CSCI 448 – Mobile Application Development
Spring 2016
Syllabus

Instructors:

William Hoff
Office: 314H Brown Hall
Phone: x3761
Email: whoff@mines.edu
Office Hours: Monday 1-3, Wednesday 1-2, or by appointment

Jeffrey Paone
Office: 280C Brown Hall
Phone: x2587
Email: jpaone@mines.edu
Office Hours: Tuesday 1-3, Thursday 10-12, or by appointment

Grader:

Cara Beutel
Email: cbeutel@mymail.mines.edu

Lectures & Labs:

Mondays, Wednesdays, Fridays 3:00 PM – 3:50 PM
Marquez Hall 026

Textbook:

Required:
Android Programming: The Big Nerd Ranch Guide, 2nd ed., Bill
Phillips, Chris Stewart, Brian Hardy, & Kristin Marsicano, 2015.

Optional:
The Busy Coder's Guide to Android Development, Mark Murphy.
https://commonsware.com

Other resources will be posted on the website.

Website:

http://eecs.mines.edu/Courses/csci448/

Assignments:
- There will be **individual** programming assignments throughout the semester. Due dates will be specified and are firm. Individual programming assignments **must** represent individual student work.
- All programs (homework assignments, programs, projects, labs) must be submitted in electronic form through BlackBoard.
- Every submission must include a README file which describes how to build and run the assignment. Any other information that is required to run the program should be included here as well.
- Your goal should be to make it easy for us to see how great you did. If we make a couple of good faith attempts but fail to get your program running, we will try once to contact you to help us. If we still cannot get your work to compile, it will receive a zero grade.

Late Policy:
- All assignments and projects are due at the date and time specified on the item handout.
- Items received less than 24 hours past the due date will receive a 25% grade reduction.
- Items received less than 48 hours past the due date will receive a 50% grade reduction.
- Items not turned in or received more than 48 hours past the due date will receive a grade of zero.

Projects:
- There will be one semester long project, to be completed in teams..
- In addition to the programming project, each team will give a brief in-class presentation showcasing their project.
- All code and files required for the projects must be submitted to one of the team member's dropbox. A text file must be placed in the other members' dropbox stating who their partners were and where the files are located.

Labs:
- Each week there will be a laboratory sessions that will cover the new techniques discussed in lecture each week. The lab assignments will be checked for completion in class and must be shown to one of the instructors to receive credit.

Quizzes:
- There will be in-class quizzes during the semester.
- Make-up quizzes will be allowed only in accordance with University policy. Make-up quizzes resulting from illness require notification (email is fine) the day of the quiz and a doctor's note when well. Make-up quizzes due to legitimate travel require advance notice and instructor's approval.

Piazza
- Be polite. This also applies to assignment clarifications (e.g. writing "This requirement makes no sense" may not be the best phrasing. Try something like: "I'm not clear what requirement X means. Should I do [x] or [y]?")
- A Piazza post is not a text message; use complete sentences and correct spelling, punctuation, and grammar.
- When asking a question, do not post large blocks of code. A single line of code, to clarify your question, may be appropriate. Before posting, ask yourself: would this be giving most of the answer to another student? Thinking about how to phrase the question may help you solve the problem.
- When answering a question, do not post the exact code from your homework solution. Possible exception would be something that takes one line and is primarily a syntax question. E.g., to a question like "How do I set the color of my rectangle" you might answer with something like "You need to set the color before drawing. If g is a Graphics object, you can do g.setColor(Color.CYAN);".
- Using pseudocode is an **excellent** way to answer questions.

Grading:
The final course grade will be computed from the following course percentage breakdown:
- 40% Programming Assignments
- 30% Final Project
- 20% Quizzes
- 10% In-class assignments and participation

Final grades will be determined using a *straight scale*. The straight scale assigns letter grades as follows:

- [93, 100]   --   A
- [90, 93)   --   A-
- [87, 90)   --   B+
- [83, 87)   --   B
- [80, 83)   --   B-
- [77, 80)   --   C+
- [73, 77)   --   C
- [70, 73)   --   C-
- [63, 70)   --   D
- [0, 63)   --   F

Participation:

- A portion of the student's grade will be comprised of in-class and online participation.
- After the first week of class, students are expected to participate by making regular forum posts, either asking a question or responding to an existing topic. From time to time, there may be specific discussion topics.
- Regular course attendance is mandatory. If attendance is low, the instructor reserves the right to administer pop quizzes for credit, to be determined.
- You are expected to read the assigned reading **prior** to the start of class.

Discrepancies:

- If you have any questions regarding how any assignment, project, or exam is graded and you think you deserve more points than you received, you must contact the instructor within **one week** of the day the item was returned to class. No claims, justifiable or not, will be considered after this deadline.
- Any assignment returned to the instructor is subject to total **one** re-grading.

Computing:

- This class will involve extensive use of Android Studio and Android devices (both virtual and physical). You may use any platform to develop your program. The computers in Marquex 026 are set up for your use. If desired, you may set up your own personal computer to mimic this build environment so you can work at your convenience outside of the lab. Tutorials will be provided to aid in your setup.
- If you have an Android device (phone, phablet, tablet) then you may use that for development. If you do not have access to a physical Android device, then you will be able to check out a device for the

duration of the semester. More details and specifics will be discussed in class.

Collaboration Policy for Programming Projects in CS Courses
The following policy exists for all CS courses in the EECS department. This policy is a minimum standard; your instructor may decide to augment this policy.

- If the project is an individual effort project, you are not allowed to give code you have developed to another student or use code provided by another student. If the project is a group project, you are only allowed to share code with your group members.
- You are encouraged to discuss programming projects with other students in the class, as long as the following rules are followed:
  - You view another student's code only for the purpose of offering/receiving debugging assistance. Students can only give advice on what problems to look for; they cannot debug your code for you. All changes to your code must be made by you.
  - Your discussion is subject to the empty hands policy, which means you leave the discussion without any record [electronic, mechanical, or otherwise] of the discussion.
- Any material from any outside source such as books, projects, and in particular, from the Web, should be properly referenced and should only be used if specifically allowed for the assignment.
- To prevent unintended sharing, any code stored in a hosted repository (e.g. on github) must be private. For group projects, your team members may, of course, be collaborators.
- If you are aware of students violating this policy, you are encouraged to inform the professor of the course. Violating this policy will be treated as an academic misconduct for all students involved. See the Student Handbook for details on academic dishonesty.

Academic Code of Honor:
- All students are expected to follow the University's Academic Code of Honor.
- A student or assigned team working on a program may discuss high-level ideas with other students or teams. However, at time of submission all work submitted **must be his/her/their own work**.
- Use of the Internet as a reference is allowed but directly copying code or other information is **cheating**. It is cheating to copy, allow another person to copy, all or part of an exam or a project, or to fake program output. It is also a violation of the Code of Honor to observe and then fail to report academic dishonesty. *You* are responsible for the security of your own work.

- We will provide, as part of the course, functional code examples for most of the topics covered. While you are encouraged to examine these examples, your submissions must represent a good-faith effort to complete the assignment. Merely copying and pasting code from the examples will result in a failing grade. Furthermore, relying too heavily on the given examples will fail to prepare you for the much more open-ended final project.