

Course Outline (F2019)

COE528: Object Oriented Eng Analysis and Design

Instructor(s)	Faezeh Ensan [Coordinator] Office: ENG324 Phone: (416) 979-5000 x 4904 Email: fensan@ryerson.ca Office Hours: Tuesdays, 1 pm - 3 pm
Calendar Description	This course deals with the analysis and design of complex engineering systems. In particular, students will be asked to create requirement specifications prior to the design and implementation of such engineering systems. Case studies from software development projects will be used to illustrate the design process. Development of expertise in analyzing, designing, implementing, and testing industrial-quality, reusable software systems. Project work include practice with an object-oriented programming language. (Formerly COE 618)
Prerequisites	COE 318 and COE 428 and CEN 199
Antirequisites	None
Corerequisites	None
Compulsory Text(s):	<ol style="list-style-type: none"> 1. Program Development in Java: Abstraction, Specification, and Object-Oriented Design, Barbara Liskov and John Guttag, 2000, Addison-Wesley, ISBN: 0201657686. 2. Object-Oriented Software Engineering Using UML, Patterns, and Java, Bernd Bruegge and Allen H. Dutoit, 2009, Prentice Hall, Third edition, ISBN 0136061257.
Reference Text(s):	<ol style="list-style-type: none"> 1. Design Patterns Explained: A New Perspective on Object-Oriented Design, Alan Shalloway and James R. Trott, 2004, Addison-Wesley Professional, Second Edition, ISBN 0321247140. 2. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, 1995, Addison-Wesley Professional, First Edition, ISBN 0201633612. 3. Head First Design Patterns, Eric Freeman, Elisabeth Freeman, Kathy Sierra, Bert Bates, 2004, O'Reilly, First Edition, ISBN 0596007124. 4. Pankaj Jalote, Chapter-1 of "A Concise Introduction to Software Engineering", (Undergraduate topics in Computer Science) ISBN-10: 1848003013 ISBN-13:978-1848003019 Edition: 2008
Learning Objectives (Indicators)	<p>At the end of this course, the successful student will be able to:</p> <ol style="list-style-type: none"> 1. Apply Object-Oriented Software Engineering principles and concepts to solve technical problems. (1c) 2. Use the knowledge of object oriented design methodology, design patterns, and UML design tools. (4a) 3. Integrate the existing design patterns into the software design where applicable. (4b) 4. Select the most appropriate design pattern to address a software design problem. (4c) 5. (6b), (11b) 6. Produce course project report using appropriate format. (7a) 7. Demonstrate the project to the Teaching Assistant through oral communication. (7b) <p>NOTE: Numbers in parentheses refer to the graduate attributes required by the Canadian Engineering Accreditation Board (CEAB).</p>
Course Organization	3.0 hours of lecture per week for 13 weeks 2.0 hours of lab/tutorial per week for 12 weeks
Teaching Assistants	Duc-Thuan Vo Salar Razavi Sepehr Bayat Shaina Raza

Course Evaluation	Theory
	Midterm Exam 15 %
	Final Exam 40 %
	Laboratory
	Labs 20 %
	Project 25 %
	TOTAL: 100 %
	<p>Note: In order for a student to pass a course with "Theory and Laboratory" components, in addition to earning a minimum overall course mark of 50%, the student must pass the Laboratory and Theory portions separately by achieving a minimum of 50% in the combined Laboratory components and 50% in the combined Theory components. Please refer to the "Course Evaluation" section for details on the Theory and Laboratory components.</p>
Examinations	Midterm exam in Week 7, 120 minutes, multiple-choice and questions, closed book (covers weeks 1-6). Final exam, during exam period, 3 hours, closed-book (covers weeks 1-13).
Other Evaluation Information	<p>* Students must achieve passing grades in both the theoretical and the laboratory components of the course in order to pass the course.</p> <p>* All the Labs and the project have to be done individually.</p>
Other Information	None

Course Content

Week	Hours	Chapters / Section	Topic, description
1	3		Course Overview and Introduction Understanding Objects in Java. (Chapter 2 of "Program Development in Java: Abstraction Specification and Object-Oriented Design". Sections 2.1-2.8. Lecture notes. Slides).
2-3	6		Abstraction and Decomposition. (Chapter 1, Sections 1.1-1.3 and Chapter 4, Sections 4.1-4.4 of "Program Development in Java: Abstraction Specification and Object-Oriented Design". Lecture notes. Slides).
4	3		Procedural and Data Abstraction. (Chapter 3, Sections 3.1-3.6 and Chapter 5, Sections 5.1-5.10 of "Program Development in Java: Abstraction Specification and Object-Oriented Design". Lecture notes. Slides).
5	3		Modeling with UML. (Chapter 2, Sections 2.1-2.4 of "Object-Oriented Software Engineering Using UML Patterns and Java". Lecture notes. Slides).

6-7	6	Requirements Elicitation. (Chapter 4, Sections 4.1-4.6 of "Object-Oriented Software Engineering Using UML Patterns and Java". Lecture notes. Slides).
8	3	System Analysis. (Chapter 5, Sections 5.1-5.6 of "Object-Oriented Software Engineering Using UML Patterns and Java". Lecture notes. Slides).
9	3	System Design. (Chapter 6, Sections 6.1-6.4 and Information taken from Chapters 7 and 8 of "Object-Oriented Software Engineering Using UML Patterns and Java". Lecture notes. Slides).
10-11	6	Design Patterns. (Information taken from Chapter 8 of "Object-Oriented Software Engineering Using UML Patterns and Java" and Information taken from "Design Patterns Explained: A New Perspective on Object-Oriented Design" and "Design Patterns: Elements of Reusable Object-Oriented Software" and "Head First Design Patterns". Lecture notes. Slides).
12	3	Testing and Debugging. (Chapter 10 of "Program Development in Java: Abstraction Specification and Object-Oriented Design". Sections 10.1 and 10.2 and 10.4 and 10.7. Lecture notes. Slides). Software Development Life Cycle and Methodologies. (Information taken from chapters 15 and 16 of "Object-Oriented Software Engineering Using UML Patterns and Java" and Chapters 1 and 2 of "A Concise Introduction to Software Engineering". Lecture notes. Slides).
13	3	Course Review.

Laboratory/Tutorials/Activity Schedule

Week	Lab	Description
2-3	ENG411	Review Java programming and JUnit (for testing) using NetBeans IDE. (4 marks)
4	ENG411	Procedural Abstraction - Implement and specify procedures with requires/modifies/effects clauses. (3 marks)
5	ENG411	Implement an application using Interfaces and Abstract Classes. (3 marks)
6	ENG411	Data Abstraction – Provide and implement the rep invariant and abstract function for each given class. (3 marks)
7-8	ENG411	UML Modeling - Analyze system requirements of a software system. Design with UML diagrams. Implement and test the system. (7 marks)
9-11	ENG411	Project.

Policies & Important Information:

1. Students are required to obtain and maintain a Ryerson e-mail account for timely communications between the instructor and the students;
2. Any changes in the course outline, test dates, marking or evaluation will be discussed in class prior to being implemented;
3. Assignments, projects, reports and other deadline-bound course assessment components handed in past the due date will receive a mark of ZERO, unless otherwise stated. Marking information will be made available at the time when such course assessment components are announced.
4. Refer to our **Departmental FAQ** page for information on common questions and issues at the following link:
<https://www.ee.ryerson.ca/guides/Student.Academic.FAQ.html>.

Missed Classes and/or Evaluations

When possible, students are required to inform their instructors of any situation which arises during the semester which may have an adverse effect upon their academic performance, and must request any consideration and accommodation according to the relevant policies as far in advance as possible. Failure to do so may jeopardize any academic appeals.

1. **Health certificates** - If a student misses the deadline for submitting an assignment, or the date of an exam or other evaluation component for health reasons, they should notify their instructor as soon as possible, and submit a Ryerson Student Health Certificate AND an Academic Consideration Request form within 3 working days of the missed date. Both documents are available at <https://www.ryerson.ca/senate/forms/medical.pdf>. **If you are a full-time or part-time degree student, then you submit your forms to your own program department or school;**
2. **Religious, Aboriginal and Spiritual observance** - If a student needs accommodation because of religious, Aboriginal or spiritual observance, they must submit a Request for Accommodation of Student Religious, Aboriginal and Spiritual Observance AND an Academic Consideration Request form within the first 2 weeks of the class or, for a final examination, within 2 weeks of the posting of the examination schedule. If the requested absence occurs within the first 2 weeks of classes, or the dates are not known well in advance as they are linked to other conditions, these forms should be submitted with as much lead time as possible in advance of the absence. Both documents are available at www.ryerson.ca/senate/forms/reobservforminstr.pdf. **If you are a full-time or part-time degree student, then you submit the forms to your own program department or school;**
3. **Academic Accommodation Support** - Before the first graded work is due, students registered with the [Academic Accommodation Support office](http://www.ryerson.ca/studentlearningsupport/academic-accommodation-support) (AAS - www.ryerson.ca/studentlearningsupport/academic-accommodation-support) should provide their instructors with an Academic Accommodation letter that describes their academic accommodation plan.

Academic Integrity

Ryerson's [Policy 60 \(the Academic Integrity policy\)](#) applies to all students at the University. Forms of academic misconduct include plagiarism, cheating, supplying false information to the University, and other acts. The most common form of academic misconduct is plagiarism - a serious academic offence, with potentially severe penalties and other consequences. It is expected, therefore, that all examinations and work submitted for evaluation and course credit will be the product of each student's individual effort (or an authorized group of students). Submitting the same work for credit to more than one course, without instructor approval, can also be considered a form of plagiarism.

Suspensions of academic misconduct may be referred to the Academic Integrity Office (AIO). Students who are found to have committed academic misconduct will have a Disciplinary Notation (DN) placed on their academic record (not on their transcript) and will normally be assigned one or more of the following penalties:

1. A grade reduction for the work, ranging up to and including a zero on the work (minimum penalty for graduate work is a zero on the work);
2. A grade reduction in the course greater than a zero on the work. (Note that this penalty can only be applied to course components worth 10% or less, and any additional penalty cannot exceed 10% of the final course grade. Students must be given prior notice that such a penalty will be assigned (e.g. in the course outline or on the assignment handout);
3. An F in the course;
4. More serious penalties up to and including expulsion from the University.

The unauthorized use of intellectual property of others, including your professor, for distribution, sale, or profit is expressly prohibited, in accordance with Policy 60 (Sections 2.8 and 2.10). Intellectual property includes, but is not limited to:

1. Slides
2. Lecture notes
3. Presentation materials used in and outside of class
4. Lab manuals
5. Course packs
6. Exams

For more detailed information on these issues, please refer to the [Academic Integrity policy](https://www.ryerson.ca/senate/policies/pol60.pdf) (<https://www.ryerson.ca/senate/policies/pol60.pdf>) and to the Academic Integrity Office website (<https://www.ryerson.ca/academicintegrity/>).

Important Resources Available at Ryerson

1. [The Library](https://library.ryerson.ca/) (<https://library.ryerson.ca/>) provides research workshops and individual assistance. Inquire at the Reference Desk on the second floor of

the library, or go to library.ryerson.ca/guides/workshops

2. [Student Learning Support](https://www.ryerson.ca/studentlearningsupport)(<https://www.ryerson.ca/studentlearningsupport>) offers group-based and individual help with writing, math, study skills and transition support, and other issues.