



COMP 2601 A Winter 2021

MENU

HOME ([index.html](#))

Course Outline ([2601winter2021outline.html](#))

Lecture Schedule, Topics ([2601schedule.html](#))

TA's, Office Hours ([2601office_hours.html](#))

Instructor: Louis Nel (<http://www.scs.carleton.ca/%7Eldnel>)

Prof. Nel's Lecture Notes ([notes](#))

Resources ([2601resources.html](#))

Assignments ([assignments](#))

Tutorials ([tutorials](#))

USEFUL LINKS:

culearn (<http://www.carleton.ca/culearn>)

ANDROID:

developer.android.com (<https://developer.android.com/develop/index.html>)

android classes API (<https://developer.android.com/reference/classes.html>)

java 8 API (<https://docs.oracle.com/javase/8/docs/api/>)

ANDROID -Kotlin:

Kotlin sandbox

(<https://play.kotlinlang.org/koans/Introduction/Hello,%20world!/Task.kt>)

Kotlin -syntax and doc (<https://kotlinlang.org/docs/reference/basic-syntax.html>)

Kotlin -CLI (<https://kotlinlang.org/docs/tutorials/command-line.html>)

IOS:

Swift (Apple) (<https://developer.apple.com/swift/resources/>)

developer.apple.com (<https://developer.apple.com/documentation/>)

apple developer forum (<https://forums.developer.apple.com/welcome>)

© L.D. Nel 2021

COMP 2601 Mobile Applications

COURSE OUTLINE

Lectures/Tutorials	Monday, Wednesday 11:30-1:30 Online. There will not be synchronous lectures. Official class time will be used for Quizzes and student demo's of completed work. You need to be prepared to attend a class-time event if needed.
Tutorials	-same as lectures
Instructor	Louis Nel (http://www.scs.carleton.ca/~ldnel)
TA's	TBA

Calendar Description:

Development of applications for mobile environments taking advantage of gesture-based input and using location and presence services. Topics include introduction to low-level network services and mobile platforms, description of architectural patterns, principles of mobile development and interaction styles for network service usage.

Prerequisite(s): COMP 1601.

Lecture/lab four hours a week.

Course Description:

The course covers the principles involved in the design and implementation of mobile applications typically on the Android platform (Java) and IOS platform (Swift). The course will focus on the application frameworks, typical patterns, network interaction and data storage and exchange.

Topics:

The follow are the topics we covered in the last offering and will be adjusted and updated as the course proceeds.

- Multi-threaded application development on Android and IOS
- Platform agnostic data representations: XML and JSON
- Platform agnostic network protocols and databases (SQLite)
- Use UI widgets.
- App state: representation, accessing and updating
- App communication mechanisms and patterns: services
- Typical programming patterns and decoupling: e.g.Reactor pattern
- Gestures

IMPORTANT NOTE ABOUT SOFTWARE AND COMPUTERS:

This course will be taught as a lab style course where there will be short lecture introductions followed by lab exercises during each scheduled lecture time. Lab exercises will relate to the homework assignments.

This is a "bring your own device" course where students are REQUIRED to come with a laptop computer capable of running the course software: current MacOS (Catalina, or Big Sur), Android Studio 4.1.x and the latest version of XCode. Typically a Macbook with up-to-date MacOS (Catalina, or Big Sur).

Android-based exercises will require an up-to-date Android Studio IDE and Java 1.8 JDK (which runs on all platforms: Windows, Mac, Linux). IOS based exercises will require Apple's XCode IDE and use the Swift programming language. This will require a Mac with latest OS (High Sierra). All the development software is available free of charge. (Android is open source, iOS is proprietary but Apple makes its development tools available free of charge.

Both environments implement simulators for running apps, and they've gotten better in recent years, you will enjoy things more if you do some testing on a real device (android phone/tablet, iphone/ipad).

Textbook and Notes:

The resources section of the course website will list recommended texts for various topics. These texts are not mandatory but references will be made to their contents. The recommended books are available in electronic form. We suggest you get the recommended texts if you don't find online resources sufficient.

Online resources will be posted in the resources section of the course web site and in the individual lecture schedule/topics section. It is expected that the resources will be updated and modified as the course progresses. Students will be expected to contribute to the list of helpful resources.

Course Material Copyright Notice:

We remind you that lectures and course materials, including power point presentations, outlines, code examples, and similar materials, are protected by copyright. The professor is typically the exclusive owner of copyright and intellectual property of the course materials unless otherwise noted. You may take notes and make copies of course materials for your own private (educational) use. You may not and may not allow others to reproduce or distribute lecture notes and course materials publicly for commercial purposes without my express written consent.

About sample code:

We often post sample code on the course web site to accompany lecture content or to use as a starting point for exercises and assignments. Often the sample code is intentionally taken from a recommended text, or online source, so that you can be referred to that source for further explanation. Because of this you need to respect the copyright of those sources as explained below.

You are free to use whatever sample code we post on the course web site as the starting point for your own work intended for submission as course assignments. Assignment submission uploaded to culearn are considered private and not published to the world at large. You may NOT however publish the sample code to the world at large. For example, using sample code from a copyrighted source in a public GitHub repository IS a copyright violation. If you are going to post any code to a public repository like GitHub make sure it is all your own work. GitHub has become a defacto repository used in much software development, however its free accounts do not allow private repositories at this time. Therefore anything placed in a free account is published to the world at large and should NEVER contain content that could represent a copyright violation.

Tutorials:

Class lectures/tutorials are compulsory. There will be weekly programming exercises to be completed and demonstrated in class and count for a large part of your course mark. You must ensure you demonstrate your work to the TA or Prof. before you leave the session to get credit for your work.

Assignments:

We will be using electronic submission of assignments using the new CULearn system culearn. (<https://culearn.carleton.ca>) Electronic submission enforces strict deadlines. Only assignments submitted through culearn will be accepted for marking. No assignments will be accepted late or directly by email or in other forms. TA's are instructed not to accept assignments directly.

Teaching Assistants:

Since the TA's will be there at every lecture/tutorial they will not hold other office hours. Some assignments might require demos to the TA's which will be scheduled as needed.

Marking Scheme:

IMPORTANT: This course is traditional done with no lectures and many demonstrated deliverables. We will make extensive use of screen-capture videos for you to demonstrate your creations. You will need to find an efficient way to record these and they need to have sound as well so you can talk us through your demo. On the Mac the built in Quicktime app is probably the easiest way to record these but any screen capture video recorder would do. When you submit your work you can either submit the video or a link to where the video can be viewed (e.g. YouTube).

deliverable	value	comment
Class Tutorial Exercises	45%	<p>We will drop the worst two tutorial marks (you get 2 free ones)</p> <p>There will be a tutorial roughly each week so expect 10-12 tutorials. A new tutorial will be posted each week (and my consist of several posted parts to complete).</p> <p>Your results from the tutorial exercises will be posted to culearn along with a screen capture video demonstrating how your code works and explaining why you believe the requirements have been met.</p> <p>Your exercise will be given a mark of 0,1 or 2 as follows.</p> <p>0 no submission (missing either the code or the demo video). 1 submission that is partly complete. 2 submission that meets all the requirements.</p>
Assignments	20%	<p>3 assignments equally weighted. Assignment submissions will include the code project and a screen capture video with sound demonstrating your work.</p> <p>Count BEST 2/3 assignments. (That is, you get one free one.)</p>
Project Assignment	15%	<p>Build an app of your own choosing in either Android or IOS. You will need to submit an proposal for the app that must be approved. This will be the last course assignment.</p> <p>Again submission will be code and a screen capture video.</p> <p>Proposal will be due when assignment #3 is due. Everyone must do this -there is no free one)</p>
Quizzes	20%	<p>Up to 4 Quizzes synchronous during class-time (datesTBA). We will drop the worst quiz mark -you get one free one</p>
Final Exam	0%	<p>There is NO final exam in this course</p>

Missed assignments, tutorials and tests for medical and other reasons: You may miss up to 2 tutorials, and 1 assignment and 1 test for medical, compassionate, or other reasons. If you miss more than that a mark of 0 will be used for the missed items when the final grade is computed. We will NOT collect doctor's notes for missed work, but if you miss more than the allowed number a mark of 0 will be used for the missed work.

IMPORTANT: If you wish to appeal a mark (assignment, tutorial, or midterm) you must make the appeal within 10 days of the mark being posted on culearn. After this we will not be obliged to entertain appeals or change marks.

Collaboration is encouraged but cheating, or copying, is not allowed. You may work together and consult but any work you hand in must be your own and judged to be unique. Any two assignments judged to be too similar will both receive a grade of 0, and will be handled as a formal academic offence -see calendar for details.

The TA's will be using the Stanford MOSS (Measure of Software Similarity) system to detect copied work (plagiarism). There is no "statute of limitations" on detecting copying meaning we will run these tests throughout the term and may deduct marks from work that was graded previously.

NEW UNIVERSITY POLICY REGARDING ACADEMIC INTEGRITY

Academic Integrity: Minimum penalties for offences starting 6 January 2020:

First offence, first-year students (< 4.0 cr)	Final grade reduction of one full grade (e.g., A- becomes a B-, if that results in an F, so be it)
First offence, everyone else	F in the course
Second offence	One-year suspension from program
Third offence	Expulsion from the University

Course Web Page:

As well as being announced in class, all important information, such as course news, assignments, TA hours, instructor office hours, will be available on the course web page at <http://www.scs.carleton.ca/~ldnel/2601winter2018> (<http://www.scs.carleton.ca/~ldnel/2601winter2018>). The course website is password protected. If you are registered in the course you can find your userid/password by logging into your culearn account. It is your responsibility to check this web page frequently for new information and announcements. Paper copies of outlines and assignments will not be provided.

Undergraduate Academic Advisor

The Undergraduate Advisor for the School of Computer Science is available in Room 5302C HP; by telephone at 520-2600, ext. 4364; or by email at undergraduate_advisor@scs.carleton.ca. The undergraduate advisor can assist with information about prerequisites and preclusions, course substitutions/equivalencies, understanding your academic audit and the remaining requirements for graduation. The undergraduate advisor will also refer students to appropriate resources such as the Science Student Success Centre, Learning Support Services and Writing Tutorial Services.

IMPORTANT UNIVERSITY POLICIES

Student Academic Integrity Policy

Every student should be familiar with the Carleton University student academic integrity policy. A student found in violation of academic integrity standards may be awarded penalties which range from a reprimand to receiving a grade of F in the course or even being expelled from the program or University. Some examples of offences are:

plagiarism and unauthorized co-operation or collaboration. Information on this policy may be found in the Undergraduate Calendar, Section 14, Page 59.

Plagiarism

As defined by Senate, "plagiarism is presenting, whether intentional or not, the ideas, expression of ideas or work of others as one's own". Such reported offences will be reviewed by the office of the Dean of Science.

Unauthorized Co-operation or Collaboration

Senate policy states that "to ensure fairness and equity in assessment of term work, students shall not co-operate or collaborate in the completion of an academic assignment, in whole or in part, when the instructor has indicated that the assignment is to be completed on an individual basis". Please refer to the course outline statement or the instructor concerning this issue.

Academic Accommodation

You may need special arrangements to meet your academic obligations during the term because of disability, pregnancy or religious obligations. Please review the course outline promptly and write to your instructor concerning any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. It takes time to review and consider each request individually, and to arrange for accommodations where appropriate. Please make sure you respect these timelines particularly for in-class tests, midterms and final exams, as well as any change in due dates for papers. You can visit the Equity Services website to view the policies and to obtain more detailed information on academic accommodation at <http://carleton.ca/equity/accommodation>

Students with Disabilities Requiring Academic Accommodations Register with the Paul Menton Centre for Students with Disabilities (PMC) for a formal evaluation of disability-related needs. Documented disabilities could include but are not limited to mobility/physical impairments, specific Learning Disabilities (LD), psychiatric/psychological disabilities, sensory disabilities, Attention Deficit Hyperactivity Disorder (ADHD), and chronic medical conditions. Registered PMC students are required to contact the PMC, 613-520-6608, every term to ensure that I receive your Letter of Accommodation, no later than two weeks before the first assignment is due or the first in-class test/midterm requiring accommodations. If you only require accommodations for your formally scheduled exam(s) in this course, please submit your request for accommodations to PMC by the deadlines published on the PMC website: <http://www2.carleton.ca/pmc/new-and-current-students/dates-and-deadlines/>

Religious Obligation

Write to your instructor concerning any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details visit the Equity Services website http://www.carleton.ca/equity/accommodation/student_guide.htm

Pregnancy Obligation

Write to your instructor concerning any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details visit the Equity Services website http://www.carleton.ca/equity/accommodation/student_guide.htm

Medical Certificate

The following is a link to the official medical certificate accepted by Carleton University for the deferral of final examinations or assignments in undergraduate courses. To access the form, please go to <http://www2.carleton.ca/registrar/forms/>