

4002-542/4002-890

**Native Application Development for Mobile Phones I**  
**Fall 2011 Course Syllabus**

The information presented in this syllabus is subject to expansion, change, or adjustment during the quarter.

**Instructor:**

Name: **Tony Jefferson**  
Office: Bldg. 70, Room 2671  
tony@mail.rit.edu

**Office Hours:**

W 2-6PM

**Course Text and Materials****Required**

**Title: iOS Programming – The Big Nerd Ranch Guide – 2<sup>nd</sup> Edition**

Author: Joe Conway and Aaron Hillegass  
ISBN: 0321773772

**Additional Resources**

Other reading such as handouts, web readings, and videos will be required.

**Important RIT Deadlines**

Last day of add/drop is **Sunday of week 1**

Last day to withdraw from a course with a W grade is the end of the 8th week of the quarter. This may now be done online and does not require the professor's signature.

**NOTE:** IT department policy states that a student has one quarter to **challenge** any **grade**. After that, grades cannot be challenged.

## Course Description

This course is an introduction to creating native applications for mobile phones. Topics covered include user interaction patterns, connectivity, interface design, software design patterns, and application architectures within the context of mobile computing. Programming is required.

## Prerequisite Courses

- 3rd year standing
- A two or three course programming sequence, plus at least two additional courses that required programming.

## Course Goals and Objectives

### General Course Goal

The purpose of this course is to provide students with the experience of creating native applications for mobile phones. Topics covered include user interaction patterns, connectivity, interface design, software design patterns, and application architectures within the context of mobile computing.

### Topics

- 1.0 **Topics** (outline):
  - 1.1 Mobile Hardware
    - 1.1.1 Capabilities and Limitations
    - 1.1.2 Touch interfaces
    - 1.1.3 Location awareness
  - 1.2 Mobile User Interaction Patterns
    - 1.2.1 Persistent Connectivity
    - 1.2.2 Single User
    - 1.2.3 Short/Frequent Use
  - 1.3 Programming
    - 1.3.1 Available SDKs and software frameworks
    - 1.3.2 Software patterns and architectures
    - 1.3.3 Native Language of Platform
  - 1.4 Interface Conventions
    - 1.4.1 Screen layout
    - 1.4.2 Data display
    - 1.4.3 Navigation systems
    - 1.4.4 Interface elements
    - 1.4.5 Animation
  - 1.5 Media
    - 1.5.1 Digital Images
    - 1.5.2 Video
    - 1.5.3 Sound
    - 1.5.4 Procedural drawing

- 1.6 Data Acquisition
  - 1.6.1 Consuming web services
  - 1.6.2 Working with data formats: XML, JSON, Text
  - 1.6.3 Posting data to remote data stores
  - 1.6.4 System Architectures
- 1.7 Data Storage
  - 1.7.1 XML
  - 1.7.2 SQLite
  - 1.7.3 Object-relational Database
- 1.8 Additional Device Sensors and Capabilities
  - 1.8.1 Location Awareness
  - 1.8.2 Accelerometer and Orientation
  - 1.8.3 Other sensors

## Course Organization

### **Attendance**

Required. The following grading will apply to the attendance grade for missed lectures: 1 absence = 85%, 2 absences = 70%, 3 absences = 40%, 4+ absences = 0%. Arriving to class late counts as ½ absence. Inappropriate classroom behavior during lectures such as texting, talking, surfing social media sites and so on counts as ½ an absence. Additional absences beyond 4 will result in a penalty of -2% to the student's final average per absence.

### **Weekly Homework**

Each week in class, students will be given small homework assignments to complete outside of class. At the beginning of each class, we will review the assignments. Assignment completion and participation in assignment discussions will be part of the final grade. Some assignments will be graded beyond just a "completion" grade.

### **Mini-projects**

Two assignments:

- 1) An application utilizing Location and Map APIs
- 2) An improved version of the above project utilizing UITableViews, UITabBarController, UINavigationController, and custom detail views.

### **Final Written Exam**

These will consist of both short answers, and longer essay style questions. They will be heavily based on the lectures and reading.

### **Final Project**

A group of students will create an iPhone application as their final project. The project will be presented to the class during finals week.

## Grading

The grading scale used along with the grading criteria is as follows:

Component	Weight
Final Exam	20
Final Project	35
Weekly HW	15
Mini projects	20
Attendance	10

Range	Grade
$\geq 90.0\%$	A
$\geq 80.0\% \ \& \ < 90\%$	B
$\geq 70.0\% \ \& \ < 80.0\%$	C
$\geq 65.0\% \ \& \ < 70.0\%$	D
$< 65.0\%$	F

NOTE: The assignments in this course are designed based upon an opened-ended lab model. This means that merely completing the requirements for an assignment is only sufficient for a grade of “B”. To receive an “A” grade, you must go beyond the basic requirements by demonstrating creativity, motivation, and initiative. In other words, the grade of “A” is truly reserved for excellent work, not just work that satisfies the basic criteria.

**Cheating:** Academic dishonesty (cheating, collusion, or duplicate submission) is misrepresenting someone else's work as your own. Academic dishonesty is a serious matter, and can result in an automatic F for the course. Please review the IT department's policy, located online at <http://www.it.rit.edu/policies/dishonesty.html>.

If, during the quarter, you ever have any questions about what does or does not constitute academic dishonesty, please come and talk to me.

## Finally...

Any or all of the previous information is subject to change or adjustment during the quarter.