

## Department of Computer Science

# CS4153/CS5153 – Mobile App Development

---

### Classes

CS4153 – CRN 27922, Web/Internet Course  
CS4153 – CRN 27923, Web/Internet Course  
CS5153 – CRN 31593, Web/Internet Course

### Instructor

**Dr. Moawia Eldow**

Stillwater Office: MSCS 224

Tel - 405.744.2607

[meldow@okstate.edu](mailto:meldow@okstate.edu)

TA: TBA

#### **Virtual Office Hours:**

Tuesday & Wednesday 3:00 PM to 5:00 PM

Available by appointment Via Zoom, Skype for Business or Canvas Conferences.

### Description

(This is an online course. All video classes, resources, assignments, discussions and announcements are accessed entirely online through Canvas at [canvas.okstate.edu](https://canvas.okstate.edu). You must use your campus O-Key username and password to access the course.)

The history of mobile apps and their implication on computing in general. Survey of the various platforms and approaches used for mobile apps. Examine the differences between "conventional" programs and mobile apps. Learn tools and techniques to develop mobile apps, and demonstrate proficiency through development assignments. May not be used for degree credit with CS4153/CS5153.

### Prerequisites

A good working knowledge of OOP, and either a good working knowledge of Java (e.g., CS2133) or C++ (e.g., CS 2433).

### Goals

Examine the similarities and differences between mobile computing and traditional computing. Discuss the effects that mobile computing has had and will have on computing and society. Learn to program mobile apps for the iOS platform and devices (iPhone, and iPad) using the Swift programming language and an IDE. Learn to program mobile apps for the Android program using the Kotlin programming language and an IDE. Work individually and in groups to develop mobile apps.

## Objectives

By the time you finish this course, you should have learned the following:

- The primary differences between traditional computer programs and mobile apps, and how to address those differences when writing an app.
- Sources of current and upcoming news and trends related to mobile computing, and evaluating your own design and development practices in terms of what you discover there.
- The use of the Xcode IDE (Interactive Development Environment) to develop, test, and debug apps for devices that run the Apple iOS operating system.
- The syntax and semantics of the Swift programming language, as well as several fundamental iOS APIs (Application Program Interfaces) necessary to develop apps.
- Use of the Kotlin programming language and Android Studio to develop Android apps.
- Several fundamental Android APIs necessary to develop apps.
- Working as part of a team to develop apps more effectively and efficiently.

## Textbook

*None!* There are online course materials which can be accessed on Canvas.

### Recommended References:

- 1- **Native Mobile Development: A Cross-Reference for iOS and Android**, Shaun Lewis, Mike Dunn, O'Reilly Media, Inc. (2019), ISBN: 9781492052876.
- 2- **Mobile App Development for iOS and Android**, Jakob Iversen, Michael Eierman, Prospect Press (2020), ISBN: 9781943153909 (eText), ISBN: 9781943153916 (print).

## Instructor Response Time

The response time is within 48 hours for all student inquiries. Grades and feedback for assignments will be posted within one week of the assignment due date. All inquires can be made Via Canvas or directly to the instructor's email. Use CS4153/CS5153: <short description> as a subject of your email.

## Participation Expectations

This is an online class. There are deadlines for items every week of the semester. The course is largely self-managed and rapidly paced. You must keep up with the course work including announcements, watching video classes, readings, assignments, discussions and exams. The easiest way to keep up is to work ahead by a week. Students need to be aware about the course outline posted on Canvas and at the end of this syllabus.

If a student has not participated in at least one assignment within the first two weeks of class then it is at the discretion of the instructor to submit an academic alert or contact the student directly to discuss options for continuing in the course.

## Grading Policy

Assignment:	Percentage of Total Grade
Programming Assignments	45%
Project	20%
Discussions	15%
Exams	20%
<b>Total</b>	<b>100%</b>

Final grades will be assigned according to the following standard scale:

Grade	Percentage Earned
A	90 – 100%
B	80 – 89.9%
C	70 – 79.9%
D	60 – 69.9%
F	0 – 59.9%

## Assignment Descriptions

The assignments for the course are as follows:

- **Individual programming assignments (45%)** – One assignment every two to three weeks.
- **Team programming assignments (Projects) (20%)**– A single programming assignment during the last 10 weeks of the course. A progress report is due 4 weeks into the assignment; completed app and documentation is due at the end of the 10-week assignment period.
- **Discussions** – 15% - One discussion during every module.
- **Examinations** (10% each) – The course includes a mid-semester and final examination.

### Programming Assignment Submission Logistics

- All programming solutions must be submitted via the Canvas classroom submission page that corresponds to the assignment.
- All files associated with a given programming submission must be zipped into a single file. The zipped files must retain their folder structures and contain all needed assets so that – once opened – they can be built and run by the grader. Solutions that are uncompressed or compressed with a format other than zip will be ignored.

### Late Submission and Make Up Policy

Individual programming assignments cannot be made up if they are missed except under extenuating circumstances such as serious illness, family emergency, etc. Please contact the instructor if the need to submit an assignment late arises. For no excuse of late submission, students may be subject to the penalties (Refer to Canvas to understand the late policies for each assignment). Students need to work ahead by several days to avoid late submissions.

## Technical Requirements

### Software/Hardware requirements:

- An Apple Macintosh Computer running macOS Catalina (10.15). iMacs are available for your use in the Stillwater lecture lab, 222 MSCS.
- The XCode 11 IDE for macOS – Register for free at the *Apple Developer* website, <http://developer.apple.com>, where you will click on the *Account* link to create an account or sign in. You then can click on the *Downloads* link and download Xcode for your own, personal Mac.
- Android Studio and Java for Android development toward the end of the semester. Optional: any iPod Touch, iPhone, or iPad that is compatible with iOS 12 or 13.
- Other software as specified during the semester.

### **Collaboration policies:**

- **Individual programming assignments:** Discussion of concepts, ideas, and techniques is acceptable. After discussion, each student must write up his/her own solution. Copying another person's work, in part or in whole, is not allowed. Giving another student your work, in part or in whole, is considered cheating as well. If you are unsure whether your collaboration is acceptable, speak with the instructor in advance. Take care that your solutions are not exposed to or by other students.
- **Team programming assignments:** Sharing of work among students on a project team is acceptable. Inter-team discussion of concepts, ideas, and techniques is acceptable, but inter-team sharing of work is not permitted. If you are unsure whether your collaboration is acceptable, speak with the instructor in advance.
- **Examinations:** During an examination period, no communication of any kind about the exam is allowed.

### **Disabilities act:**

According to the Americans with Disabilities Act, each student with a disability is responsible for notifying the University of his or her disability and to request accommodations. If you think that you have a qualified disability and needs special accommodations, you should notify the instructor and request verification of eligibility for accommodations from the Office of Student Disability Services, 315 Student Union, (405)744-7116. Please advise the instructor of such disability as soon as possible, and contact Student Disability Services, to ensure timely implementation of appropriate accommodations. The instructor of this class will respond when he receives official notice of a disability, but he does not provide retroactive accommodations.

### **University Syllabus Attachment**

Other useful information, such as important dates and further policies, throughout the semester, can be found on the following links ([You can click on Spring 2021 Syllabus Attachment](https://academicaffairs.okstate.edu/faculty-development/index.html)):

<https://academicaffairs.okstate.edu/faculty-development/index.html>

## Course Outline and Schedule

<b>Module</b>	<b>Weeks and Dates</b>	<b>Topics &amp; Activities</b>
Module 1	Weeks 1 & 2 Jan 18 – Jan 29, 2021	<ul style="list-style-type: none"> <li>• Introduction to mobile computing, Swift programming, and Xcode IDE. Source control in Xcode.</li> <li>• Assignment – 1 &amp; • Discussion – 1</li> </ul>
Module 2	Weeks 3 & 4 Feb 1 – Feb 12, 2021	<ul style="list-style-type: none"> <li>• Graphics &amp; Colors, Delegation, Gestures, Segues</li> <li>• Assignment – 2 &amp; • Discussion – 2</li> </ul>
Module 3	Weeks 5 & 6 Feb 15– Feb 26, 2021	<ul style="list-style-type: none"> <li>• Table views, Alerts, Stack views &amp; Auto layout</li> <li>• Discussion – 3</li> </ul>
Module 4	Weeks 7* & 8 Mar 1 – Mar 12, 2021	<ul style="list-style-type: none"> <li>• Multithreading &amp; Core data</li> <li>• Project – Phase 1 due</li> <li>• Discussion – 4</li> </ul> <p><b>Tuesday, March 9 (Mid-term Exam)</b></p>
Module 5	Weeks 9 & 10 Mar 15 – Mar 26, 2021	<ul style="list-style-type: none"> <li>• Swift UI, Restful services, iOS on iPad</li> <li>• Assignment – 3 &amp; • Discussion – 5</li> </ul>
Module 6	Weeks 11 & 12 Mar 29 – Apr 9, 2021	<ul style="list-style-type: none"> <li>• Introduction to Android, Android Resources, Adapters, View holders &amp; recyclers</li> <li>• Assignment – 4 &amp; • Discussion – 6</li> </ul>
Module 7	Weeks 13 & 14 Apr 12 – Apr 23, 2021	<ul style="list-style-type: none"> <li>• Tables, Anko &amp; Anko commons, Databases in Android, Android Threads &amp; Coroutines</li> <li>• Assignment – 5 &amp; • Discussion – 7</li> </ul>
	Weeks 15 & 16 Apr 26 – May 7, 2021	<ul style="list-style-type: none"> <li>• Review and exam weeks</li> <li>• Project – Phase 2 due</li> <li>• <b>Final Exam</b> (TBA)</li> </ul>

(\*) Tuesday, March 3 (**Six-week grades due**)

- The points of Discussions 1 & 2 and Assignments 1 & 2 will be collected for these grades.