Course Outline

COMP9322
Service Oriented Architectures
Teaching Period 1 2011

Contents

- Course Staff
- Course Details
- Course Aims
- Student Learning Outcomes
- Assumed Knowledge
- Teaching Rationale
- Teaching Strategies
- Assessment
- Academic Honesty and Plagiarism
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Staff

Contact Details

Staff Name	Role	Email	Phone
Helen Paik	Course Convenor	hpaik at cse.unsw.edu.au	54095

Consultations

Staff Name	Day/Time	Location
Helen Paik	Monday 1100-1300	K17 401A

Course Details

Course Code: COMP9322
Course Title: Service Oriented Architectures
Units of Credit: 6
Course WebSite: http://www.cse.unsw.edu.au/~cs9322/
Handbook Entry: http://www.handbook.unsw.edu.au/postgraduate/courses/2011/COMP9322.html

Course Aims

The Service Oriented Architecture (SOA) concept is rapidly changing the way modern enterprise systems are designed, built and managed. An SOA is not just a technical architecture but a revolutionary way to design an enterprise environment that features cross-platform compatibility, agility and cost-efficiency. An SOA can be realised using a number of technologies, amongst them Web services are considered to be the key technical enabler of SOA.

It is fast becoming one of the most important skills for an IT graduate to be able to understand the role of an SOA in different contexts and to articulate the motivation behind service-based technologies and their practical implications in terms of engineering complex software systems and automating business processes.

This module aims to provide students with a deep understanding of SOA, service-orientation paradigm, and Web services as an implementation technology.

We also aim to train the students to develop the ability to appreciate the necessity for tracking technology development and become a self-learner and an insightful predictor of future developments in this area.

Student Learning Outcomes

After completing this course, students will:

- Describe different architectural design approaches and their role in enterprise application integration

- Apply the concept of business process in a concrete setting
- Be competent in designing and developing applications using SOA concepts and related technologies

This course contributes to the development of the following graduate attributes:

Graduate Attribute	Where Acquired
the capacity for analytical and critical thinking and for creative problem solving	lab exercises and programming assignments
the ability to engage in independent and reflective learning	lab exercises and programming assignments, online Q/A, reading system documentations
an appreciation of and respect for, diversity	attending lectures, online Q/A, teamwork
the skills of effective communication	group/team assignments, online Q/A

Assumed Knowledge

Before commencing this course, students should:

- Have theoretical understanding of Web applications and Web-based architectures
- Have practical skills to develop a Web application
- Have sound knowledge of Java programming

These are assumed to have been acquired in the course COMP9321

Teaching Rationale

This course is taught the way it is because we believe in structured learning, also learning by doing. We provide timely feedback for learning via small, step-by-step assignments that gradually build up knowledge and practical skills.

Teaching Strategies

The design of the course is aligned with the simple learning model in which students are expected to go through the following stages: (i) be introduced to the idea, concept or principle, (ii) get to know more about it, (iii) try it out for oneself, (iv) get feedback and (v) reflect, adjust and try again. The lectures are used to introduce new concepts, facts and knowledge each week, gradually moving onto advanced ideas. Labs and assignments are used as opportunities for students to try, get feedback, reflect and try again. Especially, the students are encouraged to share and experience the learning with their peers through group activities. The teaching support staff will give feedback either directly during lab sessions or through the online discussion board. Students are highly encouraged to participate in the discussions as we see it as an important part of active learning.

Assessment

The assessment consists of the following components:

- 55% formal written exam: individual assessment. This component is going to assess the various facts-and-knowledge level learning outcomes. The exam is a mixture of multiple choice questions and written answer questions.
- 45% on laboratory work: group assessment (group of 2 maximum). This component assesses the practical-skills-and-tools level learning outcomes. The assessment activities include four programming assignments. Each assignment is designed for students to explore each main technology from the lectures. Labs and assignments are released roughly every two weeks to encourage students to progressively develop their skills.

Note: There is a hurdle requirement in this course. If you do badly in the final exam, you would not pass the course. We enforce this by the following rule:

- If you score 40% or more in the final exam, your total score for grading is (final exam + assignments),
- if you fail to score 40% or more in the final exam, you will get UF* grade regardless of your final score.

*UF (Unsatisfactory Fail) - Unsatisfactory performance in an essential component of the course.

Academic Honesty and Plagiarism

Plagiarism is defined as "using the words or ideas of others and presenting them as your own". UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several on-line sources to help you understand what plagiarism is and how it is dealt with at UNSW:

- Learning Centre: Plagiarism and Academic Integrity
- MyUNSW: Plagiarism and Academic Misconduct

- CSE: Addendum to UNSW Plagiarism Guidelines
- CSE: Yellow Form (whose terms you have agreed to)

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism.

Note: All submissions for the assignments go through plagiarism checks. If you are found to have plagiarised, you will automatically score 0 for the component.

Course Schedule

The topics are organised and delivered by week. The following shows a tentative schedule:

No schedule is available (yet)

Resources for Students

There is no textbook for this course. Recommended reading lists will be provided through the online Q/A (messageboard) throughout the semester.

Course Evaluation and Development

This course is evaluated each session using the CATEI system.

In the previous offering of this courses, students remarked that the five assignments were interesting, but posing a high work load.

Based on their comments, we have removed one of the assignments, as well as simplified the first assignment. We will also introduce BPMN and BPMN examples in the lectures.