

EECS 395

Day 19

Nathan Nichols

Today

- Optional rap session
 - Grocery list persistence
- Recap
- Final Project
- Mobile App Development

Recap

- Camera
- CoreLocation
- Multitouch
- Address Book
- Accelerometer

Final Project

- 40% of your final grade in this project
 - 5% is from demonstrations you will give
 - 10% is from documents you will turn in
 - 25% is from the project itself

Final Project

- You can work in small (two or three people tops) groups
- I will go through your proposal and make sure we agree that what you're doing is an appropriate amount of effort
- Your app should be on the order of the Recipe app
- You will have three and a half weeks to do the project
 - Recipes was four
 - One of the weekends is Memorial Day
- It's better to have a minor app than a buggy one

Important Dates

- Friday, May 15 - Two minute proposal to class
- Monday, May 18 - Written proposal due (through email)
- Monday, June 1 and Wednesday, June 3 - Demonstration, discussion, status update (5 or 6 minutes each)
- Wednesday, June 10 - Final demonstrations, turn in project, turn in writeup

Remaining class schedule

- The rest of the semester is going to go quickly
- Week 7
 - Final project, Todd Warren / WinMo, proposals
- Week 8
 - Simeon Peebler/iPhone gaming, no class, Instruments
- Week 9
 - Memorial Day, device comparison, ?
- Week 10
 - Project updates, project updates, ?

Remaining class schedule

- The rest of the semester is all discussion days
- I will be taking attendance, remember 5% of your final grade is participation
- Especially this Wednesday and next Monday, don't be late

Project Proposal

- Due one week from today, Monday the 18th
- Written document you email to me
- I won't be here next Monday or Wednesday
- Still due by start of class on Monday

Project Proposal

- Four parts
 - App description
 - Three or four sentences about what your app is and what it does
 - Competitive Analysis
 - I need three apps already available on the App Store that are related to your app, and give me a sentence for each about how your app is different

Project Proposal

- Technical due diligence
 - Give me two or three sentences about what is going to be the most technically challenging part of your project and why you are confident you will be able to do it
 - “My project requires making secure HTTPS POST requests, which I don’t have any experience with. I found a few good iPhone/HTTPS tutorials online, though, and I’ve seen some other apps that use it, so I am confident I can get it working.”

Project Proposal

- Project Plan
 - A list of the major features/programming constructs in your app
 - A short description
 - Whether it will be done by the midway point or by the final point

Project Proposal

- Recipe Project Plan
 - RecipeTableController - searchable list of recipes. Midway.
 - RecipeCategoryTableController - user selects a category, then browses recipes within that category. Midway.
 - Menu - user can add recipes to the menu, and the menu is automatically saved. Final.
 - Grocery list - Recipes in the menu automatically have their ingredients added to a grocery shopping list. Final.

Project Proposal

- Yours should be more detailed than that (more entries) but that's the idea
- We'll talk about the midway demo and final document turn in later
 - They'll be along the same lines
 - I'm going to require a Magic Touch
 - And there will be brownies at the final thing

Mobile Application Design

- Nate's algorithm for designing mobile applications
- Ask yourself these questions:
 - “What would be a really cool thing to do on a phone?”
 - “How would this work?”
 - “Why would this be on a phone?”
 - “What features of the device would be useful here?”
 - “What parts of this are inessential?”

Mobile Application Design

- Then
 - Throw out everything that's not essential
 - Build it
 - Add your Magic Touch
 - Polish polish polish

Mobile Application Design

- I've spent a lot of time trying to figure out how to organize today's talk
- The default thing I was going to do was compare desktop/laptop computers with mobile devices
- For example, did you know that the iPhone has a smaller screen than a desktop PC?
 - Der der der
 - “Der der der” aside
 - ““Der der der’ aside” aside

Mobile Application Design

- These things were so boring to type out, I can't imagine how boring it would be for you guys to hear me read them
- I've also decided that a comparison like that is the wrong approach
- Don't worry about how programming for the iPhone or Android is different than programming for Windows or OSX
 - It doesn't matter

Mobile Application Design

- Go native
- JLB's The Warrior and the Captive Maiden
- To help drive this home, we will play a little game for the rest of the class
- Everytime I compare a phone to a computer, bang your fist on the desk
- Test
 - Computers have physical keyboards, but the iPhone does not!

Mobile Application Design

- “What would be a cool thing to do on a phone?”
- What are the usage characteristics of mobile apps?
 - Shorter use sessions
 - More varied use environments
 - On a bus, in bed, at the doctor’s office
 - Less use overall

Mobile Application Design

- What are the design implications of these usage characteristics?
- Shorter use sessions
 - Needs to start up quickly, save automatically
 - Display useful information as quickly as possible
 - Saving state is probably a good idea
 - In your recipe app, if the user is looking at a recipe and then exits the app, it should probably show that recipe when the user restarts the app
 - (That's what `animated:NO` is for)

Mobile Application Design

- More varied environments
 - Be polite about sounds
 - Allow for your users to put your app down for five minutes and then pick it up again
- Less use overall
 - Can't charge as much
 - There's less buy-in from your users
 - Only 30% of paid apps are used after the first day
 - Only 5% are actively used after twenty days
 - (Pinch Media)

Mobile Application Design

- What does this mean for you?
- Simplify simplify simplify simplify simplify
- “Figure out the absolute least you need to do to implement the idea, do just that, and polish the hell out of the experience.” - John Gruber (DaringFireball.net)

Mobile Application Design

- Password Engine

The screenshot shows a mobile application interface for a password engine. The status bar at the top indicates 'SoftBank' service, signal strength, Wi-Fi, and the time '10:25 AM'. The app title 'THE PASSWORD ENGINE' is displayed in a dark header. The main interface consists of several input fields and controls:

- Global Code:** A text field containing 'testcode'.
- Provider:** A button labeled 'Provider' next to a text field containing 'Acme service provider'.
- Username:** A button labeled 'Username' next to a text field containing 'myusername'.
- Entry Code:** A text field with a 'Keep' checkbox and a toggle switch set to 'OFF'.
- Length:** A text field containing '12'.
- Punctuation:** A text field containing '!@#%\$^&*?_~'.
- Notes:** A large empty text area.
- Password Ready:** A section indicating the generated password: 'Password: poscal0@q7g8'.
- Strength:** A progress bar showing 'Good' strength.
- No Changes to Save:** A message indicating no changes are pending.
- Criteria Section:**
 - Pronounceable Strings:** A toggle switch set to 'ON'.
 - Use Words:** A toggle switch set to 'OFF'.
 - OTHER CRITERIA:**
 - Uppercase:** A toggle switch set to 'OFF'.
 - Lowercase:** A toggle switch set to 'ON'.
 - Punctuation:** A toggle switch set to 'ON'.
 - Numbers:** A toggle switch set to 'ON'.
- Bottom Bar:** A row of buttons: 'Save', 'Revert', 'Options', a plus sign '+', and 'Help'.

Mobile Application Design

- Because there is less buy-in from your users, they are less willing to put time into learning your app
- It needs to have as few options as possible
- Follow the familiar UI as much as possible
- It's UI needs to be discoverable
 - The chances of getting your user to read any documentation is exactly 0%
- Scatter bits of documentation around
 - If the user's menu is empty, the (also empty) grocery list should say why it's empty and how to fill it

Mobile Application Design

- “How would this work?”
- Sketch out possible user interfaces
 - Remember that the more native UI elements you use, the less your users have to think about and learn
 - These sketches don't have to be formal

Mobile Application Design

- Make use cases
 - It's very tempting to build use cases for infomercial audience participants
 - A great way to build use cases is to talk to users
 - Put the “user” back in “use cases”
 - For the workout app, ask someone who works out:
 - “What's the first thing you would want the app to show you?”
 - “How would you use it in the weight room, would you choose the machine or the muscle?”

Mobile Application Design

- This doesn't mean you allow some dude at the gym to design your app
- But he IS going to be your user/customer
- It's probably a good idea to talk to him

Mobile Application Design

- “Why is this on a phone?”
- Doing an app on the iPhone takes longer than doing it as a website or desktop app
- It has a smaller audience

Mobile Application Design

- There are two reasons for making an app mobile:
 - App-driven:
 - The app would work best on a mobile device
 - Programmer-driven:
 - You want to build a mobile app
 - You want to because someone is paying you, you're getting graded on it
- You need to understand why you're doing it
- Ocean's II

Mobile Application Design

- “What features of the device would be useful here?”
- Cool hardware and services is what separates Man from the animals
- Accelerometers, cameras, location, are all easy to use
- Keep your eye out for anytime you can use some of that hardware

Mobile Application Design

- The world is moving towards integrated services/mashups/APIs
- Data from Facebook put on a Google map with a song from iTunes and a picture of your Contacts, etc.
- With 3.0, iTunes can programmatically access the Contacts, Calendar, Music, Maps, and Email app
- The story is similar on Android, Pre, etc.
- And because they're always online, it's easy to connect to Facebook, Flickr, Twitter, Yelp, Last.fm, etc.

Mobile Application Design

- “What parts of this are inessential?”
- Life is too short, users are too distracted, and RAM is too low to crowd your app with inessential features
- Find the spine of your app, and ditch the rest
- What would you ditch in Recipes?

Mobile Application Design

- Conclusion
- Design mobile apps as mobile apps
- Expect shorter, more distracted usage sessions
- Keep the UI and any options/settings as simple as possible
- Users won't read any documentation or help
- Understand why you're building it for a phone
- Leverage as many mobile features as you can