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    tools:layout="@layout/fragment2" />
</LinearLayout>
```

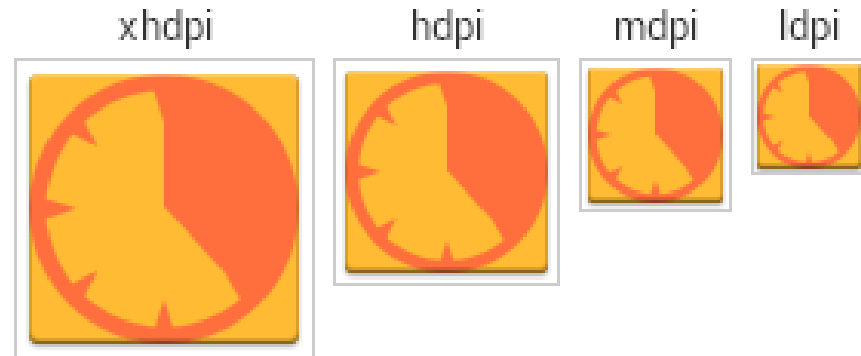
Alternative Resources

- `res/<resources_name>-<qualifier>`
- You can append more than one `<qualifier>`, separate each one with a dash
- The qualifiers must be in the right order

Examples:

Configuration	Qualifier
Language and region	en, fr, ja
Screen size	small, normal, large, xlarge
Screen pixel density	ldpi, mdpi, hdpi, xhdpi nodpi, tvdpi
Screen orientation	port, land

Icon Generator



Android Asset Studio

<https://romannurik.github.io/AndroidAssetStudio>

Save your icons in the respective res/mipmap directories.

Application: Internationalization

res/values/strings.xml

- Default text for all strings

res/values-fr/strings.xml

- French text for all strings

res/values-ja/strings.xml

- Japanese text for all strings

res/values-ja-land/strings.xml

- Japanese text for all strings when the screen is landscaped

Exercise

Write the MathInspector (2p).

- Fragments: portrait/landscape layout
- LongClick
- Toast/Snackbar messages
- Restore red number after restart

Add the following features (1p).

- fancy design
- toolbar: About
- personalized launcher icon
- second language

