CIS 3760 – Software Engineering (Winter 2017)

Lectures: MWF 11:30am-12:20pm, Room XXX
Laboratories: **Section 1:** Wed 3:30-5:20pm, Room XXX
Section 2: Thu 9:30-11:20am, Room XXX
Section 3: Thu 2:30-4:20pm, Room XXX

Instructor: Prof. Stacey D. Scott, Associate Professor, School of Computer Science
Office: MacLachlan 226; Email: stacey.scott@uoguelph.ca; Phone: 519-824-4120 ext. 54153

Contact: Use email function in CourseLink or make an appointment to meet outside of class time.
(For emergency issues, email, call, or contact in person)

Teaching Assistants: XXXXXXX and YYYYYYY
Email: ta3760@socs.uoguelph.ca

Course Website: http://courselink.uoguelph.ca (Login with Central login ID and password)
The course website will be used to provide:

- ***informational materials*** (e.g., course notes, assignment handouts, course updates, course contacts)
- ***electronic drop boxes*** for course deliverable submissions
- ***course discussion boards*** for asking course-related questions and discussing issues related to course material

Course Communications:
To facilitate timely and accurate communication between people with busy schedules (you, professor, TAs) we will use both electronic and face-to-face communication:

1. For ***lecture, project, or general course questions***, post to the *"CIS 3760 Course Questions" Discussion board* on the course website. This will be monitored by your TAs and Prof. Scott. Or talk to Prof. Scott or your TAs in person before/after lectures, or during laboratories.
2. For ***personal issues***, email (use the email function in CourseLink) or talk in person to Prof. Scott. The best time to chat with her is immediately before/after class.

When communicating with fellow students, TAs or the professor, **you are expected to use professional practices, including respectful, clear messages, especially via email**. See "Email Policy" section below.

Required Text:
Freeman, E., Freeman, E., Sierra, K. & Bates, B. (2004). Head First Design Patterns. Cambridge: O'Reilly (eText available at O'Reilly Online: shop.oreilly.com/product/9780596007126.do)

Required Lab Manual:
Kniberg, H. (2014). Scrun and XP from the Trenches, 2nd edition. Stockholm: C4Media. Free Downloadable e-Book available at https://www.infoq.com/minibooks/scrum-xp-from-the-trenches-2.

Recommended Texts:
Bruegg, B. and Dutoit, A.H. (2010). *Objected-Oriented Software Engineering: Using UML, Patterns, and Java, 3rd edition*. Toronto: Prentice Hall. (eText available at Pearson Online: www.pearsonhighered.com/program/Bruegge-Object-Oriented-Software-Engineering-Using-UML-Patterns-and-Java-3rd-Edition/PGM58934.html)

Pressman, R.S. and Maxim, B.S. (2015). *Software Engineering: A Practitioner's Approach, 8th edition*. New York: McGraw Hill Education. (eText available at McGraw Hill Education Online: www.mheducation.com/highered/product.M0078022126.html?searchContext=software+engineering)

Calendar Description:

This course is an examination of the software engineering process, the production of reliable systems and techniques for the design and development of complex software. Topics include object-oriented analysis, design and modeling, software architectures, software reviews, software quality, software engineering, ethics, maintenance and formal specifications.

Prerequisites: CIS 2750, (CIS 3110 recommended); Restriction(s): CIS 3200

Course Objectives:

The main goal of CIS 3760 is to familiarize students with the complexities involved in software engineering projects and to provide tools and techniques for planning for and managing successful team-based software projects. Achieving this goal requires learning how to *understand the problem* and *design effective solutions,* as well as learning how to apply these analytic and design skills using specific *engineering modeling and design tools* (e.g. unified modeling language (UML) and Java programming language). Additionally, this course aims to provide students with the appropriate project management, communication, and professional skills to apply their software skills in a professional software engineering context. By the end of the course, you should be able to:

a. *Explain* and *Use* the procedures involved in software design and development;
b. *Select* and *Use* appropriate modeling techniques for planning and designing a software project;
c. *Analyze* problem specifications and *Integrate* into appropriate software requirements and design models;
d. *Select* and *Use* appropriate software architectures and design patterns for common software design problems;
e. *Identify* sources of risk in software projects, and *Select* and *Use* risk mitigation strategies in solution designs;
f. *Select* and *Use* appropriate testing methods for verifying and validating a software project;
g. *Describe* effective project management and team behaviours and dynamics and *Evaluate* your contributions.

The significant challenge in successful software engineering is learning the potential sources of complexities involved in software projects and how to manage those complexities. Whether you are interested in becoming a software developer, software engineer, or project manager in the software field, this course will supply you with a range of practical as well as theoretical knowledge related to software engineering.

Grading Scheme:

The following table describes the grading scheme, and the corresponding learning objective.

			Learning Objectives						
	Marking Scheme		**a**	**B**	**c**	**d**	**e**	**f**	**g**
Evaluated through:	**Indiv.**	**Team**	Software Design Concepts	Software Modeling	Software Reqs.	Design Patterns	Software Risks	Software Testing Concepts	Team Dynamics
Midterm Exam	**20%**		x	x	x	x			
Final Exam	**20%**		x	x	x	x	x	x	
Design Project	**10%**	**50%***							
Team Contract		*5%*							x
Sprint 0 Milestone		*10%*	x		x		x		x
Sprint 1 Milestone		-	x	x	x	x	x	x	x
Sprint 2 Milestone		*10%*	x	x	x	x	x	x	x
Sprint 3 Milestone		*10%*	x	x	x	x	x	x	x
Final Project Demo		*10%*	x	x		x		x	x
Project Post-Mortem		*5%*	x	x		x		x	x
Accountability Report	*10%*								x
Course Total	**50%**	**50%***							

***NOTE: Due to the significant amount of group work required for the team deliverables, the instructor reserves the right to apply a fraction of the grade to an individual group member without sufficient evident of contribution to the group effort. See Required Coursework Policies below for more detail.**

Estimated Weekly Course Commitment (15-18 hours per week for this 0.75 credit course):

- 3 hours lectures
- 3-4 hours course preparation (e.g. readings, academic tasks)
- 2 + 2-3 hours laboratories + lab preparation (lab activities will typically support your design project)
- 5-6 hours design project (this includes in-laboratory activities, which will all relate to your design project)

The University of Guelph expects students to spend 10-12 hours per week on a 0.5 credit course, and more (~15-18 hrs/wk) for 0.75 credit course. This time commitment represents student workload rather than contact hours. In CIS 3760, the main learning vehicle is a team-based software design project. Class members must make a minimum commitment of 5-6 hours per week to the design project, in addition to relevant laboratory activities. It is recommended that project teams meet at least 1 hour a week outside of lecture / lab sessions to discuss their project status.

Policies on Required Coursework:

Late or Missed Assignments/Deliverables:

- **0%** will be given for any late or missed assignments or deliverables. **No extensions** will be given after a submission date.

Missed Exam:

- **0%** will be given for a missed **midterm exam**. If accommodation is warranted under the "Accommodation" policy below, the midterm marks will be redistributed to the final exam (i.e. the final exam will then be worth 40% instead of its original 20%).
- If you miss the **final exam**, you must talk with your academic counsellor and follow University of Guelph policies on this matter. Prof. Scot cannot do anything in this situation.

Accommodation:

- If you are unable to meet a course requirement due to medical, psychological, or compassionate reasons, please make an appointment to meet Prof. Scott. Please see below for specific details and consult the undergraduate calendar for information on regulations and procedures for Academic Consideration.
- If you are unable to meet a course requirement due to religious obligations, please email Prof. Scott **within two weeks of the start of term** to make alternate arrangements. See the undergraduate calendar for information on regulations and procedures for Academic Accommodation of Religious Obligations.

Marking of Team-Based Deliverables:

- By default, marks for all team-based deliverables will be applied to all group members. However, the **instructor reserves the right to assign individual students a fraction of the total group mark** on any specific team-based deliverable without sufficient evidence of that student's contribution to the group effort. The onus is on each individual student to provide evidence of their individual contribution to the team effort. The instructor reserves the right to apply fractional grading retroactively at the end of the term to one or more team deliverables if the **accountability report** of an individual team member lacks sufficient evidence of contribution to specific or overall team efforts.
- If a team feels that lack of effort by one or more group members is negatively impacting the group's ability to meet the progress or outcome expectations for the course, the situation **MUST be identified to the course instructor or TAs PRIOR to any impacted deliverable** to be considered for accommodation in deliverable grading. Otherwise, default grading criteria will be applied (e.g., project progress and outcomes will be assessed on the basis of a full team effort). Group members must submit a written request for accommodation, PRIOR to the deliverable deadline, detailing the situation. Note that submission of an accommodation request is not in and of itself approval.
- **It is the responsibility of ALL team members to ensure the accuracy and quality of all aspects of submitted team-based deliverables**. Therefore, any academic integrity offences arising from a team-based deliverable will impact ALL group members.

Failed Midterm + Final Exam:

- You must achieve a combined passing grade (50% or more) on the midterm and final exams to pass the class. If you do not achieve a combined passing grade on both exams, your final grade will be your combined midterm and final exam grade.

Use of Laptops / Personal Computing Devices during Lectures:
Use of laptops, tablets, and other personal computing devices during lectures will be at the instructor's discretion. The devices can be highly distracting to both the student using the device and to other students in the class. Use of these devices must be restricted to course related activities (e.g., note taking, viewing course materials, etc.). Distractions resulting from the use of these devices may result in a student being asked to leave the classroom.

Email Policy:
All course email should follow the following guidelines:

- Use the email function in CourseLink when emailing Prof. Scott or the TAs. This authenticates your email and also includes the course number, in order to give some context to the recipient for your email.
- Address your email appropriately (i.e. "Dear Prof. Scott / Professor / Corey /…")
- Sign your email with your first and last name, and your student number. If you have a nickname, include that also, e.g., Yu-Ling (Betty) Chang. (Tip: you can set-up your CourseLink profile to use an email signature that is automatically included at the end of your CourseLink emails)
- Use professional and respectful language. Email containing crude or coarse language will not be answered.
- Allow 24-48 hours for a response to your email.
- Email should be used for brief questions that can be answered quickly. Please make an appointment to see Prof. Scott or a TA, or talk to them before/after lecture or during the lab, for detailed discussions/explanations.

Roles and Responsibilities

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.

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.

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.

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. Login with your central login credentials. The course key is “imhonest”.

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.

Accessibility
The University of Guelph is committed to creating a barrier-free environment. Providing services for students is a shared responsibility among students, faculty and administrators. This relationship is based on respect of individual rights, the dignity of the individual and the University community's shared commitment to an open and supportive learning environment. Students requiring service or accommodation, whether due to an identified, ongoing disability, or for a short7term disability should contact Student Accessibility Services (SAS) as soon as possible.

For more information, contact SAS at 519782474120 ext. 56208 or email: csd@uoguelph.ca or see the website: http://www.uoguelph.ca/csd/

Course Timetable/Content Schedule (tentative; check CourseLink for updates)

Wk – Dates	**Topic(s)** *[more details available on CourseLink]*	**Deliverables / Exams** *[submit Deliverables to CourseLink]*
1–Jan 9, 11, 13	Introduction, Software Engineering Processes	Jan 11: Background Survey (indiv.)
2–Jan 16, 18, 20	Agile Software Development	
3–Jan 23, 25, 27	Project Management and Communication	
4–Jan 30, Feb 1, 3	Modeling with UML (Common Diagrams)	Feb 1/2 (in-lab): Team Contract & Sprint 0: Project Initiation (team)
5–Feb 6, 8, 10	Architectural Patterns	
6–Feb 13, 15, 17	Design Patterns	Feb 15/16 (in-lab): Sprint 1 Review (team)
7–Feb 20-24	***Reading Week, No Classes***	
8–Feb 27, Mar 1, 3	Software Quality and SQ Assurance, Software Metrics	Mar 1/2 (in-lab): Midterm Exam
9–Mar 6, 8, 10	Design Reviews	Mar 8/9 (in-lab): Sprint 2 Review (team)
10–Mar 13, 15, 17	Software Testing	
11–Mar 20, 22, 24	Maintenance and Re-Engineering	Mar 22/23 (in-lab): Sprint 3 Review (team)
12–Mar 27, 29, 31	Software Security	
13–Apr 3, 5, 7	Formal Methods	Apr 5/6 (in-lab): Sprint 4, Final Project Demo (team) Apr 7: Individual Accountability Report (indiv.)
Final Exam Period		Final Exam