

CSCI 0079 - MOBILE DEVICE PROGRAMMING

Catalog Description

Formerly known as CSCI 309

Prerequisite: Completion of CSCI 12 with grade of "C" or better

Hours: 72 (54 lecture, 18 laboratory)

Description: Introduction to creating applications for mobile devices including Apple iPhone, iPad, and Google Android. Topics include touch interfaces, GUI elements, sensor input, simple animation and game play, network communication, and database access. (CSU)

Course Student Learning Outcomes

- CSLO #1: Write code to draw a custom view for a mobile device application using lines, rectangles, ellipses, and/or bitmap image files.
- CSLO #2: Implement a mobile device application that contains multiple views controlled by user-interface elements.
- CSLO #3: Identify the tools, steps, and components used to create a mobile device application containing common features such as standard user-interface controls, tables, and custom graphical views.

Effective Term

Fall 2020

Course Type

Credit - Degree-applicable

Contact Hours

72

Outside of Class Hours

90

Total Student Learning Hours

162

Course Objectives

Lecture Objectives:

1. List different sensor types found on modern mobile devices (e.g. GPS, accelerometer, gyro).
2. Identify standard techniques to display and manipulate data on a mobile device.
3. Design a GUI for a simple application using buttons and text fields.
4. Compare and evaluate different methods of storing persistent data (text files, JSON, SQLite, etc.).
5. Evaluate and choose from different strategies to implement animation.

Laboratory Objectives:

1. Use development tools to build a sample application and install it on a mobile device for testing.
2. Interpret the function of code samples in a text programming language (such as Lua, Javascript, C#, etc.).
3. Use vector graphics to display text, rectangles, and circles on the screen.

4. Use standard User-Interface controls to display and edit data.
5. Test a mobile application to discover flaws and solve them.
6. Experiment with and analyze the effect of device orientation on mobile applications.
7. Develop a simple game or utility application with at least two different screen views.
8. Design a simulation or animation using physics (gravity, collisions, etc.).

General Education Information

- Approved College Associate Degree GE Applicability
- CSU GE Applicability (Recommended-requires CSU approval)
- Cal-GETC Applicability (Recommended - Requires External Approval)
- IGETC Applicability (Recommended-requires CSU/UC approval)

Articulation Information

- CSU Transferable

Methods of Evaluation

- Objective Examinations
 - Example: Which of the following sensors will allow your app to detect whether the device is being held in portrait (vertical) or landscape (horizontal) orientation? a. GPS b. Accelerometer (correct answer) c. Gyroscope d. Ambient light sensor
- Problem Solving Examinations
 - Example: Interpret the function of code samples in a text programming language (such as Lua, Javascript, C#, etc.). Rubric Grading.
- Projects
 - Example: Write a program which displays a picture. The picture should fill the entire mobile device screen. Your picture should involve at least two types of shapes (lines, rectangles, ovals, etc.) and at least two colors. You should use for-loops to draw multiple copies of the shapes, not just a lot of separate draw statements. For instance, you might draw 6 blue squares running down a diagonal and 5 horizontal green lines across the image (using two for-loops). Completing the bare minimum program will earn you a B on this assignment. To get a higher grade, you should do something extra to show off. What you do is up to you, but it should demonstrate that you have thought more about your program. For this assignment, possibilities might include changing the sizes of the shapes across the screen, drawing some fancier pattern, alternating colors, etc. Rubric Grading.

Repeatable

No

Methods of Instruction

- Laboratory
- Lecture/Discussion
- Distance Learning

Lab:

1. Following an instructor discussion on the use of the integrated development environment (IDE) using the projection equipment in the lab, students follow each step on their computer resulting in a simple mobile application. Explain the significance of each step and

the available options. Distribute a quick start handout that will assist students in creating their own applications. (Laboratory Objective 1)

- OER Link:
- OER:

Lecture:

1. Following an instructor lecture on buttons and labels, students will create their own applications incorporating buttons and labels. Demonstrate how to write code to handle button taps. Explain the interaction between the button taps and the messages displayed in the labels. Finish by having each student demonstrate the application that they created using the projection equipment in the lab. (Lecture Objective 3)

Distance Learning

1. Following a video lecture, demonstrate how to write code to handle button taps. Explain the interaction between the button taps and the messages displayed in the labels. Have students create their own applications incorporating buttons and labels. Finish by asking each student upload the application that they created. (Lecture Objective 3)

Other materials and-or supplies required of students that contribute to the cost of the course.

Typical Out of Class Assignments

Reading Assignments

1. Read the chapter on Data Persistence. Analyze the different data saving strategies and choose one to study in detail. Develop a strategy for saving and restoring key values and be prepared to discuss in class.
2. Read the chapter on Table Views. Learn how to display a table of names and how to bring up a new view when a name in the table is tapped and be prepared to discuss in class.

Writing, Problem Solving or Performance

1. Using the IDE, create a small app to display the text "Hello World" on its screen. Give the application an icon to make it look more like a real app. Upload a zip file of your source code to the instructor.
2. Using the IDE, write a slightly more complex application, one with two buttons as well as a label. When the user taps either of the buttons, the label's text changes to indicate which button was pressed. Upload a zip file of the source code to the instructor.

Other (Term projects, research papers, portfolios, etc.)

Required Materials

- Beginning Mobile App Development with Corona
 - Author: Brian G. Burton
 - Publisher: Burtons Media Group
 - Publication Date: 2014
 - Text Edition: 1st
 - Classic Textbook?: No
 - OER Link:
 - OER:
- Learn Corona SDK Game Development
 - Author: Frank Zammetti
 - Publisher: Apress
 - Publication Date: 2013
 - Text Edition: 1st
 - Classic Textbook?: No