**Course Description:** In this project-based course, students will learn to develop iOS applications using the Swift programming language and the Apple toolset. Additionally, students will learn basic concepts about designing intuitive and usable interfaces.

**Course Learning Outcomes (From Apple’s Teaching App Development with Swift Course Overview):**
The primary learning outcome for this course is that students will be able to design and create iOS apps. Students will leverage Swift, the iOS SDK, and Apple developer tools. With iOS as the platform, students will learn object-oriented programming, design patterns, type systems, functional language features, user interface design, best practices in programming, and problem analysis.

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

1. Define key programming terms relevant to Swift and iOS programming.
2. Describe the process of creating iOS apps.
3. State the purpose of the Apple developer tools, such as Xcode, Instruments, debugger, analyzer, and iOS Simulator.
4. Distinguish well-written code from poorly written code.
5. Recognize patterns and idioms present in the Cocoa Touch API and other Apple frameworks.
6. Employ the Apple developer tools to create an iOS app.
8. Examine and subdivide app functionality into properly designed components.
9. Explain and summarize iOS API features including location, mapping, sensors, gestures, multimedia and user interface components.
10. Plan, prepare and build an original iOS app, from concept to working program.

**Prerequisites:** COP 3502 and 3503. Contact me if you have any questions.

**Text (Optional):** The Swift Programming Language

**Tentative List of Topics (From Apple project and lesson overviews):**

* Number of updates is subject to change
• Running and modifying an iOS app
• Gaining a comfort level with Xcode
• Apply Auto Layout constraints to create adaptive user interfaces
• Discovering how to connect user interface controls to controller code
• Understanding the tools and technologies used to create iOS apps
• Practicing the fundamentals of Swift syntax
• Understanding object-oriented programming with Swift
• Discovering Swift data types and collections
• Analyzing code quality, and discovering advanced Swift topics
• Practicing applying Xcode, Interface Builder, and MVC
• Applying UILabel and UIPickerView components, IBOutlets and IBActions
• Demonstrating Arrays, ranges and the map function
• Describing protocols and delegates
• Using NSUserDefaults and property list files for persistence
• Describing object-oriented inheritance and subclassing
• Discovering how to respond to touch events
• Discovering Core Graphics contexts and procedural drawing idioms
• Distinguishing the intents of UIViewController and UILabel objects
• Practicing establishing connections between controllers and views
• Describing how frameworks provide additional app functionality
• Defining URLs and the NSURL class
• Combining additional frameworks in an Xcode project configuration
• Discovering the fundamental features of the MKMapView API

Grading:
• 10% -------- Programming assignments
• 10% -------- Group Project Pitch
• 40% -------- Project Updates (4)*
• 20% -------- Final Project Submission
• 10% -------- Final Project Presentation
• 10% -------- Group Evaluations
**No Final Exam

Programming Assignments: At the end of a section, students will be given a programming and/or writing assignment that will be due the following week.

Group Project Pitch: Students will form groups of two (2) to three (3) students. Online students will work by themselves. Each group will propose a project on which the group will work throughout the semester. The pitch can be a written document or video presentation that explains the proposed project, each group member’s role and their projected progress for each of the four (4)* project updates.

* Number of updates is subject to change
Project Updates: Throughout the semester, students will submit four (4)* project updates. For each update, each group will present a written document or video presentation that compares their actual progress to their projected project. Groups will describe what they were able to accomplish, what they did not accomplish, problems they faced and, if necessary, how they will get back on track.

Final Project Submission: Each group will submit their completed project including all code and assets. Additionally, each group will submit a written document that compares what they proposed to what they actually accomplished. Groups will describe what they were able to accomplish, what they did not accomplish, problems they faced, and what they would do differently to improve their project.

Group Evaluations: Each group member will submit a written document that describes their team dynamics. This document should describe each group member’s contributions to the project. Your grade for this assignment will be based on how much you contributed to your group. I suggest you keep track of contributions as a group and individually. Before each project update, be sure to reconcile group and individual contribution lists. Online students will submit a document that describes extra features that they could have implemented if they were part of a group. Additionally, this document will include what they feel are the pros and cons of working as individuals.

Final Project Presentation: Each group will give a presentation (5 slide maximum) that describes their final project.

A 20% PENALTY will be assessed for any assignment submitted one day after the announced due date. An additional 10% penalty will be assessed for each additional day. Any project that does not compile will result in a 0. All extensions must be requested in writing at least three (3) days before the announced due date.

NOTE: This syllabus may change depending on the pacing of the class. Any changes will be beneficial to students and will have minimal impact to student grades, the grading categories and to the student workload.

Tentative Due Dates:
September 11 – Group Project Pitch
September 25 – Project Update 1
October 16 – Project Update 2
October 30 – Project Update 3
November 13 – Project Update 4
December 4 – Final Project and Group Evaluation Submission
December 4, 7, 9 - Project Presentations

Important Dates
• NO CLASS
  o September 7 (Labor Day)
  o October 13 & 15 (Grace Hopper Conference)

* Number of updates is subject to change
November 6 (Homecoming)
November 11 (Veteran’s Day)
November 25 & 27 (Thanksgiving)
• November 23 – Last day to withdraw with a W
• December 9 – Last day of classes

Score (Rounded to the nearest point) | Grade
--- | ---
100 - 92 | A
91 - 90 | A-
89 - 88 | B+
87 - 82 | B
81 - 80 | B-
79 - 78 | C+
77 - 72 | C
71 - 70 | C-
69 - 68 | D+
67 - 62 | D
61 - 60 | D-
59 - 0 | E

A C- will not be a qualifying grade for critical tracking courses. In order to graduate, students must have an overall GPA and an upper-division GPA of 2.0 or better (C or better). Note: a C- average is equivalent to a GPA of 1.67, and therefore, it does not satisfy this graduation requirement. For more information on grades and grading policies, please visit: [http://www.registrar.ufl.edu/catalog/policies/regulationgrades.html](http://www.registrar.ufl.edu/catalog/policies/regulationgrades.html)

**Accommodation for Students with Disabilities:** Students requesting classroom accommodation must first register with the Dean of Students Office. That office will provide the student with documentation that he/she must provide to the course instructor when requesting accommodation.

**Incompletes:** Incompletes will not be granted except under previous agreement of the professor. Students must inform the instructor of his/her intent to receive an incomplete and provide adequate documentation to support the request.

**Honor Code and Collaboration:** Students MUST document all reused code. Failure to document code found online or provided by other students WILL be viewed as plagiarism and an honor code violation.

Each student is expected to read, know, and uphold the UF Honor Code and Honesty Policy. Information about the honor code can be found here [http://www.dso.ufl.edu/SCCR/honorcodes/honorcode.php](http://www.dso.ufl.edu/SCCR/honorcodes/honorcode.php)

ANY and ALL suspected violations WILL be reported to the Dean of Students.

* Number of updates is subject to change