

CS 4381/5391: Topics in Software Engineering - Mobile Game Programming Summer 2019

CRN: 36189 (CS 4381), 36190 (CS 5381)

Lecture: MTWRF 2:00-4:10 pm in CCSB 1.0702

Website: <http://www.cs.utep.edu/cheon/cs4381>

Instructor: Yoonsik Cheon (x-8028, ycheon@utep.edu); office hours: MTWR 4:10-4:50 pm in CCSB 3.0606

Prerequisite: CS 3331 or instructor's approval

Description

This course is targeted for students who want to start writing mobile games on Android platforms. It is for Java programmers who want to adapt their programming skills to make exciting Android 2D games. The course will provide a solid foundation for developing Android games through hands-on learning. We will get started with the basics of Android programming by understanding its development framework. We will then develop three complete games built step by step, starting with a simple game and gradually increasing the complexity. Through these developments, we will learn how to implement the following game features:

- Sprite sheet character animation
- Scrolling parallax backgrounds
- 2D rotation, velocity, and collision
- Platform game levels
- Multi-touch screen input
- Other features like pickups, firing weapons, HUDs, sound effects, scenery, level transition, and high scores

Textbook

The textbook—John Horton, *Android Game Programming by Example*, Packt Publishing, 2015—should be available at the UTEP bookstore, and students are expected to acquire a copy for their use in this course, as reading assignments will be taken from the textbook. The following books are also recommended for supplementary reading.

Mario Zechner, Jerome F. DiMarzio and Robert Green, *Beginning Android Games*, 3rd edition, Apress, 2016.

Jerome F. DiMarzio, *Beginning Android Programming with Android Studio*, 4th edition, Wrox, 2017.

Dawn Griffiths and David Griffiths, *Head First Android Development: A Brain-Friendly Guide*, 2nd edition, O'Reilly, 2017.

Electronic books of the required textbook and the recommended references are available to authorized UTEP users through UTEP Library; use VPN from outside the UTEP domain (see the course website for the links to e-books).

Examinations

There will be one mid-term exam and the final. The mid-term exam will take place during the regular class session, and the final exam will take place on the date specified by the university. Makeup exams will be given only when you have unusual circumstances, such as incapacitating illness or presenting a research paper at a conference. If you believe that you have an unusual circumstance that warrants a makeup exam, notify us as soon as possible. If you will be attending a conference or other event, you must make arrangements for a make-up exam *in advance*. Under any circumstances, you may be required to provide official documentation before a make-up will be administered.

Homework Assignments

There will be several homework assignments, and most assignments will be programming assignments. Some of the assignments may be done in pairs or teams. No late submission will be accepted for homework assignments.

Semester Project

You should do a semester-long class project. The purpose of the semester project is to apply concepts and techniques learned in the course and develop a more realistic Android game that is feature-rich and may be publishable in the Android market. Sample project topics will be suggested by the instructor or you'll have a chance to propose your own project topic. In either way, your project must be approved by the instructor. You are expected to write a project proposal, demo a prototype, submit a final project report, and present the project result. Depending on the size and the complexity of the project, it can be an individual, pair, or team project; however, the initial proposal has to be written individually.

Technical Paper Presentations or Tutorials on Android (CS 5381)

CS 5390 students are required to:

- (a) present a technical paper on mobile game development, or
- (b) give a lecture/tutorial on additional topics or features related with Android game programming.

The presentation/lecture should be 15-20 minutes. You may present any tutorial or technical paper related with course topics; however, it has to be approved by the instructor.

Grading

Your grade is independent of anyone else's grade. We do not grade on a curve, and everyone can earn an A. The purpose of grading is not to rank you, but to uphold a standard of quality and to give you feedback. Your final letter grade will be calculated based on a combination of quizzes, in-class work, homework assignments, project assignments, exams, and class participation. The approximate percentages are shown below:

Homework:	40%
Semester project:	40%
Exams:	20%

There are also up to 5% bonus points for class attendance and participation. To earn this, you must arrive at classes on time and participate in class discussions in a constructive and prepared manner, e.g., by asking or answering questions that demonstrate that you have read and attempted to understand the material.

The nominal percentage-score-to-letter-grade conversion is as follows:

90% or higher:	A
80-89%:	B
70-79%:	C
60-69%:	D
below 60%:	F

The instructor reserves the right to adjust these criteria downward, e.g., so that 88% or higher represents an A, based on overall class performance. The criteria will not be adjusted upward, however.

Attendance

Class attendance is required; you should understand that your success in the course will improve greatly by attending classes regularly. The instructor reserves the right to penalize unexcused absences; e.g., your final grade may be lowered by one point for each unexcused absence above three. The following is excerpted from the 2017-2018 Undergraduate Catalog.

The student is expected to attend all classes and laboratory sessions. It is the responsibility of the student to inform each instructor of extended absences. When, in the judgment of the instructor, a student has been absent to such a degree as to impair his or her status relative to credit for the course, the instructor can drop the student from the class with a grade of W before the course drop deadline and with a grade of F after the course drop deadline.

Standards of Conduct

You are expected to conduct yourself in a professional and courteous manner, as prescribed by the Handbook of Operating Procedures: Student Conduct and Discipline. All graded work (homework, projects, exams) is to be completed independently and should be unmistakably your own work, although you may discuss your work with others in a general way. You may not represent as your own work material that is transcribed or copied from another source, including persons, books, or Web pages. "Plagiarism" means the appropriation, buying, receiving as a gift, or obtaining by any means another's work and the unacknowledged submission or incorporation of it in one's own academic work offered for credit, or using work in a paper or assignment for which the student had received credit in another course without direct permission of all involved instructors. Plagiarism is a serious violation of university policy and will not be tolerated. All cases of suspected plagiarism will be reported to the Dean of Students for further review.

Disabilities

If you have a disability and need classroom accommodations, please contact The Center for Accommodations and Support Services (CASS) at 747-5148, or by email to cass@utep.edu, or visit their office located in UTEP Union East, Room 106. For additional information, please visit the CASS website at www.sa.utep.edu/cass.

Course Outline

As shown below, the course will have four phases: boot camp, introduction to Android, game programming, semester project; refer to the next page for a detailed schedule.

Boot camp: Review (1 day)

1. Java GUI programming

Introduction to Android (3 days)

1. Android Studio
2. Basics of Android programming
3. Android 2-D graphics

Assignments: Connect Four

Game Programming (12 days)

1. Tappy Defender (Chaps 2-4)
2. Platformer (Chaps 5-8)
3. Asteroids simulator (Chaps 9-11)
4. Exam and CS 5391 presentations

Assignments: Tappy defender, platformer

Semester Project (3 days)

1. Project proposal
2. Prototype demo
3. Final presentation

Schedule

The following table shows a planned schedule for the course. The schedule is subject to change, and an up-to-date schedule will be available from the course website.

Dates		Topics	Readings	Assignments
Day 1	Jul. 9	Introduction Review: Java GUI programming		
Day 2	Jul. 10	Android Programming basic Lab: Android Studio	Chaps 1-2 of [DiMarzio17] or Chap 2 of [Zechner16]	
Day 3	Jul. 11	Android programming basics Lab: Timer		
Day 4	Jul. 12	Android 2D Graphics Lab: Connect Four	Basic Graphics Programming (p. 133 of [Zechner16])	HW1
Day 5	Jul. 15	Tappy Defender – First Step	Chap 2	
Day 6	Jul. 16	Taking Flight	Chap 3	
Day 7	Jul. 17	Going Home	Chap 4	HW2
Day 8	Jul. 18	<i>Project proposal</i>		
Day 9	Jul. 19	Exam 1		
Day 10	Jul. 22	Platformer – Upgrading Game Engine	Chap 5	
Day 11	Jul. 23	Bob, Beeps and Bumps	Chap 6	
Day 12	Jul. 24	Guns, Life, Money and the Enemy	Chap 7	
Day 13	Jul. 25	Putting It All Together	Chap 8	HW3
Day 14	Jul. 26	<i>Prototype demo</i>		
Day 15	Jul. 29	Asteroids at 60 FPS with OpenGL ES	Chap 9	
Day 16	Jul. 30	Move and Draw with OpenGL ES	Chap 10	
Day 17	Jul. 31	Things That Go Bump – Part II	Chap 11	
Day 18	Aug. 1	CS 5381 presentations		
Day 19	Aug. 2	<i>Project presentations</i>		
Day 20	Aug. 6	Final at 4:00 pm – 6:45 pm		

[DiMarzio17] Jerome F. DiMarzio, *Beginning Android Programming with Android Studio*, 4th Edition, Wrox, 2017.

[Zechner16] Mario Zechner, Jerome F. DiMarzio and Robert Green, *Beginning Android Games*, Apress, 2016.

Important Dates

July 9: Class begins
 July 11: Census day
 July 19: Exam 1
 July 26: Drop/withdrawal deadline
 August 2: Last day of classes
 August 6: Final on Tuesday at 4:00 pm – 6:45 pm