*SCMP391 Software and System Design*

*Syllabus*

*Spring 2018*

I. *Title*: SCMP391 Software and System Design

II. *Time and Location*: TBD

III. *Instructor*: Jim Skon, Hayes 309C, skonjp@kenyon.edu

IV. *Course description*:

A study of many software design projecta that requires planning, analysis, design, implementation, testing and maintenance. Different methods of planning, definition, requirements analysis, and cost estimation are considered. A central component of the course is a semester long team project which engages a team of 3-5 students in the analysis, design, implementation, and documentation of a significant applied project. The goal of this team project is for the students to engage with the material as they work to solve a real world problem. These projects are real needs of organizations in the surrounding community (including Gambier, Knox county, and at times, beyond). Prerequisite: SCMP118 Introduction to Programming, SCMP318 Software Development or SCMP218 Data Structures or permission of Instructor.

V. *Position of the course in the college curriculum*:

This course is a first year through senior and junior level course required of all majors interested with a background and interest in applied computer science.

VI. *Objectives of the course for the student*:

A. To recognize the significance of using engineering methodologies in software design and development.

B. To learn how to apply software engineering several techniques and methodologies to the development task.

C. How to develop and construct a software specification.

D. How to develop and construct a software design.

E. How to verify and validate software with care.

F. To apply various techniques of software management.

G. To work as the member of a design and development team to develop a product for a real world entity in the community.

H. To establish connections between academic knowledge and civic life.

J. To develop professional communication skills by working within and across communities, context, and social structures.

VII. *Texts*:

1. (IS) Ian Sommerville. *Software Engineering, Tenth Edition.* Addison-Wesley Publishing Company, 2015

X. *Outline*: See Handout

XI. *Method used in the course*:

This course will mostly use a combination of lecture, discussion, team projects work, team project presentations. The team’s projects are a central activity in the course. The project is specifically design to require you to function as a member of a team. Each team is responsible for internal organization and assignment of member responsibilities. People in the group should be assigned to have primary responsibilities for specific aspects. This does NOT mean they do all the work on that aspect, rather that they coordinate, and facilitate the work they are responsible for making sure it is complete and coherent. You should plan for regular (at least weekly) scheduled meetings.

This course is a higher level 4-hour course. Student should expect to spend between 8-12 hours outside of class per week on the class work and project. If you are spending less then 8 hours a week outside of class, you will not be able to keep up with the assignments and work.

The two hour lab will be used for project activities, including group meetings, project milestone presentations, and project user interviews.

**Daily quizzes:** As mentioned above, this class will be discussion oriented. The textbook was specifically chosen for being easy to read. Also – it breaks up networking into many, small chapters, most of which can be covered in a single class period. You will be expected to read EACH CHAPTER prior to each class in which it is covered. A short quiz will be given at the beginning of each class. The quiz will 3-5 questions on the days reading. You will ONLY be allowed to use you class notes as you take the quiz.

**Midterm and Final exams**: There will be a midterm and a comprehensive final examination.

**Software Engineering Team Project**. Each student will need to participate on a software engineering team, working through the processes as a team. As seen in the grading below there are a variety of activities, some team oriented, and some individual. Participation in all team activities, and completion of individual work is essential to this aspect of the course. The activities the team engages in will include the following critical phases, with each phase having an associated document, and presentation.

|  |  |
| --- | --- |
| **Phase** | **Document and presentation** |
| Initial Project Research | Project Proposal |
| Requirement Engineering | System Requirements Specification |
| Research Solution Options | Project Options overview and table |
| Risk Assessment and Management | Risk Tables |
| Project Planning | Project Plan |
| System Design | * System Architecture * System Design Document |
| System Prototype | System Code and Documentation |
| System Test Planning | Test Plan |
| System Delivery | * Operation Documentation * Installation Documentation * System Demo * System Presentation |

**Reflection**. An extremely central part of this course is critical reflection, reflection about the processes used, reflections about preparing for meetings with community partners, reflections about the quality and effectiveness of interactions with community partners, reflections about the methods, advantages, and effectiveness of team work, and reflections about the processes of proof reading, critiquing, and unifying the work produced by the team. This is seen below in the reflection component of each phase.

The goal will be to, for each of the phases above, engage in the following reflective activities:

1. Describe: The student will be asked to describe their engagement with the activity. They will be asked to consider:
   1. How did you prepare before meeting with project principals to gather information for this phase?
   2. Who did you interact with in order to proceed with the assignment? Why with this person? How useful was the information gathered?
   3. In what ways did you as a team? In particular, did you divided the work among team member, and if so, how did you divide it? Alternatively, did you actively work together as a group, and on what parts?
   4. Did team members proof and critique each other’s work? If so, on what aspects/documents?
2. Examine:
   1. How did your interactions with project principals go? Were you appropriately prepared? Did you miss anything important?
   2. Did you use your time, and the time of the project principals effectively as you gather information? How effective were you in recording the information from these meetings in ways that are useful for moving the projects forward.
   3. Compare and contrast the value of working individually on components of the project verses working together actively as a team.
   4. How effective were you in working together to produce output that was well integrated, complete, and error free? Was there unnecessary duplication of error?
3. Articulate Learning:
   1. What did you learn about preparing for interactions with project principals? What will do to prepare for future phases to improve effectiveness and productivity of these interactions.
   2. What did you learn about having effective and productive interactions with project principals? How with you improve record keeping? How will you prepare for follow-up queries?
   3. What did you learn about effective team work? Can you categorize and describe the types of work that is best done individually verses done as a team?
   4. How can you work more effectively to check and improve each other’s work, and to better integrate the corporate work product into a unified, cohesive document?

One unexcused absence is allowed. After that, your final grade will be reduced by 2% per each unexcused absence. Also coming late or leaving early (without a written excuse) will result in a ‘late’. Three of these will count as a full unexcused absence.

XII. *Method of evaluation*:

Grading: Your final grade will be determined in the following way:

Final exam 15%

Midterm Exam 10%

Presentations and demos 20%

Deliverables 20%

Implementation 15%

Reflections for each phase 15%

Attendance, participation 5%

----------------------------------------------

TOTAL 100%

### Calendar

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date** | **Topic** | **Notes Links Readings** | **Quiz** | **Slides** | **Assignment Due** | **Team Assignments Due** | |
| 1-16 | Introduction | 100 Interview Questions for Software Developers |  | Ch1.pptx |  |  | |
| **1-18** | **Lab**: Project presentations |  | Link |  |  |  | |
| **1-20** | **Software Processes**: Software process models Process activities |  | Link | Ch2.pptx | Project Team Application |  | |
| **1-23** | **Software Processes**: Coping with change The rational unified process |  | Link | Ch2.pptx | 2.1, 2.2, 2.3 on page 54 |  | |
|  | Agile software development | Update Wiki with test plan and Risk Assessment |  | Link  Ch3.pptx | Chapter 3 Homework 3.1, 3.5 3.6, 3.8 on page 78 | Resume on wiki | |
|  | **Lab**: Project Brain Storming, Discussion, and student presentations | Project Application Assignment |  |  | Five minute research presentation on at least 2 projects concepts. | | |
|  | Requirements Engineering | System Requirements Specification | Link | Ch4.pptx | Homework 3.1, 3.5 3.6, 3.8 on page 78 | | |
|  | Requirements Engineering |  | Link |  | Project Application   1. Top three project choices (Allocate 10 votes among 3 choices) 2. Rationale for each (qualification, interest) | | |
|  | System modeling |  | Link | Ch5.pptx |  |  | |
|  | **Lab**: Team Formation, teams meet with domain experts Lucent Chart Demo | lucidchart |  |  |  | | |
|  | System modeling |  | Link |  |  | | |
|  | Architectural design |  | Link | Ch6.pptx |  | | |
|  | Design and implementation |  | Link | Ch7.pptx |  | Meeting with notes Moodle Link | |
|  | **Lab**: Intro to UML, UML Lab 1 | System Requirements SpecificationUML.pdf UML Lab 1 | Link |  |  |  | |
|  | **Lab**: UML lab 2 - Use Case | UML Lab 2,  Avoid pitfalls with use cases, Weather Station Use Cases |  |  |  |  | |
|  | **Lab**: UML lab 3- Sequence Diagram | UML Lab 3 |  |  | UML Lab 1 | User Requirement Draft System Requirements Specification Moodle Link | |
|  | Project management |  |  | Ch22.pptx | UML Lab 2 Due | Use Cases on Wiki | |
|  | **Lab**: Present User Requirements and Use Cases Team Meanings | Risk Analysis Assignment |  | Ch22.pptx |  | Team Presentation ruberic  Turn in slide presentations with requirements and use cases | |
|  | Project management, Risk Assessment |  |  | RiskTables.txt |  |  | UML Lab 3 Due |  |
|  | Project Planning |  | Link | Ch23.pptx |  | Update Wiki with Sequence Diagrams |  |
|  | Project Planning |  |  | Ch23.pptx |  | System Requirements Specifications Complete | |
|  | **Lab**: Complete System Specification Presentation, Risk Analysis Presentation |  |  |  | Risk Assessment complete on Wiki | | |
|  | Configuration Management |  | Link | Ch25.pptx |  |  | |
|  | Quality Management |  | Link | ch24.pptx |  | | |
|  | Software Testing |  | Link | Ch8.pptx | System Requirements complete on Wiki | | |
|  | **Lab**: System Architecture Presentations |  |  |  |  | System Architecture Slides | |
|  | Software Testing |  | Link |  |  | System Architecture on Wiki | |
|  | Software Evolution |  | Link | Ch9.pptx |  | | |
|  |  |  |  |  |  | | |
|  | Sociotechnical systems |  |  | Ch10.pptx |  | | |
|  | **Lab**: System Design, Progress Report |  |  |  |  | System Design and Progress Slides | |
|  |  |  |  |  |  | System Design on Wiki | |
|  | ACM Code of Ethics |  |  |  |  | | |
|  | ACM Case studies | Case Studies Assigned: Code Of Ethics Case Studies |  |  |  | | |
|  | **Lab**: System Implementation Plan and Updated Risk Assessment |  |  |  |  | System Implementation Plan and Updated Risk Assessment | |
|  | Sociotechnical Systems |  |  |  |  | Update Wiki with System Implementation Plan and Risk Assessment | |
|  |  |  |  |  |  | | |
|  | Dependability and security |  |  | Ch11.pptx |  | | |
|  | **Lab**: System Test Plan and Progress Report |  |  |  |  | Test Plan and Feature Status Presentation | |
|  | Dependability and security specification |  | link | Ch12.pptx |  | Update Wiki with test plan and Risk Assessment | |
|  | Dependability engineering |  |  |  |  | Ch13.pptx |  |
|  | ACM Case Study Presentations and Discussions | Code Of Ethics Case Studies |  |  | Code Of Ethics Case Studies Presentation | | |
|  | \*Lab\*: Prototype Demo, Project Status Team Meetings | GitLab |  |  |  | Demo Presentation Slides | |
|  | Security engineering |  | link | Ch14.pptx |  |  | |
|  | Software Failure Case Studies | Failure Group Assignment  Software Failure Group Assignments: 1. Ariane 5 Study 2. Therac 25 Study |  | therac25.ppt |  | | |
|  | Software Reuse |  | link | Ch16.pptx |  | | |
|  | \*Lab\*: System Security Plan Presentation System Status Report Team Meetings |  |  |  |  | Security and Status report slides | |
|  | Embedded Software | IoT and Embedded Systems | link | Ch20.pptx |  |  | |
|  | Software Failure Case Presentations | Students Presentations |  |  | Presentation Slides | | |
|  | Embedded Systems - Class Activity | Embedded Class Activity |  |  |  | | |
|  | Final System Presentation |  |  |  |  | FInal Wiki Update, Presentation Slides | |
|  | Final System Presentation |  |  |  |  | | |