Summer 2016 Summer Programs for High School Students iPhone and iPad App Programming From the Ground Up

Course Description: In this course students learn to program in the Objective-C programming language and write native apps for Apple's iOS platform, which includes the iPhone and iPad.

The iOS platform offers many opportunities for students to explore such as the multi-touch user interface, GPS locations services, and social media. Students learn to use Apple's Xcode development tools to prototype their app's user interface and then write code to make their apps work. They write apps that can post a message to social media, work with the GPS and maps, and draw graphics.

By the end of the course, students will have created several apps which they will be able to demonstrate from their iPhone, iPod Touch, or iPad.

Participants are expected to bring an Apple laptop and either an iPhone or iPad for this class.
Course Prerequisites: One year of previous programming experience in C, C++, or Java or by consent of the instructor
Course Webpage: «http://gamble.ecs.fullerton.edu/teaching/summer16/scehr-ios/»
Course Mailing List: «None»
Course Calendar: «None»
Course Meeting Information:

Room:	MUD 327 (morning), FLS 601B (afternoon)
Time:	Mon.–Fri. 10:10–12:00 Mon.–Fri. 14:10–16:00
Midterm Exam:	June 27, 2016
Final Exam:	None

Important Dates:

June 27	First Day of Class
July 4	Holiday–Independence Day
July 15	Last Day of Class

Instructor: Michael Shafae

Phone:	310-526-3842
Email:	«michael@shafae.com»
Office:	N/A
Office Hours:	By appointment

Teaching Assistant: *None*

Textbooks:
This is not an exhaustive list of texts that may be used during this course. Additional texts may be recommended or required as the course progresses. All the texts listed below are published by Apple at no cost.

Required:

Object-Oriented Programming with Objective-C, Apple, inc., Apple, inc.
ISBN: `http://bit.ly/19PWVKz`

Programming with Objective-C, Apple, inc., Apple, inc. ISBN: `http://bit.ly/154MpGB`

Start Developing iOS Apps Today, Apple, inc., Apple, inc. ISBN: `http://bit.ly/19PXunC`

iOS App Programming Guide, Apple, inc., Apple, inc. ISBN: `http://bit.ly/19PXzYw`

Apple's developer website has many online texts which a student should read before committing to purchasing a popular text or reference book.

Apple's iTunes U has a wealth of video podcasts focused on iOS development. Students are encouraged to subscribe to these podcasts to supplement this course's lectures and lab exercises.

Additionally, many popular technical books may be read online through the campus's subscription to Safari Books Online. The Safari Books Online service can be accessed directly from any computer on the campus network, «`http://www.columbia.edu/cgi-bin/cul/resolve?clio4136562`».

Learning Goals: In order to achieve a satisfactory grade in this course the student must display competency in the following course objectives as measured by assignments, exams, quizzes, and participation.

1. Identify the unique constraints placed on mobile applications
2. Model, implement, & debug an application for a mobile device
3. Understand and apply the Model-View-Controller design pattern
4. Gain fluency in the mobile device vendor's preferred programming language, runtime, and API documentation.
5. Improve proficiency with development tools such as text editors, distributed version control, and code navigation.

Course Outline: This weekly outline will be adjusted accordingly during the semester and is given here as a guideline.

1. Introductions, Objective-C, Objective-C, Foundation Classes, Memory Management, MVC, Protocols, Categories, & Xcode
2. Labels, Text Fields, & Views, Actions, Outlets, Quartz2D, & Image View, Application State Machines
3. ScrollView & Table View, Core Location & Map Kit, Game Programming & Engines

Grading: Plus and minus grading will be used when determining final grades. Approximately the upper third of each grade bracket will be assigned a plus and the lower third a minus.

Final grades are computed by first finding the average score in each category described in the table below on the right. All scores are normalized to a scale of 0 to 100 before being averaged. The average score for each category is then used to compute the weighted average according to the weights in the table below on the right.

Grade	% of Total Points
A	90–100%
B	80–89%
C	70–79%
D	60–69%
F	Below 59%

Category	% of Final Grade
Projects	50%
Participation	50%

Assignments: Programming and written assignments will be handed out in class and typically posted on the course website in advance of their due dates. Each assignment description will include the assignment's grading rubric. Reading assignments are outlined in the syllabus and it is the responsibility of the student to stay up to date with the reading. Should there be a final project, it will be due on the last week of class, which will include an in class presentation and a short essay.

Written assignments must be typeset and presented in a professional manner. Presentation, spelling and grammar can be worth up to 30% of an assignment's grade.

All programming assignments must be written in the Objective-C programming language, unless specified otherwise. Coding style must conform to professional norms. At a minimum, code must be commented, have descriptive names for identifiers, and contain a header with pertinent information. Students are encouraged to use inline documentation systems such as Doxygen. A plain text `README.txt` must be included with each assignment submission summarizing and documenting the work submitted.

This course does not require you to use any particular computing platform however, your work must adhere to *open* standards. At the start of every semester, the instructor will detail the platform and tools used to grade student assignments. It is the student's responsibility to ensure that the assignments execute to his or her satisfaction on the instructor's grading platform.

Exceptions are made on a case by case basis given enough time and evidence to weigh the merits of the application.

The development tools are *only* available on the Apple computers. Participants are expected to bring an Apple laptop and either an iPhone or iPad for this class. Microsoft Windows, Unix, Unix-like, and other computer platforms cannot be used to complete the homework assignments. Virtual machine software can host OS X however it is not recommended.

The course uses iOS 9, OS X 10.[10|11] (Yosemite|El Capitan), and Xcode >7. (The earliest release of OS X 10.10 that supports Xcode 7 is 10.10.4.)

In order to sign up for Apple's iOS Developer program, you must fill out the online form at «`http://bit.ly/1Ui1LEg`» by the third day of instruction. After filling out the form, each student will receive an email invitation to Apple's iOS Developer University program. You will need to complete the registration process within 30 days of receiving the invitation code in order to be *blessed* by Apple. This will give you the privilege of loading your own software on a stock Apple i(Contraption). At the end of the summer you will be removed from the iOS Developer University program.

You can *sideload* your application without jailbreaking your smartphone or tablet with Xcode 7 even if you are not part of the developer program.

Beyond the assigned readings, there will be approximately three projects in total. Each project will progressively increase in complexity. The projects planned for this summer are:

- BMI & BMR calculator
- Currency converter
- Arthmetic calculator

Participation: in the context of this course participation is defined as the following:

- Arriving to class prepared and on time. Taking notes.
- Actively listening to the lecture and asking questions when appropriate.
- Annotating code listings and handouts.
- Bringing any required materials to class.
- When needed/desired, seeking assistance to complete assignments.
- Barring an emergency, not leaving the class session early unless the instructor consents.

Quizzes & Exams: There may be a small number of quizzes that will be administered in class. The quizzes will cover material that has been discussed over the past 10 calendar days prior to the quiz, including any assigned reading. Quizzes may or may not be announced. A quiz score will be counted as an assignment.

The dates for the exams are listed on the syllabus. Unless stated otherwise, all quizzes and exams are closed-book. Students are not permitted to use anything that uses electricity during an exam, such as electronic calculators, cellular phones, and portable computers.

Quizzes and exams cannot be taken after the test date.

Academic Dishonesty: Students are encouraged to assist one another and discuss the course materials with your peers. It is your responsibility to be aware of and follow the spirit of Columbia University's academic honesty policy which can be found at «`https://www.college.columbia.edu/academics/academicintegrity`». Academic dishonesty will not be tolerated.

By submitting work for evaluation, you acknowledge that you have adhered to the spirit of the university's academic honesty policy and that your submission is an original work by you unless otherwise directed to work in groups. Failure to follow the spirit of the academic honesty policy will result in a severely negative evaluation of the work in question and may result in involving the program's administration to seek a disciplinary remedy.

Course Rules:

- All students must have an email address, be subscribed to the course mailing list and read email regularly.
- Attendance at all regularly scheduled lecture and discussion section is mandatory.
- Do not eat during lecture.
- If it makes noise, silence it.
- Portable computer use is *not* allowed in lecture except for taking notes.
- Quizzes and Exams cannot be taken after the test date.
- The student is responsible to be aware of any course announcements including changes to due dates and requirements.
- Never submit an assignment via email.
- Never send attachments to the instructor unless directed to do so.
- Work submitted must be appropriately prepared and typeset.
- A student's final project may not be submitted late. Any assignment submitted late will have 10% deducted from the total points possible for every day that it is late. For example:
  - 1 day late → max. score of ${}^{90}/_{100}$
  - 2 days late → max. score of ${}^{81}/_{100}$
  - 3 days late → max. score of ${}^{74}/_{100}$
- Third party code may not be used in student work without prior instructor consent. Failure to gain and document instructor consent will be construed as willful academic dishonesty.
- When third party code is incorporated into student work, failure to wholly document the code's origin, copyright and license will be construed as willful academic dishonesty.
- Due to an act of nature, medical emergency, an act of violence by a non-state actor, severe civil unreset, or hostilities between nations, students have 10 calendar days to petition the instructor to retake any exam/quiz or submit an assignment without late penalty.
- Exceptions will be made on a case by case basis, provided there is time to evaluate the merits of such an application.

Student Checklist: The checklist below summarizes the actions that a student must take in the first week of the course.

□ Submit your UUID for your iOS computers «`http://bit.ly/1Ui1LEg`»

□ Download the PDF books recommended for this course and start reading.

□ Set up your development environment: Xcode with command line tools.

□ Bookmark the course web page, «`http://gamble.ecs.fullerton.edu/teaching/summer16/scehr-ios/`».

□ Bookmark the course GitHub page, «`https://github.com/mshafae/SCEHR-iOS`».

□ Make sure you have `git` installed.

$Id: iPhone_and_iPad_App_Programming_From_the_Ground_Up.tex 6143 2016-06-27 02:42:33Z mshafae $