CMPE 275 Enterprise Software Components

Instructor:	John Gash
Email*:	john.gash@sjsu.edu
Office Hours:	Immediately following class (Mon R/337, Wed R/331) or by appointment
Class Days/Time:	Monday: Lecture 6:00 – 8:45 pm, integrated lab and discussion included
Classroom:	Engineering, R/337
Group:	Desire to Learn is our primary distribution of information and project submissions.
Prerequisites:	• CMPE 273 or strong equivalent experience • Java development proficiency** • OOA/D, UML and software engineering concepts/experience

*Email subject must start with **275 Wednesday - subject**
**Java is used extensively (but not exclusively) to demonstrate concepts and in the preparation of class
 projects. You will be expected to learn/use multiple programming languages including Java.

Faculty Web Page and MYSJSU Messaging

You are responsible for regularly checking the following: 1) the messaging system through MySJSU; 2) the class' D2L (Desire2Learn online learning management system) site.

Course Catalog Description

Applications development using components and distributed objects; introduces commercial Java EE component infrastructures and component frameworks; integration technique; lab uses commercial component construction tools.

Course Notes

CMPE 275 explores application development using distributed software architectures, practices, and technologies. The course covers advanced distributed concepts of component design, scalable architectures, communication, and integration practices. Prerequisite: Java Programming, CMPE 273 or instructor consent. Lecture 2.75 hours/lab 3 hours.

Note: This course requires significant commitment and effort; you will work hard, but you will learn a lot. Information and skills required to complete the assignments require investigation and research that extend beyond the material presented in lecture. This includes, and not limited to, design practices for components and software systems, technologies, tools, standards, and thinking.

Course Learning Objectives

The learning objectives of CMPE 275 is to provide individuals with an understanding of and experience with:
• Software component design, development, and testing
• Research in enterprise software development practices and methodologies
• Proficiency in designing and developing sustainable enterprise architectures
• Hands on experience with the current and emerging technologies for constructing distributed systems

Required Texts/Readings

Online research and investigation

The primary source of information will be research and reading from online papers, specifications, and discussions. While no specific textbooks are required, you may find several books useful, they are listed below.

Supporting books

This is not an inclusive list; students are expected/required to research and utilize information from many sources (papers, online, peers, etc.). If you select to use purchase/borrow a book(s) for this class, a couple of suggestions are listed below.

Directly applicable:

- The Ruby Programming Language, by Flanagan, Matsumoto, 2008
- Python Essential Reference, 4th Ed., by David Beazley, 2009

Moderately useful:

- EJB3 In Action, by Panda, Rahman, Lane, 2007
- Pro JPA 2, Keith, Schincariol, 2009
- Enterprise JavaBeans 3.1, Rubinger, Burke, Haefel

Supporting:

- Software Systems Architecture, by Rozanski, Woods, 2005
- Software Architecture in Practice, Second Edition, by Bass, Clements, Kazman, 2003
- UML Distilled, Third Edition, by Fowler, 2003
- Applying UML and Patterns, 3rd Edition, by Larman, 2004

Other equipment / material requirements

Additional references included at the end of each lecture. Suggested papers/information are provided at the end of each lecture to provide additional sources of research and investigation. These references are not required though they enrich your learning experience.

Classroom Protocol

1. Ty to arrive on time. Arriving early can give you time to talk to your team members regarding the projects, or hold general discussions on technologies, design, or coordination.
2. Cell phones must be muted during the lecture.

Dropping and Adding

Students are responsible for understanding the policies and procedures about add/drop, grade forgiveness, etc. Refer to the current semester's Catalog Policies section at *http://info.sjsu.edu/static/catalog/policies.html*. Add/drop deadlines can be found on the *current academic calendar* web page located at http://www.sjsu.edu/academics/. The Late Drop Policy is available at *http://www.sjsu.edu/aars/policies/latedrops/policy/*. Students should be aware of the current deadlines and penalties for dropping classes.

Information about the latest changes and news is available at the *Advising Hub* at *http://www.sjsu.edu/advising/*.

Assignments and Grading Policy

Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of forty-five hours over the length of the course (normally 3 hours per unit per week with 1 of the hours used for lecture) for instruction or preparation/studying or course related activities including but not limited

to internships, labs, clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus (See additional information).

Grading (A curve is applied)

100 - 90	A
89 - 80	B
79 - 70	C
69 - 60	D
59 - 0	F

Distribution

(70 pts with approximate distribution, %)

```
Projects              40 pts (57.2%)
Paper                 15 pts (21.4%)
Mid-Term Examination  15 pts (21.4%)
Final Examination     15 pts (21.4%)
```
(121% allows you to choose 2 of the 3 15 pts)

University Policies

Academic integrity

Students should know that the University's Academic Integrity Policy is available at http://sa.sjsu.edu/judicial_affairs/faculty_and_staff/academic_integrity/index.html. Your own commitment to learning, as evidenced by your enrollment at San Jose State University and the University's integrity policy, require you to be honest in all your academic course work. Faculty members are required to report all infractions to the office of Student Conduct and Ethical Development. The Student Conduct and Ethical Development website is available at *http://www.sa.sjsu.edu/judicial_affairs/index.html*.

Instances of academic dishonesty will not be tolerated. Cheating on exams or plagiarism (presenting the work of another as your own, or the use of another person's ideas without giving proper credit) will result in a failing grade and sanctions by the University. For this class, all assignments are to be completed by the individual student unless otherwise specified. If you would like to include in your assignment any material you have submitted, or plan to submit for another class, please note that SJSU's Academic Policy F06-1 requires approval of instructors.

Campus Policy in Compliance with the American Disabilities Act

If you need course adaptations or accommodations because of a disability, or if you need to make special arrangements in case the building must be evacuated, please make an appointment with me as soon as possible, or see me during office hours. Presidential Directive 97-03 requires that students with disabilities requesting accommodations must register with the Disability Resource Center (DRC) at *http://www.drc.sjsu.edu/* to establish a record of their disability.

Tentative Schedule (subject to change)

Dates & Assignments	Topics and Objectives	

Jan 28	**Introduction (1)**
	   1. Overview
	   2. What you will need (Java, C++, Python, Ruby)
	   3. Key focus areas: Data and Communication	

Feb 4
Project 1 handout
 	
Feb 11
 	
Feb 18
 	
Feb 25
Project 2 handout
 	
Mar 4
 	
Mar 11
	Distributed concepts (1)
	   1. Review MOM concepts and strategies
	   1. Architecture patterns of distributed communication
	   2. Synchronous and Asynchronous behavior
	
	Communication Pattern Study: Netty - Java, C++ (2)
	   1. Introduction to JBoss Netty and Google ProtoBufs
	   2. Patterns and behaviors of asynchronous communication
	
	Distributed Component Model: Enterprise components (1.5)
	   1. Introduction to EJBs
	   2. Patterns and behaviors of synchronous communications
	
	Distributed Persistence (2)
	   1. Patterns of distributed storage
	   2. Java Persistence API (JPA – EJBs)
	   3. TBD
	
	JEE part 2: Message-based communication (.5)
	   1. JEE Message Driven EJBs
	   2. Interceptors and lifecycle
		Project 1 (Data and Communication)

Mar 18	Mid-Term Exam	

Mar 24 (before break)
Project 1 due
 	
Mar 25 - 29
Spring recess
 	
Apr 1
 	
Apr 8
Paper due
Project 2 part A due (TBD)
 	
Apr 15
	RESTful communication and overlay network concepts
	
	Web Services (SOAP and REST) (2.0)
	   1. RESTful web services
	   2. Overlay network design
	
	Advanced topics (1)
	   1. TBD - if time permits
		Project 2 parts A & B (RESTful and web frameworks)

Apr 22
Project 2 part B due
 	
Apr 29
 	
May 6
Project 2 part C due
	Concepts in presentation frameworks and technologies (Ruby on Rails, Django).
	
	Introduction to Ruby and Ruby on Rails (2)
	   1. Models and ActiveRecord
	   2. Views, Controllers and routing
	   3. Integration using web services
	
	Introduction to Django (Python) (1)	

May 13	**Project 2 poster session**	
May 20	Final Exam	

Note: topics span multiple lectures and are subject to revisions and changes due to factors such as time constraints, travel, or extended discussion (except for the exam date, which is fixed).

Additional information

The following information is provided to help in increasing your overall experience with CmpE 275.

How This Course Will Be Conducted

As stated earlier, student expectations are to a minimum of 3 hours per unit or This course includes interactive discussions and hands on software development and research relating to computing methodologies and technologies for distributed systems. Consequently, a considerable level of effort and time will be required for research and software development (multiple software projects). Details follow.

Class/Lecture. Class discussion is an interactive exploration of concepts and ideas focusing on real world situations (businesses, institutions, and research), which includes participation in critical problem solving, articulating concepts, defending positions, and presenting ideas within a group environment. You are required to prepare for each meeting by researching and investigating topics; this may include literature searching, prototyping, and Internet investigation.

Grading. End of term grades are assigned using a curve based on accumulated points from the following areas: Final Exam (maximum 15 pts), a research component of projects (20 pts), and three Class Projects (10-25 pts each).

Task	Point Value
Project 1	15
Project 2	25
Individual paper, 20 pts	20
Final Exam	20
	80

Note: percentages and points may change due to course/project/time adjustments.

Projects

Projects are a key component to the class. Projects provide you a challenging, real, problem to apply your engineering skills and concepts of the course. In order to maximize interactions, a team and individual approach is used to foster perspectives and collaboration. Each project is composed of a team and individual effort. They are (Note: Additional details will be provided in class):

1.   Team – design, problem solving, project implementation and report.
2.   Individual – assigned work, participation in design, development, and supporting other team members.

Teams. Please create teams as soon a possible as projects will begin/be due as early as the second or third lecture. Teams are self-forming with each team composed of three, or four persons; larger or smaller teams will not be accepted. In addition to interactions within a team, select projects will require cross team interaction to solve distributed challenges.

Topics. Projects are organized with one or more key concepts or scenario (see below). Guidelines, objectives, and expectations will be provided prior to the start of each assignment (projects will be structured to allow completion in 2-3 weeks).

Concepts	Scenarios
• Distributed concepts in complex systems	• Legacy Integration
• Distributed workflow – delegation, discovery	• Collaboration
• Distributed persistence models	• Research and Exploration
• Designing asynchronous architectures	• Highly Scalable, Fault Tolerant
• Distributed component concepts in presentation (web) frameworks	• Open Source
	• Distributed data repositories

Deliverables. Each project (unless noted) will include two deliverables, an report, and a team project (source code, test cases, and supporting data). Projects are submitted in electronic format within a 48 hour window of the due date.

In order to facilitate grading and prompt feedback, projects should be submitted one per group and in the following format:

- Project directory must include the group ID/Name (e.g., project1-caffeine)
- Within the project directory include the report, source code, and test data. If a large quantity of data is to provided, only include data for testing. Do not include libraries (jars) used to build your project. The report should include a list of dependencies and how to retrieve/install/configure.
- Contain all group members in the report cover

For example:

> Team Caffeine is submitting their project 1. They have provided the following files and source directory:
>
> **project1-caffeine/**
> - **Installation notes**
> - **caffeine-project-report.doc (.pdf)**
> - **project-1/** *(do not send class or jar files)*
>
> The archive submitted (emailed) is **caffeine-project1.zip** (.jar, .rar) of the <u>directory:</u>
>
> **zip –r project1-caffeine.zip project1-caffeine**
>
> Upload to D2L

Papers

The research paper assignment provides the <u>individual</u> an opportunity to explore in greater depth or a facet of specific concepts related to the technologies or processes discussed during class. Research papers are also a chance to explore ideas relating to (or supporting) Master's project topics.

1. Papers are associated or complementary to the topics under study for each project
2. Choose any one of the three project's technologies/concepts to conduct your investigation
3. **Papers are due the same time as the project and should be submitted separately**

Assignments and Grading Policy

Grading is based on both group (projects) and individual contributions (papers, final) – see above for point breakdown.

Your grade (letter) is determined using a curve and varies from class to class. There is no fixed percentage of assigned As, Bs, etc. – grades are awarded on where one falls within the letter tiers. This means that if everyone in the class scores 85 or better (curve applied), everyone gets an A. The converse holds true as well. More information will be provided during the first lecture.

Notes:
1. None participation in group projects will affect the offending individual's grade – determined by instructor
2. Your grade is your responsibility not your group's.
3. Late assignments are assessed a late fee unless otherwise noted

Academic integrity

Instances of academic dishonesty will not be tolerated. Cheating on exams or plagiarism (presenting the work of another as your own, or the use of another person's ideas without giving proper credit) will result in a failing grade and sanctions by the University. For this class, all assignments are to be completed by the individual student unless otherwise specified. If you would like to include in your assignment any material you have

submitted, or plan to submit for another class, please note that SJSU's Academic Policy F06-1 requires approval of instructors.

Miscellaneous

<u>Work/Life Balance.</u> Life is not predictable and occasionally we need to rebalance family, class, and work obligations. There are times when no amount of planning allows one to satisfy all requirements and conflicts arise; if you find yourself in such a situation, please talk with me to see if there are options or adjustments that will allow you to be successful.

<u>Philosophy.</u> As mentioned previously, this class is built upon interactive research and discussion. This format requires individuals to perform investigations and research prior to each discussion topic. Sessions or lectures are for the discussion and examination of the topic at hand. Everyone is required to fully participate in all discussions.

<u>Projects.</u> Real world projects are not completed in isolation ("bring me a rock"), they are interactive, exploratory, evolving, and advisory in nature; our class projects mimic these traits.

In support of project assignments, lectures and lab include time to foster and support discussions on project approaches, strategies, and implementation. Please plan your time accordingly; **projects are software intensive, which require significant investment of time and research.** They are also rewarding, as they provide a unique opportunity to practice and validate one's research and studies of the domain and technologies.