RUTGERS
THE STATE UNIVERSITY
OF NEW JERSEY

Mobile App Engineering & User Experience: Introduction, Syllabus

Gradeigh D. Clark

Department of Electrical and Computer Engineering
Rutgers, the State University of New Jersey

September 6th, 2016

General Information…

- Instructor: Gradeigh D. Clark
  - Email: gradeigh.clark@Rutgers.edu
  - Office hours: TBD, let's do appointments only for now
  - Office location: CoRE 501
- Teaching Assistant: Can Liu
  - Office hours: TBD
  - Office location: TBD
- Classes meet Tuesday & Fridays, 12:00PM – 1:20PM in DSV Lab (is that the name?)

Course Learning Outcomes

- From the ECE handbook….
- 1.An ability to develop both user-oriented and systems-oriented Android programs
- An ability to conduct user-centered design for mobile applications
- An ability to understand programming constraints with mobile platforms
- Understanding how different program hierarchies affect energy-efficiency, user experience, and security
- An understanding of the Android programming models and development tools

Course Outline

- Previous iterations of the course were:
  - To create mobile applications?
  - To learn aspects of mobile devices?
  - To do human-centered design of applications?
- Trying to do too much all at once with two groups (grad, undergrad) of different skill levels

Blast from the past…

- [Open up previous lecture slides, 7th – Security]
- [Open up previous lecture slides, 8th – Location]

New course outcomes

- New goals:
  - Re-tooled focus entirely on Android programming & APIs
  - Students should know how to:
    - Create and manage various UI elements: Button, TextView, LayoutInflaters, View, Canvas, etc …
    - Perform FileIO on Android with Java APIs
    - PROPERLY! Manage the Android activity lifecycle and cause sustained connections between screens, learn how NOT to block UI threads
    - Deal with local databases / SQL on device
    - Generate notifications, manipulate sensors, manage Services
    - …. And more
- Some topics still being determined: Multithreading, Intents, Drawing, Input methods, Widgets …

What about graduate students?

- Do the course outcomes differe?
  - Answer: not really…
  - Extra work will be given, but objectives not different

Course Organization

- Planning for 4~6 homework assignments, one per major topic covered
  - One every 2 to 3 weeks, depending on difficulty
  - Given 10-14 days to implement
  - Individually done
    - You are being warned ***right now*** that we are using a code similarity algorithm to catch cheaters. Heaven holds the faithful departed that go down this path
    - You will be reported immediately to Academic Integrity office for both Undergraduates / Graduates
    - You WILL fail the class if you are caught
    - **Graduate students, getting an F can get you EXPELLED from the university. This is not a game. Be smart.**
    - http://academicintegrity.rutgers.edu/

Late Policy

- • Can use ONE 72hr grace period during the semester without penalty on HOMEWORK only
  - (intended for illness and other unforeseeable issues).
- No permission, notification needed.
- Afterwards, 20% penalty per beginning 24hr period.
- No assignments will be accepted after 72hrs.
  - No exceptions.

Course Organization

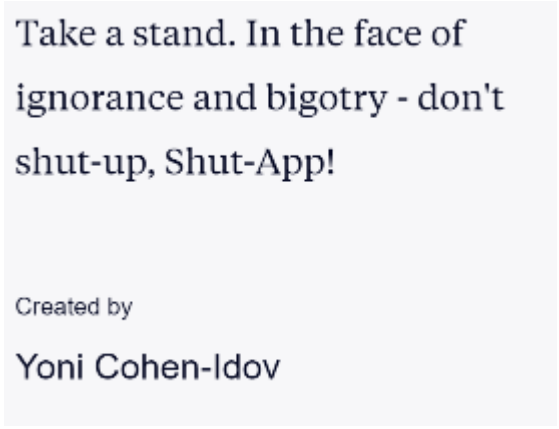
- This course will have unannounced quizzes
  - Test attendance, mostly, they are not difficult
  - Sometimes will have more than one quiz per lecture
  - If you can't make the quiz, you should be informing the instructor or the teaching assistant ahead of time with a valid, **verifiable** excuse
- Will have up to two written exams **or** programming practicums…
  - Depends on our progress
  - Also depends on how gerrymandered the computing setup can be

Final Course Project

- We will have a final project substituting in for a final exam
- Idea is to work collaboratively on an idea / concept application for Android
- You have will soon receive an assignment to do:
  - Form a group of between 2~4 people
  - Name your team, create accounts on Github
  - Create a private Github repository (note: you can get free private repos through Github Student using your Rutgers Email)
  - Link myself & the TA to your private repository
  - More details to come …
- **Note: a list of BANNED application ideas will be sent out as well. List is made of previous year student projects.**

Aside: What apps are popular now?

- Went to Kickstarter to look…
- Found the following: https://www.kickstarter.com/projects/1826759255/shut-app
- Shut-App!
- Description box ->



Aside: Shut-App

Have you ever felt yourself lost for words or too flustered to react in the face of overwhelming hate, stupidity, misogyny, ignorance or bigotry?

Aside: Shut-App

Too often, we reluctantly stay silent in those situations. We're too flustered or exhausted by sheer absurdity, or don't know the facts well enough, or intimidated by the loudmouth idiot, or know what we want to say but can't quite put it together, or always think of the perfect response 10 minutes too late.

Aside: Shut-App

The end result is the same - we're frustrated, they get the alleged upper-hand, and sometimes even sway others around.

Aside: Shut-App

Whether it's the racist candidate and his supporters, your homophobic uncle at Thanksgiving dinner, misogynistic colleagues at work, or Facebook "friends" denouncing everything from science to feminism to civil rights -

Aside: Shut-App

I am sick and tired of racist / sexist / xenophobic / homophobic / fact-repellent / logic-resistant people 'winning' arguments and gaining traction just by dint of being louder, allegedly-tougher, or spewing falsehoods without consequence

That's why I came up with "Shut-App!"

It's time to take a stand. Don't shut-up. Shut-App!

Aside: Shut-App

I want to make an app that puts the best answers, most persuasive arguments and most rhetorically-powerful responses to all common asinine arguments on all the hot topics - at the tips of your fingers.

It's a tool that empowers anyone and everyone to take a compelling stand, regardless of access to reading materials or possession of rhetorical savvy.

Aside: Shut-App

Aside: Shut-App (last)

Prerequisites

- Programming Methodology I/II, or equivalent course knowledge:
  - Data Structures (stacks, queues, linked lists, sorting)
  - Basics of object oriented programming
  - Virtual functions, virtual base classes, OO design…
- **Primarily: do you know Java?**
  - **!!! Prerequisite Quiz is coming for this !!!**
- Other things needed:
  - Android Studio
  - Reference textbook
  - Github account, working knowledge of git commands
  - A central nervous system

Helpful Resources:

- Sites:
  - developer.android.com/index.html
  - https://developer.android.com/studio/install.html
  - https://desktop.github.com/
  - http://learngitbranching.js.org/, http://tom.preston-werner.com/2009/05/19/the-git-parable.html
- Some reference textbooks:
  - Android Programming, the Big Nerd Ranch Guide (2nd edition)
  - Android Programming Concepts
  - Learning Moble App Development: A Hands On Guide to Building Apps with Ios and Android
  - Introduction to Android Application Development, Android Essentials (4th Edition)
  - https://commonsware.com/Android/
  - https://developer.android.com/reference/packages.html

Lecture Slides

- Most lecture slides in the beginning will be sourced from CS193A @ Stanford, courtesy of Marty Strepp.
- We really appreciate that the course authors have released their content under a creative commons attribution 2.5 license. To comply with the license, we link to the license, give Stanford credit, do not use the materials for commercial purposes, and distribute our own version of the materials under the same license. We have modified the originals in a variety of places.