## Course Info

**Course:** CSCI 4237 – Software Design for Handheld Devices, Spring 2022 **Instructor:** Dr Michael Cobb  
**Contact:** [Michael.cobb@gwu.edu](mailto:Michael.cobb@gwu.edu) 540.834.9569  
**Location & Time:** TOMP 402, Tuesdays, 6:10 – 8:40 PM

## Bulletin Description

Design of interactive software for handheld devices. Event driven programming, user interface design practices, memory management, handheld debugging techniques.

## Detailed Description

In this course, students will learn the basics of Android app development. Android currently holds the majority of the global market share for smartphone operating systems and additionally will run on smartwatches, vehicle dashboards, TVs, and even embedded systems. For consumers, Android runs on over 20,000 unique device models – which hit different price points, feature sets, designs, performance, and form factors. For developers, Android is a flexible platform: allowing apps to be programmed in either Java or Kotlin (and even C/C++ at times!), has strong library support from Google and third parties, offers access to numerous hardware sensors, is free to develop for, and cheap to deploy to the Google Play store.

Android development often requires knowledge in many different areas, which we will cover in this course: UI design, networking with API calls, threading, data persistence, localization, analytics, third-party libraries, design patterns, and more!

## Prerequisites

* CSCI 2113 – Software Engineering or CSCI 6221 – Advanced Software Paradigms.
  + Familiarity with Java (or similar object-oriented language).
  + Basic concepts and usage of Git / GitHub.
  + A laptop which you can use to access learning materials and use to work on assignments.
  + Some of the tools we use in this class are CPU-intensive (like running the Android emulator). This means older machines or those with very low specs may experience difficulty.
    - In the event you cannot run the Android emulator, you will need to purchase a physical Android device to test with (next section).
    - [System Requirements - Android Studio](https://developer.android.com/studio#Requirements)
    - [System Requirements - Android Emulator](https://developer.android.com/studio/run/emulator#requirements)

## Materials

* **No book is required**, but the following are decent references if you want something:
  + Google has many good [Android Kotlin codelabs](https://developer.android.com/courses/kotlin-android-fundamentals/overview).
  + *Kotlin in Action*
  + *The Big Nerd Ranch Guide* (has Android and Kotlin variants)
* **An Android device will** **not be required**, as the Android emulator will come bundled with the Android development environment (which I will also use for all lectures).
  + But, if you are *unable* to [run the Android emulator](https://developer.android.com/studio/run/emulator#requirements) on your machine, you will ***need*** to purchase a device to test with.
  + Otherwise, it’s also a nice experience to test on a real device with “real” data. If you’d like to buy a cheap device, **aim for Android 6.0 (Marshmallow)** or higher.
    - Amazon has many Android devices available and you can often find ones under $100 (read reviews or stick with notable brands like Samsung / Google / Nokia / etc.).
    - **You will not need a SIM card / cellular contract.**

## 

## 

## 

## Expected Outcomes

By the conclusion of the course, students should expect to be able to:

* Recognize the complexities of development on the Android platform, including: UI design, handling user input, handling screen rotation or configuration change, app architectures, user data persistence, threading, and supporting Android’s device fragmentation - many of these extend to iOS and hybrid development, as well.
* Work with data originating from another system -- using RESTful API calls and data parsing to utilize data from sources such as Twitter and Google.
* Develop an application resilient to errors -- handling a spotty network connection, handling a user who denies runtime permissions, and implementing crash reporting.
* Measure basic user engagement through analytics.
* Apply the Kotlin programming language for the purpose of Android development, demonstrated over two major app projects.
* Use Git and GitHub for working on projects, collaborating with others, and giving / receiving code reviews.
* Complete two fairly comprehensive software projects -- the class contains two app projects, which require students to develop fully-functional and error-resilient mobile applications within a set of constraints.

## Grading

The course will be graded **on the standard A-F scale, no plus-minus**, with this breakdown:

* 15% - In-Classwork/Homework
* 10% - Quizzes
* 15% - Final Exam
* 60% - Projects (two large app projects split 30% / 30%)

**Classwork/Homework** carries a total weight of 15% and will include small assignments to help you prepare for upcoming lectures or projects.

**Projects,** two with a combined weight of 60%, will have you develop an Android app within a set of constraints. The first app project will be constrained to a particular set of functionality that I will specify and the second app project will be very open-ended for you to choose, more in line with a “final project” where you may be allowed to work in pairs and will give some sort of final presentation or demo. Grading on the projects will be a combination of whether it functions properly and fulfills project requirements (most importantly), code quality, effective use of Git, any significant bugs found, etc. I will provide grading rubrics for each assignment.

**Quizzes and Final Exam**, for a total of 25%, will be announced ahead of time and will cover material presented during lectures and homework. I generally will give some guidance on what topics to study for.

## Submission

All assignments will be submitted via Blackboard and GitHub – code through GitHub and final application APK files/Reports through Blackboard. Quizzes and exams will be also submitted through Blackboard.

**The cutoff time will generally be 11:59 PM on the due date.** For GitHub code submissions, I will use the commit timestamps.

## Late Policy / Extra Credit / Rounding

Late work **will be accepted**, however you will be docked **10 percent** from your grade per 24 hours late (**with a limit of 2 days** **late**). For example, if an assignment is due by Friday by 11:59 PM, the latest you can submit would be Sunday by 11:59 PM for a max of 80%. Presentations, quizzes, and exams cannot be done late unless arranged in advance.

There will not be separate extra credit assignments. I may offer a small bit of credit for completing course evals / surveys towards the end of the course to substitute for rounding.

## Student Expectations

There are so many topics within the realm of mobile programming and we can only cover so much during a single semester. When completing assignments, there will likely be times where you need to consult external resources / tutorials to learn what you need for a particular project. Great resources include the Android Developers site, StackOverflow, and even YouTube video tutorials. ***Small* code snippets can be used if cited in comments, but otherwise the *majority* of work submitted *must* be original.**

Coding projects will be submitted via GitHub and I will be reviewing pull requests and/or commits – looking at code style, code organization, usage of comments, and commit messages. In this way, I hope to be able to give you valuable feedback on your work and recognize when you go above-and-beyond.

You should expect to spend roughly 2.5 hours in lecture per week, plus roughly double that, 5 hours, outside of the course (totaling to 7.5 hours per week) varying by what homework / projects are currently assigned and upcoming quizzes / exams.

Finally, the app projects may take a decent amount of time and coding, with due dates roughly announced ahead of when the project was assigned. Ultimately, it will be your responsibility to effectively manage time to complete the project and all requirements. **Do not wait until the final weekend to work on projects**. You will **not** have enough time to fully complete them if you value your sleep schedule.

## Plagiarism

**I take plagiarism very seriously**, I have *not* had a semester where plagiarism hasn’t been a problem – it happens to freshmen all the way up to seniors and graduate students. I plan to heavily use Git for students to check-in their work and as a way for me to review code. If caught plagiarizing, you will be heavily penalized for the project and I will follow up with the CS department for additional action as necessary.

I have no problem with code taken from StackOverflow or learned via some video tutorial, however these instances should generally be minimal and cited in your code in comments.

## Communication

I will send announcements through both Blackboard and email. Course information, homework assignments, quizzes, exams, and grades will be posted to Blackboard.

To reach me, I’d prefer you to utilize email or text me.

## Office Hours

You can feel free to ask questions or get help by directly messaging me and I will often be available before & after lecture for in-person questions.

## 

## Tentative Schedule of Topics and Due Dates

On days that items are due, they are due by 11:59 PM EST, unless specified otherwise.

|  |  |  |
| --- | --- | --- |
| **Week** | **Lecture Day** | **Lecture Topic & Due Dates** |
| 1 | 01/17 | Course Intro, Why Android?, Kotlin, Git  **Homework 1 assigned** |
| 2 | 01/24 | Git, Android Basics, UI Design,  **Homework 1 due (Due by Lecture), Homework 2 assigned** |
| 3 | 01/31 | UI Design, Activities  **Homework 2 due** |
| 4 | 02/07 | Intents, Lists  **Quiz1, Project 1 assigned** |
| 5 | 02/14 | Lists, Shared Preferences, Dialogs  **Project 1 check-in #1** |
| 6 | 02/21 | Google Maps, Geocoding, Threading  **Homework 3 assigned** |
| 7 | 02/28 | Networking, JSON  **Project 1 check-in #2, Homework 3 due** |
| 8 | 03/07 | Networking, OAuth **Quiz 2, Project 1 due, Project 2 Assigned** |
| 9 | 03/14 | **Spring Break-No Class** |
| 10 | 03/21 | Firebase: Authentication, Realtime DB |
| 11 | 03/28 | Firebase: Realtime DB, Analytics, Crash Reporting  **Homework 4 assigned, Project 2 proposals due** |
| 12 | 04/04 | Permissions, Location  **Quiz 3, Homework 4 Due** |
| 13 | 04/11 | Handling Rotation, Notifications  **Project 2 check-in #1** |
| 14 | 04/18 | Sensors, Automated Testing  **Quiz 4** |
| 15 | 04/25 | PRESENTATIONS  **Project 2 check-in #2** |
| 16 | 5/02 | **Make up day**  **Project 2 due** |
| 17 | 05/09 | Final Exam (TBD) |

## University Policies

[https://registrar.gwu.edu/university-poliies](https://registrar.gwu.edu/university-policies)

### Religious Holidays

In accordance with University policy, students should notify faculty during the first week of the semester of their intention to be absent from class on their day(s) of religious observance. For details and policy, see: [students.gwu.edu/accommodations-religious-holidays](http://students.gwu.edu/accommodations-religious-holidays).

### Academic Integrity

Academic dishonesty is defined as cheating of any kind, including misrepresenting one's own work, taking credit for the work of others without crediting them and without appropriate authorization, and the fabrication of information. For details and complete code, see: studentconduct.gwu.edu/code-academic-integrity

### Disability and Mental Health Services

**Disability Support Services** - Any student who may need accommodation based on the potential impact of a disability should contact the Disability Support Services office at 202-994-8250 in the Rome Hall, Suite 102, to establish eligibility and to coordinate reasonable accommodations. For additional information see: <https://disabilitysupport.gwu.edu/>

**Mental Health Services** - The University's Mental Health Services offers 24/7 assistance and referral to address students' personal, social, career, and study skills problems. Services for students include: crisis and emergency mental health consultations, confidential assessment, counseling services (individual and small group), and referrals. Phone: 202-994-5300. <https://counselingcenter.gwu.edu/>

### On-Campus Safety and Security

In the case of an emergency, if possible, the class should shelter in place. If the building that the class is in is affected, follow the evacuation procedures for the building. After evacuation, seek shelter at a predetermined rendezvous location.