

ECE 5010: Software Design

Instructor **Andrew Vardy**
E-mail **av at mun dot ca**

Office Hours **Mondays and Wednesdays 1:00 – 2:00 on Webex (via Brightspace)**

E-mail to arrange an appointment outside of office hours.

Note: The Wednesday office hour may have to be cancelled a few times throughout the semester to accommodate Departmental/Faculty meetings.

Website **<https://online.mun.ca/d2l/home/389120>**

CALENDAR ENTRY:

5010 Software Design (same as the former ENGI 5895) examines the development process: requirements analysis, design, iterative development, design documentation; an introduction to the Unified Modelling Language: use cases, class diagrams and sequence diagrams; an introduction to software design patterns: creational patterns, structural patterns and behavioral patterns; object oriented, modular decomposition. The course includes a major design project.

CR: the former ENGI 5895

LC: 25 lecture hours per semester

LH: six 3-hour sessions per semester

OR: meetings with project supervisor as required

PR: ECE 4400 or the former ENGI 4892

CREDIT VALUE: 3 credit hours

COURSE TYPE: compulsory (Computer Engineering)

CONTENT CATEGORIES: (expressed as %, no category can be $0 < c < 25$)

| Math | Natural science | Complementary Studies | Engineering Science | Engineering Design |
|------|-----------------|-----------------------|---------------------|--------------------|
| | | | | 100 |

COURSE DESCRIPTION:

In this course, students will combine technical mastery with creativity to create complete software applications. The student will learn about modelling software using UML, object-oriented design principles, design patterns, and a selection of software technologies. There will be a major project completed in groups of 2-3.

| | | | |
|------------------|----------------|---------------|---------------|
| SCHEDULE: | LECTURE: | Mon, Wed, Fri | 12:00 – 12:50 |
| | MEETING / LAB: | Thur | 9:00 – 12:00 |

Lectures, meetings, and presentations will take place using Brightspace's Online Rooms unless otherwise specified.

The Thursday morning lab slot will primarily be used for meetings between students and the instructor and TAs. It will also be used as the time slot for student presentations. The lab slot will only be utilized on the dates when these activities are ongoing (see Assessment Schedule).

RESOURCES:

TEXT BOOK

- ***THERE IS NO REQUIRED TEXT BOOK FOR THIS COURSE***

REFERENCES

- Robert C. Martin, Agile Software Development, Prentice Hall, 2003.
- Martin Fowler, UML Distilled, 3rd ed, Addison Wesley, 2004. (Available electronically from the library)
- Erich Gamma, Richard Helm, Ralph Johnson, and Jon Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison Wesley, 1995.
- Craig Larman, Applying UML and Patterns, Prentice Hall, 2005.
- Eckel, B., Thinking in Java, (3rd edition available for free online at <http://www.mindview.net/Books/TIJ>).

MAJOR TOPICS:

- Java programming
- UML
- Software Design principles
- Object-Oriented Design patterns
- Software Engineering processes
- Application of all the above to a major project

LEARNING OUTCOMES:

Course Level Graduate Attribute Focus: Des-D, PA-D, Tools-D

1 A knowledge base for engineering 2 Problem analysis 3 Investigation 4 Design 5 Use of engineering tools

| | LEARNING OUTCOMES At the appropriate milestones, the student will be expected to be able to: | GRADUATE ATTRIBUTES. LEVEL OF COMPETENCE | Methods of Assessment |
|---|---|--|---|
| 1 | Understand and document software designs using appropriate linguistic and visual formalisms and tools. Short version: Document their designs well. | 1.A Knowledge base – Applied 4.A Design – Applied. 5.A Use of tools – Applied. 7.A Communication – Applied. | Project design documents. Presentations, Exam. |
| 2 | Explain the basics of assertion-based design (design by contract). | 1.D Knowledge base – Developed. 4.D Design – Developed. | Exam. |
| 3 | Apply the principles of object-oriented design. | 1.3 Knowledge base – Applied. 4.3 Design – Applied. | Project design documents and code. Presentations. Exam. |
| 4 | Apply major object-oriented design patterns and show familiar with others. | 1.2 Knowledge base – Developed. 4.2 Design – Developed. | Project design documents. Exam. |
| 5 | Manage a modest sized software design project. | 6.D Individual and team work – Developed. 7.D Communication – Developed. 11.I Economics and project management – Introduced. | Project meetings and design documents. |
| 6 | Understand the advantages of sound modularization and well-defined interfaces. | 1.D Knowledge base – Developed. 4.D Design – Developed. | Project meetings and design documents. Exam. |

*Each Graduate Attribute for each learning outcome is rated at a Content Instructional Level of

I=Introduced, D=Developed, or A=Applied.

See www.mun.ca/engineering/undergrad/graduateattributes.pdf for definitions on the 12 Graduate Attributes and the Content Instructional Levels.

ASSESSMENT SCHEDULE:

| | | |
|------------------------|------|---|
| Assignments (2) | 15% | Wed Jan 20, Wed Jan 27 |
| Project | 60% | |
| | 0% | Thurs Jan 21 (Project Ideas Meeting) |
| | 7.5% | Wed Feb 3 (Project Proposal / Use Cases Report Due) |
| | 7.5% | Thurs Feb 11 (Technology Presentation) |
| | 5% | Wed Feb 17 (Initial Deployment Submission) |
| | 10% | Thurs Mar 11 (Design Presentation) |
| | 10% | Tues Mar 23 (First Iteration Submission) Thurs Mar 25 (First Iteration Demo) |
| | 20% | Tues April 6 (Final Submission) Thurs April 8 (Final Presentation) |
| | | |
| Quiz | 5% | Mon Mar 1 |
| Mid-term Exam | 20% | Mon Mar 8 |

A Google calendar with all of the above dates is available here:

<https://calendar.google.com/calendar/u/0?cid=djZncDk2ZjQ1dDUzaGdzZGNudDlxdmEwZ2tAZ3JvdXAuY2FsZW5kYXluZ29vZ2xlMnVbQ>

PROJECT:

This course is focussed around a major project, which is to be completed by groups of 2-3 students. To allow more time to complete the project, lectures will end prior to the quiz on March 1. The project is a group project to be completed in teams of 2 - 3. Note that the performance expectations for groups of 3 will be greater than for groups of 2.

Students can exercise considerable free choice in the selection of their project topic. Prior to the Project Ideas Meeting on January 21, students should have identified their group and have 2 – 3 different project ideas. A list of project ideas from previous years will be provided. However, due to remote delivery the way in which projects are evaluated will differ from previous years. In particular, each group's project should be a desktop application without any hardware or cloud computing dependencies. Your application should be easily portable between different operating systems. For example, you might develop it in Windows but the instructor/TAs should be able to test it on Linux or Mac OS.

You are recommended to use Java for the project but you are free to use another language with object-oriented features (e.g. C++ or Python). However, it must be possible for the instructor and TAs to quickly and easily build your application without having to download or install any additional packages. That is, the application must be easily deployed.

Groups may be asked to complete a **peer review assessment**. If the contributions of the 2 (or 3) students are significantly unbalanced, then marks may be adjusted accordingly. It is therefore important for all team members to strive towards an equal contribution of efforts.

ASSIGNMENTS:

The assignments are to be completed individually or in pairs. You are not required to work with the same people for all assignments and the project. However, once the project teams are formed, you should do your best to work collaboratively with your teammates for the duration of the term.

TOOLS:

We will be using the Java language for the assignments. The Visual Paradigm CASE tool will be introduced for UML diagrams, although you are free to use other software or handwritten diagrams.

ACADEMIC INTEGRITY AND PROFESSIONAL CONDUCT:

Students are expected to conduct themselves in all aspects of the course at the highest level of academic integrity. Any student found to commit academic misconduct will be dealt with according to the Faculty and University practices. More information is available at <http://www.mun.ca/engineering/undergrad/academicintegrity.php>

Students are encouraged to consult the Faculty of Engineering and Applied Science Student Code of Conduct at <http://www.mun.ca/engineering/undergrad/academicintegrity.php> and Memorial University's Code of Student Conduct at <http://www.mun.ca/student/conduct/>.

INCLUSION AND EQUITY:

Students who require accommodations are encouraged to contact the Glenn Roy Blundon Centre, <http://www.mun.ca/blundon/about/index.php>. The mission of the Blundon Centre is to provide and co-ordinate programs and services that enable students with disabilities to

maximize their educational potential and to increase awareness of inclusive values among all members of the university community.

The university experience is enriched by the diversity of viewpoints, values, and backgrounds that each class participant possesses. In order for this course to encourage as much insightful and comprehensive discussion among class participants as possible, there is an expectation that dialogue will be collegial and respectful across disciplinary, cultural, and personal boundaries.

STUDENT ASSISTANCE: Student Affairs and Services offers help and support in a variety of areas, both academic and personal. More information can be found at www.mun.ca/student.