Las Positas CurricUNET

LAS POSITAS COLLEGE

Las Positas College
3000 Campus Hill Drive
Livermore, CA 94551-7650
(925) 424-1000
(925) 443-0742 (Fax)

Course Outline for CS XX

MOBILE COMPUTING FUNDAMENTALS

Effective: Fall 2017

I. CATALOG DESCRIPTION:
CS XX — MOBILE COMPUTING FUNDAMENTALS — 3.00 units

This programming course is intended for those students who already have completed an introductory programming course. It presents an overview concepts and skills for common mobile programming platforms and helps students develop applications for mobile devices.

2.00 Units Lecture 1.00 Units Lab

Strongly Recommended
CS 1 - Computing Fundamentals I

Grading Methods:
Letter or P/NP

Discipline:

	MIN
Lecture Hours:	36.00
Lab Hours:	54.00
Total Hours:	90.00

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: 1

III. PREREQUISITE AND/OR ADVISORY SKILLS:

Before entering this course, it is strongly recommended that the student should be able to:

A. CS1
   1. Modify and expand short programs that use standard conditional and iterative control structures and functions.
   2. Choose appropriate conditional and iteration constructs for a given programming task.
   3. Apply the techniques of structured (functional) decomposition to break a program into smaller pieces.
   4. Analyze and explain the behavior of simple programs.
   5. Describe, interpret and apply the mechanics of parameter passing.
   6. Discuss and apply the concept of algorithms in problem-solving processes.
   7. Judge the correctness and quality of algorithms, identifying necessary properties of good algorithms.
   8. Describe and apply effective debugging strategies.
   9. Design, implement, test, and debug programs using basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.

IV. MEASURABLE OBJECTIVES:

Upon completion of this course, the student should be able to:

A. Explain and apply design patterns used in mobile application development.
B. Design, create and debug mobile applications using a standard development environment and programming language features for mobile development.
C. Design and create effective user interfaces using a wide range of typical controls and visual elements.
D. Interpret and apply data types and fundamental control structures within a programming language used for mobile application development.
E. Interpret and apply classes and objects within a programming language used for mobile application development.
F. Interpret and apply collections within a programming language used for mobile application development.
G. Apply techniques for timing and asynchronous events within a mobile application.
H. Apply techniques and structures for persistent data storage and retrieval within a mobile application.

V. CONTENT:

A. Overview of Mobile Applications
   1. Mobile Application Development Environments
   2. Acquiring, Installing, Configuring a Development Environment
   3. Fundamental Components of Mobile Application Frameworks
B. Design Patterns in Mobile Programming
   1. Builder

        2. Singleton
        3. Adapter
        4. Observer
        5. Delegation
        6. Model-View-Controller (MVC)
    - C. Programming Language Concepts and Features
        1. Data Types
        2. Collections
        3. Functions and Closures
        4. Error/Exception Handling
        5. Classes and Objects
        6. Inheritance and Polymorphism
    - D. Application Essentials
        1. Structure of an Application
        2. Application Lifecycle
        3. Application Resources
        4. Application Content and Relationships
    - E. Building User Interfaces (UIs)
        1. User Interface Design Concepts
        2. Controls
            - a. Text/Labels
            - b. Buttons
            - c. Lists and/or Pickers and/or Menus
            - d. Grids and/or Tables
            - e. Date/Time controls
            - f. Maps
        3. Dialogues and Alerts
        4. Managing Layout
    - F. Other Application Features
        1. Audio and Video
        2. Timed and/or Asychronous Events
    - G. Security
        1. Overview of Security Concepts
        2. Application Signing
        3. Security Settings and Runtime Checks
    - H. Data Persistence
        1. Loading and Saving User Preferences and/or Files
        2. Using and Managing Databases (e.g., SQLLite) Within an Application

VI. METHODS OF INSTRUCTION:

- A. **Classroom Activity** - Student team-based analysis and implementation tasks to reinforce the content presented
- B. **Lab** - Including programming and user interface design tasks
- C. **Lecture** - Including interactive activities
- D. **Demonstration** -
- E. **Discussion** -

VII. TYPICAL ASSIGNMENTS:

- A. Create a mobile application that will maintain a list of user's friends, including names, birthdays, addresses, emails and phone numbers. It should allow the user to enter new friends, and to delete or updating existing ones. The program will alert the user three days before the birthdate of each friend in the list.
- B. Create an mobile application that lets the user create and use vocabulary flash cards. The program will allow the user to enter the contents for a each card, including a word and a definition. In practice mode, the program will show the each card's word, asking the user to select showing the definition or advancing to the next card.

VIII. EVALUATION:

- A. **Methods**
    1. Exams/Tests
    2. Quizzes
    3. Projects
    4. Class Participation
- B. **Frequency**
    1. At least two in-class midterm examinations, or one in-class midterm examination and several quizzes
    2. One in-class comprehensive final examination
    3. Several programming assignments of sufficient size and complexity to incorporating all concepts in the course.

IX. TYPICAL TEXTS:

1. *Android Programming for Beginners.* First ed., Packt Publishing, 2015.
2. Manning, Jonathon, Paris Buttfield-Addison, and Tim Nugent. *Learning Swift: Building Apps for macOS, iOS, and Beyond.* 2nd ed., O'Reilly Media, 2017.

X. OTHER MATERIALS REQUIRED OF STUDENTS:

- A. USB based memory device for storing classroom work and projects for use outside of the classroom environment