

CS 5150 Software Engineering

10. User Interfaces for Web Sites and Mobile Devices

William Y. Arms

Systems and Networks

computer model	view
	controller
	model
	computer systems and networks

Computer Systems and Networks: Device-Aware Interfaces

Interfaces must take into account physical constraints of computers and networks:

- How does a desk-top computer differ from a laptop?
- What is special about a smartphone?
- How do you make use of a touch-sensitive screen?
- What works well with a digital camera?

Constraints that the interface must allow for:

- => **performance** of device (e.g., fast or slow graphics)
- => limited **form factor** (e.g., small display, no keyboard)
- => **connectivity** (e.g., intermittent)

Networks

Operations that transfer data over the network have unpredictable response times and are subject to delay.

- Large data transfers should run asynchronously in a separate thread.
- Provide visual feedback to indicate that the operation is in progress.



- Provide a way for users to cancel long running data transfers.

Smartphones and Tablets: Screen Size

Smartphones and tablets have small screens. Every small area is important.

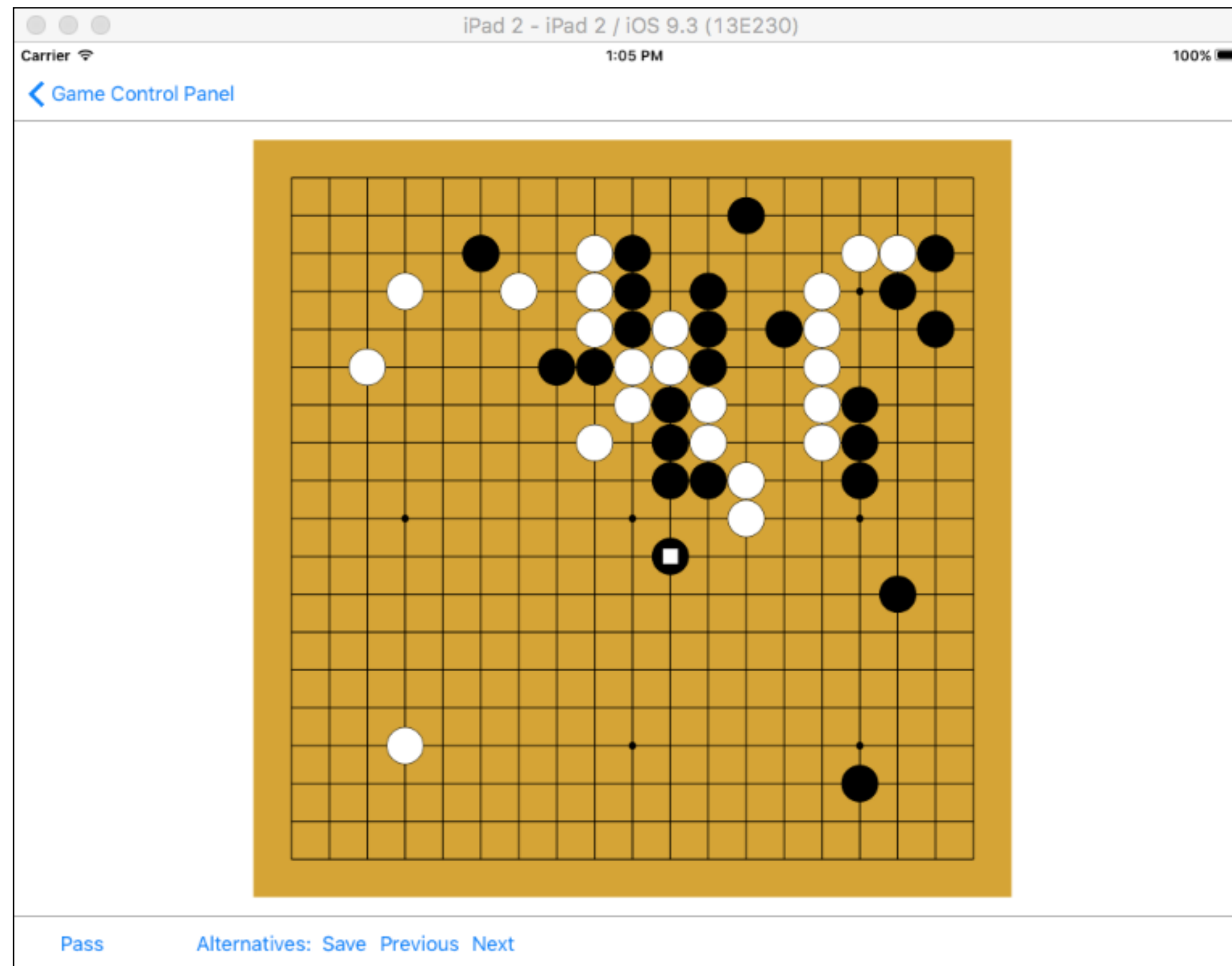
- There is big difference in screen size between a small smartphone and a large tablet.
- There may be different layouts or even different versions for different screens.
- Apps need to have different layouts for portrait and landscape.

Apps need to be tested with a full range of screen sizes and orientations.

Some apps may be optimized for a large screen.

Use a simulator to see how your app looks on various devices.

Screen Size: Simulators for Mobile Development



Development environments for mobile apps provide simulators for a variety of mobile devices.

This example is the simulator for Apple's Xcode showing how an app would appear on an iPad.

This app has too much detail to be effective on a small smartphone.

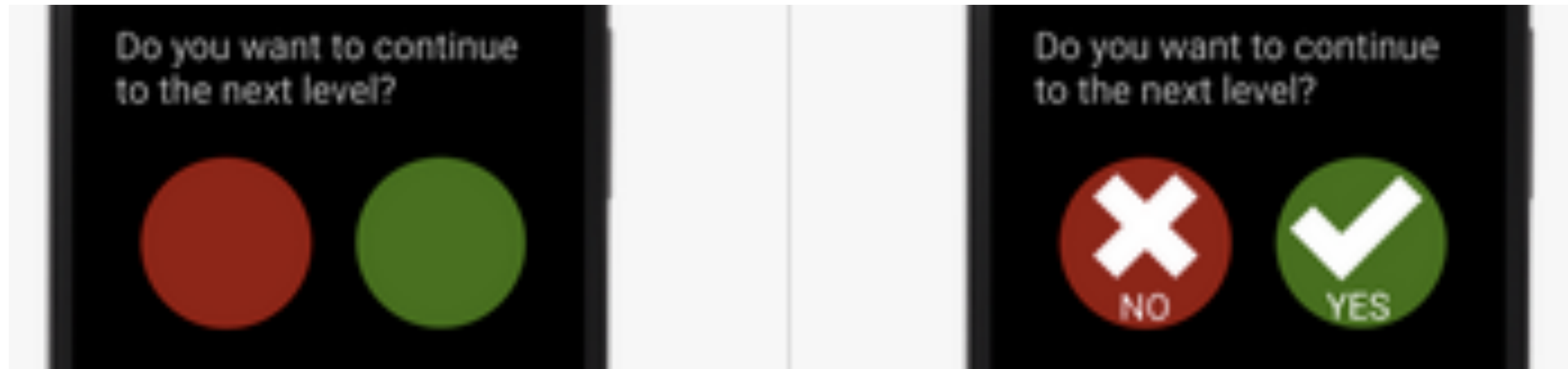
Smartphones: Accessibility

**Smartphones have small displays and small virtual keyboards.
Some apps rely on speech or other sound signals.**

People with poor eyesight, color blindness, hearing loss, or clumsy fingers may have difficulty using your applications.

Android and iOS provide numerous accessibility features and provide online advice about how to build accessible apps.

For your user testing try to find people who do not have perfect eyesight, hearing, etc. Have testers of various ages. Older people are often less able to use touch sensitive screens.



Screen Size: Responsive Web Design

Responsive web design adjusts how web pages are viewed based on the size of the screen, or other characteristics of the device or browser used to view the page.

- Media queries in CSS allow the page to use different CSS style rules based on characteristics of the device the site is being displayed on, e.g., the width of the browser window.

For example. You develop a web site using a laptop computer. How will it look on a smartphone?

Responsive Web Design

CS 5150

**CS 5150: Software Engineering
Spring 2018**

Course Description

This course is an introduction to the practical problems of specifying, designing, building, testing, and delivering reliable software systems. Special topics include professionalism, project management, and the legal framework for software development. As a central part of the course, student teams carry out projects for real clients. Each project includes all aspects of software development from a feasibility study to final delivery.

Offered:	Spring semester
Prerequisites:	Computer Science 2110 or equivalent

Browsers such as Chrome, Firefox, and Safari have options that allow you to see how your site would appear on other devices.

This Chrome example shows the development version of the CS 5150 web site as it would appear on an older Android device.

Responsive Web Design

The ideal:

A single web site adapts to any device by using a mix of flexible grids and layouts, and careful use of CSS media queries.

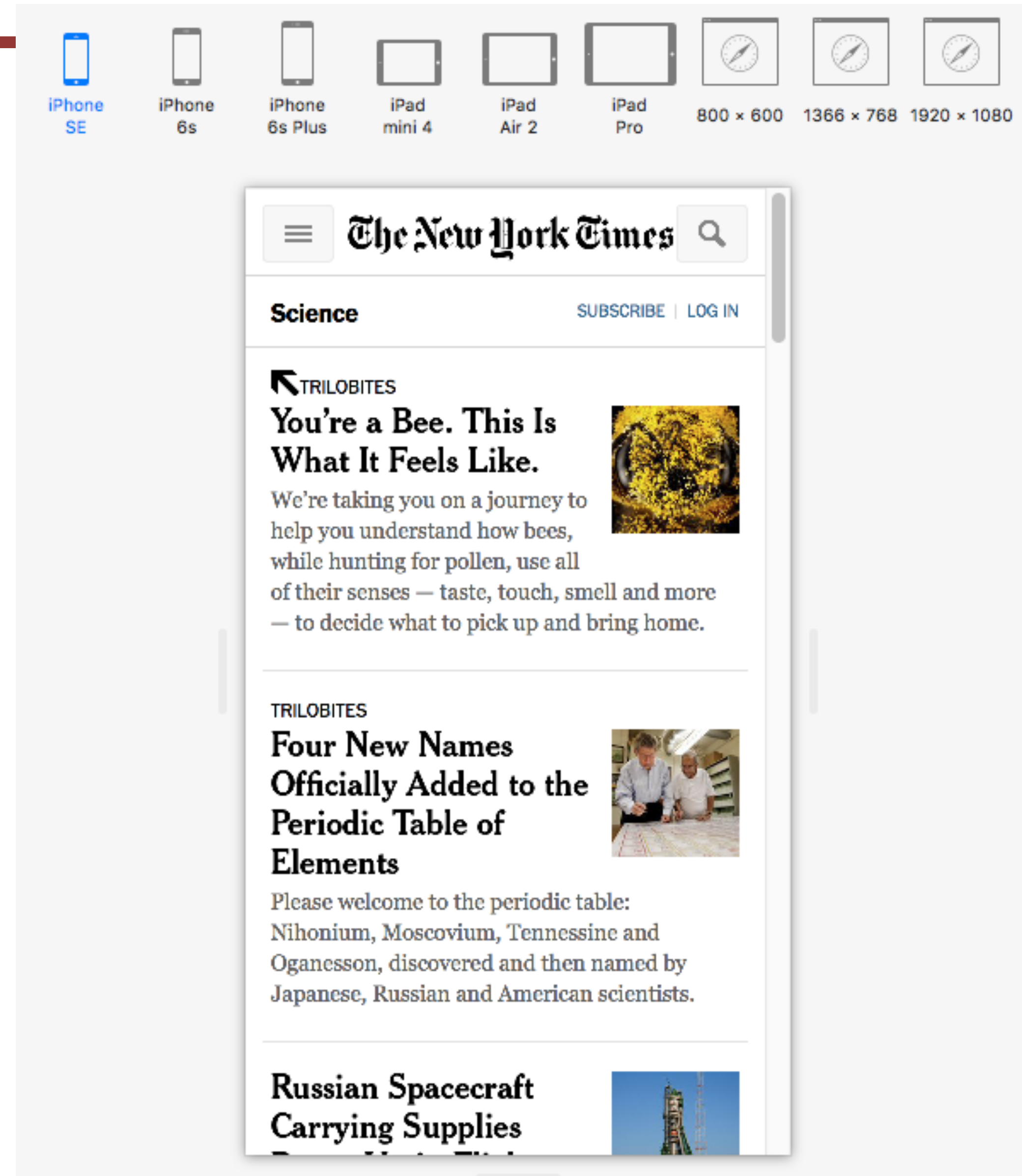
In practice:

Mobile devices, such as smartphones, are so different from regular computers that it is extremely difficult to create a web site that works well on all devices.

For example:

- Less content can be displayed on a small screen than on a large screen.
- Smartphones are usually used in portrait; computer screens are landscape.
- A touch screen has different characteristics from a mouse, e.g., the user cannot hover over a menu item.
- A virtual keyboard needs screen space.

Responsive Web Design



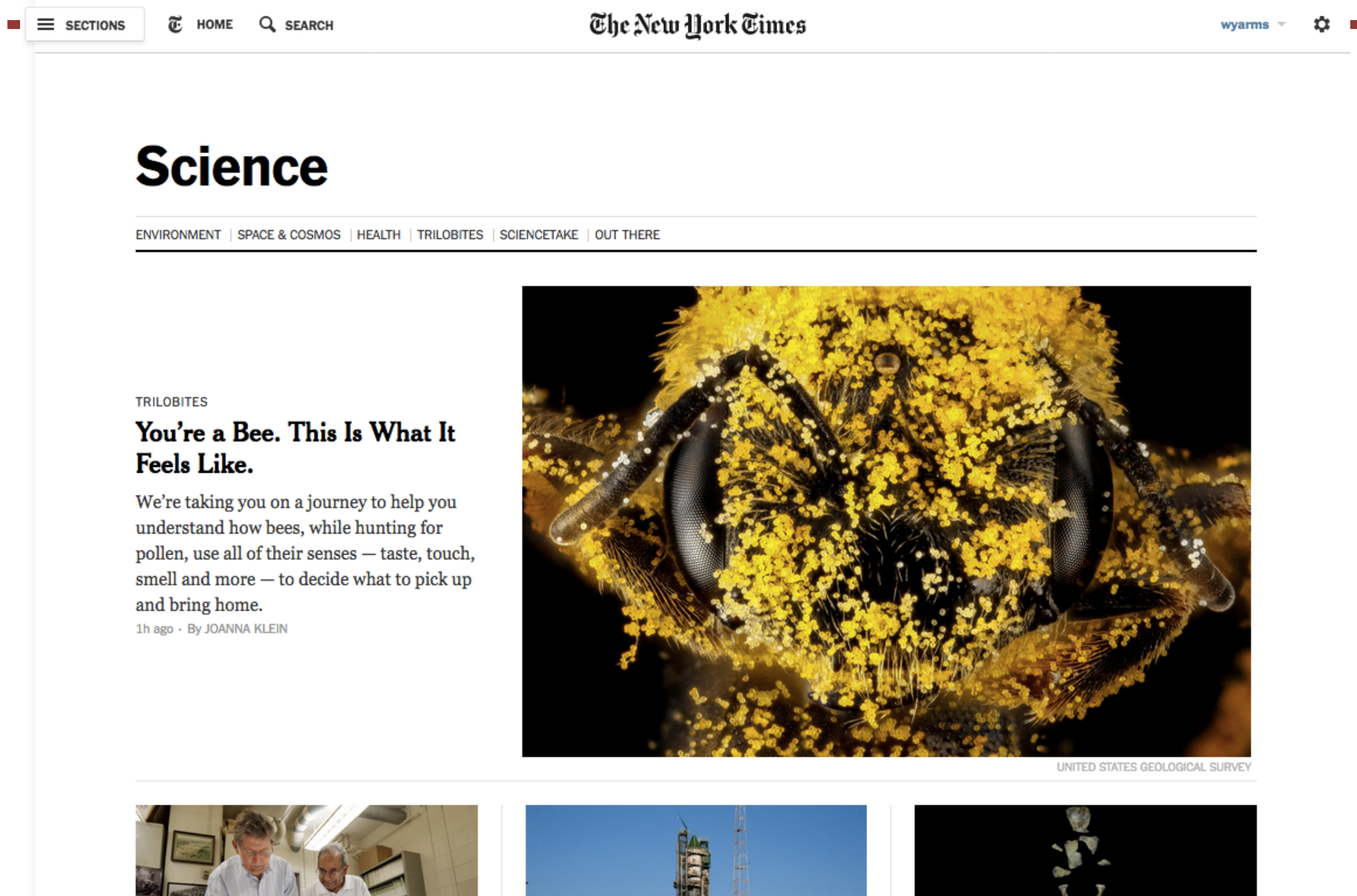
Major web sites have several versions of the site for different devices.

They identify the user's device and automatically provide the version designed for that type of device.

This Safari example shows the New York Times mobile version of the web site on a small iPhone.

The next slide, from Firefox, shows the same page on a laptop computer.

Responsive Web Design



User Interface Design: Navigation

Getting the navigation right is central to the user experience.

Applications consist of one or more pages (web) or screens (mobile apps).

- Web: loading a new page uses the network and is therefore slow
- Smartphone or tablet: loading a new screen is usually instantaneous

The first step in user interface design is to choose what pages or screens to use.

- What functionality is provided on each screen?
- How are the screens organized?
- How does the user know what screens are available?
- How does the user move between screens?

Navigation: Organization of Pages and Screens

The organization of the pages must match the user's mental model.
Keep it simple.

Organization of pages in a web application

The basic building block of the web is a hyperlink. This allows any page to link to any other page. This is flexible, but can lead to confusing applications.

Many web sites use a hierarchical tree structure.

When a user leaves a page, the state is lost unless explicitly saved.

Organization of screens in a mobile app

Both Android and iOS encourage the use of a stack based architecture.

When a user leave a screen, the state is pushed onto a stack and is available when the screen is next used.

Indicate to the user what pages are available and how to reach them.

Navigation: User Testing

Before building an application, test the navigation.

- Create simple mock-ups, e.g., rough sketches.
- Create a large number of simple scenarios. They should show all the main paths through the application, including user mistakes, system problems, etc.
- Step through the scenarios with the client and potential users.

Later in the development process, these scenarios can be used for user interface design, program design, and all forms of testing.

It is much easier to make changes in the basic navigation before beginning the detailed user interface design and implementation.

Mobile Apps: Development Environments

App development is complex. Most apps are built using development environments such as Android Studio and Apple's Xcode.

There are good tutorials and good documentation online. I strongly recommend that you work through a tutorial before attempting to build a real app.

If you are new to Android or iOS development allow plenty of learning time.

For your CS 5150 project try to have several team members with previous experience of the development environment that you are using.

Mobile Apps: Development Environments

These environments are moderately flexible for user interface design. They support a variety of ways to navigate, user interface objects, etc. But they have limitations (e.g., iOS does not have a radio button).

To build a good user experience, understand and adopt the style of the development environment.

- Users will be familiar with the style of interface from other apps.
- The interface objects are well designed with good technical implementation.
- The objects are intended to be used on a variety of devices.
- When new versions of the operating systems are released, the designs will need few modifications.

Mobile Apps: the Design Challenge

How do you design a user interface with no instructions, no user manual, no training?

Look:

Characteristics of the appearance that convey information.

Feel:

Interaction techniques that are intuitive and provide satisfactory experience.

Metaphors and mental models:

Mental models and metaphors. But there may not be an intuitive model.

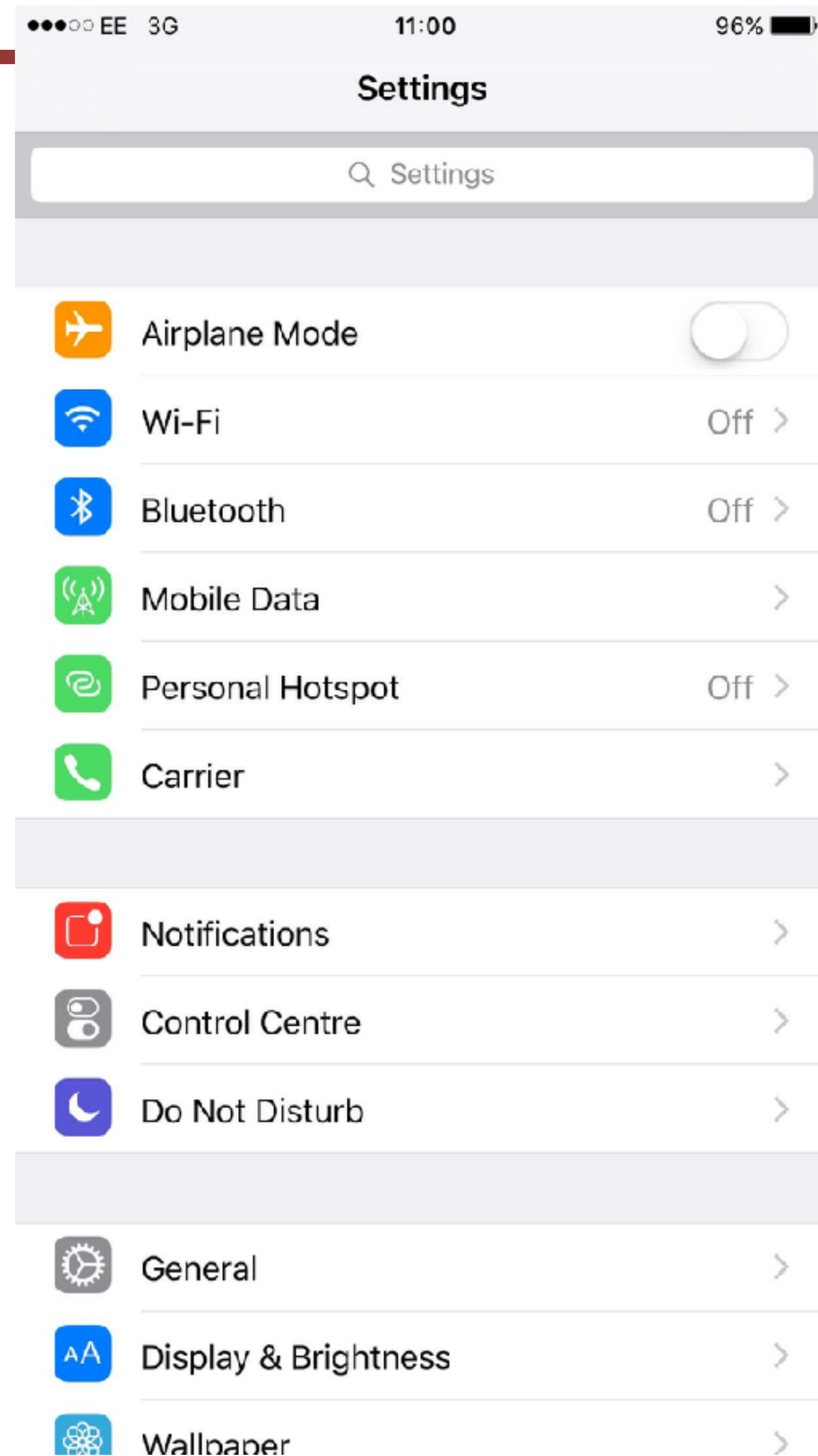
Navigation rules:

How to move among data, functions, and activities in a large space.

Conventions:

Familiar aspects that do not need extra training – good for users, good for designers, e.g., scroll bars, buttons, gestures, help systems, sliders.

Mobile Apps: Development Environments



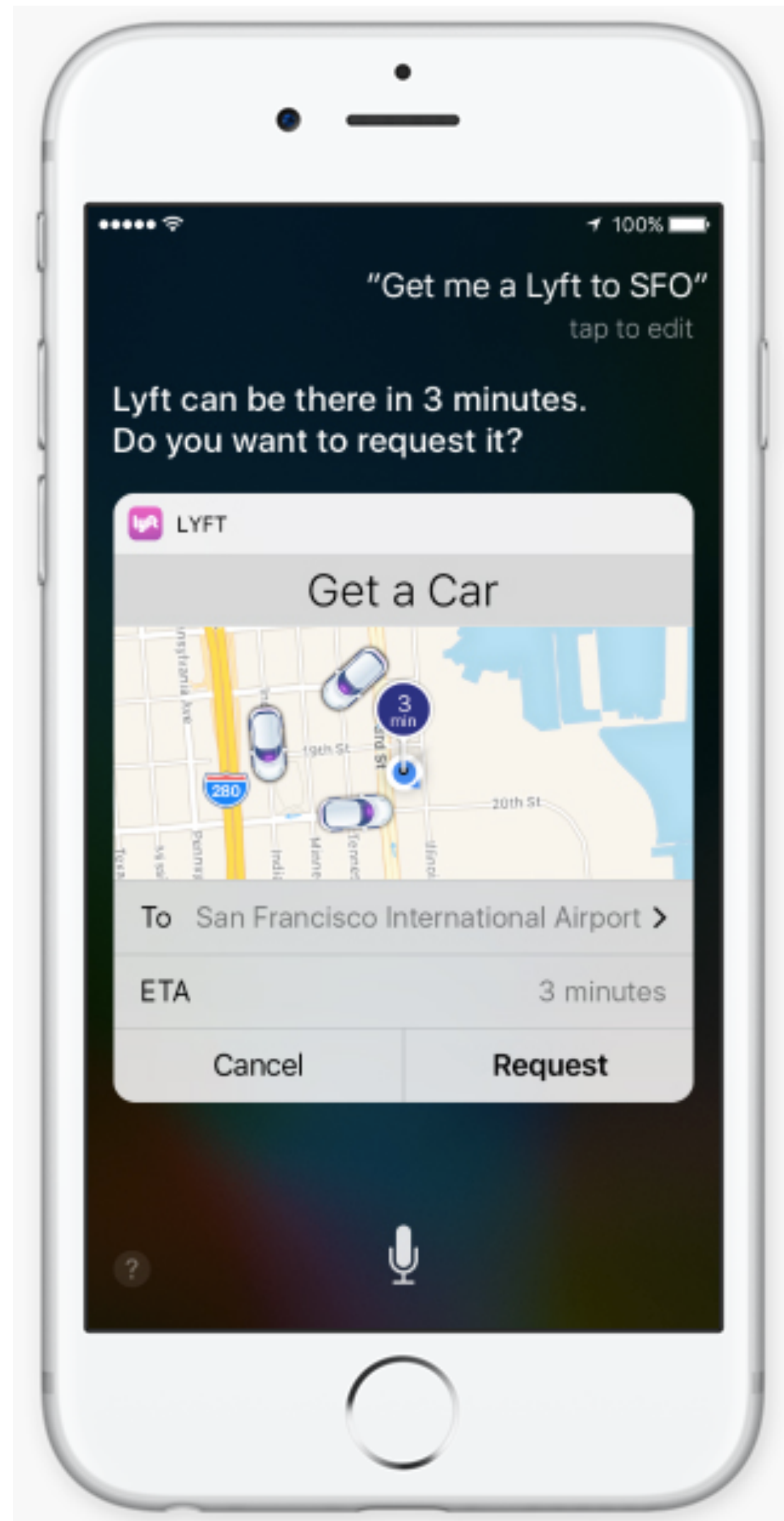
Development environments include standard classes that provide templates for navigation and the layout of screens.

Designs that incorporate these classes can use services, APIs, etc. that the operating environment provides.

This example uses an iOS UITableViewController.

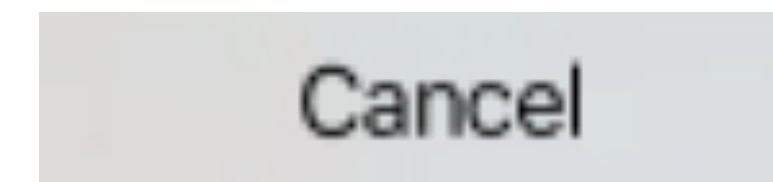
Because the length of tables can be very long, iOS implements a UITableViewController in a manner that makes efficient use of storage and processing power.

Mobile Apps: Conventions



Development environments encourage consistent apps.

You may never have used this app, but look at how many aspects feel familiar.



Design Tensions in Web and Mobile Applications

Designers wish to control what the user sees, but users wish to configure their own environments.

- Client computers and network connections vary greatly in capacity.
- Client software may run on various operating systems, which may not be the current version.
- Accessibility requires that designers do not take control of parameters such as font size.

Be explicit about the assumptions you make about the user's computer, web browser, etc.

In using style sheets, such as CSS, avoid over-riding user preferences.

Usability and Cost

- User interface development may be a major part of a software development project
- Good usability may be expensive in hardware or special software development
- Costs are multiplied if a user interface has to be used on different computers or migrate to different versions of systems

Design users interfaces that can be built with standard tools:

- Web browsers provide a general purpose user interface where others maintain the user interface software.
- Use the standard classes that the development environment provides.

CS 5150 Software Engineering

10. User interfaces for Web Sites and Mobile Devices

End of Lecture