Syllabus
Java and Android programming
Java and Android Programming

Instructor: Gongjun Yan, Ph.D.
Office: BE 2046
Telephone: (812) 228 5073
e-mail: gyan@usi.edu
Web site: Blackboard

Office hours: 1) Monday 9:00-5:00 2) Wednesday 10:00-5:00. You can also make an appointment to meet with me. Please email me or call me for an appointment.

Class Meeting Times: Tuesday and Thursday 1:30-2:45pm.

Book: not required

ANDROID: HOW TO PROGRAM-W/ACCESS

Author: DEITEL
ISBN: 9780133764031

Other reference:
   Paperback: 250 pages
   Publisher: Pragmatic Bookshelf; 2nd edition (October 23, 2009)
   Language: English
   ISBN-10: 1934356492
   Product Dimensions: 8.9 x 7.5 x 0.9 inches
   We are going to program on Android simulator and smartphone using Java!!!
3. Big Java: Compatible with Java 5, 6 and 7
   Paperback: 1168 pages
   Publisher: Wiley; 4 edition (December 30, 2009)
Course Objectives:
The purpose of this course is to teach you how to program using Java and apply this knowledge to Android Platform in smart phones. This course will give you an opportunity to learn the currently hottest technology and to use the skills you have gained as an undergraduate student to make marketable software in smart phones. We will learn the general theme of software engineering for all topics. All topics will address fundamental Java programming language faced by professionals in IT industry. Another objective of this course is to provide students an environment to experience the life-cycle of software development. We will have projects to make marketable software for smart phone by combine Android and Java together.

Topics covered: the android development environment including Android studio, android SDK and phone emulator; key programming paradigms; UI design including views and activities; data persistence including SQLite. The advanced topics which will be selected based on how the class goes include: content providers; messaging and networking; phone sensors, location based services (e.g., Google Maps), background services; broadcast receivers; cloud programming using App Engine; and publishing applications to the android market.

Android programming concepts are reinforced through a set of thematic programming exercises that introduce these topics and incrementally allow the student to build a complex application.

A key part of this course is group projects where students will work in small teams for joint problem solving.

Disability Services
If you have a disability for which you may require academic accommodations for this class, please register with Disability Resources (DR) as soon as possible. Students who have an accommodation letter from DR are encouraged to meet privately with course faculty to discuss the provisions of those accommodations as early in the semester as possible. To qualify for accommodation assistance, students must first register to use the disability resources in DR, Science Center Rm. 2206, 812-464-1961, http://www.usi.edu/disabilities. To help ensure that accommodations will be available when needed, students are encouraged to meet with course faculty at least 7 days prior to the actual need for the accommodation. However, if you will be in an internship, field, clinical, student teaching, or other off-campus setting this semester please note that approved academic accommodations may not apply. Please contact Disability Resources as soon as possible to discuss accommodations needed for access while in this setting.

Title IX
USI does not tolerate acts of sexual misconduct, including sexual harassment and all forms of sexual violence. If you have experienced sexual misconduct, or know someone who has, the University can help. It is important to know that federal regulations and University policy require faculty to promptly report incidences of potential sexual misconduct known to them to the Title IX Coordinator to ensure that
appropriate measures are taken and resources are made available. The University will work with you to protect your privacy by sharing information with only those who need to know to ensure we can respond and assist. If you are seeking help and would like to speak to someone confidentially, you can make an appointment with a counselor in the University Counseling Center. Find more information about sexual violence, including campus and community resources at www.usi.edu/stopsexualassault.

**Weekly Lecture Organization:**
Each lecture will consist of series of questions and demos. The instructor will give many exercises of each topic and make sure everyone understand how to use it. Every week, we will have a lab to practice the skills we learn. The labs are supposed to be finished in class and no assignments left after class.

**Class Participation**
We will do intensively discussion on a forum created on our course website. Every week, one student (the “session leader”) will be selected and will lead the discussion on that week by identifying questions and/or issues that should be discussed and debated.

The grading for the class participation includes: 1) class attendance (50%); 2) assignment submission (20%); 3) class discussion (20%); 4) discussion forum (10%).

**Group Project**
Group projects are made up of 3 students.

The project starts in week 5, pitches week 6. The last two weeks of class are held over for working on collaborative project -- there will be no classes during the last two weeks of term.

The grade will breakdown as follows:

- Project webpage, team and project pitch
- Reviews
- App demo day. The app/code has to work.
- Report

There is a focus on **working code** and a **demoable app**. Rules of demo day. The professor should be able to download the application from Google Play and it should work off the shelf.

The ground rules; the **teams and projects** will:
Form their own groups and come up with the theme of their app;
Make a webpage with 1) one page pitch; 2) team names; 3) project name; 4) slides;
Make a 90 sec elevator pitch in front of the complete class;
Have a design/code review;
Publish it on Google Play Store; and
Show their working app on demo or die date.
Write a report

Project Deadlines

1. Teams and project set up no later than Tuesday September 15, 2015
2. 90 seconds project pitches during x-hour Tuesday September 22, 2015

Each team gets 90 seconds to pitch. 1 slides only linked into your webpage.
Plus a 1 pager project description: What is the idea of the app?


Similar to MyRuns design document
Top Level Design Document (approximately 5 pager).
Include: Revised project description; project should focus on a single simple idea; UI, top level systems/component design diagram; and a brief description of classes


You should show working code during this checkpoint. The more the better.
See Fanglin for sign up. Each team gets 20 mins. Campbell's office.
Free format discussion. Demo some code.


All students should be able to download each groups app and play with it.
Best project will be voted on by the complete group.
App must be available for download from Play or project website.
Bonus points for Play but takes 48 hours to process.
App has to work!
5. Project report and code submitted: December 8, 2015 before midnight

**Project Grading**
The project grade will be based on four components, counting for a certain percentage of the final grade (please refer the percentage on Syllabus):

1. The creativity, accuracy, and clarity of the technical description of your project
2. The clarity and correctly of your project documents
3. The effectiveness of your project in an informative and interesting way
4. The program's correctness (i.e., whether it runs properly) and the clarity of the program document, if your project is programming.
5. The completeness of the required documents.
6. Software coding takes 50%, presentation takes 20%, the project writing takes 30% of the total project score.

**Assignments**
There will be about 12 assignments. The lowest 2 assignments score will be dropped.

**Grading**
The course grade will contain the following components:
(Note that these percentages are only approximate and are subject to change, but by no more than 10%.)

<table>
<thead>
<tr>
<th>Projects</th>
<th>30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>70%</td>
</tr>
</tbody>
</table>

**Grading**

**70% - Thematic programming exercises**

There are 6 weekly programming assignments over the first 7 weeks. These labs are designed to help students learn the android programming environment and key programming paradigms. Assignments are done individually. Each lab will receive the same percentage of the grade.

Each student will demonstrate their standalone MyRuns5 (after lab 5 is complete). We will meet on the green to assess your app. You will be asked to demo two activities: walking around the green and running (if you don't feel up to running we will find someone to run with your phone).
Your MyRuns5 app should correctly capture the activity (walking and running) and the details of the exercise. Andrew should be able to see the exercised saved in your history tab displayed on Google Maps.

Labs are graded on completeness of the required features, the correctness of the functionality, and the robustness: **note, if your lab crashes it will not be graded and you will have to resubmit a working version with 20% penalty against your lab grade.**

**30% - Group projects**

Students will develop their own app.

Projects are made up of a small (2 people) teams and require strong collaboration and a problem solving mindset. The goals of this activity are to help you develop the confidence, skills, and habits necessary to write real phone apps while part of a multi-person team.

Each team member will get the same grade assuming all goes well. The grade will breakdown as follows:

4% The Pitch

4% Design document. Similar to the MyRuns design document.

4% The Checkpoint: Design/code/project review. Important, some running code must be demoed; that is, as much running code that can be demoed during this review the better.

14% Demo and Code

4% Report

**Grades:**

Final grades will be assigned according to the following schedule:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90-100</td>
</tr>
<tr>
<td>B+</td>
<td>87-89</td>
</tr>
<tr>
<td>B</td>
<td>83-86</td>
</tr>
<tr>
<td>B-</td>
<td>80-82</td>
</tr>
<tr>
<td>C+</td>
<td>77-79</td>
</tr>
</tbody>
</table>
C = 73-76
C- = 70-72
D+ = 67-69
D = 63-66
D- = 60-62
F = 0 - 59

**Late Submissions**
The due date and time are specified for all assignments. Assignments turned in after the due date and time will be penalized 5% for each day the assignment is late.

<table>
<thead>
<tr>
<th>Turned in less than or equal to...</th>
<th>Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>24 hours (1 day) late</td>
<td>- 5%</td>
</tr>
<tr>
<td>48 hours (2 days) late</td>
<td>- 10%</td>
</tr>
<tr>
<td>72 hours (3 days) late</td>
<td>- 15%</td>
</tr>
<tr>
<td>etc.</td>
<td>etc.</td>
</tr>
</tbody>
</table>

**Grading policy:**

**Cheating is zero tolerance.** If the instructor catches one cheating in assignments, projects or exams, the grade of the student who makes the cheating will be “F” as the final grade.

**Schedule (tentative)**

Week 1

- Introduction
- Lecture 1: Getting started

Week 2

- Lecture 2: The wonderful world of Android

Week 3

- Java: using objects, input/output
- Lecture 3: Our First Android Application
Week 4

- Java: implement class
- Lecture 4: User Interface I
- Lecture 5: User Interface II

Week 5

- Java: Data type, decision
- Lecture 6: Activity Lifecycle
- Lecture 7: A very cool activity lifecycle app to play with

Week 7

- Java: Iterations (loops)
- Lecture 8: Using the Camera and Data Storage

Week 8

- Design Document due Thursday before midnight.

Similar to MyRuns design document. Top Level Design Document (approximately 5 pager). Link the document to your project webpage; Include: Revised project description; project should focus on a single simple idea; UI, top level systems/component design diagram; and a brief description of classes. Each project group should have one submission. There should be the following directories:

- Pitch -- pitch slides
- DesignDoc -- the design.pdf
- Code -- source code
- FinalPresentation -- final presentation slides
- Report -- final project

No Classes

Week 9
Java & Android Programming

- Java: Designing classes
- Lecture 9: Fragments and ActionBars
- Lecture 10: Debugging
- Lecture 11: Dynamic Layouts using the Fragment Manager

Week 10

Project Checkpoint: Each team gets 20 mins to discuss and demo the project so far.

- Lecture 12: Tools
- Lecture 13: Using PreferenceFragment to store user preferences

Week 11

- Lecture 14: Customizing Dialogs with DialogFragment

Week 12

Project Checkpoint: Each team gets 20 mins to discuss and demo the project so far.

- Lecture 15: SQLite Database

Week 13

- Lecture 16: Location-based Services

Week 14

Project Checkpoint: Each team gets 20 mins to discuss and demo the project so far.

- Lecture 17: Google Maps or http client

Week 15

- Lecture 18: Services, BroadcastReceivers and Notifications
- Lecture 19: AsyncTask

Week 16
Project Checkpoint: Each team gets 20 mins to discuss and demo the project so far.

Presentation practice

All students should be able to download each groups app and play with it. Best project will be voted on by the complete group. App must be available for download from Play or project website. Bonus points for Play but takes 48 hours to process. App has to work!

- Project report, presentation and code submitted: Thursday 29 May before midnight.

- All project products: pitch, design doc, code, final presentation and report have to be linked into your project page.

Week 17

- Final Presentation