software
studio

intro & logistics

Daniel Jackson

For now, all rights reserved. Daniel Jackson, 2012.
introductions

lead instructors
› Daniel Jackson and Jonathan Edwards

assistants
› Stephan Boyer
› Zack Drach
› Katie Everett
› Dalton Hubble
› Anurag Kashyap
› Jess Lin
reviewing 6.170 (Fa11 and Sp12)

hot?
› realistic, open-ended projects
› focus on web applications
› the MIT way, not always the standard way
› pick your own project

not!
› case-study-based lectures
› traditional grading
› DIY infrastructure
what are the goals?

for students
› develop design skills
› learn new programming idioms
› be a more thoughtful programmer
› get familiar with technologies

for everyone
› more fun, less overloaded
› so time to think more deeply...
what are design skills?

starts with the problem
  › never obvious, rarely fixed

behavior not just internals
  › designing what, not how

articulate, propose, critique
  › architect, not bricklayer

aesthetics & style
  › aim to be disruptive, not just fit in
key innovations of course

class as studio
› we’ll discuss your ongoing work

projects as evolving portfolios
› incremental, on GitHub, deployed

mastery-based grading
› roughly: B for keeping up, A for excellence
› just show mastery once, but in all four areas

flipping the classroom
› nuggets in class; tutorials for basics; book+videos+websites

standard platform
› Rails in Berkeley VM, GitHub, Heroku
Berkeley SaaS course

book
› from Amazon CreateSpace, with 20% discount code
› electronic or softcover

short videos
› see Stellar for links

virtual box VM
› should eliminate platform problems
how projects will work

solo projects
› no collaboration outside class
› phased (by functionality and deliverables)
› (1) get going (2) build MVP (3) build cool app (4) critique
› after each phase: 3 teams of 3 do peer reviews

final project
› you choose team and problem: start now!
› non-profit, or MIT, or civil society
why X?

why Ruby?
› Rails: lets you avoid SQL
› strong culture of elegance & simplicity
› cleaner syntax, more uniform than Python
› good to learn another language

why GitHub?
› most popular open source host
› good support for issue tracking too
› they gave us free private repos (thank you!)
your turn

enrollment form
› by 5pm today

read website
› general info
› solo project overview

tutorial tomorrow
› load material in advance
› pick best time (but no 3-4pm this week)

solo projects
› P0: bring to class on Monday
› P1.1: checkin by Sunday midnight
remember

6.170 is a work in progress!

expect mid-course corrections!