CIS*2750 Software Systems Development and Integration
Winter 2019

UNIVERSITY of GUELPH

School of Computer Science

1 INSTRUCTIONAL SUPPORT

Section 01-02

Instructor: Dr. Denis Nikitenko
Office: Reynolds 3322
Email: cis2750@socs.uoguelph.ca

Office hours: By appointment only. Details will be posted on the course website. **Note**: Weekly hours will vary during the semester to better accommodate student demand and instructor schedule.

Pre-requisites: CIS*2430, CIS*2520
Credit Weight: 0.75

Teaching Assistants:
Jordan Evans
Pooja Joshi
Daniel Kantor
Vicky Mohammad
Dalton Polhill

Email: cis2750@socs.uoguelph.ca

Office Hours: See lab times below

Timetable

Lectures:
Section 1: Monday, Wednesday, 05:30PM - 06:50PM, ROZH 102
Section 2: Tuesday, Thursday, 11:30AM - 12:50PM, MACN 113

Lab times:
- Mon 11:30AM - 01:20PM
- Tues 08:30AM - 10:20AM, 07:00PM - 08:50PM
- Wed: 08:30AM - 10:20AM, Wed 07:00PM - 08:50PM
- Thur 08:30AM - 10:20AM

All labs will be held in THRN 2418

Unless otherwise stated, lab times will be used for advising and consulting with the Teaching Assistants. Students do not have to come to their assigned lab, and may attend any lab if they have questions.

2 LEARNING RESOURCES

2.2 Course Website

Course material, news, announcements, and grades will be regularly posted to the CIS*2750 website which can be found at https://moodle.socs.uoguelph.ca. You are responsible for checking the site regularly.

- Lecture Information: Selected notes will be posted on the course website as instructors have time to make them available. You are expected to take your own notes during lecture.
- Labs and Tutorials: Selected tutorial and lab materials will be available on the course website.
- Assignments: Assignment descriptions are found on the course website. Assignments are submitted via the course website.

2.3 Recommended Textbook

Beginning Linux Programming, by Richard Stones and Neil Matthew, Wrox Press Ltd.
Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 by Robin Nixon (5th Ed.)

2.4 Calendar Description

This course introduces techniques and tools used in the development of large software systems. Students learn methods for organizing and constructing modular systems, manipulating files, introductory interface design, and use of databases. Software tools for managing projects, database connectivity, configuration management, and system application programmer interfaces are also covered.

The Academic Calendars are the source of information about the University of Guelph's procedures, policies and regulations which apply to undergraduate, graduate and diploma programs: http://www.uoguelph.ca/registrar/calendars/index.cfm?index

3 ASSESSMENT

3.1 Dates and Distribution

Assessments
Assignments (4, 60%) - Assignment 0 - Jan 17 (1%) - Assignment 1 - Jan 28 (17%) - Assignment 2 - Feb 15 (17%) - Assignment 3 - Mar 13 (17%) - Assignment 4 - Mar 27 (8%) Final exam (40%) - Wednesday, April 8, 11:30AM - 01:30PM. Room TBA. Bonus clicker participation - 5%

Students found committing **clicker fraud** will be handled according to the policy on academic misconduct, and are liable to forfeit their bonus mark. Possession of multiple clickers will be taken as *prima facie* evidence of fraud involving the actual owners of the clickers.

Students **must** complete the SOCS Academic Integrity Unit (see Section 6 below) before submitting their first assignment. Assignment submissions will not be graded until a student has completed the SOCS Academic Integrity Unit.

Use of Git in this course is not required. If you wish to use a Git repo for this course for your own use, you may do so. However, if you do, your repo **must** be private. Having any code for this course hosted in a public repo will constitute a breach of academic integrity. In addition, it is strongly recommended to use the GitLab repo hosted by SoCS (gitlab.socs.uoguelph.ca).

3.1.1 Final grade calculation

In order to pass the course, you must submit your assignments on time and pass **both** the assignment and the test components of the course.

If you fail either the assignment component or the test component, your course grade will be the grade for the failed component. For example, if you get 25/60 (43%) on the assignments and 35/40 (75%) on the test, you fail the course and your course grade is 43. If you fail both the assignment and the test components, your course grade will be the lowest grade of the two failed components. For example, if you get 25/60 (43%) on the assignments and 10/40 (25%) on the tests, you fail the course and your course grade is 25.

Your bonus clicker grade will be added only if you pass <u>both</u> the assignment component and the test component. So if , for example, you get 25/60 (43%) on the assignments, 30/40 (75%) on the test, and 4/5 for the clickers, the clicker marks will <u>not</u> be applied, because you have failed the assignment portion.

On the other hand, if you get 42/60 (70%) on the assignments, 30/40 (75%) on the tests, and 4/5 for the clickers, the clicker grade is applied, and your total course grade will be 76%.

Please note that the maximum course grade cannot exceed 100%. If the clicker mark results in a grade over 100%, your course grade will be capped at 100%.

3.2 Course Grading Policies

Development environment: Systems and software are provided in the labs for use in assignments. Students who choose to develop their assignments on other systems and/or with other versions of software are fully responsible for ensuring compatibility with the lab systems for marking purposes.

Individual work: Assignments must be carried out by individuals; there are no group projects in this course.

Extensions: Due date extensions are only declared for catastrophic reasons such as server failures and snow closures. Assignments and midterms in other courses are not grounds for extensions.

Late Assignments: All assignments are due at 9am on the due date, unless explicitly stated otherwise. Late assignments will be accepted for 12 hours after the deadline and penalized at 2% per hour. Assignments that are more than 12 hours late will not be accepted, and will automatically receive the grade of **zero** (0). See below for compassionate exceptions.

Compiler errors/warnings: Program code which does not compile will not be accepted for marking and a grade of **zero** (0) will be assigned. Code that does not compile "clean" will lose marks for compiler warnings. Code **must** compile and run on the SoCS servers.

Submission errors: Failure to submit assignments correctly (e.g., incorrect file names, faulty/missing makefile, etc.) will result in a mark penalty.

Resubmission: Resubmission of assignments is not normally permitted. In exceptional circumstances the instructor may allow assignment resubmission.

Regrades: Students may request a regrade of an assignment if the marker has made an error in grading. The original submission will be entirely regraded and a new mark will be assigned. It is possible for a mark to go down, go up, or remain unchanged as a result of a regrade. Students must request a regrade via Regrade link on the course website within 5 calendar days of receiving the assessment grade.

Missed Assessments: If you are unable to meet an in-course requirement due to medical, psychological, or compassionate reasons, please make an appointment to meet your course instructor. Please see below for specific details and consult the undergraduate calendar for information on regulations and procedures for Academic Consideration: http://www.uoguelph.ca/registrar/calendars/undergraduate/current/c08/c08-ac.shtml *Note:* ***There are no makeup assignments.***

Accommodation of Religious Obligations: If you are unable to meet an in-course requirement due to religious obligations, please email the course email address within two weeks of the start of the semester to make alternate arrangements. See the undergraduate calendar for information on regulations and procedures for Academic Accommodation of Religious Obligations:
http://www.uoguelph.ca/registrar/calendars/undergraduate/current/c08/c08-accomrelig.shtml

3.3 Course Learning Outcomes

1. Practice effective strategies for learning to use new software frameworks, methodologies, and programming languages.
2. Construct software components that adhere to provided specifications.
3. Integrate software components written in different programming languages to create a software system.
4. Design and implement software libraries.

5. Demonstrate systematic quality assurance and software testing techniques.
6. Employ a database or a file-based back end to implement data storage for an interactive program.
7. Identify and apply appropriate human-computer interaction techniques to the design of a graphical user interface.

4 TEACHING AND LEARNING ACTIVITIES

4.1 Lecture and Lab Schedule

Lectures	Lecture Topics	**Assignments**
Week 1	Course Introduction; XML	
Week 2	Using and designing software libraries	Assignment 0
Week 3	Additional C concepts and tools	
Week 4	Testing and debugging	Assignment 1
Week 5	Standards in computing	
Week 6	Scripting languages; multi-language applications	Assignment 2
Break week		
Week 7	Web programming and scripting languages: client side	
Week 8	Web programming and scripting languages: server side	
Week 9	Introduction to database design and SQL	Assignment 3
Week 10	Working with databases and SQL	
Week 11	Graphical user interfaces: design and evaluation	Assignment 4
Week 12	Special/advanced topics	

4.4 Important Dates

Monday January 6: First day of class
Monday February 17 - Friday February 21: No Classes (reading week)
Friday April 3: Last day of classes

5 ROLES AND RESPONSIBILITIES

5.1 Communication & Email Policy

Please use lectures, lab sessions, and the website discussion forum as your main opportunities to ask questions about the course. Questions that are specific to your particular situation may be emailed to cis2750@socs.uoguelph.ca and will be answered by one of the instructional team. Extremely private communication should be conducted in person by making an appointment with the course instructor.

Major announcements will be posted to the course website. It is your responsibility to check the course website regularly. As per university regulations, all students are required to check their <mail.uoguelph.ca> e- mail account regularly: e-mail is the official route of communication between the University and its students.

5.2 Recording of materials

Presentations which are made in relation to course work—including lectures—cannot be recorded or copied without the permission of the presenter, whether the instructor, classmate or guest lecturer. Material recorded with permission is restricted to use for that course and may not be posted on any public space unless further permission is granted.

5.3 Instructor's Role and Responsibility to Students

The instructor's role is to develop and deliver course material in ways that facilitate learning for a variety of students. Selected notes will be made available to students on the course website but are not intended to be stand-alone. During lectures, the instructor will expand and explain the content of notes and provide example problems that supplement posted notes. Scheduled classes will be the principal venue to provide information and feedback for exams and assignments.

5.4 Students' Learning Responsibilities

Students are expected to take advantage of the learning opportunities provided during lectures, labs and help sessions. Students, especially those having difficulty with the course content, should also make use of other resources recommended by the instructor. Students who fall behind due to illness, work, or extra-curricular activities are advised to keep the instructor informed as early as possible. This will allow the instructor to recommend extra resources in a timely manner and/or provide consideration if appropriate.

6 ACADEMIC INTEGRITY

The University of Guelph is committed to upholding the highest standards of academic integrity and it is the responsibility of all members of the University community faculty, staff, and students to be aware of what constitutes academic misconduct and to do as much as possible to prevent academic offences from occurring. University of Guelph students have the responsibility of abiding by the University's policy on academic misconduct regardless of their location of study; faculty, staff and students have the responsibility of supporting an environment that discourages misconduct. All students who take a SOCS course must pass the **Academic Integrity Self Test.**

For educational purposes, instructors impose conditions on assignments that may limit students' permission to collaborate with others or to utilize external sources (including, but not limited to, software, data, images, text, etc.). Any permitted utilization must be done with proper references. Aiding and abetting is a punishable offence; students must be careful not to help others commit offences by giving out solutions or providing to access computer accounts. Instructors may use automated tools to detect possible cases of academic misconduct.

Please note: Whether or not a student intended to commit academic misconduct is not relevant for a finding of guilt. Hurried or careless submission of assignments does not excuse students from responsibility for verifying the academic integrity of their work before submitting it. Students who are in any doubt as to whether an action on their part could be construed as an academic offence should consult with a faculty member.

Use of Git in this course is not required. If you wish to use a Git repo for this course for your own use, you may do so. However, if you do, your repo **must** be private. Having any code for this course hosted in a public repo will constitute a breach of academic integrity. In addition, it is strongly recommended to use the GitLab repo hosted by SoCS (gitlab.socs.uoguelph.ca).

The Academic Misconduct Policy is detailed in the Undergraduate Calendar:
http://www.uoguelph.ca/registrar/calendars/undergraduate/current/c08/c08-amisconduct.shtml

The SOCS Academic Integrity Unit:
http://moodle.socs.uoguelph.ca/course/view.php?id=2 Login with your central login credentials. The key to use is "imhonest".

7 ACCESSIBILITY

The University promotes the full participation of students who experience disabilities in their academic programs. To that end, the provision of academic accommodation is a shared responsibility between the University and the student.

When accommodations are needed, the student is required to first register with Student Accessibility Services (SAS). Documentation to substantiate the existence of a disability is required, however, interim accommodations may be possible while that process is underway.

Accommodations are available for both permanent and temporary disabilities. It should be noted that common illnesses such as a cold or the flu do not constitute a disability.

Use of the SAS Exam Centre requires students to book their exams at least 7 days in advance, and not later than the 40th Class Day.

More information: www.uoguelph.ca/sas