44-444/644: Mobile Computing - Android

Contact Information
Dr. Michael P. Rogers
Office: 2270 Colden Hall
Office Hours: MW 2:30-4:30 PM; Th 1PM-3PM; F 2-3 PM
Virtual Office Hours: Anytime you see me online
Phone: 309-825-6454 (Cell/Home/Text); 660-562-1551 (Work)
E-mail: mprogers@mac.com; Michael@nwmissouri.edu
Social Media: @mprogers (Twitter); mprogers@mac.com (Facebook)
Web Site: http://cs.mprogers.org/course/view.php?id=50

Description
44-444/44-644, Mobile Computing - Android provides an introduction to app development on the Android platform.

Prerequisites
Knowledge of object-oriented programming and an interest in exploring new technologies.

Instructional Objectives
After successfully completing this course, students will:
• be able to assimilate new technologies with greater confidence ("learn how to learn")
• be better problem solvers in general, and better coders in particular
• have experience using code versioning software
• have developed a larger project in a team environment
• have experiencing reading larger code bases (eh, maybe)
• have been exposed to the process of publishing an app to Google Play

Instructor Responsibilities
Your instructor will help you attain the aforementioned objectives by:
• explaining the material as clearly as possible
• guiding your reading
• maintaining numerous office hours, both during the day and (virtually) during the evening
• monitoring your progress via periodic assignments, quizzes and exams
• soliciting feedback

Resources
• developer.android.com
• The Busy Coder's Guide to Android Development (recommended, but not required)
• stackoverflow.com
Expectations
1. Attend all classes
2. **Ask questions during class.** If you are confused, chances are your neighbors are, too, and will appreciate your inquiry.
3. Explore and experiment! Try new things and learn from both your successes and failures. Computer Science is a **science.**
4. Keep up with the readings.
5. Do group assignments in groups; do individual assignments individually.

Assessments

<table>
<thead>
<tr>
<th>Category</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quizzes (3)*</td>
<td>10</td>
</tr>
<tr>
<td>Assignments (15-20)*</td>
<td>15</td>
</tr>
<tr>
<td>Exams (2)*</td>
<td>25</td>
</tr>
<tr>
<td>Final Group Project</td>
<td>20</td>
</tr>
<tr>
<td>Attendance</td>
<td>5</td>
</tr>
<tr>
<td>Final Exam</td>
<td>25</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
</tr>
</tbody>
</table>

A: [90.0,100]; B: [80.0,90.0); C: [70.0,80.0); D: [60.0,70.0); F: [0.0,60.0)

On Moodle, each grade interval will be divided into thirds: a + will be assigned to the top third, a - to the bottom third (e.g., if you earned a 92.0, you would get an A-). This is for your information only, and will **not** appear on your transcript: on CatPAWS only unqualified letter grades (A,B,C,D,F, I) will be reported.

Grades are weighted by category. Within each multi-item category (*), items are expressed as percentages and averaged. Say you had 2 quizzes, 15/20 & 9/10. Your average quiz score would be (75+90)/2 = 82.5%. Since the Quizzes weight is 10%, you would have earned 82.5 * 0.1 = 8.25 points (out of a total of 100), due to quizzes.

Empty categories are **ignored** when grades are calculated: weights during the semester only reflect categories for which you have one or more grades. For instance, suppose you had 82.5% in the Quizzes category, 90% in the Assignments category, and nothing else. Your weights would be 10/25 & 15/25 (not 10/100 & 15/100), so your overall grade would be 82.5 * 0.4 + 90.0 * 0.6 = 87.0 %. After the first exam, the weights would include the 25% Grades weight, and would thence become 10/50, 15/50, and 25/50.
Topics

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hello, Android!; onClick() and Viewgroups</td>
</tr>
<tr>
<td>2</td>
<td>Creating a UI Programmatically</td>
</tr>
<tr>
<td>3</td>
<td>Linking Activities; Activity Life Cycle</td>
</tr>
<tr>
<td>4</td>
<td>Customizing Adapters; Fragments</td>
</tr>
<tr>
<td>5</td>
<td>Interfragment Communication</td>
</tr>
<tr>
<td>6</td>
<td>The Action Bar</td>
</tr>
<tr>
<td>7</td>
<td>MVC Design Patterns; Dialogs, Alert Dialogs and More</td>
</tr>
<tr>
<td>8</td>
<td>JSON and the Android-nauts; Everything but the Kitchen Async</td>
</tr>
<tr>
<td>9</td>
<td>Data Persistence; SQLite</td>
</tr>
<tr>
<td>10</td>
<td>Disney, Look Out; Drawing</td>
</tr>
<tr>
<td>11</td>
<td>Motion Events</td>
</tr>
<tr>
<td>12</td>
<td>Media and Camera Magic</td>
</tr>
<tr>
<td>13</td>
<td>Google Maps</td>
</tr>
<tr>
<td>14</td>
<td>Google Maps and Genymotion</td>
</tr>
<tr>
<td>15</td>
<td>Final Project Presentations</td>
</tr>
</tbody>
</table>

Course Philosophy and Policies

1. Your instructor thrives on feedback and pertinent conversation during class: interrupt at any time if you have questions.
2. This class involves rapidly evolving technology. This presents a wonderful opportunity to practice what you will be called upon to do often during your professional career: master a new technology. How do you do so? This class will help you develop and practice techniques and strategies to answer that question. Most crucially, adopt a cheery attitude — whatever issue you are grappling with is not an annoying obstacle, but an intriguing mystery (of your own devise), just waiting for you to solve it!
3. Grades will be posted on cs.mprogers.org
   1. Grades will be posted in multiple categories One other category, called "Raw Grades", should be ignored [it represents the # of points deducted for exams and quizzes, and is used to calculate Exam and Quiz scores].
   2. Please set your time zone properly in the profile, otherwise your times will be wrong and you may miss an assignment.
   3. Please upload a selfie of yourself to help your instructor identify you.
4. Quizzes and exams must be taken at the specified times during class: there are no makeups.
5. Exams will be given during class on the 16th of February and 17th of March.
6. The final exam will be given on the day specified by the Registrar, Thursday, 28 April, 2:00-4:00 PM
7. There is an undergraduate version of this course: graduate students will be required to do one additional project and expect more rigorous grading.
8. CSIS has an official academic integrity policy, which we follow and you should therefore read.
9. No sharing of any materials (pens, pencils, erasers, paper, textbooks, answers, etc.) is permitted during any exams or quizzes. Doing so, or talking during exams or quizzes, will be interpreted as cheating. This will result in a 0 for the exam/quiz, and at the instructor's discretion the Dean will be notified.
10. Computer programming is not a spectator sport: you only benefit when you are the one getting the exercise. So, while you may discuss assignments in general terms with your fellow students, the final product must be yours and yours alone. If you do seek help (from the instructor, GA, or fellow student), you must acknowledge the fact in the comments (in the project's main class).
11. Late Assignments. Unless otherwise stated, all assignments must be uploaded to cs.mprogers.org by the date/time that they are due. Late submissions will incur a 10% penalty per 24 hour period, and will not be accepted once the assignment has been discussed in class. Emailed assignments will not be graded.
12. Coding guidelines as specified in class, must be followed. For example:
   1. All identifiers (variable names, method names, class names, etc.) should be descriptive. E.g., "temp" and "loop" and "isHotOrCold" are terribly unhelpful choices; "lastName" and "maximumSpeed" are great. Each inappropriate identifier will result in a 1% penalty.
   2. Methods should be of appropriate length (not more than approximately 30 lines: factor longer methods). Inappropriately long methods will result in a 2% penalty.
   3. Methods should perform a single task (e.g., computation or output, but not both)
   4. The name of the method should reflect its task. If a method does something it should have a verb-noun combination (e.g., rollDice(), or reduceSpeed()). If a method calculates something, a noun will usually suffice (e.g., cubeRoot()). Predicates should start with "is", as in isAwake(), isLateForClass(), etc.). Inappropriately named methods will result in a 1% penalty.
   5. Comments are required for most methods, classes, and interfaces (exceptions being the obvious lifecycle methods). Missing comments will result in a 1% penalty
   6. Comments should be appropriate and plentiful, but not pandering (e.g., don't say that a for-loop is a for-loop). Inconsistent indentation and egregious spacing will result in a 2% penalty.
   7. Use consistent indentation (Android Studio's default indentation style is fine)
   8. Your code should be efficient, elegant, and thoroughly tested.
   9. Use standard HCI conventions when designing GUIs
13. Class attendance will be taken: students who are not in their seat by the start of classes may be marked as absent.
14. Any concerns about any grade must be raised within 7 days of their posting.
15. Please silence all cellphones, etc., before class. Any time a student's cellphone rings during class, the recipient will lose 1% in the Class Attendance category. If the instructor's cellphone rings during class, all students will receive an extra 1% in the Class Attendance category (unless it comes from somebody in the class, in which case the two cancel each other out).
16. Students in this course who need disability accommodations/modifications should present a copy of their official Northwest accommodation letter from the LAP/S Committee to the instructor during private office hours as early in the term as possible. Additional information can be found online at www.nwmissouri.edu/swd.

17. **Fine print:** Your instructor reserves the right to amend this syllabus if events dictate, but will inform and confer with students before doing so.

---

**Final Project**

1. All students must complete a final project -- an app that is "Google Play" ready
2. The final project will be done in groups of 3 (possibly 1 group of a different size, depending upon enrollment)
3. The project will be managed in a git repository (bitbucket.org)
4. Students will give a presentation of their final project during the last week of class
5. Students should submit their apps to Google Play, because it will look terrific on a resume, boost their ego, and help with future employment.
6. A **rubric** will be used to grade the projects.

---

**Getting Help**

In addition to your instructor, we have a distinguished Mobile Computing - Android expert, Andrew Allen, who will be providing assistance in this course. You may ask him for help at any time (eh, except during exams).