CTC 305 – Introduction to Game and Mobile Programming

Instructor	G. Poppe	E-Mail	gpoppe@csudh.edu
Phone	310-243-2417	Office	LIB 5717 (5th floor old library)
		Office Hours	Monday 1:00PM-2:00PM
Classroom	SCC 800	Class Time	MoWe 11:30AM - 12:45PM
Phone	(310) 243-3398	URL	http://csc.csudh.edu

COURSE DESCRIPTION:

This course teaches students through lectures, discussions, demonstrations, and classroom labs. Students learn the knowledge, skills, and abilities necessary to create games and applications on devices other than typical computers such as mobile devices, game consoles, etc. This course is intended for people interested in developing games.

PRE-REQUISITES

None

TEXTBOOK

None

REFERENCE:

TBA

COURSE GOALS:

The primary objective of this course is to teach students to develop short mobile device applications and game programming. The course presents the process in developing games and mobile applications using existing tools, techniques, and programming languages.

SPECIFIC INSTRUCTIONAL GOALS:

The purpose of the course is to provide the student with an overview of the field of Game and Mobile Programming through a semester long project.

COURSE OUTCOMES:

Upon completion of this course, students will be able to:

- Understand the use and application of "cloud" computing and the design of web based games
- Understand the fundamentals of program construction including the implementation of blocks, conditional statements, loops, and switch constructions.
- Understand the principles of variable creation and data storage in a program, including the use of identifiers, variable declaration, local and class scope variables, integer and floating point storage, widening, narrowing, and casting.
- Make use of collections in programs, including arrays, lists, and dictionaries.
- Understand the use of methods and the use of value, reference, and out parameters in method calls.
- Create objects containing data fields, behaviors, and constructors.

- Create data properties to provide managed access to data fields within an object.
- Understand the fundamental difference between objects managed by value and reference.
- Create class hierarchies, which make use of method overriding to customize child class behaviors and constructor chaining to ensure proper creation of classes at each level in the hierarchy.
- Use abstract classes to create template objects to be used as the base of a class hierarchy.
- Create custom enumerated types with a range of values that reflect the problem domain.
- Understand the principles of test---driven development and the steps to be followed when creating software using this technique.
- Understand the software engineering terms coupling, cohesion, and encapsulation. Know the reasons why high cohesion is a good design aim, along with low coupling and objects with high encapsulation.
- Understand the reasons behind code refactoring and the need to reduce the duplication of statements, and to ensure that the identifiers of items in a program properly reflect the purpose of the item.
- Understand the role of existing set of resources for game development.
- Be able to construct the setup, update and draw behaviors to create a working game.
- Make use of existing resources in the creation of 2D sprite---based games, including Color, Sprite Batch, Texture2D, Sprite Font, Rectangle, Point, Vector, Sound Effect, Song, Network Session, Gamepad State, and Keyboard State.
- Use a Content Manager to add textures and sound content to a game, and content loading methods to retrieve the content when the game runs.
- Use the gamepad and keyboard as input devices, both in a level- and edge- triggered manner.
- Use of Network application programmer interface to create networked games with a game lobby.
- Create pseudo-3D displays by use of overdrawing and sprite scaling.
- Create animated sprite game elements.
- Use the random number generator in game play to introduce randomness to game play elements and behaviors.
- Create games using XNA for deployment to game consoles and mobile devices (i.e., Xbox, Windows PC, and Windows Phone).
- Manage the deployment of games to the cloud, Xbox, Windows PC, Zune, and/or other devices.
- Debug programs by the use of breakpoints, single stepping, and viewing the contents of variables.
- Create mobile applications for Android based devices using Android Software Development Kit.
- Create applications based on HTML5 coding using third party development software.
-

ATTENDANCE:

Students are expected and encouraged to attend lectures and contribute to discussions. It is the student's responsibility to contact the instructor as early as possible if he/she cannot attend class. There will be no make---up opportunities, although all classes will have companion videos available on line.

The student is responsible for materials missed during an absence, whether excused or not. Classes will start at the prescribed time and will end at the prescribed time. Instructor will be available during the posted office hours and you may make an appointment for times not posted.

GRADING BREAKDOWN:
Homework/quiz   10%

Midterm Exam	20%
Final Exam	30%
Semester Project	40%
	100%

Evaluation criteria explained:

- Students are expected to be active participants in each class meeting. Full credit for participation will be extended to students who regularly ask questions, share observations, and contribute relevant personal experiences.
- The mid-term examination will consist of objective questions and will require a technological comprehension that covers the lecture material and assigned readings.
- The assignments will consist of a number of individual in class and homework tasks.
- The semester project will be the publication of an application available for download on one of the play markets. The application will be presented to the class for a portion of the grade.

Students will be given specific guidance on the amount of collaboration permitted for each assignment. Unless otherwise specified, all assignments are individual assignments, and thus must be completely the original work of the student submitting them and include proper citations to the published work of others.

Quizzes:

Quizzes may be given throughout the semester, at a rate of approximately 1 per chapter. Quizzes will always cover the material covered since the last Quiz or Exam. The quizzes will be combinations of objective and short-answer questions. Quizzes will be administered online via Blackboard. Makeup quizzes will not be given. However, the lowest quiz grade will be dropped. Any class material missed by the student is the student's responsibility to acquire.

GRADING SCALE:

	96-100 = A	90-95 = A-
87-89 = B+	83-86 = B	80-82= B-
77-79= C+	73-76= C	70-72= C-
67-6 = D+	63-66= D	
	below 60 = F	

GENERAL POLICIES:

ACADEMIC HONOR CODE

Programming assignments must be done individually. Failure to do so will result in a violation of the CSUDH Academic Honor Code. The following cases will be considered as violations: identical code, and extremely similar code. Violations will be reported to the Office of Vice President of Academic Affairs. Disciplinary action will be taken against any student who alone or with others engages in any act of academic fraud or deceit. (Read University Regulations in University Catalog) It is the student's responsibility to ensure they fully understand to what extent they may collaborate or discuss content with

other students. No exam work may be performed with the assistance of others or outside material unless specifically instructed as permissible. If an exam or assignment is designated "no outside assistance" this includes, but is not limited to, peers, books, publications, the Internet and the WWW. If a student is instructed to provide citations for sources, proper use of citation support is expected. Additional information can be found at the following locations:

http://www.apa.org/journals/webref.html
http://www.lib.duke.edu/libguide/citing.htm
http://bailiwick.lib.uiowa.edu/journalism/cite.html
http://www.cas.usf.edu/english/walker/papers/copyright/ipdummie.html
http://www.indiana.edu/~wts/wts/plagiarism.html
http://plagiarism.phys.virginia.edu/links.html
http://www.arts.ubc.ca/doa/plagiarism.htm
http://alexia.lis.uiuc.edu/%7ejanicke/plagiary.htm
http://webster.commnet.edu/mla/plagiarism.htm
http://www.virtualsalt.com/antiplag.htm
http://www.engr.washington.edu/~tc231/course_info/plagiarism.html
http://quarles.unbc.edu/lsc/rpplagia.html

ATTENDANCE POLICY
Excessive absences will result in lowered grades. Excessive absenteeism, whether excused or unexcused, may result in a student's course grade being reduced or in assignment of a grade of "F". Absences are accumulated beginning with the first day of class.

STUDENT ACADEMIC APPEALS PROCESS
Authority and responsibility for assigning grades to students rests with the faculty. However, in those instances where students believe that miscommunication, error, or unfairness of any kind may have adversely affected the instructor's assessment of their academic performance, the student has a right to appeal by the procedure listed in the Undergraduate Catalog and by doing so within thirty days of receiving the grade or experiencing any other problematic academic event that prompted the complaint.

ADA STATEMENT
Students with disabilities, who believe they may need an academic adjustment in this class, are encouraged to contact me as soon as possible to better ensure receipt of timely adjustments.