

Android Developer Fundamentals V2

Delightful User Experience

Lesson 5



5.3 Adaptive layouts and resources



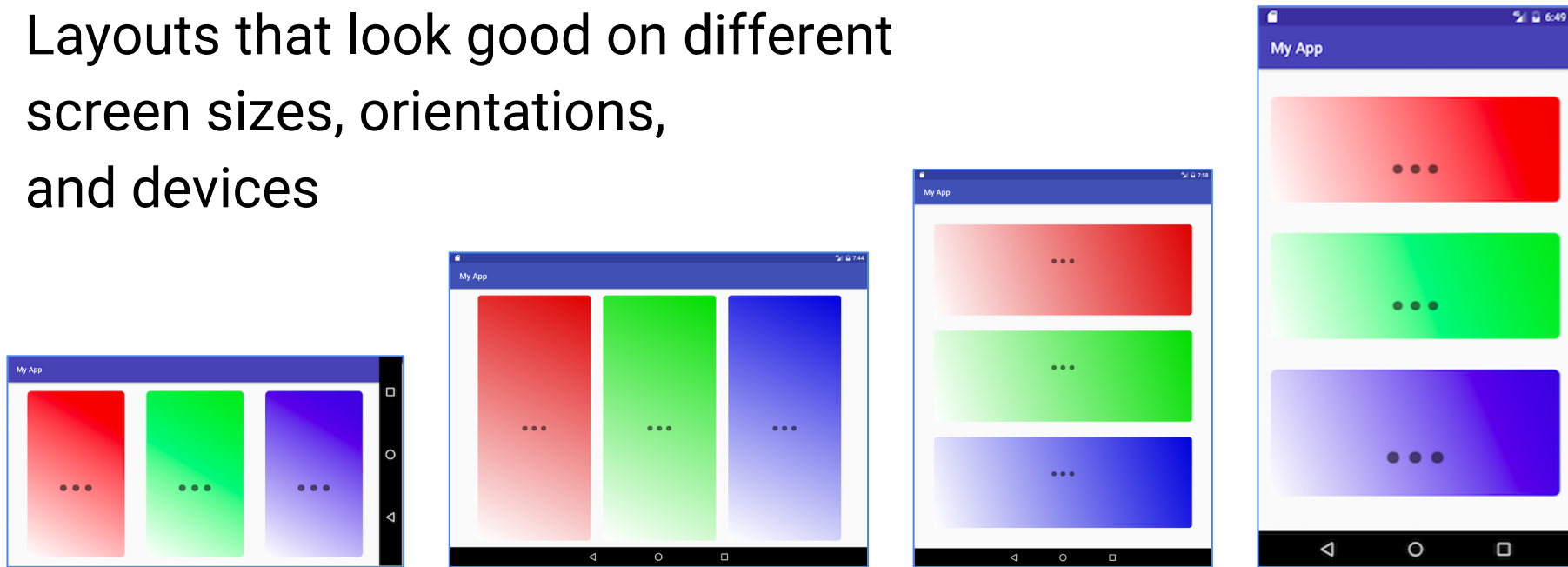
Contents

- Adaptive layouts and resources
- Alternative resources
- Default resources

Adaptive layouts and resources

What are adaptive layouts?

Layouts that look good on different screen sizes, orientations, and devices



Adaptive layout

- Layout adapts to configuration
 - Screen size
 - Device orientation
 - Locale
 - Version of Android installed
- Provides alternative resources
 - Localized strings
- Uses flexible layouts
 - GridLayout

Resource folders of a small app

```
MyProject/  
  src/  
  res/  
    drawable/  
      graphic.png  
    layout/  
      activity_main.xml  
      list_iteminfo.xml  
    mipmap/  
      ic_launcher_icon.png  
    values/  
      strings.xml
```

Put resources in your
project's res folder

Common resource directories

- drawable/, layout/, menu/
- values/—XML files of simple values, such as string or color
- xml/—arbitrary XML files
- raw/—arbitrary files in their raw form
- mipmap/—drawables for different launcher icon densities
- [Complete list](#)



Alternative resources

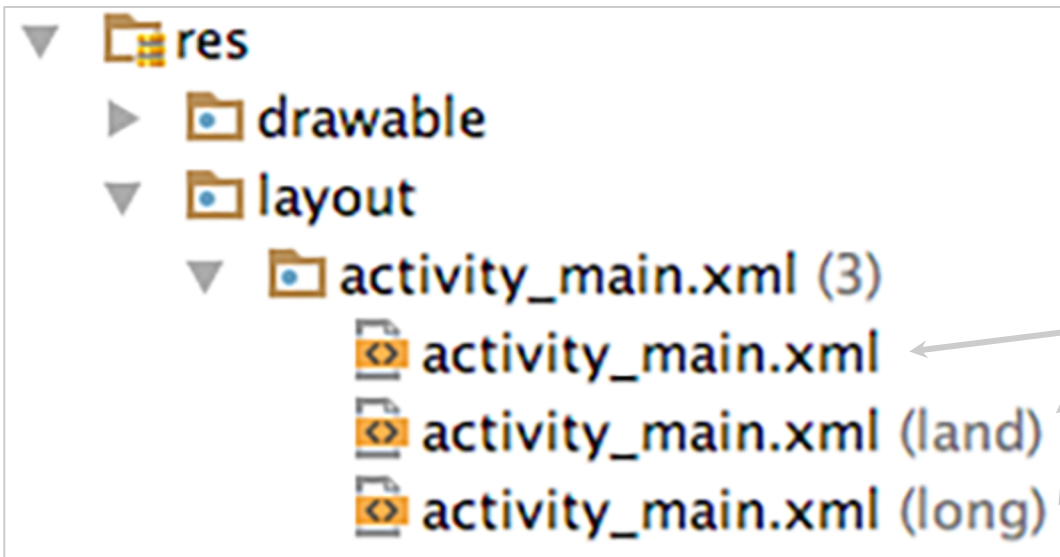
What are alternative resources?

Different device configurations may require different resources

- Localized strings
- Image resolutions
- Layout dimensions

Android loads appropriate resources automatically

Create alternative resource folders



Use alternative folders for resources for different device configurations

Names for alternative resource folders

Resource folder names have the format
resources name-config qualifier

drawable-hdpi	drawables for high-density displays
layout-land	layout for landscape orientation
layout-v7	layout for version of platform
values-fr	all values files for French locale

[List of directories and qualifiers](#) and usage detail



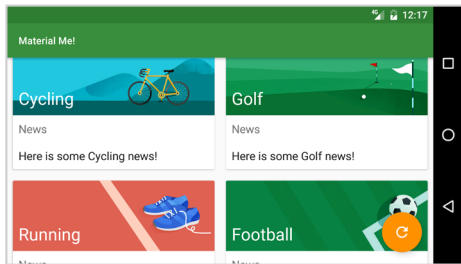
Screen Orientation

- Use `res/layout` and provide alternatives for landscape where necessary
 - `res/layout-port` for portrait-specific layouts
 - `res/layout-land` for landscape specific layouts
- Avoid hard-coded dimensions to reduce need for specialized layouts

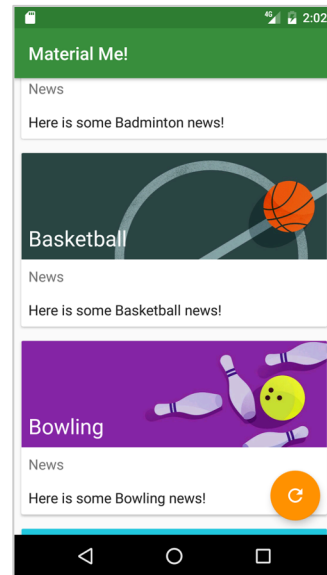
Simple adaptive layout

GridLayout

- In values/integer.xml:
`<integer name="grid_column_count">1</integer>`
- In values/integer.xml-land:
`<integer name="grid_column_count">2</integer>`



Landscape



Portrait

Smallest width

- Smallest-width (sw) in folder name specifies minimum device width
 - res/values-sw n dp, where n is the smallest width
 - Example: res/values-sw**600**dp/dimens.xml
 - Does not change with orientation
- Android uses resource closest to (without exceeding) the device's smallest width



Platform Version

- API level supported by device
 - `res/drawables-v14`
contains drawables for devices that support API level 14 and above
- Some resources are only available for newer versions
 - WebP image format requires API level 14 (Android 4.0)
- [Android API level](#)



Localization

- Provide strings (and other resources) for specific locales
 - `res/values-es/strings.xml`
- Increases potential audience for your app
- Locale is based on device's settings
- [Localization](#)



Default resources

Default Resources

- Always provide default resources
 - directory name without a qualifier
 - res/layout, res/values, res/drawables....
- Android falls back on default resources when no specific resources match configuration
- [Localizing with Resources](#)



Learn more

- [Supporting Multiple Screens](#)
- [Providing Resource](#)
- [Providing Resources Guide](#)
- [Resources Overview](#)
- [Localization Guide](#)

What's Next?

- Concept Chapter: [5.3 Resources for adaptive layouts](#)
- Practical: [5.3 Adaptive layouts](#)

END