Mobile Development

AME 435/535

School of Arts, Media and Engineering
Herberger Institute for Design and the Arts
Arizona State University
Spring 2019

Monday/Wednesday, 12:15pm-1:30pm, Stauffer B123

Loren Olson

loren.olson@asu.edu

SPRING 2019 OFFICE HOURS

Location: Stauffer B260
Time: M/T/W/Th 2:00 - 3:00pm
During office hours you are welcome to walk in, no appointment is necessary.
If you need to see me, and you can't make my office hours, please send me an email to make an appointment at another time.

This syllabus is subject to change. I will let you know when it does, always refer to this online version.

OVERVIEW

The mobile phone has become a ubiquitous device worldwide. It is a product with a real market potential of every individual in the marketplace. With the release of the iPhone in 2007, the mobile phone transformed, from phone to pocket computing platform. As of February 2018, Apple has more than 1.3 billion iOS devices in active use. Google reported the number of active Android devices as more than 2 billion in May 2017. The mobile computing market is now the most dynamic, creative, profitable, personal and transforming market in computing.

This course explores native application development for iOS for use in the domain of Media Arts. The course is project based, and covers the native Xcode development environment, and the Swift programming language. Native applications are able to achieve the highest levels of performance, they leverage the unique technologies built

for a platform, and they are able to participate in the native user experience (look & feel) that all users of a popular platform expect. To understand native apps for iOS, the course will examine a variety frameworks such as UIKit, CoreMotion, CoreLocation, MapKit, AVFoundation, AVKit, and SpriteKit.

The class assumes that you already have experience programming. However, you do not need any previous experience with iOS development. If you are concerned about your programming ability, I suggest taking Mac Development for Media Arts before taking this class.

Course Objectives
Students are able to write native iOS applications of significant complexity.
There are a wide range of deep frameworks available for creating applications, we won't have time to explore them all. So, our goal will be to cover the important fundamentals, and look carefully at some exemplar frameworks. Students should then be ready to learn additional more specialized frameworks on their own initiative.
Upon successful completion of this course, students will be able to:

- Use the Xcode development environment to develop native apps for iOS.
- Use Foundation, UIKit and other bundled frameworks in the development of native apps.
- Learn how to understand and use new frameworks for development.
- Describe key differences and relationships between mobile development and desktop or web development.
- Describe important issues unique to iOS development.
- Describe key environmental issues of mobile development.

TEXTBOOK

This semester we will use some material drawn from two ebooks available from Apple for free.

Intro to App Development with Swift [https://itunes.apple.com/us/book/intro-to-app-development-with-swift/id1118575552?mt=11]

App Development with Swift [https://itunes.apple.com/us/book/app-development-with-swift/id1219117996?mt=11]

We will also use the free online book "The Swift Programming Language" as a reference. [https://docs.swift.org/swift-book/]

ASU Spring 2019 Important Dates

Session Dates and Deadlines	Session C: 15 weeks (Jan 7 – April 26, Finals week is April 29)
Classes Begin	January 7, 2019
Drop/Add Deadline	January 13, 2019
Tuition and Fees 100% Refund Deadline	January 20, 2019
Course Withdrawal Deadline	March 31, 2019
Complete Session Withdrawal Deadline	April 26, 2019
Spring Break	March 3 - 10, 2019
Final Exam	May 1st (Wednesday), 12:10-2:00

For additional university deadlines and important dates for the spring 2019 term, please visit: students.asu.edu/academic-calendar.

EVALUATION

60% Projects

Projects will involve writing iOS applications, projects are turned in using Critviz. I expect we will do approximately 8 projects. Always check Critviz for updated project information and due dates. The final project will include an in-class demo during the finals week meeting time on Wednesday May 1st at 12:10pm.

30% Homework

There will be short homework assignments in addition to the major projects. Homework may be reading/writing assignments or short coding assignments. You should always check Critviz for updated homework information and due dates.

10% Class Participation, Attendance

There will be a class roster to sign for attendance, it is your responsibility to sign the roster every class. If you don't sign the roster, you don't get attendance credit.

We will use an online system called Critviz to turn in assignments. Critviz can be found at http://critviz.com. I will show you how to use the system in class, and we will do an example project before a "real" assignment is due.
We score every assignment, exam, quiz or any activity on a scale of 0-100. Using that scale makes it very easy to always understand the performance result for any activity. It doesn't mean that everything has equal value! The assignment, exam, or quiz will be put into one of the above categories, and given a weight to determine how it contributes to the final course grade.

Not all exercises, assignments, or projects will be weighted equally, I may change the weights during the semester.

Grading Scale

A+	97-100%
A	93-96%
A-	90-92%
B+	87-89%
B	83-86%
B-	80-82%
C+	77-79%
C	70-76%
D	60-69%
E	0-59%

LATE ASSIGNMENT POLICY

The concepts, techniques and ideas you learn in this class will be cumulative. They build on each other as the semester progresses, and you will use all of these things together throughout the semester as you learn more about programming. Therefore, it is important for your success to do all assignments in a timely fashion, in the order that they are given. The important concepts from one assignment will become foundations for subsequent coursework. If you miss an assignment, or for some reason you struggle with a particular assignment and do poorly - we want to make sure you don't miss those concepts and therefore struggle in future assignments. For this reason, we will accept late assignment submissions or resubmissions with a late penalty. Work submitted the week after due date is marked down 15 points. Each further week late means an additional 10 point penalty. After 4 weeks, late submissions will no longer be accepted.

CLASSROOM CITIZENSHIP

Attend class, and pay attention. Participate in class discussion. Be curious. Be open to working hard and trying new things. Speak up. If you have a burning questions that you

think is too simple to ask, it probably means several other people have the same question.

Be courteous to your fellow classmates, teaching assistant and teacher. Help us to create a positive and constructive learning environment that encourages everyone. Any disruptive behavior will be dealt with according to ASU policy. Please see the Student Services Manual for more details.

Title IX
Title IX is a federal law that provides that no person be excluded on the basis of sex from participation in, be denied benefits of, or be subjected to discrimination under any education program or activity. Both Title IX and university policy make clear that sexual violence and harassment based on sex is prohibited. An individual who believes they have been subjected to sexual violence or harassed on the basis of sex can seek support, including counseling and academic support, from the university. If you or someone you know has been harassed on the basis of sex or sexually assaulted, you can find information and resources at http://sexualviolenceprevention.asu.edu/faqs/students

As an employee of ASU, I am a mandated reporter and obligated to report instances of reported or suspected incidences of sexual harassment.

ACADEMIC INTEGRITY

Is writing code like solving an equation in Calculus or is it more like writing an essay in English? If two students in English turn in essays that are nearly identical, it will be flagged as plagiarism. When two students have identical solutions in Calculus, not a second thought would be given - but everyone understands that copying answers from a neighbor's paper is cheating. In fact, I think that our projects should be more closely related to the English class essay, than the Calculus example. When you work on your project, please keep in mind the essay example - you must write your own unique essay.

What about code you found to help you? (Code at developer.apple.com, or stackoverflow, or github, etc) It can be ok to use other code in projects. It is critical that you acknowledge that code. You must cite where the code came from, and what code you have used. Also, I should add that you should never add code to a project that you don't understand. It is critical that you understand code that you add, it should not be a "black box." Always cite code in comments, where you use it. Clearly mark what code is involved, and include a URL, and explanatory text. Failure to cite code in an assignment will result in an automatic 0 grade.

How much code, and what code is ok to use is also context dependent. Again, think about an English essay. You would not turn in an essay that consists of two sentences you wrote, then simply quote another writer for two entire pages of content. That wouldn't be your own essay. In this class, you cannot copy and paste many lines of code, change a few variable names, then submit the result as your own work. That would be a coding example of plagiarism.

What about study groups? A problem has arisen in the past when students work together, then turn in identical code. You should not be turning in identical code, since it becomes impossible for us to tell what is innocent coincidence due to a study group versus blatant unethical copying. I think study groups are very helpful, and do not want to discourage that community. Experience so far shows you can avoid this problem with a simple rule of thumb - always type all of your own code. Never copy a file from a friend, or allow a friend to copy one of your files.

University policy regarding copyright
Students must refrain from uploading to any course shell, discussion board, or website used by the course instructor or other course forum, material that is not the student's original work, unless the students first comply with all applicable copyright laws; faculty members reserve the right to delete materials on the grounds of suspected copyright infringement.

Attendance and Participation

Students are expected to attend all classes. In the case of absence, please inform the instructor before the class if possible, and/or after the missed class. Classroom attendance and participation is 10% of the overall grade. Any student missing more than 2 classes without formal notes (Dr. Note etc) will fail the course.

Religious Accommodations for Students

Students who need to be absent from class due to the observance of a religious holiday or participate in required religious functions must notify the instructor in writing as far in advance of the holiday/obligation as possible. Students will need to identify the specific holiday or obligatory function to the faculty member. Students will not be penalized for missing class due to religious obligations/holiday observance. The student should contact the class instructor to make arrangements for making up tests/ assignments within a reasonable time.

Special Accommodations

To request academic accommodations due to a disability, please contact the ASU Disability Resource Center (http://www.asu.edu/studentaffairs/ed/drc/# ; Phone: (480) 965-1234; TDD: (480) 965-9000). This is a very important step as accommodations

may be difficult to make retroactively. If you have a letter from their office indicating that you have a disability which requires academic accommodations, in order to assure that you receive your accommodations in a timely manner, please present this documentation to me no later than the end of the first week of the semester so that your needs can be addressed effectively.

Stauffer Media Lab

The Stauffer Media Lab - Stauffer B135 - is available for all students in this class, weekdays 8am to 8pm. All the computers in the lab have Xcode installed. There may be weekend hours this semester, check the lab to find current hours.

REFERENCE

Apple Developer Web Site. If you want to get serious about writing software for Apple platforms, you should become familiar with this site.
https://developer.apple.com/

There are many other books about Xcode and Swift available, some of them are very good. There are enough that keeping a current list here isn't practical.

If you have an iPad, Apple has released an iPad app called Swift Playgrounds. See this (marketing) web page, and take a look - its free. The app includes links to educational material. There is a Learn to Code iTunes U course that uses the Playgrounds app.

OUTLINE

1. An introduction to Mobile Computing and iOS for developers.
2. Mobile computing?
3. iOS
4. AppStore
5. Project 0
6. Xcode projects
7. Model-View-Controller
8. Interface Builder
9. Connecting Code
10. Using the iOS Simulator
11. Project 1
12. UIView & view hierarchy
13. Intro to Auto Layout
14. Text Input

15. Delegation / Protocols
16. View Controllers
17. MapKit framework
18. Creating views and constraints programmatically
19. Project 2 - Camera Project
20. UITableView
21. UINavigationController
22. Using the Camera
23. UIImageView
24. UIImagePickerController
25. Application States
26. CoreImage
27. Project 3 - Game Project
28. Touch Events
29. Recognizing Gestures
30. SpriteKit
31. Using WebServices
32. Project 4 - SceneKit
33. Elements for a virtual 3d world
34. Camera
35. Geometry
36. Lighting