Western Science

Department of Computer Science

CS 4471A & cs9549A

Software Design and Architecture

Course Outline: Fall 2024

Logistics and Instruction:

[redacted]	[redacted]
Day and Classtime	Friday 10.30 AM – 1.30PM
Instructor	Nazim H. Madhavji
TA	Haoran Wei
Emails to the instructor	<my last name> <<at>> gmail <<DOT>> com
Emails to the TA	<h> then <last name> <<at>> uwo <<DOT>> ca
Office Hours (instructor):	[redacted]
Office Hours (TA):	By email appointment.

Sessional Dates

Term begin	Thu 7th September, 2023.
First class	Fri 8th September, 2023.
Non-instructional Day	Fri 29th September, 2023.
Last class	Fri 8th December, 2023.
Term end	Fri 8th December, 2023.

Students' comments: 0_Student_Comments.doc

Noteworthy points: v2_Noteworthy_Points.doc

Quick Access:

Back to the Top of the document

Course Description

The objective in this course is to share knowledge and to learn about technology on the subject “software architectures”.

A software architecture is an abstract representation of a software system, filtering out what is traditionally considered detailed design- and implementation-level issues (such as: algorithm, design patterns, data representation and coding) and highlighting such aspects as the overall system structure, inter-relationships and interaction among the elements in the system structure, and other “run-time” or “off-line” properties of the system (such as reliability, performance, portability, inter-operability, etc.).

Whereas the *requirements* of a system generally state the functionality and the expected behaviour (quality) of the deployed system, the *architecture* of the system describes how the desired functions and qualities are to be (or have been) implemented. An architecture sets bounds for lower-level design of the system and gives a technical context to future enhancements of the system.

For a large and complex (software-intensive) system, its architecture is an essential means for controlling and evolving the system. Among many other uses of the architecture include: development team organisation, overall system test planning, training of new developers, release planning, defect analysis, system component reuse, size/cost/effort estimation, and vision sharing with various stakeholders.

Among the resources required to create a high quality system architecture include: requirements of the system; organisational context; domain and technical knowledge and experience; existing system and its architecture (if any); appropriate stakeholders; and architecting notations, methods, techniques, tools and processes.

Just as there are patterns for the design of a system, there are patterns for the architecture of the system. This helps in the selection of the entire, or parts of an,

architecture from a set of choices; increasing reuse, cost reduction, and quality development.

Work Components

In this course, the key work components include:

1) **Individual work**: Weekly assigned **readings** and creating **individual summaries**.

2) **Groupwork**: Creating a **technical domain model** from knowledge gained in the course.

3) **Groupwork: Topic research**: literature search on a specific topic, preparing a presentation, and in-class presentation.

    a. Briefly, each team will identify a focused architectural topic (approved by the instructor), search related literature, and present their topic to the class.

4) **Groupwork**: In-class **group summary creation**, and **Q&A and discussions**.

5) **Groupwork: Project**.

    a. Briefly, the project is about creating microservices, architected based on desired quality attributes, on a Cloud platform (your choice). Projects are carried out in teams that will give progress/final reports, demo, and supporting presentation.

6) **Individual work:** Takeaways at the end of the class as directed.

7) **Individual -- Show & Tell:** Individuals can "Show and Tell" about subject matter that they think will be of interest to the rest of the class. Conditional to approval from the instructor. Bonus marks.

Class Operation – IMPORTANT TO NOTE:

- There will be initial, introductory, set of lectures in the class. The class schedule will indicate the exact dates. Beyond the initial set of lectures, formal lectures will only be given if the instructor sees this as appropriate.

- By default, **all the students** are expected to read the assigned weekly literature (typically specific book chapters) and extract key elements therefrom and fill the given **individual templates**. These readings are **mandatory**. The completed templates will need to be submitted every week. They will be assessed.

- In the class, **groups will create their respective group templates** based

on individual member templates. These group templates will be posted "live" to OWL. These will then be used to drive "live" Question-Answer (QA) interactions in the class. These (QA) interactions will be assessed in real-time as "class participation" marks.

- *Active learning:* Please note that this is not a course where students receive lectures throughout the term. Rather, it is an active course where students collaborate within their groups to produce target results. If still not sure about the expectations, please consult the instructor in person for advice.
    - In this course, you will need to read and understand the weekly assigned material and extract pertinent information before you come to the class.
    - In the class, in your designated group, you will then collectively create your group's vision for the assigned material.
    - Then, in the class, you will be expected to respond to the questions posed and discuss related material. You will be assessed on the quality of your responses.
    - Clearly, what you will get out of this class is heavily dependent upon what you will put into it.
        - Not following this advice will result in wasting your time and in poor learning experience and outcome.

Equipment needed in the class:

- You will need your **laptop** for discussing the individual templates and creating group templates among the members of your group.

Learning Outcomes:

- Familiarity with the notion of software architectures, their importance, and different types of architectures.
- Understanding of the role architectures play in software-intensive systems and in system development.
- Understanding of, and modelling experience with, system and architecture qualities, tactics and patterns (through a class project).
- Understanding of management and governance issues and the relationship between architectures and business.
- Experience with: reading assigned literature and extracting key

elements therefrom; formulating trigger questions; creating own comments concerning the “content”; debating technical issues through discussions.

- Experience with highlighting lessons learnt.
- Experience with creating a technical domain model from the concepts learnt in the course.
- Experience with a cloud platform.
- Experience with researching and synthesizing architecture-related material on a focused topic.
- Experience with presenting findings and results through presentations.

Expectations

- Some expectations were already described earlier (see the important note on **Class Operation)**
- This is not a course where one writes thousands of lines of code, characterised typically by intense effort of getting the program to work during the days approaching the project deadline. Rather, it is an intellectual course where system-level decisions are represented in a high-level notation (e.g., UML) that together form a system architecture. These decisions are implemented and demonstrated to be operational.
- Such decisions require circumspection of a wide variety of issues (user needs, domain issues, business goals, technical issues, management, regulatory and legal issues, socio-political issues, and others) that can affect the feasibility and quality of the system.
- It also requires collaboration among the project team members to achieve the project-goals. (Typically, in industry, it requires collaboration with various stakeholders such as users, requirements analysts, verification specialists, project managers, product and release managers, designers and integrators, and others.)
- In those attending the course, it requires dedication, self-motivation, teamwork, willingness to learn from diverse sources, and ability to communicate and share with others.
- In-class discussions are “triggers” for awareness of concepts in this subject. Students are expected to learn from identified sources, problem solving and group work. In other words, “active learning” also

occurs outside the class time.

Prerequisites

- For undergraduate students: CS3307 -- Object-Oriented Design and Analysis. For graduate students, an undergraduate degree with software engineering course(s).

- Also, please note the following regulation from the university:
  Unless you have either the requisites for this course or written special permission from your Dean to enroll in it, you will be removed from this course and it will be deleted from your record. This decision may not be appealed. You will receive no adjustment to your fees in the event that you are dropped from a course for failing to have the necessary prerequisites.

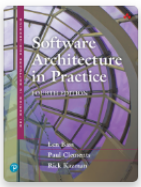
Note: Students who have been admitted to this course without the normal prerequisites may not have been exposed to some of the background material expected for this course; it is the responsibility of these students to gain familiarity with this material on their own. These students are encouraged to speak to the instructor for advice.

For the students who are not familiar with Requirements Engineering, it is strongly recommend that you read at least the following:

Requirements Engineering, by Gerald Kotonya and Ian Sommerville, Wiley, 1998.
- *Chapters 1, 2, 3, 4, 6, 8*

Textbooks:

Required Text book:



Software Architecture in Practice, 4th edition

Published by Addison-Wesley Professional (August 3, 2021) © 2022

Len Bass Software Engineering Institute , Software Engineering Institute| Paul Clements | Rick Kazman

ISBN-13: 9780136886099

(e-book is acceptable)
Author:
Bass
Edition:
4E
SKU:

CEB EBOOKID=9746045

https://bookstore.uwo.ca/textbook-search?campus=UWO&term=W2024A&courses%5B0%5D=001_UW/CSC4471A

Supplementary Text books:

DevOps -- A software Architect's perspective

Len Bass, Ingo Weber, and Liming Zhu

Addison Wesley, 2015

http://ptgmedia.pearsoncmg.com/images/9780134049847/samplepages/9780134049847.pdf

Designing Software Architectures: A Practical Approach

Humberto Cervantes and Rick Kazman

Publisher: Addison-Wesley, 2016 | **ISBN:** 0134390784

Documenting Software Architectures: Views and Beyond

Second Edition

Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Robert Nord, Judith Stafford

Addison-Wesley Professional, 2011

ISBN-10: 0321552687
ISBN-13: 9780321552686

Software Architecture: Foundations, Theory, and Practice

Richard N. Taylor, Nenad Medvidovic and Eric Dashofy

Wiley, 2009

ISBN-10: 0470167742

Software Engineering, An Object-oriented perspective

(Chapter 5 particularly)

Eric J. Braude

Wiley, 2001

ISBN: 0-471-32208-3

Evaluation

- All material covered in the course (including lectures, discussions, assignments and projects, books and other cited resources) is examinable.
- The teaching staff reserve the right to adjust (lower or raise) a student's marks for the tabulated components below based on their judgment of the student's knowledge and understanding of the subject matter during the term.
- Project logistics:
    - Projects will be carried out in groups.
    - The membership of a group will be self-assigned based on the background skills claimed. Once formed and consolidated, the groups will not be changeable for the rest of the term. The instructor has the right to adjust the membership of a group at anytime during the term.
    - Rules for group behaviour, responsibilities, constraints, consequences, etc., will be presented in the class by the instructor.
    - **EXTREMELY IMPORTANT**: In the event a group member is removed from his or her group for reasons of discontentment, please note that placement of that individual in another group will *__not__* be possible. **In this case, the student concerned would have no choice but to withdraw from the course**. PLEASE note that there is nothing else that can be done within the parameters of operation of this course. The project and other groupwork MUST be done as a group and not individually.
- The grading criteria, as applied to each evaluation component, will be described with the details of the component.
- **Scholarship:** the general level of scholarship expected in the course components is that of the 4th year student (or of a graduate student, if so registered), which implies that the student is expected to use technologies, solutions, methodologies, concepts, and the like learned in ALL the courses to date. The student is also expected to self-learn new solutions, concepts, methodologies, and technologies, as deemed required in the course component, through publicly available sources (e.g., Open source, youtube, corporate tutorials, educational sites, and others).
- **Regrading Policy:** If a student has any concern regarding marks or feedback received for one or more specific aspects of an assignment or project and is asking for re-grading the identified aspect, the re-grading

policy is that the **entire assignment** or the **entire project (with possibly multiple parts)** will be re-graded, not only the identified aspects. The net result of such re-grading could be that the marks for the entire assignment or entire project **could go up or down or stay the same**. In the event the mark goes down, the re-graded mark will be the new mark and there is no possibility to revert to the original mark given. Also, in the case of re-grading, the revised mark will be delayed.

- **Attendance in class is mandatory**. See the table below for consequences of absenteeism.
- Those who miss the in-class question and answer session, etc., will receive zero marks for this component (exceptions only as per the university policy).
- *There will be no makeup Quiz or Test, except for students requesting a Special Quiz or Test for religious reasons. These students must have notified the course instructor, by email, at least 2 weeks prior to the Quiz or Test.*

  If you miss the Quiz or Test for any other reason, follow the procedure for Academic Accommodation for Medical Illness. If accommodation is approved by your Dean's office, the Quiz or Test component will be redistributed to the other evaluation components of the course.
- IMPORTANT: grading will begin according to the schedule very early on in the term. There will be no make-up mark for days missed.

Component	Max. %	Dates
Summaries of assigned readings (pre-add/drop date – Fall term) (TEMPLATES)	15**	Weekly till 30th Nov., 2024
**Summaries of assigned readings (Post-add/drop date till end of the term). This subsumes the Pre-add/drop date mark obtained. (TEMPLATES)	20	Weekly after 30th Nov., 2024 till the end of the term.
Topic research and presentation.	15	As scheduled.
Questions and Answers (in-class interactions)	10	4th Dec (last day of class).
Technical domain model resulting from the knowledge gained in the course.	10	4th Dec (last day of class).
System architecting project (PR1: 10 pts; PR2: 30 pts; Final: 70 pts)	40	As scheduled.
INDIVIDUAL Takeaways -- created in the class and posted on the designated storage while in the class. Not permitted to log Takeaways *after* the class time.	5 **(Missing Individual Takeaways get -5% for each class)**	As scheduled..
Show and Tell (S&T) class presentation (volunteers – accepted on a FCFS basis and by instructor's approval whose decision is final and not appealable). **Admin of S&T gets up to 3% bonus.** **Bonus level decided by the instructor and is NOT disclosed.**	**BONUS** **Up to 3%.**	As scheduled.

Please Note:

- Class attendance is *mandatory* (see penalty above for missing takeaways).
- Thus conflict with any other course will just not work .... It will not be acceptable that you attend the class for only part of the class-duration (of 3 hours).

- It will not be acceptable that you miss the class due to reasons such as job interviews, extra-curricular activities, family trips, events, etc. Please arrange your other events accordingly or be prepared to lose 5% per class missed.

- University policy on absenteeism will prevail.

Topics

The topic list is tentative. The actual topics will be dictated by the dynamics of the class schedule.

Topic*
Course Overview & Preliminaries
Introduction to software architectures
Quality attributes
Architecture in the life cycle
Architecture and Business
Newer Architectures

* **Note**: other literature may be added to these topics as deemed appropriate.

Email and OWL Policy

Staff contacting students:

- *We may need to send email messages to the whole class, or to students individually. Email will be sent to the UWO email address assigned to students by Western Technology Services (WTS), i.e. your email address @uwo.ca. It is each student's responsibility to read his/her email on a frequent and regular basis.*

  However, note that email at WTS (your UWO account) and other email providers such as google, hotmail.com or yahoo.com establish quotas or limits on the amount of space available to you. If you let your email accumulate there, your mailbox may fill up and you may lose important email from your instructors. Losing email is not an acceptable excuse for not knowing about the information that was sent. Similarly, emails may end up in your "Junk" mail folder for whatever reason. It is the student's responsibility that no email from the instructing team gets placed into the Junk mail folder.

- *OWL will be used to post announcements, assignments, material in the resource folder, etc. It is the student's responsibility to check OWL frequently and regularly to ensure that no course-related material is missed.*

Student contacting Staff

- *For **technical issues** concerning the course (e.g., topic research and presentation, class project, domain model, summaries, etc.), students must create communication channels (e.g., Slack) within their group or for class-wide forums to post their questions/answers. Emails to staff for this purpose will **not** be responded to.*
- *For **administrative issues** (e.g., absenteeism, marks, and course registration) emails to the staff are acceptable.*
    - *Email **subject line** MUST include: "cs4471A/cs9549A:" <subject>. Without this, the email may be trapped by SPAM filter and may not be available, read or responded to.*

Accommodation and Accessibility

Religious Accommodation

When a course requirement conflicts with a religious holiday that requires an absence from the University or prohibits certain activities, students should request accommodation for their absence in writing at least two weeks prior to the holiday to the course instructor and/or the Academic Counselling office of their Faculty of Registration. Please consult University's list of recognized religious holidays (updated annually) at

https://multiculturalcalendar.com/ecal/index.php?s=c-univwo.

Accommodation Policies

Students with disabilities are encouraged to contact Accessible Education, which provides recommendations for accommodation based on medical documentation or psychological and cognitive testing. The policy on Academic Accommodation for Students with Disabilities can be found at:

https://www.uwo.ca/univsec/pdf/academic_policies/appeals/Academic Accommodation_disabilities.pdf.

Academic Policies

The website for Registrarial Services is http://www.registrar.uwo.ca.

In accordance with policy,

https://www.uwo.ca/univsec/pdf/policies_procedures/section1/mapp113.pdf,

the centrally administered e-mail account provided to students will be considered the individual's official university e-mail address. It is the responsibility of the account holder to ensure that e-mail received from the University at their official university address is attended to in a timely manner.

Scholastic offences are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site:

http://www.uwo.ca/univsec/pdf/academic_policies/appeals/scholastic_discipline_undergrad.pdf.

Support Services

Please visit the Science & Basic Medical Sciences Academic Counselling webpage for information on adding/dropping courses, academic considerations for absences, appeals, exam conflicts, and many other academic related matters: https://www.uwo.ca/sci/counselling/.

Students who are in emotional/mental distress should refer to Mental Health@Western (https://uwo.ca/health/) for a complete list of options about how to obtain help.

Western is committed to reducing incidents of gender-based and sexual violence and providing compassionate support to anyone who has gone through these traumatic events. If you have experienced sexual or gender-based violence (either recently or in the past), you will find information about support services for survivors, including emergency contacts at

https://www.uwo.ca/health/student_support/survivor_support/get-help.html.

To connect with a case manager or set up an appointment, please contact support@uwo.ca.

You may wish to contact Accessible Education at

http://academicsupport.uwo.ca/accessible_education/index.html

if you have any questions regarding accommodations.

Learning-skills counsellors at the Student Development Centre (https://learning.uwo.ca) are ready to help you improve your learning skills. They offer presentations on strategies for improving time management, multiple-choice exam preparation/writing, textbook reading, and more. Individual support is offered throughout the Fall/Winter terms in the drop-in Learning Help Centre, and year-round through individual counselling.

Additional student-run support services are offered by the USC, https://westernusc.ca/services/.