

# Mobile Application Development

## Computer Science 344

<b>Instructor:</b>	Erik Krohn
<b>E-mail:</b>	krohne@uwosh.edu
<b>Text Message Only:</b>	608-492-1106
<b>Class/Lab Times:</b>	Tuesday: 5:10pm - 6:40pm & 6:40pm - 8:10pm
<b>Class/Lab Room:</b>	Tuesday: Clow 232 followed by Clow 016C
<b>Office Location:</b>	Halsey 216
<b>Office Hours:</b>	Tuesday & Thursday: 2:00pm - 3:00pm Wednesday: 10:30am - 12:30pm
<b>Prerequisites:</b>	Computer Science 262 with a grade of C or better.
<b>Course Website:</b>	<a href="http://uwosh.instructure.com">http://uwosh.instructure.com</a>
<b>Number of Credits:</b>	3

### Course Information

An introduction to the tools for developing mobile applications. Topics covered include: history of mobile development, using an appropriate IDE, emulating a mobile device, building a flexible user interface, understanding the application lifecycle, creating and managing multiple threads, creating and using web services and encrypting a completed project. A large mobile applications will be created throughout the course.

### Course Website

You should check canvas on a regular basis - it will contain lecture notes, handouts, assignments, announcements, and grades. Ill do my best to let you know when something new and important comes up, but it is your responsibility to check the web site frequently for information that you might not get otherwise.

### Project

**Your project must be written using Android Studio.** One of your goals (during this class and beyond, in Java or any programming language) should be to write understandable, readable code. You should be making every effort to comment anything that might be confusing to a reader unfamiliar with your program, to name variables intelligently, to use indentation that reflects the codes organization, and so on. All of this will be taken into account during grading: poorly organized or written code may have a negative impact on your grade, even if the resulting program works fine.

Keep this in mind when writing programs: write your code in small pieces, making sure each piece works before moving on to the next one. It is much better to turn in a project that is not finished but has many working pieces than to turn in one that doesn't work at all, even though most of the code is written.

All labs must be submitted electronically via canvas. It is your responsibility to ensure that your submission was submitted correctly. You must double check to ensure your program was uploaded correctly. **There are no late submissions.**

## Grading

Course grades will be based on labs and a large programming project. Your final grade will be computed with the following percentages:

- 30% - final project iterative steps
- 30% - final project draft submission
- 15% - final project evaluations
- 25% - final project final submission

Grading will be on a plus/minus system. Grading may be done on a curve depending on the overall performance of the class. If no curve is used, your grade will be computed based on the following:

Percentage	Grade	Percentage	Grade
$\geq 92$	A	72 - 78	C
90 - 92	A-	70 - 72	C-
88 - 90	B+	68 - 70	D+
82 - 88	B	62 - 68	D
80 - 82	B-	60 - 62	D-
78 - 80	C+	< 60	F

## Course Outcomes

1. Understand the mobile platform and development environment and how it differs from traditional application development.
2. Understand the basic components on which mobile applications are created.
3. Be able to create a manifest file to present essential information to the system about an application.
4. Use permissions correctly to restrict access to a part of code or to data on a device.
5. Be able to define and create applications with sophisticated user interfaces including using layouts, menus, action bars, dialogs, notifications, drag and drop and toasts.
6. Provide the user with notifications when the application is running in the background.
7. Be able to handle concurrency using threads and handlers.
8. Create an application where the application can access data over a network from a mobile device.
9. Create an application using GPS locational services.
10. Be able to animate an application and create applications that respond to custom gestures.

## Other Information

1. Attendance is not taken in this course. However, you are unlikely to do well if you miss lecture and/or lab.
2. **Academic dishonesty** of any kind will not be tolerated. All assignments, labs, mini assignments and exams are to be completed individually. While discussion of ideas and problems with fellow students is encouraged, all projects and labs must be done individually. In certain circumstances, code fragments from the instructor may be provided to eliminate tedious coding or to provide a common framework for all students. **All other code must be original.** Online resources may be used to help you understand the material, but you may not copy online code nor can you “borrow” code from other students, past or present.

Any suspected academic dishonesty will be dealt with on a case-by-case basis. Any clarification of what does or does not constitute academic dishonesty must take place **before** you turn in questionable work. For clarification on what constitutes academic dishonesty, contact me or consult the printed policy in the UWO Student Discipline Code, Chapter UWS 14.

3. If you are in need of accommodations for this course, please see the Dean of Students for assistance: <https://www.uwosh.edu/deanofstudents/Accessibility-Center/student-resources>
4. Students are advised to see the following URL for disclosures about essential consumer protection items required by the Students Right to Know Act of 1990: <https://uwosh.edu/financialaid/consumer-information/>.
5. If any substantive changes are made in the course syllabus, such as changes in schedule or assignments, notification will be provided in a timely manner and a revised syllabus will be made available.