



**03 60-322 Object-Oriented Software
Analysis/Design
School of Computer Science,
University of Windsor
Fall 2010**

Instructor

Rahmatullah Hafiz

PhD (c), School of Computer Science, University of Windsor

Contact info: Office - Erie 3146, Phone - (519) 253-3000 ex: 3794

E-mail - hafiz@uwindsor.ca

Course Logistics

Duration : 6th July 2010 to 12th August 2010

Meetings : Mondays and Wednesdays, 2:30 – 5:20, Erie 2127

Consultation Hours

The **instructor** can be reached at the **Java lab (3rd Floor, Erie)** on **Fridays from 12:00 pm to 2:00 pm**. Student e-mails, originated from University of Windsor accounts, will be replied as early as possible. Please mention '60-322' in the subject line of your e-mail.

One **graduate teaching assistant**, Satish Panigrahi (panigra@uwindsor.ca), will be available to assist students in **Java lab** on **Mondays from 12:00 pm to 3:00 pm**. He will be also assisting the instructor and the students during in-class activities.

Course Website

The University of Windsor's **online learning environment CLEW** will host the course website. Please login at <https://clew.uwindsor.ca/xsl-portal>.

The course website in CLEW should be considered as the primary communication medium between students, and the instructor. This site will provide related resources for each lecture, assessment methods and grades, rubrics, support for feedback and announcements. These resources will be uploaded before, during or right after the class meetings. Students need to visit the site on regular basis.

Prerequisites

Official course pre-requisites - 60-212, 60-254, 60-256

Participating students should know or should be able to learn quickly the fundamental concepts of Object Oriented Programming. They also should be able to program fluently in Java, a major Object Oriented programming language that is the primary tool to teach and learn Object Oriented programming in the School of Computer Science of the University of Windsor.

Course Description

The University of Windsor calendar description of 60-322:

“This course builds on the knowledge of object-oriented programming, data structures, and systems programming. Students are introduced to object-oriented software analysis and design concepts (such as cohesion and coupling), and design practices currently used in industry (such as design patterns and refactoring). These concepts and practices will be discussed through case studies and programming exercises.”

Learning Outcomes

Participants who successfully complete this course will be able to:

1. Recognize benefits of **iterative developments** for analyzing and designing.
2. Determine and record productive **use cases** of problem domains.
3. Apply **inspection** and **elaboration** phases of **Unified Process** for requirement analysis and for designing solution-models of given problem scenarios.
4. Apply **UML notations** in analysis and design for
 - a. Identifying and documenting concepts, associations, and attributes in conceptual models.
 - b. Identifying and documenting system events, responsibilities and behaviours in domain and design models.
 - c. Using UML tools that are available in public domain.
5. Transform design models and artifacts into **program code** in an object-oriented programming language - Java.
6. Verify and develop implementation by agile practices like Test Driven Development and Refactoring using testing tools e.g. JUnit.
7. Assign responsibilities and design collaborations between objects by identifying , applying and implementing appropriate **design patterns** from
 - a. General Responsibility Assignment Software Patterns (GRASP).
 - b. A few of key “Gang of Four” and related patterns.
8. Collaborate with others to analyze, design, implement, evaluate and document development components whenever required.

Teaching and Learning Environment

Students will have the opportunity to meet with the instructor, with other students, and with teaching assistant through the following:

- Two three-hour-long interactive class meetings every week – Tuesdays and Thursdays, from 2:30 pm to 5:20 pm.
 - In-between delivering lectures, the instructor will explain difficult concepts, illustrate the application of the concepts, and demonstrate problem-solving with examples.
 - The instructor and the G.A will assist students to participate in small-group or individual problem solving or discussion exercises for a short period, which will reflect their understanding of the currently-discussed topic.
- One-to-one consultations with teaching staffs as required. Discussion and interaction with other students through the CLEW discussion board.
- The course web-site where each week's lecture schedule will be accompanied by a set of reading materials (e.g., pointer to a chapter of the textbook, or an online tutorial etc.), handouts and lecture notes. If necessary, the handouts will be distributed during the lecture.
 - The CLEW site will also contain resources for assignments, which will include example solutions, links for downloading tools and sample code, suggestions and tutorials for better practical outcomes.

Teaching Evaluation

The official Student Evaluation of Teaching (**SET**) forms will be administered during the last week of the class schedule.

The instructor will request the participating students to provide written feedback during the time period of the course e.g. mid-course feedback. The students will be provided a set of questions related to the course content, teaching, learning and assessment practices. Students are also welcome to provide the instructor any form of written or verbal feedback in order to enhance their learning experience.

Required Reading

1. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition)

By Craig Larman

University of Windsor students can read the entire book **free of charge** through the University of Windsor's Leddy Library's **Safari Tech Books Online** service. The direct link to the portal is :

<http://ledauth.uwindsor.ca/authdist/check?url=http://safari.oreilly.com>

After logging in with your UWinID and password, search the book in the Safari site.

Alternatively, students can buy a new or used copy of the book from the University of Windsor bookstore, or from any other legitimate sources.

2. The course website (CLEW) will announce other open-source internet resources or book-chapter/papers if they are required for any specific lecture or assignment.

Test, Assignments and Assessments Methods and Rational

Assessment methods are divided into couple of diverse methods to reflect the fact that different students learn in different ways. Below is a tentative breakdown of the evaluation scheme:

Final Exam	: 30%
Midterm Exam	: 20%
Take-home Assignments	: 35% (7 *5)
In-class Activities	: 15% (10*1.5)

The assessment approaches for this course are diverse, and are constructed to motivate and excite the students to actively participate in the learning process. The students will be constructively challenged to solve problems, and to explain understanding and synthesizing of concepts in forms of writing, discussions, collaborations, designing artifacts and programming. These activities will require demonstration of critical thinking and creative analysis. Evaluation of these activities, assignments and tests will contain constructive feedback so that students will have chances to identify and correct their mistakes. To obtain marks for any form of missed exams, tests and required activities one needs to provide legal form of written evidence e.g., doctor's note, employer's note etc. The students' success will be reflected by how closely they have achieved the intended learning outcomes of the course, and they will have chances to demonstrate their expertise of the learning outcomes through the assignments and tests. The assignments and tests will be evaluated by following pre-announced rubrics.

In-class Activities and Take-Home Assignments

- A set of problems will be developed for realization of theories and applications of concepts, which students are required to solve in different forms of assignments. In-class and take-home assignments will reflect students' understanding of currently-discussed conceptual topics, and skills needed to utilize them in practice by analyzing and designing models, and/or implementing them using Java. These assignments are open-book so that students can spend more time on the assigned problems, rather than memorizing concepts and theories.
- The **take-home assignments** will be graded based on pre-defined grading rubrics. The grading for in-class activities would depend on how well students have been attempting to reach the solution by understanding the requirement specifications, and by applying currently-discussed topics/concepts.
- The **in-class activities** will take place during the scheduled class hours, and will be distributed throughout the class period. Except the midterm, there will be 11 class meetings; hence there will be 11 in-class activities. Students can miss at most 3 of such activities (i.e., 3 classes) and still can obtain full mark. If anyone misses any of the required 8 activities (and the take-home assessments) for a valid reason (e.g., for medical reason with legitimate doctor's note) then alternative measures of assessment will be taken by the instructor.

Midterm and Final Exams

- Key topics of the course, that are covered in in-class and take home assignments, will cumulatively reflect in the **time-constrained midterm and final exams**, which will be held according to pre-assigned date and time. The primary format of these exams will be designed following in-class and take-home assignments, and will be discussed with students beforehand so that there will be no hidden surprises. The grading rubrics will be discussed with students as well before exams.
- Students who wish to appeal an exam mark should wait until they have received their final grade at the end of the semester and then follow the procedure outlined in the University Calendar for the appeal of that grade. No remarking of the final exam will be undertaken unless a formal grade appeal is submitted at the end of the semester after the student has received the final grade for the course. Numerical errors in adding marks on the final exam will be corrected when identified.
- If anyone misses the **final exam** for a valid reason (e.g., for medical reason with legitimate doctor's note) then **alternative measures will be taken by the instructor**. There will be **no make-up exam for the midterm exam**.

Alternative Measure of Assessment

For the missed in-class activities and take-home assignments, the instructor solely will make decisions to arrange a discussion session with students to evaluate their understanding of concepts, or assign new assignments to test their skills, or assign more weight to their other forms of assessment. For a **missed final exam**, the instructor will arrange **make-up test**, and for **missed midterm exam**, instructor will **assign more weight** on the final exam.

To avoid being penalized for missing exams, assignments and in-class activities due to illness, students and their doctors need to fill up the new **Student Medical Certificate** form. The form can be downloaded from the course website in CLEW.

The **final letter grade** will be calculated from the raw scores using the following table:

≥ 93	<100	A+	≥ 63	<67	C
≥ 86	< 93	A	≥ 60	< 63	C-
≥ 80	< 86	A-	≥ 57	< 60	D+
≥ 77	< 80	B+	≥ 53	< 57	D
≥ 73	< 77	B	≥ 50	< 53	D-
≥ 70	< 73	B-	≥ 35	< 50	F
≥ 67	< 70	C +		<35	F-

Estimated Division of Learning Hours

Task	Duration
Reading of class-related resources and preparation	3 hour
Class meetings + in-class activities	6 hours
Take home assignment	3 hours
Weekly TOTAL (excluding revision for the midterm and final exam)	12 hours

Note that this is only a suggested learning hour's breakdown. An individual may need more or less than this breakdown.

Teaching Methods and Philosophy

My overall teaching goal for this course would be to help you to understand and apply some well-practiced concepts and theories of Object Oriented Analysis and Design. This is a difficult topic to teach and to learn; primarily because there are too many practices exist in real life scenarios. I will be attempting to explain difficult and well-documented concepts with examples, demonstrations, and by making topics relevant to your own purposes.

I believe that unless you are involved, then there are not many ways for you or me (as your instructor) to determine the level of your learning. Hence, I will be incorporating your active participation practices during the class meetings. I believe that as a third year CS student, you will bring and share enough positive knowledge and experience in the classroom for everyone's betterment. You will be actively engaged in learning often by participating/collaborating with your peers. I will challenge you constructively to solve problems, and to explain understanding and synthesizing of concepts in the form of writing, discussions, collaborations, designing artifacts and programming. These activities will require demonstration of critical thinking and creative analysis by requiring you to apply them into scenarios that are not always obvious.

I also believe that if I provided you constructive feedback about your work (whether about what you could have done to avoid a mistake or to improve your suggested solutions) then that would positively contribute improving your next practice/task. I will design your exam questions in a way that they will require you to memorize less but to understand deeply.

At the end, I want you leave the course with a satisfactory learning experience so that you will be confident enough to identify you strengths regarding Object Oriented Analysis and Design for realistic purposes.

Tentative Assessment Schedule

Assignments/Assessments	Important Date	Comments
Eight in-class activities <ul style="list-style-type: none">- Individual or small group tasks- to test the currently-discussed topic- $2.5 * 8 = 20$ marks- There will be total 11 activities; one student can skip at most 3 activities to get full mark.	Starting from 1 st week to 6 th week, except the 6 th class when we will have the midterm exam.	In-class submission. Assessment would be formative . Students' participation and attempt to address problems are the key criteria for evaluation. ** The GA will assist the instructor with grading.

<p>seven take home assignments</p> <ul style="list-style-type: none"> - Individual or small group problem solving tasks related to currently-discussed topic of OOAD - 7 * 5 = 35 marks 	<p>Assignments are to be submitted before the class in CLEW.</p> <p>13th July'10– Assign. 1 due 20th July'10 – Assign. 2 due 29th July'10 – Assign. 3 due 5th August'10 – Assign. 4 due 12th August'10 – Assign. 5 due</p>	<p>Mostly CLEW submission. (The assignments may potentially be linked with each other in order to form a bigger task.) ** The instructor will be responsible for grading.</p>
<p>One mid-term exam</p> <ul style="list-style-type: none"> - Written test - Time constrained - Would require solving problems related to the concepts and skills practiced in the assignments and in-class activities up to the 20th of July, 2010. - 20 marks 	<p>22nd of July, 2010 (Erie 2127)</p>	<p>Summative assessment (with constructive feedback) that will be conducted followed by pre-defined rubric. ** The instructor will be responsible for grading.</p>
<p>One final exam.</p> <ul style="list-style-type: none"> - Written test - Time constrained - Would require solving problems related to all concepts and skills practiced in the assignments and in-class activities. - 30 marks 	<p>21st August, 2010 (TC 100)</p>	<p>The test will cover most of the topics and related learning outcomes. Summative assessment that will be conducted followed by pre-defined rubric. ** The instructor will be responsible for grading.</p>

Tentative Course Schedule
(Subject to change - depending on the course context)

Class Meetings	Topics	Relevant Outcomes
<p>1. 6th July 2010</p>	<p>Introduction to Object Oriented Analysis Design. Overview of Iterative and Agile development and management. Intro to UML and UML tools. Inception phase of UP. Requirements analysis.</p>	<p>1,2,4,8</p>
<p>2. 8th July 2010</p>	<p>Creating Use Cases. Inception to Elaboration of UP. System Behaviour Model – Domain Model, SSD. Operation Contracts.</p>	<p>2,3, 4,8</p>

	Requirements to Design. UML Interaction Diagram, Class Diagram.	
3. 13 th July 2010	Assignment 1 Due before the class in CLEW. Review Architecture—subsystems, layers, model-view separation.	3,4,8
4. 15 th July 2010	Designing Objects with Responsibilities, Object interactions, Patterns (GRASP 1 to 5) - 1	3,4,5,8
5. 20 th July 2010	Design to implementation. Related Programming topics. Assignment 2 Due before the class in CLEW. Mid-course feedback for the instructor.	4,5,6,8
6. 22 nd July 2010	Midterm Exam Midterm Exam will be held at the beginning of the class for 2 hours.	
7. 27 th July 2010	Concepts of Extreme Programming (XP). Test-driven Development and Refactoring.	5,7,8
8. 29 th July 2010	Designing Objects with Responsibilities, Object interactions, Patterns (GRASP 6 to 9) - 2 Assignment 3 Due before the class in CLEW.	5,7,8
9. 3 rd August 2010	Designing for Visibility, Activity and State Machine Diagram and Modeling. More Patterns.	5,6,8
10. 5 th August 2010	Gang of Four and related design patterns – 1 Assignment 4 Due before the class in CLEW.	5,7,8
11. 10 th August 2010	Gang of Four and related design patterns – 2	5,7,8
12. 12 th August 2010	Review of key topics. Assignment 5 Due in CLEW.	
17 th August 2010	No class meeting.	
13. 21 st August 2010	Cumulative Final Exam at TC 100 (21st August 2010, Saturday. 3:30 pm to 6:30 pm)	Most of them

Expectations from Students

- **Attendance at all lectures is expected.** Please come to the class regularly and on time. If you have to leave early then please inform me in advance. All of the concepts covered would require active participation (20% of the total assessment will take place during class meetings). You should read the related resources ahead of lectures. In the class, you may require to discuss and share ideas/solutions with peers for in-class assignments.
- You must **attempt to complete take-home assignments by the suggested completion dates** and attend the exams during the pre-schedule dates. Otherwise, please provide legitimate written evidence for missing written tests and deadlines for assignments.

Answers and related feedback for assignments will be made available so that you can review the mistakes and discuss any confusion with the teaching staffs. You will need to provide your picture ID during the final exam, and you may not leave the exam room before 30 minutes into the exam.

- **The University of Windsor allows students to take a course only twice.** If a student fails to get the specified minimum grade in a required course after two attempts, the student may be asked to withdraw from the program. Prepare to the best of your ability to do well in the course.
- **Mutual respect between the teaching staffs and students, and also among all students** will be fundamental in the learning process. Students are required to provide feedback to other students so that they can attempt to learn from each other. Students are also expected to provide constructive feedback to the instructor in various forms throughout the term in order to make their learning experience satisfactory.

Access to Resources

- A distributed-computing network, accessible through a graphic interface provided by X-terminals and Sunrays located in various labs that are supported by state-of-the-art parallel-processing Sun UNIX compute servers. The computing network can be accessed through Sunrays and X-terminals in the Java-lab and the X-lab on the 3rd floor of Erie Hall (open from 9am to 10 pm weekdays), and in an X-lab in room 305 in the West wing of the Leddy Library (open from 9am to 10pm weekdays and 12 noon to 5pm Saturday and Sunday), through PCs in an open laboratory in the Computer Centre, or, for students who have the necessary equipment, from home via the Internet.
- The Java (TM) compiler is installed in the servers so that students can access them through any terminals from school and home. For your personal computers, you need to download respective compiler from <http://java.sun.com/j2se/1.4.2/>. You can use any IDE that are available for Java. One useful IDE that comes with many helpful tools is Eclipse and can be downloaded from <http://www.eclipse.org/>.
- The UML and testing tools and their sources will be discussed during respective class meetings.
- Students who have difficulties with the network, or who have questions concerning access from home, etc., should first of all contact the Computing Consultants in IT Services (go to <http://web4.uwindsor.ca/its/students> for contact information). If you are having a problem with a Computer Science system or account, you can contact any of the **technicians in the School**. The technicians are located in room 3104 Lambton Tower (go to http://cs.uwindsor.ca/ma_154 for contact information).
- **The Academic Writing Centre (AWC)** offers free writing assistance for University of Windsor undergraduates and graduates. Please go to <http://www.uwindsor.ca/awc/> for appointments, location and contact information.
- **Student Disability Services (SDS)** provides a variety of services and supports to students with documented disabilities (including: learning disabilities, attention deficit/hyperactivity disorder, acquired brain injuries, vision, hearing and mobility

impairments, chronic medical conditions and psychiatric issues), who have registered with SDS. If you have, or think you may have a disability, you may wish to visit SDS to learn how best to meet your academic goals. The SDS office is located in Room 117, Dillon Hall, (519) 253-3000 ext. 3288 or online at <http://uwindsor.ca/disability/>

Policy on Plagiarism

The instructor and the teaching assistant for this course will put a great deal of effort into helping students to understand and to learn the material in the course. However, any form of cheating will not be tolerated. The instructor and teaching assistant will report any suspicion of cheating to the Director of the School of Computer Science. If sufficient evidence is available, the Director will begin a formal process according to the University Senate Bylaws. The instructor will not negotiate with students who are accused of cheating but will pass all information to the Director of the School of Computer Science. The following behaviour will be regarded as cheating (in addition to other acts that would normally be regarded as cheating in the broad sense of the term):

- Copying assignments (program code or other kinds) from web/book or from other students or allowing another student to copy an assignment from you and present it as their own work.
- Copying from another student during a test/exam.
- Referring to notes, textbooks, etc. during a test/exam when text/exam is not declared as “open book”.
- Talking or distracting others or communicating with another student in any way or not sitting at the pre-assigned seat during a test/exam.
- Attempting to have access to assignments/ program code or the exam/test paper prior to the assessment.
- Asking a teaching assistant for the answer to a question during an exam/test.
- Modifying answers after they have been marked.
- Any other behaviour which attempts unfairly to give you an advantage over other students in the grade-assessment process.
- Refusing to obey the instructions or the officer in charge of an examination.

Please checkout the **Academic Integrity** at the University of Windsor’s website for further information at <http://www.uwindsor.ca/aio/>.