Course Alignment Matrix

Originating College: Collin College

Course Title: Introduction to iOS Mobile Development: Overview, iTunes, Xcode, Objective-C, and Cocoa

Course Number: ITSE1370AA

Recommended Text/Materials: Mac with OS (Lion or Mountain Lion) capable of running XCode

Competencies (these can be from the institution, national standards, industry standards, etc.)	Lesson(s)/Modules presented & assessed	Associated Open Resources & Development Ideas (Optional - this column is helpful when the document is used as a course design worksheet for new development)
1.0 Create basic template-based iOS applications using current iOS SDK.	3, 4, 5	
2.0 Create user interfaces for the iPhone/iPod Touch and iPad that follow Apple Human Interface Guidelines.	3, 4, 5	
4.0 Describe development cycle and approval process for iOS applications.	1	
5.0 Evaluate iOS applications from the iTunes App Store.	2	
6.0 Experience in foundation of developing appropriately using Objective-C and Cocoa Touch	4, 5	
7.0 Knowledge of IPhone and iPad development environment	1	
8.0 Experience with object-oriented design, design patterns,	3, 5	

Authoring Organization: Collin College
Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Xcode		
9.0 Prioritize and handle multiple tasks	1, 3, 4, 5	
10.0 Excellent analytical and creative problem-solving skills	1, 3, 4, 5	
11.0 Work independently on assigned tasks	1, 3, 4, 5	
12.0 Adept at conducting research into software-related issues and products	2	

Authoring Organization: Collin College
Written by: Original Author, Elizabeth Pannell; Edited Version, Susan Sands

Lesson/ Module	Topics	Lesson Objectives	Assessment & Points	Competencies/Learning Outcomes
1	Introduction to iOS Development and Xcode	1. Identify current mobile device iOS version. 2. Identify the features introduced in iOS versions. 3. Identify the different iOS devices including major features and hardware of each device. 4. Identify the operating system versions that work with different iPhone, iPod Touch, and iPad generations. 5. Identify limitations when developing iOS apps. 6. Identify advantages when developing iOS applications.	Practice - Howdy World (10 pts) Quiz 1 (10 pts)	4..0, 7.0, 9.0, 10.0, 11.0
2	iTunes App Store	1. List the iTunes App Categories and recognize an example of the type of app that would fit in that category. 2. Identify the app category with the most submissions overall and the app category with the most submissions per month. 3. Evaluate an app based on user reviews, functionality, design, popularity and utility. 4. Discuss the differences between a free app, a paid app, and why some free apps are actually in the top grossing app category. 5. Identify different web resources for reviewing iOS apps and/or for iOS and Apple news. 6. Identify the different sections in an app's page when selected in iTunes App Store and the purpose of each.	Exercise – App Research (10 pts) Quiz 2 (10 pts)	5.0, 12.0
3	Xcode	1. List the steps in creating a new project. 2. Create a new project using a single-view template. 3. Identify the areas of the Xcode window including navigator pane, utility pane, editor pane, and debug area. 4. Utilize Xcode to create iOS application using a template.	Practice - Howdy World 2 (10 pts) Practice - Show Me (10	1.0, 2.0, 8.0, 9.0, 10.0, 11.0

		5. Demonstrate building an app and use of the simulator. 6. Use Interface Builder to create the user interface. 7. Use Code Editor to declare instance variables and methods in the .h file and implement in the .m file. 8. Differentiate content that goes in the .h file, the .m file and the storyboard file. 9. Explain the use of Standard editor view, Assistant editor view and Organizer window. 10. Identify the different navigation views including Navigator view, project navigator, symbol navigator, search navigator, issues navigator, debug navigator and breakpoint navigator, and log navigator. 11. Identify the different inspectors including attribute inspector, identity inspector and connections inspector. 12. Differentiate between the folders and files seen in Xcode versus the project's folders and files on the drive. 13. Identify the purpose of outlets (IBOutlet) and actions (IBAction). 14. Create an app that uses outlets and actions.	pts) Quiz 3 (10 pts)	
4	Objective C	1. Discriminate among a superclass, a class and a subclass. 2. Identify in a message, the instance variable, the method and whether there are arguments included. 3. Explain why comments are important in a program. 4. Identify the two Boolean values used in Objective-C. 5. Identify the differences between a class method and an instance method. 6. Identify the use of pointers in iOS applications. 7. Identify primitive data types. 8. Differentiate content and code that goes in the header (.h) file, the implementation (.m) file and the xib (.xib) (or storyboard) file. 9. Identify the use of id as a data type.	Practice – Caption (10 pts) Quiz 4 (11 pts)	1.0, 2.0, 6.0, 9.0, 10.0, 11.0

		10. Identify the purpose of the viewDidLoad method.		
5	Cocoa Touch and Design Patterns	1. Identity the three frameworks included in almost every iOS application. 2. Identify the different iOS architecture layers. 3. Identify the characteristics of the Model-View-Controller design pattern. 4. Identify the characteristics of the target-action design pattern. 5. Demonstrate using outlets and actions in an app. 6. Demonstrate using textfields for user input. 7. Demonstrate using label for user output. 8. Demonstrate changing the class of a UI object. 9. Demonstrate the use of FirstResponder in an app. 10. Demonstrate setting up a keyboard for number input. 11. Demonstrate changing the class of a view (UIView) to a subclass (UIControl).	Practice - Convert Inch (10 pts) Quiz 5 (10 pts)	1.0, 2.0, 6.0, 8.0, 9.0, 10.0, 11.