



COURSE OUTLINE
SENG 422 – Software Architecture
Summer 2014

Instructor:

Dr. Issa Traore
Phone: 250-721-8697
E-mail: itraore@ece.uvic.ca

Office Hours:

Days: Monday, Thursday
Time: 2:30-4:00pm
Location: EOW 415

Lectures:

A-Section(s): A01 / CRN 30718
A02 / CRN 30719
Days: Monday, Thursday
Time: 11:30am-12:50pm
Location: CLE A202

Labs:

B-Section(s): B01 / CRN 30720
B02 / CRN 30721
Days: Friday
Time: 2:30-5:20pm
Location: ELW B220

Required Text:

Title: Software Architecture in Practice
Author: L. Bass, P. Clements, R. Kazman
Publisher: Addison Wesley
Year: 2013, 3rd edition

Optional Text:

Title:
Author:
Publisher:
Year:

References:

Assessment:

Attendance/Class Participation:	5%
Project	35% (Part 1- 8%; Part 2- 12%; Part 3 – 15%)
Mid-term	20%
Final	40%

Date: July 3, 2014

Note: Failure to complete all laboratory requirements will result in a grade of N being awarded for the course.

Due dates for assignments:

- Project Part 1: due May 30, 2014
- Project Part 2: due June 27, 2014
- Project Part 3: due August 1, 2014

The final grade obtained from the above marking scheme will be based on the following percentage-to-grade point conversion:

Passing Grades	Grade Point Value	Percentage for Instructor Use Only	
A+	9	90 – 100	
A	8	85 – 89	
A-	7	80 – 84	
B+	6	77 – 79	
B	5	73 – 76	
B-	4	70 – 72	
C+	3	65 – 69	
C	2	60 – 64	
D	1	50 – 59	
Failing Grades	Grade Point Value	Percentage for Instructor Use Only	Description
E	0	35 - 49	Fail, conditional supplemental exam. (For undergraduate courses only)
F	0	0 – 49	Fail, no supplemental.
N	0	0 – 49	Did not write examination, Lab or otherwise complete course requirements by the end of term or session; no supplemental exam.

The rules for supplemental examinations are found on page 80 of the current 2013/14 Undergraduate Calendar.

Term in which E Grade Was Obtained	Application Deadline for Supplemental Exam	Supplemental Exam Date
First term of Winter Session (Sept – Dec)	February 28 in the following term	First week of following May
Second term of Winter Session (Jan – Apr)	June 30 in the following term	First week of following September
Summer Session (May – Aug)	October 31 in the following term	First week of following January

Deferred exams will normally be written at the start of the student's next academic term; i.e., approximately 4 months following the deferral of the exam.

Course Description

Course Objectives

This course focuses on the decision-making process underlying software architecture design, and the selection and application of suitable architecture design strategies, mechanisms and technologies. Software architecture is viewed as a collection of software components interacting or communicating via a collection of software connectors. The focus of the course will be on identifying and designing software components and connectors by adopting appropriate design strategies, architectural styles, and technologies.

Learning Outcomes

By the end of this course, students should have a good grasp of the following:

- The need for software architecture and fundamental concepts underlying software architecture
- Major software design techniques
- Architecture documentation using architecture description languages
- Architectural styles and patterns
- Techniques, models, and metrics used to analyze, design, and evaluate large-scale software architecture

Syllabus

Unit 1. Introduction to Software Architecture

Define the discipline of software architecture and introduce fundamental concepts underlying this discipline. A particular emphasis will be placed on the following:

- Notions of Views and Viewpoints.
- Place of software architecture in the whole software life cycle.
- Discussion of the issues surrounding modern software architecture design: abstraction, reusability etc.

Unit 2. Architecture and Requirements

Introduce the concept of Architecturally Significant Requirements (ASR) and present systematic way to capture and express quality attributes requirements under the form of quality attributes scenarios.

Unit 3. Architecture Design

Provide a general overview of software design techniques, and then focus in detail on the Attribute-Driven Design (ADD) technique. Emphasis will be placed on the identification of software architectural drivers or tactics and the use of these drivers to guide the design of software architecture.

Unit 4. Documenting Software Architectures

Present notions and concepts of software components and connectors and their place in software architecture. Emphasis will be placed on components/connectors identification, design, and modeling using a standard architectural description language (ADL). Emphasis will also be placed on selection and application of components communication and control patterns.

Unit 5. Architectural Styles

Define the notions and needs for architectural styles as major building for software architecture, and introduce major architectural styles such as data flow styles, independent component styles, c and service-oriented architectural (SOA) style.

Unit 6. Performance Quality Attribute

Introduce methods and techniques to evaluate and improve software architecture from performance standpoint. Present techniques and models for software performance analysis, prediction, and design.

Unit 7. Availability Quality Attribute

Introduce techniques and models for software availability analysis and design. Present basic notions of software fault tolerance and give an overview of software fault tolerance techniques and architectures.

Unit 8. Security Quality Attribute

Introduce techniques and models for software security analysis and design.

Unit 9. Modifiability Quality Attribute

Introduce techniques and models for software modifiability analysis and design. Present modifiability metrics and structural analysis techniques.

Unit 10. Economic Analysis of Architectures

Introduce techniques and models for cost-benefit analysis based on architectural requirements and design decisions.

Note to Students:

Students who have issues with the conduct of the course should discuss them with the instructor first. If these discussions do not resolve the issue, then students should feel free to contact the ECE Chair by email or the ECE Chair's secretary to set up an appointment.

Accommodation of Religious Observance

See <http://web.uvic.ca/calendar2013/GI/GUPo.html>

Policy on Inclusivity and Diversity

See <http://web.uvic.ca/calendar2013/GI/GUPo.html>

Standards of Professional Behaviour

You are advised to read the Faculty of Engineering document Standards for Professional Behaviour at <http://www.uvic.ca/engineering/current/undergrad/index.php#section0-25> which contains important information regarding conduct in courses, labs, and in the general use of facilities.

Cheating, plagiarism and other forms of academic fraud are taken very seriously by both the University and the Department. You should consult

<http://web.uvic.ca/calendar2013/FACS/UnIn/UARe/PoAcI.html> for the UVic policy on academic integrity.

<p>Plagiarism detection software may be used to aid the instructor and/or TA's in the review and grading of some or all of the work you submit.</p>
--