

Designing Mobile User Interfaces

Info 152 Spring 2010

Guest Lecturer: Jeffrey Nichols

February 22, 2010

How do we make **good**
mobile user interfaces?

What is usability?

Ease of learning

- Faster the second time and so on...

Recall

- Remember how from one session to the next

Productivity

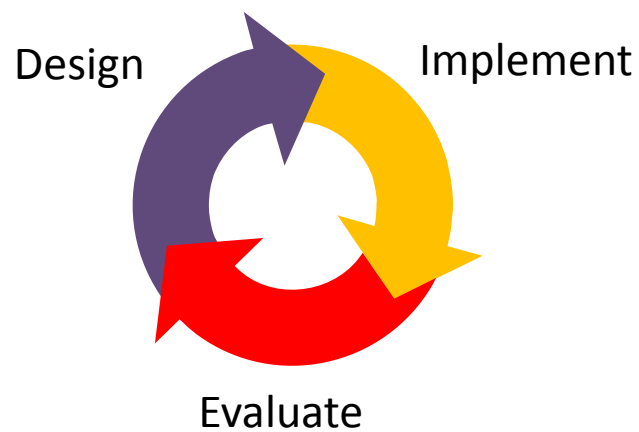
- Perform tasks quickly and efficiently

Minimal error rates

- If they occur, good feedback so user can recover

High user satisfaction

- Confident of success

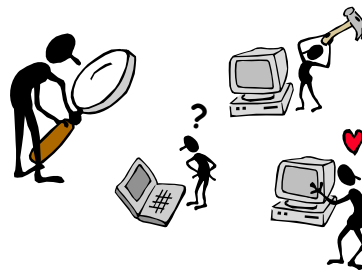


Building good user interfaces

- Task analysis & contextual inquiry
- Rapid prototyping
- Evaluation
- Repeat (iteration)

Task Analysis & Contextual Inquiry

- Observe existing work practices
- Create scenarios of actual use
- Try out new ideas before building software



Rapid Prototyping

Build an interface mock-up

Low-fidelity techniques

- Sketches
- Paper prototypes
- Video prototypes

Interactive prototypes

- HTML, JavaScript, Flash, etc.

UI Builders

- Visual Studio, etc.



Evaluation

Test with real people (intended users)

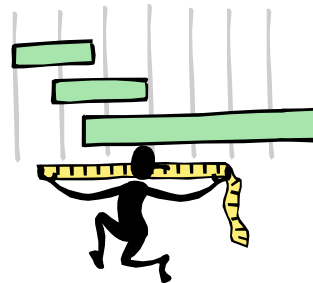
Build models

Low-cost techniques

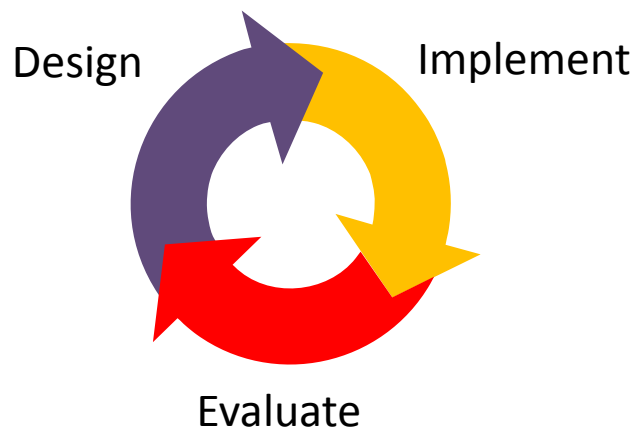
- Expert evaluation
- Walkthroughs

Higher cost

- Controlled usability studies



Iteration



Mobile Design



*Based in part on chapters from *Designing the Mobile Experience* by Barbara Ballard and *Strategic Mobile Design* by Cartman & Ting

What's the same?

- Rectangular bitmapped screens
- Text entry
- Pointer
- Audio I/O
- Network access
- Accelerated graphics hardware



What's different?

- Smaller screen
- Touch-sensitive screen
- Slower Processor
- Motion Sensor
- Location Sensor



Other Differences...

Personal

- Belongs to only one person

Communicative

- Connects with the network in many ways

Handheld

- Portable and operable on-the-go

Wakable

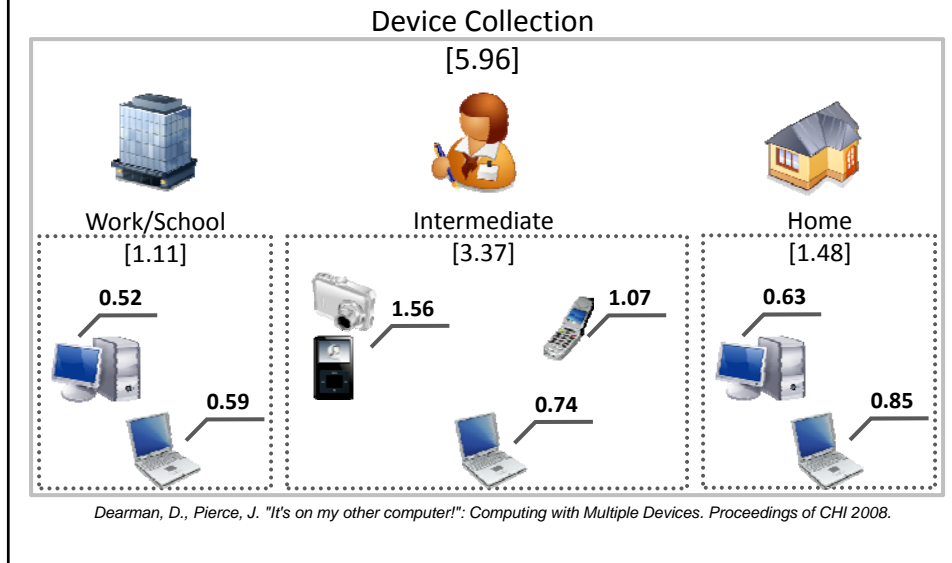
- Can be turned on quickly, either by user *or network*

The Carry Principle

Users carry the device with them all the time

- The mobile user is more available than a desktop user
- The mobile device is used with the user's other less mobile devices (desktop, laptop, etc.)
- The mobile user will be splitting their focus between the device and other external stimuli
- The mobile users social experiences may be modified both positively and negatively

People Use Multiple Devices



The Carry Principle

Users carry the device with them all the time

- The mobile user is more available than a desktop user
- The mobile user uses other devices (desktop, laptop, etc.)
- The mobile user will be splitting their focus between the device and other external stimuli
- The mobile users social experiences may be modified both positively and negatively

Final Important Difference

There is a great deal of variability across mobile platforms

- Many different kinds of devices
- Different screen sizes
- Different input devices
- Different web browsers
- Different UI conventions
- Different Carriers

Mobile User Interface Types

- Interactive Voice Response (IVR) Systems
- SMS/MMS
- Mobile Web
- Native Applications
- Hybrids

Interactive Voice Response (IVR)

E.g. Phone trees

Not the sexiest approach,
but works on all phones

Works for illiterate users

Still plenty of opportunity
for innovation

- Better Voice Recognition
- FonePal
- Etc.

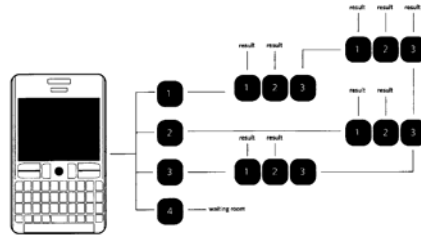


FIGURE 1.11 A sample IVR user flow.

SMS or MMS

Text or Image based interfaces

Compatible with most phones

Some apps:

- Google Search
- Twitter
- Voting apps (e.g. American Idol)
- Donations (e.g. Haiti)
- Text-based phone trees

Often used as an entry point:

- Web to SMS Push
- Text Keyword to Short Code

May be more viable in other markets
where data and voice rates are high,
but SMS is cheap (e.g. India)



Mobile Web

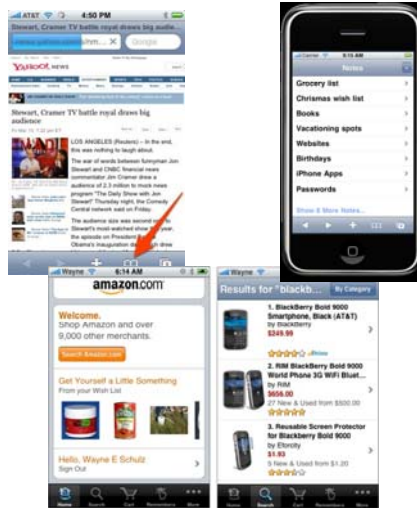
No modification needed for some devices/sites

No access to native hardware

JavaScript support reduced and slower

Can attempt to replicate native experience

- Must design new sites for each device
- Hard to do perfectly



Native Applications

Access to built-in hardware and software

- Location
- Accelerometer
- Graphics Acceleration
- Storage
- Native Widgets

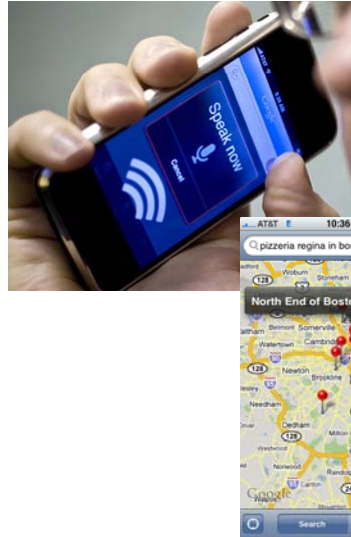
Better user experience, some restrictions

- Little or No Portability
- App Store Review Process



Hybrids

- Native Voice Applications
- Augmented Mobile Web Applications



Translating to Mobile

Mobilize, don't Miniaturize

- Applications should not work the same way in different contexts
- Direct translation does not work

“Rethink what is desirable and possible in the mobile environment”

- How do tasks and user needs change?
- How can extra hardware help?

Mobile Design Focus Points

- One-Handed Operation
- Text Entry
- Small Screen
- Consistent User Interface Style
- Personal Data
- Always On, Always Connected
- Battery-Powered
- Inconsistent Connectivity
- Changing User Context

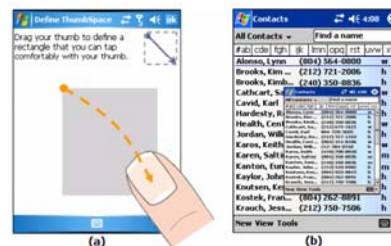
One-Handed Operation

Especially likely when the user is on-the-go

Also implies limited visual focus

Design guidelines:

- Use physical buttons or device features that can be felt
- Use larger controls that work with imprecise pointing
- Place interactive elements at bottom of screen
- Employ specialized thumb input techniques if needed (e.g. ThumbSpace)



ThumbSpace, A.K. Karlson and B. Bederson, INTERACT'07

Text Entry

“Trying to type on mobile is like trying to remove a contact lens with a cotton ball – it’s just not fun.”

–Aza Raskin

Solutions:

- Good defaults (e.g. current date, time)
- Auto-completion and predictive text entry (e.g. Fennec Awesome Bar)
- Use information from other sources (e.g. desktop computer, laptop)
- Alternate input methods, such as bar codes, camera images, speech
- Gesture-based text entry methods (e.g. ShapeWriter, Swype)



Small Screen

Limited support for multiple windows

Sharing data between apps is hard

Reading large amounts of text is difficult

Notifications cause context switch

Solutions:

- Provide customization options for large text displays (font size, number of page breaks)
- Do the work of sharing inside your application rather than force copy/paste (e.g. detect addresses and provide link to map application)
- Provide cues to re-orient user when returning from an interruption



Consistent User Interface Style

Each device has its own physical design and user interface style

Support for soft-keys may be important

General Solutions:

- If you must match style:
Be Thorough
- If matching is unnecessary:
Be Different



Personal Data

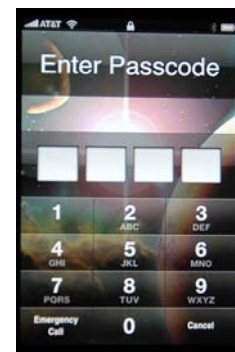
Most mobile devices are highly personal

Sensitive data and access credentials are likely to be stored on it

Loss will be noticed

Implications:

- Security is needed
- Text passwords are hard to enter
- Not all information and tasks require security
- Trade-offs are tricky



Always On, Always Connected

The user can be interrupted by a notification at any time

Users may want to use your application when it is socially unacceptable

Implications:

- Use “push” to inform users of *important* happenings (but know what important means)
- Provide cues to help a user reorient if they return to your application after an interruption
- Don’t require voice input, provide a fallback
- Avoid potentially disruptive feedback, such as audio cues or screen flashes

Battery-Powered, Inconsistent Connectivity

Wireless signal may drop at any time

Using the network eats battery power

Solutions:

- Avoid network usage on every interaction
- Store commonly-used data on the device
- Cache content as much as possible
- Pre-fetch content when power or network are readily available
- Don’t misrepresent age of data or completion of task when working around network problems



Changing User Context

Mobile applications will be used in different settings and contexts

Data and sensors are available to guess the current context (calendar, accelerometer, location, etc.)

Implications:

- If your application will be used differently in different contexts, design separate switchable interfaces for each context of use
- If you are trying to be intelligent:
 - Manage users' expectations
 - Make the internal state of the application clear
 - Provide an easy override for when you're wrong

Device Proliferation

How do we address the large variety of mobile devices?

- Targeted Design
- Least Common Denominator
- Class-Based Design
- Automatic Translation
- End-User Translation