CSE 340

Winter Quarter 2020

1/17/2020

Professor: Dr. Jennifer Mankoff, Lauren Bricker

Note-taker email: jial8@uw.edu

Week 2 Lecture 3

Introduction to Layout
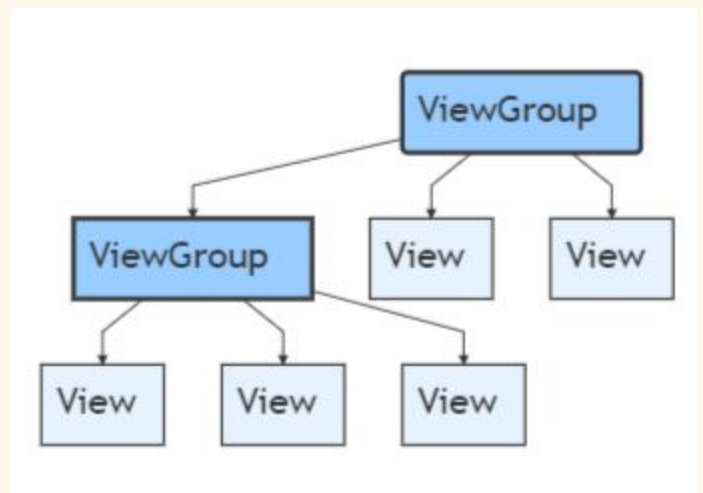
Notes

Badly Designed Website Example

"Responsive" page: page that scales to the user's screen. Static is the opposite. The website has very small text, clashing colors, and strange grouping of interactors.

Why do we need layout?

- Be able to center something
- Lock it to an edge
- Have a reactive design layout (be able to shrink and expand without looking weird)

Layout 1 - Worksheet: Distributed during class

- **Views**: Base class for all UI elements, interactors
- **ViewGroups:** groups >=1 views, can define specific layout properties
- **Vertical Container Components** (in the example)
    - **Linear** **layout** (every component on the same line in the tree). Always places its kids, in order in that direction
- **Spacers**
    - Works well with layout containers
    - **Struts:** fixed spaces, hold things fixed distance apart

○ **Stretchy** **Space (springs):** push things apart as much as possible

What is XML? How to use Android GUI editor?

- Extensible Markup Language
- Is a superset of HTML
  ○ Tag names surrounded by brackets like "< …. >"
  ○ Generally in pairs
  ○ Can have attributes
- Use to describe widgets on the screen
- Android has a GUI layout editor
  ○ Can change the background color, size, location…etc of different Views
  ○ Many different attributes!
  ○ **strings.xml**
    ■ What if I want to change a lot of text to another language, or just something else?
    ■ TextViews' attributes set to: @string/app_name
    ■ Change the text in strings.xml
  ○ Like the above example, Views can grab content from a resource file
- How to create two buttons next to each other?
  ○ Create a new **LinearLayout** (horizontal)
  ○ Make sure layout_width looks okay
- **Each thing on the screen has an ID, a meaning! Please name your buttons reasonably (not button1, button2…)**
  ○ Good names: nextButton, prevButton
  ○ Access a component programmatically by using [Component].getId() and setID()
    ■ Use this → implement tab navbar
- In code, you can set variables that we can access later
  ○ Ex: R.(file name).(value name)
- **PARAMS**: a class content, specifies layout parameters

Layout Containers

Automates layout tasks, make it simpler to ensure reliable results.

- **Constraints:** very general, can operate on multiple axes, has attachment options