

**San José State University**  
**Department of Computer Science**  
**Spring 2020**  
**CS 175 – Mobile Device Development**

**Course and Contact**

<b>Information Instructor:</b>	Ramin Moazeni, PhD
<b>Class Hours:</b>	TTh: 6:00PM - 7:15PM
<b>Office Hours:</b>	TTh: 5:30PM – 6:00PM, DH 282
<b>Email:</b>	<a href="mailto:Ramin.Moazeni@sjsu.edu">Ramin.Moazeni@sjsu.edu</a>
<b>Classroom:</b>	MH 233
<b>Prerequisites:</b>	CS 047, and knowledge of Java equivalent to that of CS 046A or CS 049J

**Course Description**

Design and implementation of applications running on a mobile platform such as smart phones and tablets. Programming languages and development tools for mobile SDKs. Writing code for Peripherals-GPS, accelerometer, touchscreen. Optimizing user interface for a small screen. Effective memory management on a constrained device. Embedded graphics. Persistent data storage.

This course introduces the use of SDKs released by Google to facilitate the development of applications for the Android Phone. Android Phones are Linux based and are programmed in Java. The Linux OS is the most powerful and easiest to manage of all operating systems. The Java programming language with its superior GUI development capabilities provides a good platform for Android development. Knowledge of SDKs is certainly an advantage when developing for the Android platform. This course will cover the current Google SDK, Android platform and Java programming features.

For the official catalog description, please visit [the online catalog](http://info.sjsu.edu/web-dbgen/catalog/courses/CS175.html) at <http://info.sjsu.edu/web-dbgen/catalog/courses/CS175.html>

**Learning Outcomes**

Upon successful completion of this course, students should be able to:

- Gain an understanding of the capabilities of several popular mobile operating systems and their development environments. The student will be able to compare and contrast environments, with respect to tools, runtime environments, and supported peripheral devices
- Develop mobile applications for one or more platforms. The student will write applications using the development tools and environment provided by the manufacturer, developing a fundamental understanding of the platform. The student will become familiar with the use of debugging tools and emulators in the development process
- Become familiar with view management and UI layout. The student should understand good principles for UI design in embedded applications and apply those principles to real-world examples.
- Gain exposure to peripheral-based development. Modern mobile operating systems allow access to a number of embedded peripherals, such as the accelerometer and GPS. The student will get experience interfacing with these devices by understanding and using manufacturer-supplied APIs
- [if time permits] Develop graphical applications. The student will understand how to integrate sophisticated graphical libraries, including OpenGL, into mobile applications.

## Text

- The most comprehensive and up-to-date information can be found on <http://developer.android.com/>
- Android Programming: The Big Nerd Ranch Guide 4th Edition, Bill Phillips, Brian Hardy <https://www.bignerdranch.com/books/android-programming-the-big-nerd-ranch-guide-4th/>
- The Busy Coder's Guide to Android Development (Mark L. Murphy) <https://commonsware.com/Android/>

## Course Mechanics

### Laptops

You will be required to bring a wireless-enabled laptop running Windows, Mac OSX, or a version of Linux to all classes and exams. It must be capable of installing and running the course software

An Android phone maybe needed for some of the assignments.

## Course Requirements

### Exams (50%)

One in-class mid-term (25%) and a final exam (25%). Exams cannot be made up, except for reasons of illness, as certified by a doctor, or documentable extreme emergency.

- The exams are based on lectures, quizzes, homework/lab assignments, and reading materials covered before the exam's date.
- Absolutely NO items may be shared during the exams, including books, notes, and calculators.
- Absolutely NO usage of cell phones during exams. Cell Phones must in off or silent mode and not within your reach.

### Programming Assignments (20%)

Schedule your time well to protect yourself against unexpected problems. I suggest starting early so you have time to ask questions if you need helps. Late work is accepted with a penalty of 10% per day. Late homework is not accepted one work past it's due date **All homework is due at 11:59PM** on the due date specified.

### Exercises (5%)

Exercises related to the topics covered and is assigned on a per topic basis. They are due within ONE week of the date they are assigned. Late work is accepted with a penalty of 10% per day. Late work is not accepted one work past it's due date

### Quizzes (5%)

Quizzes will be given throughout the course covering the required material discussed.

### Final Project (20%)

A mobile development programming group project of your choice related to the course. At the end of the semester you will present the project in the class. Detailed guidelines including milestones and their due dates for the project will be posted on Canvas. The schedule for final project presentations will be announced on Canvas as well.

Absolutely NO late submission for the Final project.

## Grading Policy

Your grade for the course is based on the exams, the homework, and quizzes. Grades are calculated by weighting the scores as defined below.

At least	Letter Grade
93	A
90	A-
87	B+
83	B
80	B-
77	C+
73	C
70	C-
67	D+
63	D
60	D-
below 60	F

## Individual Work

All homework, exercise and exams must be *your own individual work*. It is OK to have general discussions about the assignments or read other material for inspiration. You may *never* copy anything from anyone **without attribution**. This means if you find code on Stackoverflow or another web site, you need to give the URL where you found the code in a comment at the top of your class so that I can look at it if necessary.

You may copy from the textbook, the labs, or anything we do in class without attribution. For assignments and exams, you may not copy anything from any other student at all, and you may not collaborative produce results in pairs or teams. Your work must be entirely your own.

**It is never okay to share your code with other students.** If the other person submits your work, both students will receive a 0.

First incident of cheating will result in a 0 on that assignment or exam. Second incident will result in a F for the class.

## BSCS Program Outcomes supported by this course:

- (a) An ability to apply knowledge of computing and mathematics to solve problems
- (b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- (c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
- (i) An ability to use current techniques, skills, and tools necessary for computing practice
- (j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices
- (k) An ability to apply design and development principles in the construction of software systems of varying complexity

## **University Policies**

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' [Syllabus Information web page](http://www.sjsu.edu/gup/syllabusinfo/) at <http://www.sjsu.edu/gup/syllabusinfo/>

## CS 175, Mobile Device Development, Tentative Schedule

Week	Lesson	Class Date	Topics	Comments
1	1	23-Jan	Intro	
2	2	28-Jan	Android Basics, Android SDK	
	3	30-Jan	Android Lifecycle, First Android Program	
3	4	4-Feb	Android GUI, Layouts	
	5	6-Feb	Android GUI, Layouts (Contd)	
4	6	11-Feb	Resources	
	7	13-Feb	List Based Views	
5	8	18-Feb	Action Bar, Dialogs	
	9	20-Feb	Fragments	
6	9B	25-Feb	Intents	
	10	27-Feb	Intents (Contd)	
7	11	3-Mar	Sensor	
	12	5-Mar	Sensor (Contd)	
8	13	10-Mar	Android Testing, Junit	
	14	12-Mar	Rotation, Localization, Exam Review	
9		17-Mar	<b>Midterm Exam</b>	
	15	19-Mar	Data Storage	
10	16	24-Mar	SQLite, Databases	
	17	26-Mar	Location	
11	18	31-Mar	<b>No Class – Spring Recess</b>	
	19	2-Apr	<b>No Class – Spring Recess</b>	
12	20	7-Apr	Location and Maps	
	21	9-Apr	Content Providers	
13	22	14-Apr	Content Providers (Contd)	
	23	16-Apr	XML and JSON	
14	24	21-Apr	Multithreading	
	25	23-Apr	Multithreading (Contd)	
15	26	28-Apr	Android Services	
	27	30-Apr	Android Services (Contd)	
16	28	5-May	Project Demos	
	29	7-May	Project Demos, Final Review	
17	30	14-May	<b>Final Exam</b>	

