AUTO LAYOUT

PART I

Zeke Abuhoff

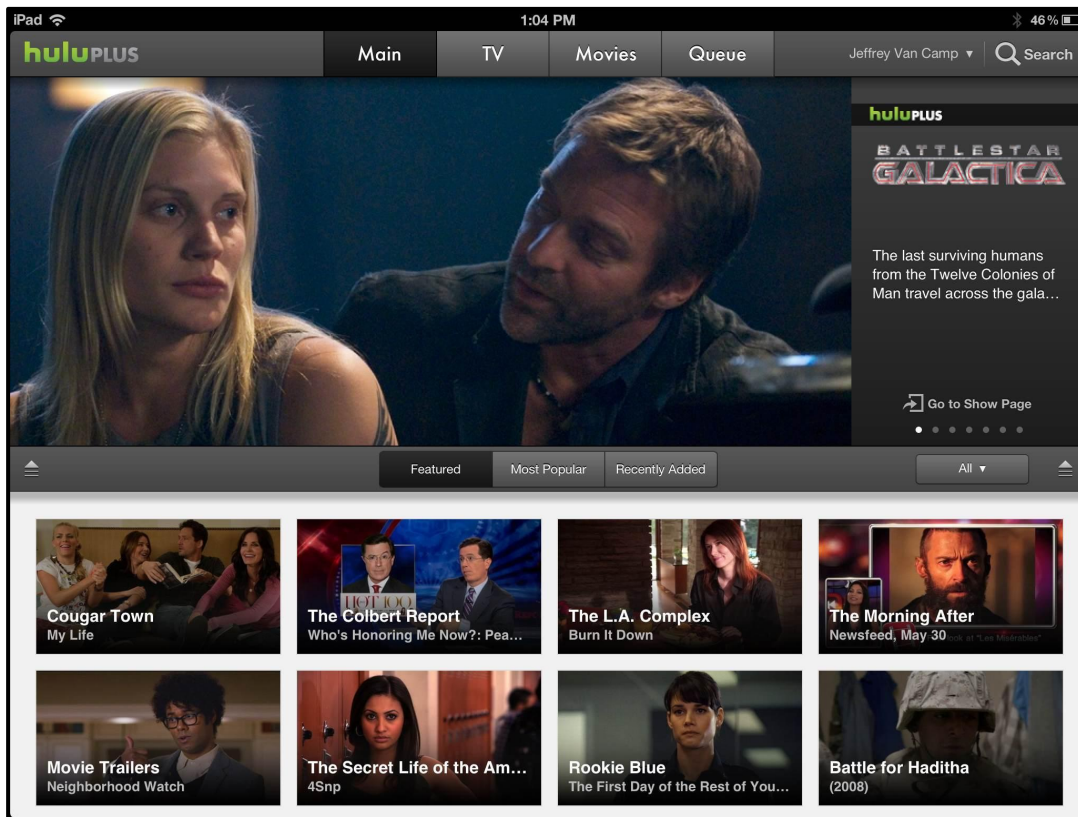*Lead iOS Instructor, General Assembly*

LEARNING OBJECTIVES

+ Identify situations in which Auto Layout is needed

+ Create Auto Layout constraints

+ Resolve Auto Layout conflicts

AUTO LAYOUT

ADAPTING UI

Exercise:

1) Adapt the UI seen in this screenshot to a portrait orientation.

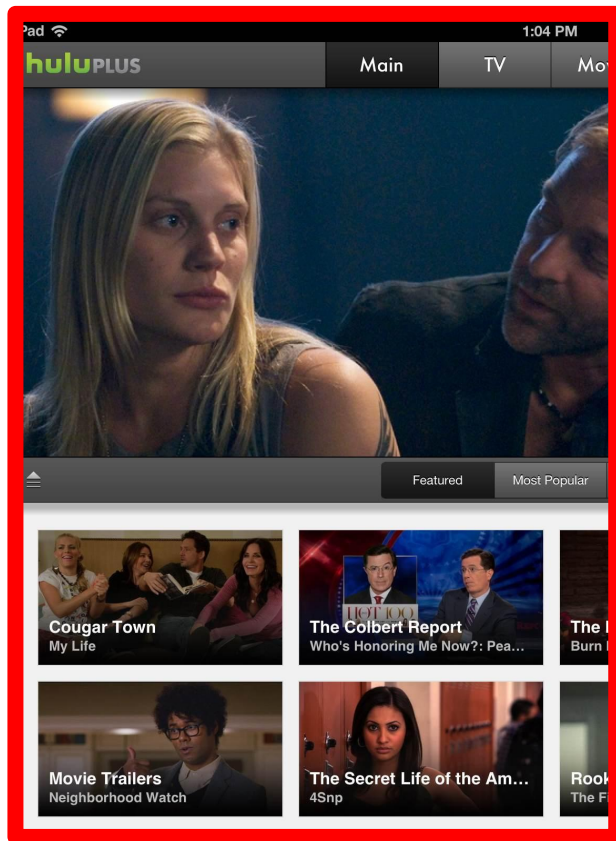2) Adapt both orientations to an iPhone screen (smaller, narrower).

AUTO LAYOUT

ADAPTING UI

Here's an example of a very simple adaptation from landscape orientation to portrait orientation.

As you can see, buttons, labels and other important information is lost from the right side of the screen.
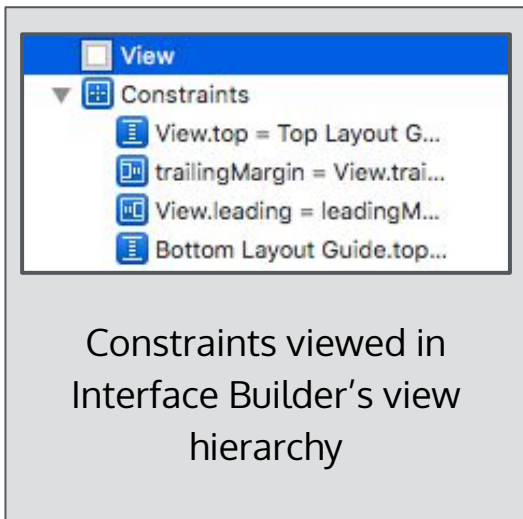

Without specific, carefully chosen rules for HOW to adapt a UI to a new context, things quickly go awry.

We need a powerful, intuitive framework to help us establish the right rules.
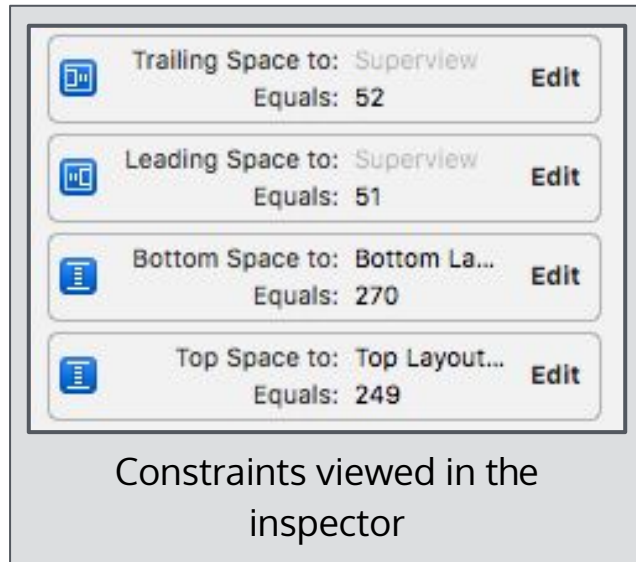
AUTO LAYOUT CONSTRAINTS

Auto Layout is the iOS system responsible for resizing and rearranging UI elements according to changes in orientation and device.

Constraints are the rules that Auto Layout follows in order to do its job correctly.

Constraints viewed in Interface Builder's view hierarchy

Constraints viewed in the inspector

AUTO LAYOUT CONSTRAINTS

Each constraint describes a relationship between two items.

Its 'constant' and 'multiplier' properties give scale to that relationship.

Constraints can be attached to outlets or given reuse identifiers.

First Item	Superview.Trailing Margin
Relation	Equal
Second Item	View.Trailing
Constant	52
Priority	1000
Multiplier	1
Identifier	Identifier

AUTO LAYOUT CONSTRAINTS

Auto Layout constraints can be created by either ctrl+click+dragging from one view to another

or

using the Auto Layout buttons in lower right of Interface Builder.

You can add whatever constraints you want, as long as they convey consistent logic to Auto Layout.

Auto Layout will give you warnings and errors if your constraints conflict.

AUTO LAYOUT CONSTRAINTS

Practice

1) Create a new view controller and place two views in it.
2) The first view should be orange. Give it constraints that make it 20 pts away from its container on every side.
3) The second view should be green. Give it constraints that make it 40 pts away from its container on every side.
4) Run your app in an iPhone simulator and rotate it. Then run it in an iPad simulator and rotate it.

AUTO LAYOUT CONSTRAINTS

Practice

1) Copy the view controller from the previous practice set three times.
2) In each copy, recreate the constraints suggested by the diagrams below.