Course Outline

COMP9323
e-Enterprise Project
Teaching Period 2 2011

Contents

- Course Staff
- COMP9323 : Course Details
- Course Summary
- 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
Fethi Rabhi	Course Convenor	f.rabhi at unsw.edu.au	93854179
Aarthi Natarajan	Course Admininstrator	aarthin at cse.unsw.edu.au	
Lawrence Yao	Course Admininstrator	lawry at cse.unsw.edu.au	
Weisi Chen	Course Admininstrator	chenw at cse.unsw.edu.au	

Consultations

Staff Name	Day/Time	Location

COMP9323 : Course Details

Course Code/Title: COMP9323 e-Enterprise Project
Units of Credit: 6
Course WebSite: http://www.cse.unsw.edu.au/~cs9323/
Handbook Entry: http://www.handbook.unsw.edu.au/undergraduate/courses/2011/COMP9323.html

Course Summary

This course is the last part of the services engineering stream. The aim of this course is to allow students to pursue a group-based software development project based on service oriented architecture

Course Aims

This course is the third instalment of the three-part series of services engineering stream. Enterprise developers and architects aiming at providing SOA based solutions in the industry today are focusing on creating reuseable services and assembling those services into flexible composite applications. Because of the wide choice of technology for building these services, the SOA industry addresses this challenge through the use of an SOA model based on a Service Component Architecture (SCA), using SOA suites such as IBM Websphere Process Server, Oracle SOA Suite. This course guides students through learning the concept of an SCA (Service Component Architecture) and develop skills to developing composite SOA based applications using suitable SCA-based tools. This course will begin with three lectures to provide students an understanding of the SCA architecture and Tuscany runtime. Students will then be required to use their acquired knowledge to implement a group-based project.

Student Learning Outcomes

After completing this course, students will be able to:

- Understand the Service Component Architecture (SCA) concept and its role in the development of Service Oriented Architecture (SOA) style applications
- Be familiar with a particular SCA-based run-time environment and its API (e.g. Apache Tuscany)
- Build a composite application based on SCA-based run-time environment (e.g. Apache Tuscany SCA Java API)
- Learn about process of developing technical documentation in an SOA context.
- Acquire additional skills working as part of project team, implementing quality software system with strict time constraints

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

Graduate Attribute	Where Acquired
the skills involved in scholarly enquiry	yes
an in-depth engagement with relevant disciplinary knowledge in its interdisciplinary context	yes
the capacity for analytical and critical thinking and for creative problem solving	yes
the ability to engage in independent and reflective learning	yes
the skills to locate, evaluate and use relevant information (Information Literacy)	yes
the capacity for enterprise, initiative and creativity	yes
an appreciation of and respect for, diversity	no
a capacity to contribute to, and work within, the international community	no
the skills required for collaborative and multidisciplinary work	yes
an appreciation of, and a responsiveness to, change	yes
a respect for ethical practice and social responsibility	no
the skills of effective communication	yes

Assumed Knowledge

Pre-requisites of the course are:

- Java programming language
- XML, XML Schemas and use of XML Namespaces
- Web services, Web Service Definition Language, Use of tools for transformation between WSDL and Java classes

Teaching Rationale

The teaching strategy adopted in this course are geared towards preparing students for real-life SOA projects. The course will supplement the lectures with a practical hands-on group based project that helps students to gain the skills necessary to design, model and implement software development projects using technologies & methodologies that are considered mature and proven in the SOA industry. Students will attain a firm grasp of SOA from an industry perspective and software development in a group situation is encouraged with the lecturer and other mentors guiding and providing continuous feedback to each group.

Teaching Strategies

This course and all of its materials are tuned to represent SOA as it exists today. The course will commence with lectures in the first three weeks, where students will be exposed to the methodologies, techniques and tools widely adopted in the SOA industry today. This will be followed by a group-based project.
A set of intermediate deliverables for this project will be specified by an allocated mentor. Weekly mentoring sessions are used to elaborate on the deliverables and answer general questions. During these meetings, teams are encouraged to discuss their progress and demonstrate work-in-progress. Teams can also arrange additional meetings with their mentor. All teams will be publicly demonstrating their work few weeks before the end of the workshop to give them opportunities to improve their project based on the feedback received prior to the final demonstration in the last week of the course.

Assessment

Assessment	Category & Percentage	Learning Outcome	Due Date
Sample Application	Demo(10%)	1,2,3	Week5
Design Report	Report(10%)	1,4	Week7
Initial Prototype	Code & Demo (20%)	1,2,3,5	Week8
Final Prototype & Report	Code,Demo & Report(60%)	1,2,3,4,5	Week12

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.

Course Schedule

- Week 1: Lecture 1: (Introduction to SCA (http://www.open-soa.org))
- Week 2: Lecture 2: (Introduction to Apache Tuscany)
- Week 3: Lecture 3: (Building, deploying & testing composite applications)
- Week 4: Lecture 4: Domain Brief + Initial Session with Mentor
- Week 5: Session with Mentor & Demonstration of sample application based on Eclipse Helios & Apache Tuscany (Sample application)
- Week 6-7: Session with Mentor (Group project). Design Report due (Week 7)
- Week 8: Public demonstration of first prototype (Reference to existing news/price services)
- Week 9-11:Session with mentor (Group project)
- Week 12: Demonstration of final prototype. Final report Due

Resources for Students

Books about Tuscany and SCA

- Tuscany SCA In Action
  by Simon Laws, Mark Combellack, Raymond Feng, Haleh Mahbod, Simon Nash
- Apache Tuscany Site defined as
- SOA for the Business Developer - Concepts, BPEL, and SCA

Articles

- What is SCA? A quick view of concepts through and an example walkthrough , Java Developer Journal, Article by Simon Laws, Raymond Feng, Haleh Mahbod
- Building SOA with Tuscany SCA, A simple service-oriented infrastructure, Java Developer Journal, Article by Simon Laws, Raymond Feng, Haleh Mahbod
- Service-Component Architectures-A programming model for SOA, Dr Dobb's Portal, August 2007
- Futureproofing With SCASee how using SCA makes your services more adaptable for the future, IBM DeveloperWorks, by Bert Lamb
- Deploy an SCA application using the Tuscany domain manager, IBM DevloperWorks, by Alla Segal, Michael Beisiegel, Jean-Sebastien Delfino
- Building a Geospatial Information System, Part 2: A real world application using Web 2.0, SOA, and SCA principles, IBM DeveloperWorks, by Angel Tomala-Reyes and Murali Vridhachalam
- Design and develop SCA components using the Spring Framework, Part 1: The trifecta: Spring, SCA, and Apache Tuscany, IBM DeveloperWorks,Ramkumar Ramalingam

Presentations

- Introducing SCA, David Chappell and Associates ** Excellent introduction
- OASIS SCA Tutorial Part 1, Given at the 2007 OASIS Symposium held in San Diego in April 2007
- OASIS SCA Tutorial Part 2 - focus on policies,Given at the 2007 OASIS Symposium held in San Diego in April 2007

Links to on-line documentation and examples will be added shortly ...

Course Evaluation and Development

Feedback was generally positive during the last running of the course. The main comment was to provide information better. This has been achieved in the current offering by: 1. Releasing information as early as possible about the requirements of each project 2. Choosing a common theme amongst all projects (financial data analysis) with background information provided 3. Requiring teams to use existing Web services (provided with documentation) Another improvement in this offering is the introduction of a new technology (Tuscany) which enables service composition