CS 235AM, Mobile Application Development: Android
Spring 2017

Sections	CRN 43837 & 43673 (online)	**Credits**	4
Classroom	Building 19, room 132	**Day & Time**	M, W 8:00–9:50

Instructor	Brian Bird	**Office**	Building 19, Room 152
Office Phone	541–463–3024	**Office Hours**	M–Th 2:00 – 3:00pm
E-mail	birdb@lanecc.edu		

Course Description
This course introduces students to applying object oriented programming to mobile application development and the Android System Devlopment Kit. Cross-platform mobile app developpment will be done using the Mono .NET framework and the Xamarin IDE.

Learning Outcome
Design the core logic for a mobile software application. Write, debug, and test the code for the core logic for an application. Design User Interfaces for two platforms. Integrate the UIs with the core logic on two platforms. Evaluate mobile app designs and architectures in terms of user experience, performance, and maintainability.

Course Content

Technologies

C#	Xamarin Studio	Mono (.NET) framework
Android APIs	AXML (Android declarative UI)	SQLite
Android Emulators	Android SDK	UI Controls & Widgets

Concepts

Cross platform development	Android Activity lifecycle	Cross platform architecture
Mobile UI design	Separation of concerns	Android application architecture
Android Services	Using device resources	Native apps vs. Web apps

Skills

Use Xamarin Studio to write, compile and run C# Android applications.
Test and debug Android applications using the Android emulator and on Android devices.
Design and create Android UIs using AXML and the Xamarin Studio UI designer.
Design and create apps that consume web services
Use Android device resources such as the file system, GPS, and camera in your apps
Read and write data using SQLite in your apps

Learning Resources

Texts

There is no textbook required for this class. We will be using resources provided on the Xamarin Web site: http://developer.xamarin.com/guides/cross-platform/getting_started/introduction_to_mobile_development/

Optional textbooks:

Murach, Joel, Murach's Android Programming (2nd Ed). 2015. Murach. ISBN 978-1-890774-93-6
Reynolds, Mark. *Xamarin Mobile Application Development for Android.* 2014. Packt.

Course Web Site

http://classes.lanecc.edu is the address for the LCC Moodle site. Course learning materials and activities will be managed through the course Moodle page.

Computers and Tablets

Computers with Xamarin Studio, Visual Studio, and the Xamarin Android Player are provided in the classroom. There are also Google Nexus 7 tablets with Android available for student use in the classroom. Students may also bring their own computers and Android phones or tablets for use in class.

Computers with the software required for the course are also available to all students in the CIT Main Lab on campus.

You may install all the software needed for this course on your own computer. On a Windows PC, you must have Windows 7, 8.1 or 10. On a Mac you need at least OS-X 10.10 (Yosemite). All of the software used in the class is available for free download.

Software and licenses

Xamarin Studio will be used as the IDE (Integrated Development Environment) in class and in the CIT Main Lab. Xamarin Studio is free and can be downloaded from http://xamarin.com/download.

The Xamarin.Android license is free for individuals and small teams.

GenyMotion is an optional Android emulator that you may want to use. You can download for free (select the version "for individuals") from https://cloud.genymotion.com/page/launchpad/download/

Visual Studio Community Edition (free) or Visual Studio Professional (or higher) may also be used for Android application development Xamarin extensions. The professional edition is available to CIT students. It can be obtained by downloading it from Microsoft Imagine Premium or students may borrow installation DVDs from the CIT computer lab.

Microsoft also provides an Android Emulator for Visual Studio which is faster and easier to use than the emulator in the Android SDK. It requires Windows 10.

Learning Activities

Lab Assignments

These are programing projects that you will do on your own- either using the computers and software in the CIT lab, or using your own computer and software. Students will submit both a *beta* version and a *release* version of the software solution for each lab assignment. The *beta* version should be essentially complete. "Complete" means having working code that fulfill all the core requirements for the assignment, but it does not need to be bug free. The *release* version should fulfill all requirements and be bug fee.

Code Reviews

Students will be paired with a code review partner. The partners will evaluate each-other's coding practices as well as the functionality of each software solution. Students will use input from the code review to revise their code prior to submitting the *release* version of their software solution.

Quizzes

Weekly quizzes are given over the reading and exercises in the textbook as a way to focus students on the most important concepts in textbook chapters covered.

Term Project

The requirements for the term project will be presented during the first week of class.

Weekly Learning Activities

- *Tuesday by 11:55 pm*
    - *Finish this week's reading and take the Quiz*
    - *Complete a code review of last week's lab app for your partner*
- *Saturday by 11:55 pm*
    - *Send the beta version of this week's lab app to your code review partner*
    - *Submit the release version of last week's lab app*

Assessment and Grading

Specific grading criteria will be applied to each of the labs, quizzes, and exams you will be working on in this class.

The table below summarizes the percentage of your total grade given for each assessment task:

Learning & Assessment Activities	Number	Total Percentage
Labs (release version)	9	40%
Code Reviews	9	10%
Quizzes	9	10%
Term Project	1	40%

Letter grades for the course will be determined by the following percentages:

	-		+
A	90 - 91	92 – 97	98 - 100
B	80 - 81	82 – 87	88 - 89
C	70 – 71	72 – 77	78 - 79
D	60 - 61	62 – 67	68 - 69
F	Below 60		

Late Work

- Grades for code reviews will be reduced by 20% if either the beta version or code review are submitted late.
- Grades for lab release versions submitted after the due date will be reduced by 10%
- Quizzes and exams cannot be taken after the due date. Plan ahead! Exceptions will only be made for illness or emergency situations.

Academic Honesty

While students are encouraged to discuss labs and to use each other as resources, each student is responsible for his/her own work. In other words you can help each other, but you can't copy any part of someone else's work. The end product must be each student's own individual work.

Attendance

Attendance is not graded, but to succeed, you must engage in the lectures and participate in class activities either online or in the classroom.

No Show Drop

The college's "no show, drop" policy requires that: during the first week, on-campus students must physically attend at least one class session. Online students must complete at least one activity (a quiz or assignment) otherwise the student will be dropped from the class.

Academic Calendar for Spring Term 2017

Term begins	4/3/17
Last day to receive refund	4/19/17
Spring Conference, College closed	5/5/17
Last day for schedule changes	5/26/16
Memorial Day holiday	5/29/16
Finals week	6/6/16 - 6/11/16

Accessibility and Accommodations

To request accommodations contact the Center for Accessible Resources at (541) 463-5150 or AccessibleResources@lanecc.edu

Please be aware that any accessible tables and chairs in this room should remain available for authorized students who find that standard classroom seating is not usable.

CS 235AM, Mobile Application Development: Android
Spring 2017

Table 1: Tentative Schedule

Week	Topics	Reading	Assignments	Due
1 4/2 – 4/8	Intro to the Xamarin mobile platform. Android application development. Android project resources. C# new and review topics	Intro to Mobile Development Setting up the Developer Tools Intro to Android Development Android Resources	Reading quiz 1 Lab 1: Hello Android • Beta • Code review • Release	4/3 4/8 4/11 4/15
2 4/9 – 4/15	Multi-screen applications Intent objects and LaunchMode C# new and review topics	Multiscreen Apps	Reading quiz 2 Lab 2: Multi-screen apps: Poke-a-screen, 99 Bugs • Beta • Code Review • Release	4/11 4/15 4/18 4/22
3 4/16 – 4/22	More on the Activity life-cycle Saving and restoring Activity state	Activity Lifecycle	Reading quiz 3 Lab 3: Activity lifecycle app: Random Quotes, Who said it quiz, • Beta • Code Review • Release	4/18 4/22 4/25 4/29
4 4/23 – 4/29	Displaying data in lists	ListViews and Adapters	Reading quiz 4 Lab 4: ListView app – Tide table v1 • Beta • Code Review • Release	4/25 4/29 5/2 5/6
5 4/30 – 5/6	UI Layouts and Orientation	User Interface (Section on Layouts)	Reading quiz 5 Lab 5: layout and orientation app – Pig game v1 • Beta • Code Review • Release	5/2 5/6 5/9 5/13

6 5/7 – 5/13	UI Fragments	Fragments ActionBar UI Element ?	Reading quiz 6 Lab 6: Fragments app – Pig game v2 • Beta • Code Review • Release	5/9 5/13 5/16 5/20
7 5/14 – 5/20	Data access with SQLite Read Ch. 3 *Brian will be at Google I/O this week*	Managing SQLite Data	Reading quiz 7 Lab 7: SQLite Database – Tide table • Beta • Code Review • Release	5/14 5/20 5/23 5/27
8 5/21 – 5/27	Consuming web services	Intro to Web Services	Reading quiz 8 Lab 8: Web Service – Tide table • Beta • Code Review • Release	5/23 5/27 5/30 6/3
9 5/28 – 6/3 *Memorial Day*	Using Android device resources Geolocation *No class on Monday 5/29*	Maps and Location	Reading quiz 9 Lab 9: Geolocation app – Tide table • Beta • Code Review • Release	5/30 6/3 6/6 6/10
10 6/4 – 6/10	Work on term projects		Term Project: beta Code Review	6/5 6/7
11 6/11 - 6/17	Present term projects		Term project: Release version	6/12