COEN 168/268

Mobile Web Application Development

Introduction

Peter Bergström (pbergstrom@scu.edu)

Santa Clara University

Today's Agenda

- Introduction
- Native vs. Web
- Course Objectives
- Administrative stuff
- Demos

Who am I?

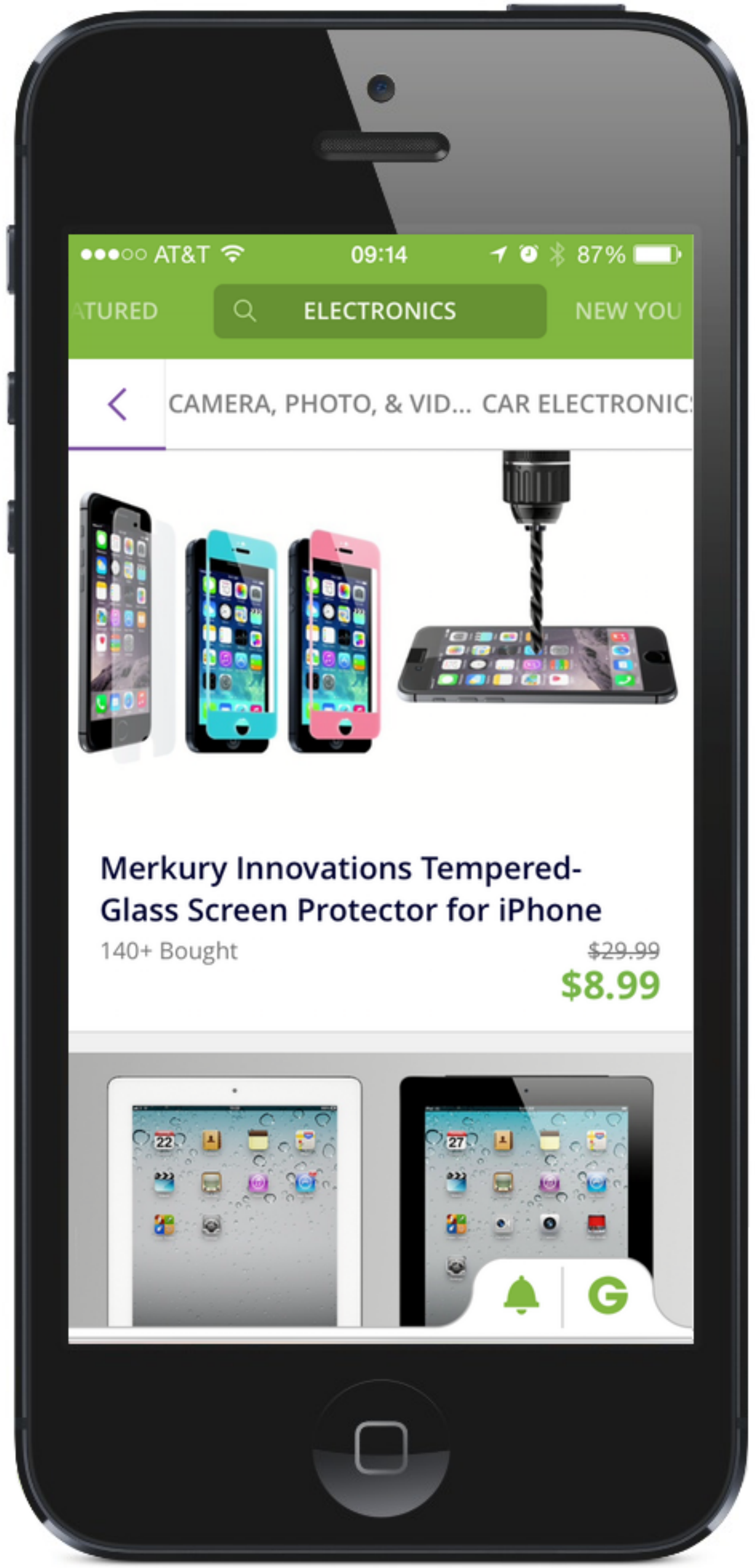
Peter Bergström

pbergstrom@scu.edu

http://www.peterbergstrom.com

I lead the development of the Groupon iOS app

- Over the 2.5 years I've been working on the app, I've see the Groupon transform into **mobile first** company
- One of the most popular apps in the Apple App Store. Named one of the top free apps of 2014
- More than 50% of Groupon's global business is mobile



But, I'm rooted in the web

In the past, I developed HTML5, native-style mobile and desktop web apps

- @Groupon (2011-2012), lead development of Groupon Scheduler, a web-based calendaring app using **Ember.js**

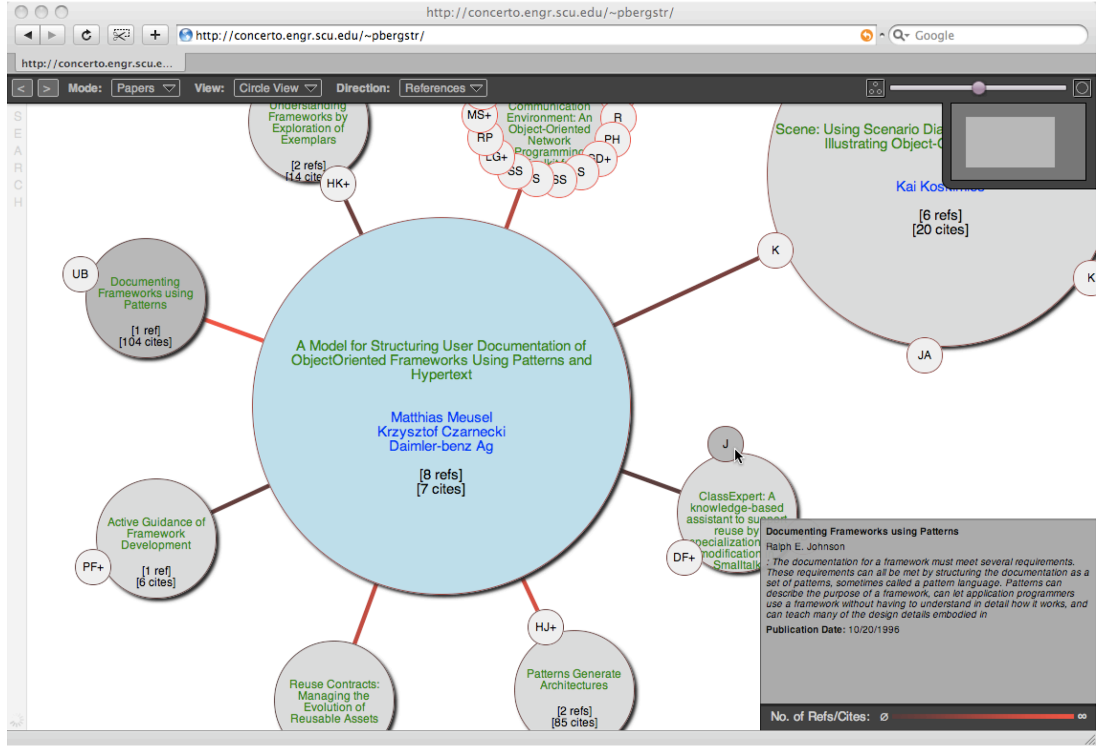
- @Strobe (2011), worked on eBay's tablet web site and Apple's iTunes U using **Ember.js/SproutCore 2.0** and **SproutCore**

- @Apple (2005-2011), lead and architected the development of MobileMe/iCloud.com calendar client using *SproutCore**

Also, I got my MSCE from SCU in 2009 while working at Apple

If you're working while getting your degree, I know your pain

My thesis project was a JavaScript-based visualization tool: PaperCube



http://concerto.engr.scu.edu/~pbergstr/

Mode: Papers View: Circle View Direction: References

Documenting Frameworks using Patterns

Ralph E. Johnson

: The documentation for a framework must meet several requirements. These requirements can all be met by structuring the documentation as a set of patterns, sometimes called a pattern language. Patterns can describe the purpose of a framework, can let application programmers use a framework without having to understand in detail how it works, and can teach many of the design details embodied in

Publication Date: 10/20/1996

Google
SEARCH
A Model for Structuring User Documentation of ObjectOriented Frameworks Using Patterns and Hypertext
Matthias Meusel
Krzysztof Czarnecki
Daimler-benz Ag
[8 refs]
[7 cites]
Documenting Frameworks using Patterns
[1 ref]
[104 cites]
UB
Active Guidance of Framework Development
[1 ref]
[6 cites]
PF+
Reuse Contracts: Managing the Evolution of Reusable Assets
Patterns Generate Architectures
[2 refs]
[85 cites]
HJ+
ClassExpert: A knowledge-based assistant to support reuse by specialization and modification in Smalltalk
DF+
J
Scene: Using Scenario Diagrams and Active Text for Illustrating Object-Oriented Programs
Kai Koskimies
[6 refs]
[20 cites]
K
JA
Understanding Frameworks by Exploration of Exemplars
[2 refs]
[14 cites]
HK+
Communication Environment: An Object-Oriented Network Programming
MS+
RP
LG+
SS
S
R
PH
D+
No. of Refs/Cites: Ø ∞

http://papercube.peterbergstrom.com

That's me, how about you?

- Your name
- What do you want to get out of this course?
- What's your experience with web technologies?

Mobile Application Development

Native vs Web!

Which is Best?

It depends as they both have pros and cons

Long live **native**, the web is dead.

Why Native?

- Deliver an optimized user experience
- Mature and full featured APIs
- Direct access to the hardware
- Fast and multithreaded
- Advanced graphics capabilities

Why **NOT** Native?

- You have to write different apps per platform: iOS, Android, Windows Phone, etc
- Expensive to develop
- Advanced and specialized technology stack
- App store approval
- Slow to patch

Long live the **web**, native is dead.

Why Mobile Web?

- Uses standard web technologies such as HTML, CSS, and JavaScript
- Write one app to be used in all platforms through responsive design
- Lighter weight
- Can still be deployed to app stores as a **hybrid app**
- Faster development cycles
- Update any time on the server

Why **NOT** Mobile Web?

- Cannot perfectly duplicate the native experience
- While you write one app, not all OSes and browsers are created equal
- You have to load your assets from a server real time
- JavaScript in browsers are typically not as fast as natively compiled code
- Can be slow if done wrong
- Hard to really optimize

Native vs. Hybrid vs. Web

	Device Access	Speed	Dev Cost	App Store	Approval Process
Native	Full	Device Access	Costly	Available	Mandatory
Hybrid	Full	Device Access	Not bad	Available	Low Overhead
Web	Partial	Fast	Not bad	Not Available	None

Taken from: http://www.slideshare.net/ErikPaulsson/hybrid-mobile-apps-16266715

This course will explore the mobile web-based approach

The good, the bad, and the ugly

Objective 1

Gain an understanding of how to create mobile web applications using **HTML5, CSS3, and JavaScript**.

Use technologies available in modern browsers such as the canvas tag, SVG, web workers, local storage, hardware integration, and more.

Objective 2

Develop **native-style, single page** mobile web applications using the **Ember.js** JavaScript framework that can run on multiple types of devices such as iPhones, iPads, Android phones and tablets, and even the desktop browser.

Ember.js is a JavaScript framework that was started at Strobe when I worked there and it evolved from SproutCore, which I worked on while at Apple

Objective 3

Use industry-wide design patterns for native application development and apply them to make JavaScript-based web applications more powerful and easy to manage.

Objective 4

Learn how to not only deploy these apps to run through a web-browser, but also in native app wrappers for **iOS** and **Android**.

High Level Course Logistics

- Tuesdays and Thursdays
- 7:10 - 9:00 AM
- Kenna Hall 214
- Course web site: http://coen268.peterbergstrom.com
- Assignments submitted electronically though Camino
- Office hours upon request (email me: pbergstrom@scu.edu)

My Office Hours

As I don't have an office, we have to coordinate office hours

- Available after class
- If you need to arrange another time, talk to me
- Always available via email

I work in Palo Alto so it can be difficult to coordinate meetings during the middle of the day unless I have advanced notice.

TA Sessions

- Also, Pratyusha Joginipally, my TA will be available to help!
- Lab is being scheduled right now and I will announce it in class.
- Hoping for the Design Center on Mondays and Wednesdays at 3-5 PM.

Course Schedule (Estimated)

- **WEEK 1:** Course introduction. HTML5.
- **WEEK 2:** Cascading Style Sheets. Desktop vs Mobile. Project Pitching.
- **WEEK 3:** JavaScript. Canvas tag using JavaScript.
- **WEEK 4:** REST, JSON, APIs. JS design patterns.
- **WEEK 5:** JavaScript MVC with Ember.js intro. Ember object model and templates.

Course Schedule (Estimated)

- **WEEK 6:** Ember naming conventions and routing.
- **WEEK 7:** Ember controllers and components.
- **WEEK 8:** Ember Data and testing.
- **WEEK 9:** Optimizing apps for production. iOS development overview and how to include mobile web apps in native app wrappers as a hybrid app
- **WEEK 10:** Project demos

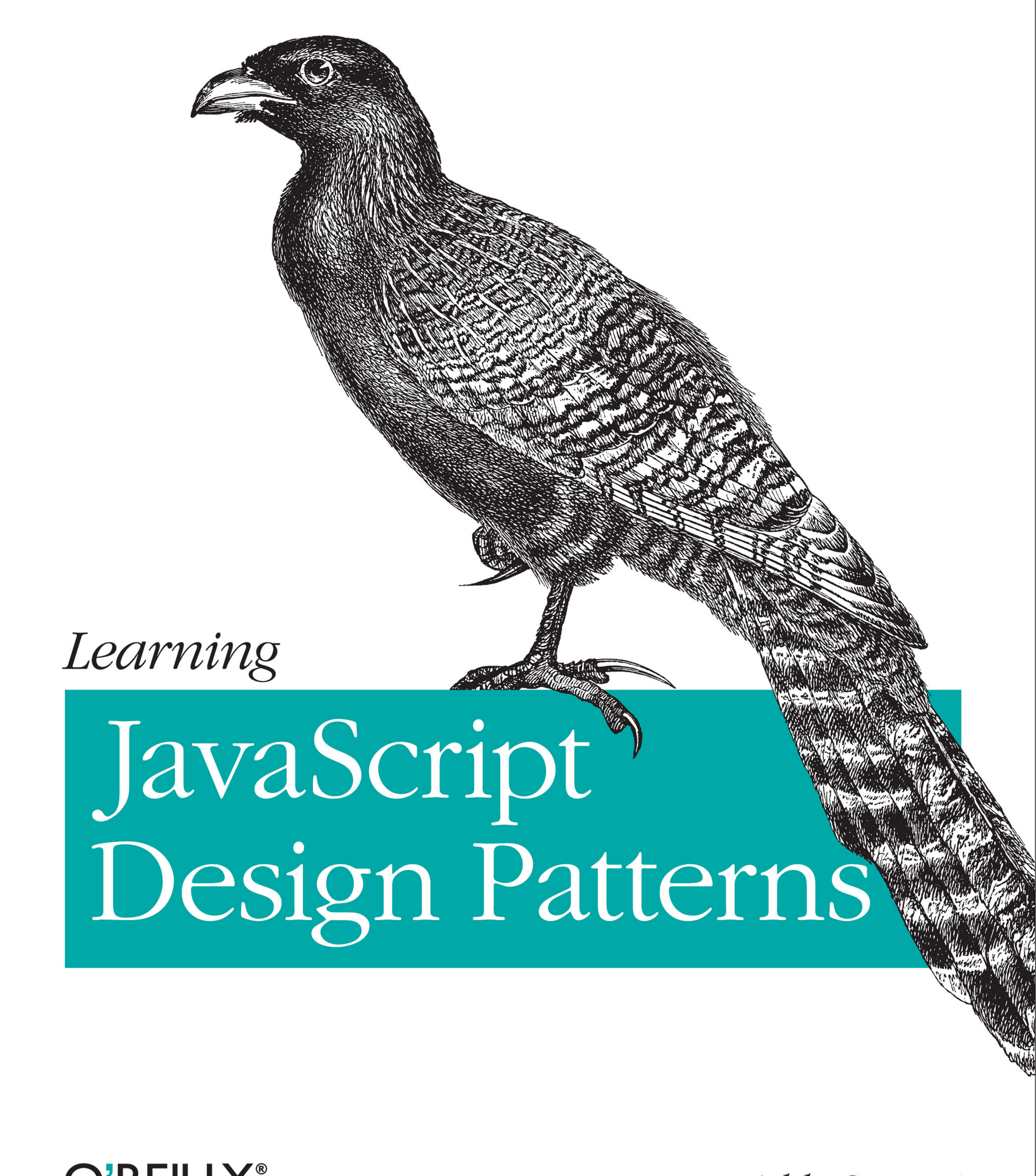
Recommended (but Optional) Textbooks

JavaScript: The Good Parts

By Douglas Crockford
Publisher: O'Reilly Media, 2008
ISBN:0596517742

Learning JavaScript Design Patterns

By Addy Osmani
Publisher: O'Reilly Media, July 2012
ISBN:1449331815



Grading

- Programming assignments: **20%**
- Quizzes: **20%**
- Project proposal: **5%**
- Project checkpoints x2: **5% each**
- Final Project Submission: **20%**
- Class project Presentation: **5%**
- Final Exam: **20%**

Programming Assignments

- No collaboration on assignments!
- Will be give frequently to get you familiar with mobile web app development
- There will be 3 programming assignments
- No collaboration on assignments!

Submitting Programming Assignments

- Submitted electronically though Camino
- All assignments will be due at **6:00am on the due date.** (1 hour before class)
- Late assignments will be deducted 1% per minute late, up to a maximum penalty of 50%.
- No late assignments accepted after the last assignment is due.

Quizzes

- Every two weeks on Thursdays
- 4 Quizzes
- Not meant to be tricky, but help you review what is important

Academic Integrity

- **Be proud of the code that you craft**
- Do your own work!
- Properly attribute open source projects and other sources
- Do not share code except on collaborative projects
- If you are caught cheating, you will get **0%** on the assignment

What development tools will you need?

- A text editor such as TextMate for writing your apps
- Chrome web browser for development and debugging
- XCode for the iOS portion of the course
- Git for source control is recommended

Final Project

- 40% of the total grade
- 1 or 2 students per team (recruit next class on Thursday)
- Create a **mobile web application** of your own choosing
- Last week of the course, you demo for the whole class
- Finals week, you submit your final code to me
- See **http://coen.peterbergstrom.com/project** for more info

Project Need to Use JavaScript Frameworks

- Should be written in a JavaScript MVC framework
- The use of Ember.js is required for this project
- JavaScript gaming frameworks are also OK if you want to do that

Project Need to Be Mobile Focused!

- Remember, you're making a **mobile** web application
- Should run on a phone or tablet
- Although more difficult, try to also have it run as a hybrid app on mobile device

Final Project Schedule

- **Project Pitch:** Due by Jan 13th, 2015 @ 6AM (5%)
- **Milestone 1:** Due by Feb 3rd, 2015 @ 6AM (5%)
- **Milestone 2:** Due by Feb 24th, 2015 @ 6AM (5%)
- **Presentations:** In class - Mar 10th & Mar 12th, 2015 (5%)
- **Final Submission:** Due Mar 19th, 2015 @ 6PM (20%)

Total: 40% of course grade

Project Pitch - January 13th, 2014

- Write a one page document describing your mobile web app and submit on Camino
- Talk about the UI and the functionality
- Also, talk about why you want to make this particular app
- Submit one per team. Include names of team members and a descriptive name of the project
- I will create Camino groups for each app with those app names

Project Milestones

- These are checkpoints on the project
- You should submit your current progress (code, plans, questions, etc)
- I will give you feedback on the project so far and answer questions you may have

Final Project Submission

- Like the milestone submission, but your final cut of your code
- Worth up to 20% of total grade (depending on difficulty multiplier)

Project Difficulty Multiplier

- To reward students who work on challenging projects, the final submission grade has a multiplier.
- A total of 20% of the final grade is possible but only with the difficulty multiplier:
    - 5% x (difficulty level) (1-4, 4 being the most difficult)
- As part of project pitch and first milestone, I will give feedback on the difficulty of the project to maximize your scores

Project Difficulty Example

- Tim is working on a very challenging project that will take him 100 hours of coding, he would be able to get a multiplier of 4x making his maximum possible final submission score be 20% of the total course grade.
- Andy is working on a very easy project that he can get done in 2 hours right before the deadline, he would only get a multiplier of 1x making his maximum possible final submission score be 5% total course grade.

Backend technologies

- I don't care what you use as long as it talks JSON
- However, I do suggest using services such as Parse to reduce the amount of work you will have to do
- But, feel free to roll your own, if you want to

Project Tips

- Most successful apps either:

  1. Solve a meaningful problem

  2. Entertains

- Try to make an app that is useful to you
- Use something like Git or even Github for version control
- Have fun

I think you can write *almost* any type of mobile app using JavaScript

So, come up with some great ideas!

Let's see some demos showing what you can do with mobile web apps

Student Project Demo

Museum App

Demo

Groupon iOS App

Demo

GrouponLite Ember App

COEN 168/268

Mobile Web Application Development

Introduction

Peter Bergström (pbergstrom@scu.edu)

Santa Clara University