CS 4330/5390: Mobile Application Development
Fall 2020

CRN: 17029 (CS 4330), 16750 (CS 5390)
Live session: MW 4:30 pm - 5:50 pm in Blackboard
Instructor: Yoonsik Cheon (ycheon@utep.edu); office hours: MW 4:30 pm - 5:50 pm in Blackboard)
Prerequisite: CS 3331 or instructor's approval

Description

This course is targeted for students who want to start writing mobile applications on Android platforms. Android became a formidable mobile operating system, and this course will provide a solid foundation for developing Android apps through hands-on learning. We will get started with the basics of Android programming by covering the most recent version of Android and understanding its development framework. We will learn both the fundamentals and the nuts and bolts of Android and have an exciting opportunity to write feature-rich Android applications that may be published in the Android market (see Appendix for learning outcomes).

Textbook

The textbook—J. F. DiMarzio, *Beginning Android Programming with Android Studio*, 4th edition, Wrox (John Wiley & Sons), 2017—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.

> Dawn Griffiths and David Griffiths, *Head First Android Development: A Brain-Friendly Guide,* 3rd edition, 2021. Provides detailed instructions and explanations for beginners.
>
> Paul Deitel, et al., *Android 6 for Programmers: An App-Driven Approach*, 3rd edition, Prentice Hall, 2015. Includes eight different example apps.
>
> Bill Philips, et al., *Android Programming: The Big Nerd Ranch Guide*, 4th edition, Big Nerd Ranch Guides, 2019. Excellent guide on writing professional apps.
>
> Trish Cornez and Richard Cornez, *Android Programming Concepts*, Johns & Bartlett Learning, 2015. Read Chapters 6 (threads and handlers).

Electronic copies 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).

Course Structure

This course is entirely online and delivered through the Blackboard Learn virtual learning environment. You will use your UTEP account to login to the course from the Blackboard link in the My UTEP page (https://my.utep.edu).

In Blackboard Learn, you will access online lessons, course materials, and resources. You will participate in a blend of self-paced and group-paced activities using Blackboard Learn and alternative Internet-based technologies. There will be learning modules containing tasks and activities such as readings, exercises, coding, discussions, quizzes, blog, and presentations/demos. You will need to complete these semi-weekly lessons, or learning modules (see Figure 1 below). The asynchronous nature of most tasks and activities will permit you to complete them, or take part, at times most convenient to you.

Examinations

There will be one mid-term exam. The mid-term exam will be online and will be available at 4:30 pm – 5:50 pm on the exam date (see the Schedule section below). There will be no final exam.

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.

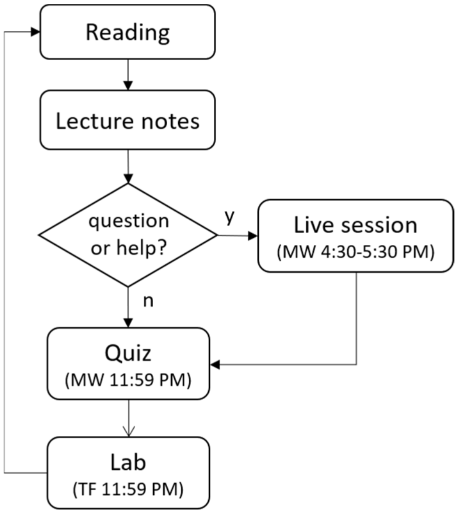

Figure 1. Complete semi-weekly lessons, or learning modules

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 are expected 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 application 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 complexity of a project, it may be done individually, in pairs, or in teams; however, the initial proposals must be written individually.

CS 5390 Presentation

If you are registered for CS 5390 (graduate course), you are required to:

(a) present a technical paper on mobile application development, or

(b) give a mini lecture/tutorial on additional topics or features of Android programming.

The presentation/lecture should be 15-20 minutes. You may select any technical paper related with course topics but it has to be approved by the instructor. The topics or features for the lecture include, but are not limited to, touch gestures, sensors, camera, tabbed interface, navigation drawer, and Kotlin language (see page 4).

Grading Policy

Your grade is independent of anyone else's grade. We do not grade on a curve, and everyone can earn an A in this course. 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 lessons, homework assignments, semester project, and exam. The approximate percentages are shown below:

Activities	Percent (%)
Lessons (readings, quizzes, exercises, etc.)	40
Homework	20
Semester project	20
Exam	20

There are also up to 5% bonus points for class attendance and participation; attendance is determined by participation in the learning activities of the course. To earn this bonus, you must participate in most online activities as listed in the course content and complete them on time. We will monitor, track, and score your participation in the course using Blackboard tracking tools, discussions, blogs, chat sessions, and group work.

Be sure to pay close attention to deadlines—there will be no makeup assignments or quizzes, or late work accepted without a serious and compelling reason and instructor approval. All work and assignments for this course will be submitted electronically through Blackboard Learn unless otherwise instructed. They must be submitted by the given deadline or special permission must be requested from instructor before the due date. Extensions will not be given beyond the next assignment except under extreme circumstances.

Final letter grades assigned for this course will be based on the percentage of total points earned and are assigned as follows. The nominal percentage-score-to-letter-grade conversion is as follows:

Letter grade	Percent (%)	Performance
A	90-100	Excellent
B	80-89	Good
C	70-79	Average
D	60-69	Poor
F	0-59	Failing

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/Participation

Attendance in the course is determined by participation in the learning activities of the course. Students are expected to participate in all online activities as listed in the course content. The asynchronous nature of most activities will permit you to take part at times most convenient to you.

You should understand that your success in the course will improve greatly by participating/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 2020-2021 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.

Accommodations

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.

COVID-19 Precautions

You must STAY AT HOME and REPORT if you (1) have been diagnosed with COVID-19, (2) are experiencing COVID-19 symptoms, or (3) have had recent contact with a person who has received a positive coronavirus test. Reports should be made at screening.utep.edu. If you know of anyone who should report any of these three criteria, you should encourage them to report. If the individual cannot report, you can report on their behalf by sending an email to COVIDaction@utep.edu.

For each day that you attend campus—for any reason—you must complete the questions on the UTEP screening website (screening.utep.edu) prior to arriving on campus. The website will verify if you are permitted to come to campus. Under no circumstances should anyone come to class when feeling ill or exhibiting any of the known COVID-19 symptoms. If you are feeling unwell, please let me know as soon as possible, and alternative instruction will be provided. Students are advised to minimize the number of encounters with others to avoid infection.

Wear face coverings when in common areas of campus or when others are present. You must wear a face covering over your nose and mouth at all times in this class. If you choose not to wear a face covering, you may not enter the classroom. If you remove your face covering, you will be asked to put it on or leave the classroom. Students who refuse to wear a face covering and follow preventive COVID-19 guidelines will be dismissed from the class and will be subject to disciplinary action according to Section 1.2.3 *Health and Safety* and Section 1.2.2.5 *Disruptions* in the UTEP Handbook of Operating Procedures.

Course Outline

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

Boot camp: introduction and reviews (1.5 weeks)

1. GUI programming
2. UML class diagram
3. Design patterns (optional)

Assignment: UML, Java dialog

Android Programming (12.5 weeks)

1. Introduction to Android and Android Studio (Chapters 1 & 2)
2. Activities, fragments and intents (Chapter 3)
3. Android GUI (Chapter 4)
4. Views and menus (Chapters 5 & 6)
5. Data persistence (Chapter 7)
6. Multithreading (Chapter 6 of Cornez-Cornez15)
7. Messaging and networking (Chapters 9 & 11)
8. Content providers (Chapter 8)
9. Location-based services (Chapter 10)
10. Developing android services (Chapter 12)
11. Others: Touch gestures, sensors and camera, tabbed interface, navigation drawer, Kotlin language, etc.

Assignments: Android studio, GUI, activity and intents, multithreading, and networking

Semester Project (2 weeks)

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 in Blackboard.

Dates		Topics	Readings	Assignments
Week 1	Aug. 24, 26	Introduction Review: GUI programming	Handout	
Week 2	Aug. 31, Sep. 2	Review: UML Intro to Android programming	Chap 1	
Week 3	Sep. 7, 9	*Labor day – no class* Android Studio	Chap 2	HW1
Week 4	Sep. 14, 16	Activities and intents	Chap 3	
Week 5	Sep. 21, 23	UI (basics and layouts)	Chap 4	
Week 6	Sep. 28, 30	Views and widgets	Chap 5	
Week 7	Oct. 5, 7	Menus Fragments	Chap 6 Chap 3 (fragments)	HW2
Week 8	Oct. 12, 14	Data persistence	Chap 7	
Week 9	Oct. 19, 21	***Project proposal*** **Exam 1**		
Week 10	Oct. 26, 28	Multithreading	Chap 6 of [CC15]	
Week 11	Nov. 2, 4	Messaging and networking	Chap 9 & 11	HW3
Week 12	Nov. 9, 11	***Prototype demo*** Content providers	Chap 8	
Week 13	Nov. 16, 18	Location-based services CS 5390 presentations*	Chap 10	
Week 14	Nov. 23, 25	Android services	Chap 12	
Week 15	Nov. 30, Dec. 2	***Project presentation***		
Week 16	Dec. 7	~~Final at 4:00 pm – 6:45 pm~~		

[CC15] Trish Cornez and Richard Cornez, *Android Programming Concepts*, Johns & Bartlett Learning, 2015.

*Possible topics: touch gestures (7.1-7.4 of [CC15]), drag & drop, fling, multi-touch (7.5-7.9), sensors (8.1-8.7), camera (8.8-8.9), tabbed interface, navigation drawer, Google Firebase, Kotlin language, etc.

Important Dates

August 26:	Classes begin
September 7:	Labor day – university closed
September 9:	Census day
October 21:	Exam 1
October 30:	Course drop/withdrawal deadline
November 26-27:	Thanksgiving holiday - university closed
December 3:	Last day of classes
December 4:	Dead day
December 7:	~~Final on Monday at 4:00 pm – 6:45 pm~~

CS 4330/5390: Mobile Application Development
Learning Outcomes

Level 1: Knowledge and Comprehension

Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. The material has been presented only at a superficial level. Upon successful completion of this course, students will be able to:

1a. Describe the unique characteristics and features of (Android) mobile applications.
1b. Define Android-specific programming concepts such as activities, intents, fragments, services, broadcast receivers, and content providers.
1c. Describe various user interface components of Android, including layouts, widgets, views, and menus along with the UI events associated with them.
1d. Describe both the benefits of multithreading on Android and the ways to implement multithreading in Android applications.
1e. Explain how to detect and respond to various touch gestures.
1f. Articulate the communication and network programming features and capabilities of Android platforms.
1g. Describe sensors---motion sensors, environmental sensors, and positional sensors---most commonly embedded in Android devices along with their application programming interfaces.
1h. Explain the structures of Android projects, including manifest, source code, and various resources such as assets, drawables, layouts, menus, and values.

Level 2: Application and Analysis

Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details. Upon successful completion of this course, students will be able to:

2a. Differentiate techniques for persisting user data, such as shared preferences, traditional file systems (internal and external storage), and SQLite database.

2b. Design and create the user interface of a moderate-sized application using various views, widgets, menus, 2-D graphics, animations, and multimedia.

2c. Create a small-sized application that utilizes Android communication/networking capabilities---phones, SMS messages, emails, HTTP, JSON web services, TCP/IP Sockets, and Bluetooth sockets---to communicate with the outside world.

2d. Create a small-sized application that utilizes embedded sensors such as accelerometers, gyroscopes, cameras, and GPS, e.g., a simple location-based service (LBS) application using Google Maps to obtain, monitor, and track geographical locations.

Level 3: Synthesis and Evaluation

Level 3 outcomes are those in which the students can apply the material in new situations. This is the highest level of mastery. Upon successful completion of this course, students will be able to:

3a. Develop an Android application of moderate size that utilizes various features of Android platforms, such as user interfaces, multithreading, data persistency, content providers, messaging and networking, embedded sensors, LBS, and background services.
3b. Express the design of a moderate-sized Android application using a standard modeling notation such as UML and evaluate its quality by relating to software engineering design principles such as cohesion and coupling, encapsulation, reusability, and maintainability.