

SE 581: Architecture 2

Dr. Jane Cleland-Huang

Office: CDM 836

Phone: 312-362-8863

Open office hours:

2:45PM - 4:15PM and Thursday 2:00PM - 3:30PM

Course Description

This is the second in a series of courses on Software Architecture. As such, this course covers intermediate and advanced concepts of Software Architecture. All students taking the course are expected to have passed SE480 which covers basic concepts of Software architecture.

Students will be exposed to case studies of real-world software architectures for which reliability, performance, availability, scalability and other such concerns drive the architectural design. Advanced architecture topics such as self-adaptation, software-product lines, domain-specific architectures, and agent-based systems will be covered. Students opting for practical experience will have the opportunity to gain hands-on practice in reverse engineering, designing, and assessing existing systems including performance testing and will build on their prior knowledge to design complex systems in principled and systematic ways. Students opting for a research experience will identify an open problem and then design, execute, document, and publish original research in the area of software architecture. PREREQUISITE(S): SE 480 (Prerequisites: **SE480 strictly enforced**)

Text Books

There are no text books for this course. Readings will be assigned throughout the course and will be posted each week on the D2L website in the form of a study guide. Each student will also develop their own individual reading list for the course in accordance with their selected practicum or research project. This list will be approved by the instructor.

Course Format

SE581 will be taught as a topics-based course in which lectures will cover a variety of advanced Software-Architecture topics. While no direct homework assignments will be given for these topics, students are expected to keep a journal discussing all topics covered during the quarter. 20% of the course grade will be assigned to the journal.

Each student in the class is expected to take on a significant architecture project which can be implementation based, analysis based, or research based.

- **Implementation projects** will involve creating executable architectural prototypes for a selected problem. Students are expected to implement and comparatively evaluate at least two different approaches (i.e. patterns or frameworks) for the same problem. As each student will enter the course with different knowledge, the project should be selected to stretch and push each person's knowledge in new ways. Students choosing an implementation project will also have the opportunity to contribute towards a group research goal (which will be explained by the Instructor on the first night of class). This means that selecting the implementation option does not necessarily preclude involvement in an external publication.

- **Analysis based projects** will be for students who wish to participate in a full-blown architectural design and assessment process. Students selecting this option will not only design and evaluate an architectural solution for a complex system, but will host an ATAM assessment of it for the other students (this can be done online or in person during class). This project would be ideal for a student who wishes to develop their practical architectural process skills. Students taking this option are expected to include some programming i.e. architectural spikes to demonstrate that major risks have been mitigated. The expected outcome is a full-length report similar in stature to the Track-Manager report studied in SE480. The major difference between the analysis vs. the implementation option is the ratio of development vs. documentation.
- **Research projects** are for students who not only wish to learn about software architecture, but who wish to contribute to the state of knowledge in the area of software architecture. There will be several different types of research projects that students can participate in. The instructor will present several options during the first week of class; however students are free to propose their own topics. All research projects will require a high-quality project report at the end of the quarter and a conference-style paper to be written. It should be noted that one of the most difficult aspects of research is finding a novel problem to address. Students may also choose to reproduce previous studies; however these may not necessarily lead to publications.

The instructor will present ideas for at least 5 different novel research projects. Each of these projects will involve conducting a literature survey, formulating hypotheses, implementing an environment to test the hypotheses, running experiments, analyzing results, and reporting results. Some of these projects will be part of the SAREC research group's ongoing research projects. In this case the student would write up an individual course report and material from the report would be included in a publication. The student would then be a co-author of the broader publication. All research projects have the potential for publication (if they address an interesting research question and return interesting results). The research projects represent a very broad spectrum of research areas that include architectural tactics, frameworks, visualization, reverse engineering architectural knowledge, architectural practices, and usability studies of architectural-related techniques (and anything interesting an individual can think up).

- **Systematic Literature Survey:** This represents a very thorough systematic survey of existing literature with the goal of answering a unique research question. A systematic literature survey does not involve unique experimentation, but instead answers research questions (of a certain nature) through exploring and analyzing existing literature. This differs from typical student reports in several respects which include the systematic nature of the study, the rigor of the study, and the way in which research questions are formulated and addressed. Students choosing this option are expected to write up a report similar in quality to an ACM Survey (examples will be shown in class).

Grading

Grades for the course will be distributed as follows:

- 10% Topic selected.
Study plan (contract) completed and approved.
(Due week 3)
- 40% Intermediate project deliverables
Defined by each student's individual approved contract
- 20% Individual journal based on topics presented throughout the quarter.
- 30% Final project report

Grading for this course will be contract-based. That means that by week 3 of the course each student must have an approved contract in place which identifies their project and establishes the specific deliverables worth 40% of

their final grade. The student and instructor will agree upon how points will be distributed across the deliverables. Students are strongly encouraged to discuss their ideas and plans with the instructor during the first two weeks of the quarter.

Grades will be assigned as follows:

If the final numeric grade is less than:	and greater than or equal to:	the final letter grade is:
-	94	A
94	90	A-
90	87	B+
87	83	B
83	80	B-
80	77	C+

If the final numeric grade is less than:	and greater than or equal to:	the final letter grade is:
77	73	C
73	70	C-
70	67	D+
67	63	D
63	60	D-
60	-	F

Weekly Schedule:

This is a **tentative schedule** which will **DEFINITELY** be adjusted and reordered as the course proceeds. However, it represents the primary topics that will be covered during the class. A detailed reading guide will be posted for each week on D2L under 'documents.' Several guest speakers have been invited and the schedule will need to be adjusted to accommodate their availability.

In addition, several supporting topics will be briefly covered including statistical analysis techniques that will be needed for some of the projects.

Week 1 (January 10th)

Course introduction.

Explanation of individual course-work contracts.

Research/Project ideas discussed

Topic 1: Architectures for Self* Systems (i.e. self-aware, self-healing, self-adapting, self-managing)

Week 2 (January 17th)

Topic 1: Self* Systems (i.e. self-aware, self-healing, self-adapting, self-managing) continued.

Also – Overview of research methods for Empirical Software Engineering

Week 3 (January 24th)

Topic 2: Software product lines and architectures

Week 4 (January 31st)

Topic 2 (continued): Evolution of Software Architectures in Product Lines

Lightening talk: 5 minutes per student on planned topic.

(Online students will record and post to the online forum).

Week 5 (February 7th)

Topic 3: Software Architecture in Agile/Lean environments

Guest speaker is invited for part of the class.

Week 6 (February 14th)

Topic 4: Technical Debt from an architectural perspective

Week 7 (February 21st)**ONLINE ONLY THIS WEEK**

Dr. Huang will be giving a presentation at the Indian Software Engineering Conference in Delhi, India and all materials for this week will be recorded and available via the D2L website.

Topic 5: Online Recording - Reverse engineering, refactoring, rejuvenating, and replacing architectures

Week 8 (February 28th)

Topic 6: Agent-based architectures

Week 9 (March 7th)

Topic 7: More on scalability, extensibility (extension from SE480). Guest lecturer invited on Performance Analysis.

Week 10 (March 14th)

Presentations: All students who opted for the research option are required to give a presentation of their results. Additional topics covered as time allows (depending on how many students take the research option).

Final Week (March 21st)

Final deliverables are due.

Anyone who has opted for a practical project must arrange a time to demo the project to the instructor during finals week.

Academic Integrity

Students should note that this is a zero-tolerance course with regard to academic integrity violations. All students are expected to read and fully comply with DePaul's Academic Integrity Policy, the text of which is available at: <http://academicintegrity.depaul.edu/>

Any reuse of code and/or components **must be clearly specified in the presentations and/or final documents.** Reuse is allowed in this course as long as the reused code is publicly available (i.e. open sourced or otherwise used with permission). However, it should be noted that if code is reused from other sources, then it is expected that the scope of the project will be far more extensive than if code is programmed from scratch. Any questions about scope of the project or code re-use can be discussed with the instructor.

A discussion of academic integrity is presented during the first week of class. If you have any questions about what constitutes an academic integrity violation or what its consequences might be, please be sure to have these questions answered before your first coursework submission.

Communication

All correspondence and communication, such as email and phone messages, must include your full name and course number. The subject line must include SE581.

Exceptional Circumstances

Every effort is made to accommodate students who encounter exceptional personal circumstances during the quarter. Students who experience unanticipated personal, work, health, or family emergencies should notify the instructor by email or phone as soon as possible with a brief explanation of the circumstances and any anticipated impact these might have on coursework. Students who have anticipated exceptional circumstances such as secular

or religious holiday observances, medical treatment, or work-mandated travel should notify the instructor as early as possible of these circumstances and any anticipated impact these might have on coursework. In both unanticipated and anticipated cases, a suitable plan for dealing with the coursework impact is agreed upon by the student and instructor. In some cases, suitable documentation of the exceptional circumstances may be requested by the instructor. In both unanticipated and anticipated cases, students must inform their fellow team members of their inability to participate in the project and make suitable arrangements to make up for missed contributions to the project.

Grade Responsibility

Every effort is made to provide the student with the resources and support needed to succeed in the course. Grades are assigned fairly and impartially based on the coursework submitted by the student, without regard to external circumstances such as GPA goals or employer tuition reimbursement minimum grade requirements. It is the student's responsibility to earn his or her final grade. Please do not ask for a grade which you do not earn.