

Iowa State University

Software Engineering

Simanta Mitra, Ph.D.
Department of Computer Science
Iowa State University

Overview

- Contact Information
- Course Description
- Course Origin
- Course Design
- Policies

Contact Information

- Contact Info
- Catalog Desc
- Objectives
- Origin
- Design
- Policies

- Instructor: Simanta Mitra
 - B21 Atanasoff
 - smitra@iastate.edu
 - 294-3463
 - Office Hours: MWF 1:10-1:50
- TA: Kai-shin Lu (kslu@iastate.edu)
- Website: On WebCT (not yet up)...
- NO recitations

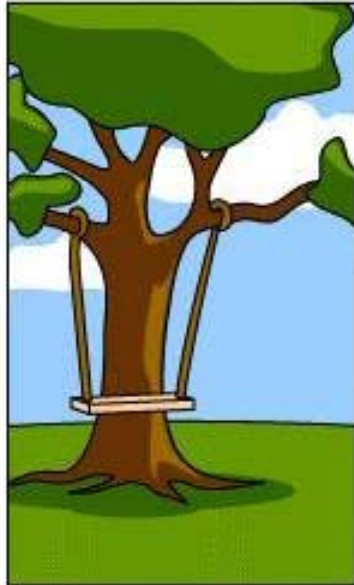
Catalog Description

- Contact Info
- Catalog Desc
- Objectives
- Origin
- Design
- Policies

- Com S 309. Software Development Practices. (3-1) Cr. 3. F.S. *Prereq: Com S 228 with C- or better, Com S 229 or Cpr E 211, Engl 250.*
- *A practical introduction to methods for managing software development. Process models, requirements analysis, structured and object-oriented design, coding, testing, maintenance, cost and schedule estimation, metrics. Programming projects. Nonmajor graduate credit.*



How the customer explained it



How the Project Leader understood it



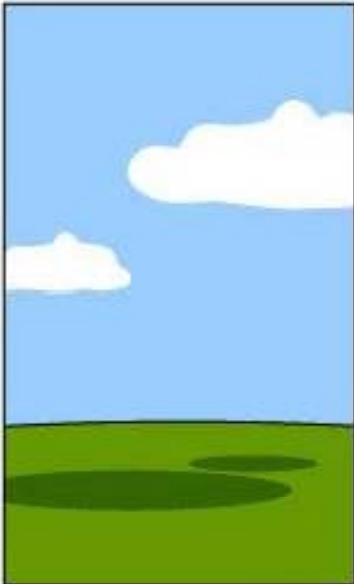
How the Analyst designed it



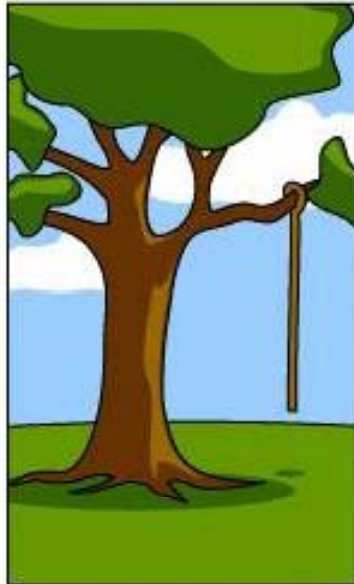
How the Programmer wrote it



How the Business Consultant described it



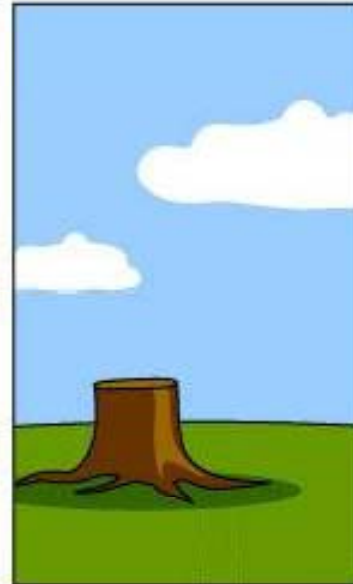
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

Objectives

- Contact Info
- Catalog Desc
- Objectives
- Origin
- Design
- Policies

- A primary objective is to have students **experience** development from concept to release of a reasonably complex software project, following good practices for each of the different stages of the development process, and using the following tools: IDE, Source-Control, Modeling, Test execution, and coverage.
- Another primary objective is to have students be knowledgeable about common software engineering techniques and concepts.

Origins

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

Course Design: Text

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

- ***Software Engineering – An Object Oriented Perspective, Eric J Braude, ISBN: 0-471-32208-3, John Wiley and Sons, Inc***
- Software Engineering, Ian Sommerville, Sixth edition, ISBN: 0-201-39815-X
- Software Engineering – A practitioners approach, Roger S. Pressman, Fifth Edition, ISBN: 0-07-365578-3

COM S 309 LIST OF TOPICS

Basics	Requirements	Architecture	Testing
Introductions			
Course policies	Basics	Basics	Basics
Processes	steps in requirements process	what is	intro to bugs and main issues
Software Lifecycle	types of requirements	4+1 views	SPRAE
major problems in SE		layering and modularity	types of testing
Inheritance and polymorphism	Analysis	abstraction leak	
	context model		Black Box techniques
	structured analysis & DFDs etc	Steps	boundary value
Project Management	object oriented analysis & UML	decomposition	equivalence class
Source Control/Configuration mgmt		specifying services	decision table
Source Control, checkin, checkout	Modeling and UML	specifying interfaces	random testing
commit, update, conflict	UML diagrams	design for NF	
branching/merging, tags	Class Diagram	eval of architecture	White Box techniques
	Sequence Diagram		statement, decision, multi-condition
Project	Collaboration Diagram	Architecture Styles/Patterns	basis path
Project Planning		repository; layered;	
Project Execution	Elicitation	virtual machine; server-client	Other
Project Termination	interviewing/questionnaire	pipes-and-filters	Integration testing
	use-cases	decorator pattern	bug lifecycle & CCB
Risk management	use-case diagram + writeup	façade pattern; proxy pattern	when to stop testing
risk assessment	prototyping and screenshots	factory pattern; builder pattern	reviews and inspections
risk control			Test plan document
	Specification	Design Document	
Cost Estimation	C-Requirements	Architecture Document	The Project
Object Points	D-Requirements	Qualities of Good Arch document	Project Roadmap
Cost Estimation Formula	Attributes of D-Reqs		
COCOMO-II stages	Qualities of Good SRS document		A1 Project proposal
	Inspections		A2 Screenshots
Process Models			A3 SRS-1 Usecases
Waterfall model			A4 SRS-2 Specific Requirements
Iterative Model			A5 DD1 Architecture Components
Spiral Model			A6 DD2 Interfaces
Agile models			A7 DD3 Design Rationale
CMM			A8 TestPlan
			A9 Final Report
			A10 DEMO (incl. Test code)
			BPA

Course Design: Organization

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

- Team Project
 - The course is organized around a team project.
 - Most of the issues in Software Engineering manifest themselves only on larger/complex projects.
 - That is why we require the projects be somewhat complex.
- Lectures
 - Introduces students to the issues, techniques, and practices of software engineering.
 - Part of many lectures is reserved for group work, where each team works to make progress on their project assignment.
 - In addition to providing interaction time, it alleviates scheduling problems that some teams face when their team members have heavy course/job workload during the week.

WK	Week Starts	WHAT YOU NEED TO WORK ON THIS WEEK AS A TEAM	DUE FRI	SPECIAL MEETINGS
1	24-Aug	Forming Teams; Proposals	A0 - form teams due Wed	
2	31-Aug	source control + Context model + Screenshots	A1 - proposal due Tue	Proposal Mtg
3	7-Sep	Screenshots	A2- screenshots	
4	14-Sep	SRS-1 - Use cases etc + de-risking		Risk-1 Mtg
5	21-Sep	SRS-1 - Use cases etc + de-risking	SRS1 - use cases	
6	28-Sep	SRS-2 - Detailed Specs + de-risking		
7	5-Oct	SRS-2 - Detailed Specs + de-risking	SRS2- detailed reqs	
8	12-Oct	Design Doc-1 (Decompose) + de-risking		Risk-2 Mtg
9	19-Oct	Design Doc-1 (Decompose) + de-risking	DD1 - components	
10	26-Oct	Design Doc-2 (Interfaces)		
11	2-Nov	Design Doc-2 (Interfaces)	DD2 - Interfaces	Risk-3 Mtg
12	9-Nov	Design Doc-3 (Rationale)		
13	16-Nov	Prepare for Finals	Test Plan + Final Report	
	23-Nov	BREAK		
15	30-Nov	Design Doc-3 (Rationale)	DD3 - design rationale	DEMO WEEK
16	7-Dec	Prepare for Finals	Test Plan + Final Report	

Course Design: Organization

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

- For UG courses, each credit requires 3 hrs outside class work – i.e. at least **9 hours total outside class for this course per week.**
- Soft skills and Hard skills
- Hands-on class
- Mostly work in groups
- Recent Changes this semester/**year**
 - Poster! (Fall 2008)
 - Switch to SVN (spr 2008)
 - Early and multiple prototyping/coding (spr 2008)
 - Late Architecting (spr 2008)
 - More Instructor/TA/Team meetings (spr 2008)

Course Policies: General

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

- Attendance
 - Full attendance is expected. Part of many classes will involve working with your team to make progress towards development of your project related artifacts. Your absence will hinder your team's ability to make progress. Excused absences will need to be requested in writing (by EMAIL) and will need evidence.
- Academic Honesty
 - Do all assignments by yourself – UNLESS otherwise instructed.
 - Do not talk or peek at others papers or cheat sheets during exams.
 - Develop your own code for project (can look at others codes).
 - University has strict guidelines and we will simply enforce these.

Course Policies: Grading

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

- Grading Scale

We use the guideline of A for excellent, B for good, C for average, D for poor, and F for fail when awarding grades for any assignment.

- Gradable work and Point Distribution

Project Assignments, Quizzes, Final Exam.

Project is worth 70% whereas Quizzes, Exams are worth 30%.

+ points for extra-credits

- points for poor team work, missing classes etc.

- Submission

Work-Distribution form MUST be submitted for All TEAM assignments. Failure to do so will result in a penalty of approximately 5% of the assignments total worth.

Timely submission is expected on all assignments.

Course Policies: Grading

- Contact Info
- Catalog Desc
- Objectives
- Origins
- Design
- Policies

- Extra Credit

- Extra credits are awarded for any assignment that clearly exceeds expectations.
- These extra credits may be awarded to the entire team OR to just a few individuals, depending on the team's work-distribution.
- All extra credit points are totaled and used to make decisions during awarding of final grades.

- Special Accomodations

Any student who feels s/he may need an accommodation based on the impact of a disability should contact me privately to discuss your specific needs. Please contact the Disability Resources Office at 515-294-7220 in room 1076, Student Services Building to coordinate reasonable accommodations for students with documented disabilities.