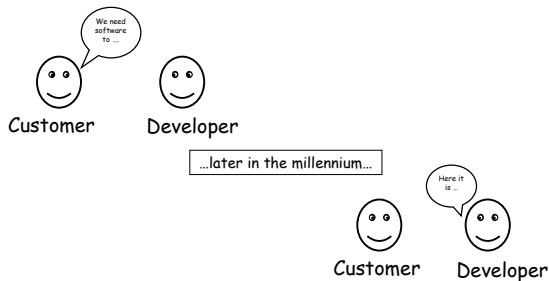


CS121: Software Development

Today

- Overview
 - What is "software development?"
 - What is this course about?
- LAC computers/accounts

"Software Development"



Key Processes

- Requirement specification/analysis
- Design
- Implementation
- Testing

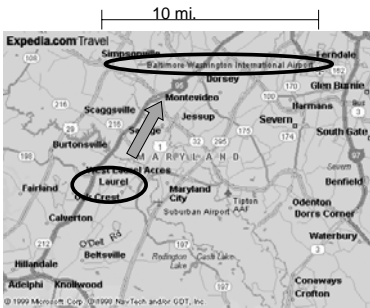
"Software Development"

- Requirement specification/analysis
- Design
- **Implementation** (focus of CS70)
- Testing

Why study software development?

- Society has become increasingly dependent on software systems.

Expedia Maps:
I need to go to the airport



Output reported in *The Risks Digest* Oct. 1, 1999

Excerpts from Expedia Maps directions:

From: Laurel, Maryland

To: Baltimore-Washington International Airport, Maryland

Driving Distance: 5865.1 miles

Time: 9 day(s) 3 hour(s) 22 minute(s)

Time (hour:minute)	Instruction
0:00	Depart Laurel, Maryland
1:01	Entering Delaware
1:17	Entering New Jersey
3:24	Entering New York
3:51	Entering Connecticut
5:51	Entering Massachusetts
7:29	Entering New Hampshire
7:44	Entering Maine
12:20	Entering New Brunswick
20:20	Take the North Sydney-Argentia Ferry
34:32	Entering Newfoundland
36:35	Turn left onto Local road(s) (4543.1 mi)
219:22	Arrive Baltimore-Washington International Airport, Maryland

Why study software development?

- Society has become increasingly dependent on software systems.
- Failures in software systems can be dangerous and costly

Therac-25

- Linear accelerators create high- energy beams that can destroy tumors with minimal impact on the surrounding healthy tissue
- Therac 25 was the first linear accelerator with dosage controlled solely by software (as opposed to hardware)

1983: Pre-release Safety Analysis

- Programming errors have been reduced by extensive testing on a hardware simulator and under field conditions on teletherapy units. **Any residual software errors are not included in the analysis.**
- Program software does not degrade due to wear, fatigue, or reproduction process.
- Computer execution errors are caused by faulty hardware components and by "soft" (random) errors induced by alpha particles and electromagnetic noise.

and then ...

- 1983: First Therac 25 installed
- 1985-1987: Six massive-overdose accidents due to "software error" are reported. Overdoses caused severe burns and death.
- 1987: Recalled for extensive design changes, including hardware to safeguard against software errors in dosage.

Therac-25 Software Errors

- Bugs in program modules **UNIT TEST**
- System errors due to misinterpretations of module interfaces **INTEGRATION TEST**
- Errors in users' guide **ACCEPTANCE TEST**

Why study software development?

- Society has become increasingly dependent on software systems.
- Failures in software systems can be dangerous and costly
- Software design/development is a hard problem

FAA

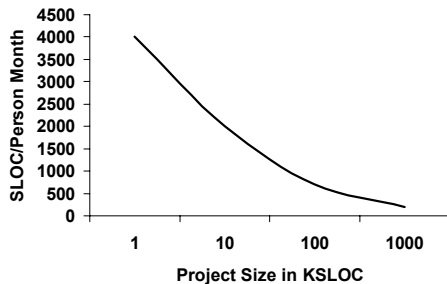
- 1981: FAA announced plans to modernize air-traffic control.
- 1985: IBM awarded contract. System estimate to have 1.5 million lines of code, cost **\$2.5 billion**, and be deployed by 1991.
- 1987: Revised cost **\$4.3 billion**, deployment slipped to 1995.
- 1994: FAA decided that the **project would never be completed**, and cancelled it. Net loss **\$1.5 billion**

Stats on software projects

J. Johnson, "Creating Chaos," American Programmer, July 1995

- 31.1% are canceled before they are finished
- 52.7% overrun their cost estimates by at least 189%
- 33.3% overrun their time estimates by 100%-200%
- 94% of all projects do a "restart"

Large vs. Small Steps: Productivity



Source: Measures For Excellence, Putnam, 1992. Based on 1,600 systems.

[there are] no silver bullets" ... that will do for software productivity, reliability, and simplicity what electronics, transistors, and large-scale integration did for computer hardware

- Frederick J. Brooks, Jr.
The Mythical Man-Month

"Wicked problems are problems that are fully understood only after they are solved the first time."

Rittel and Webber, Dilemmas in a general theory of planning, 1983

Software is a wicked problem...

DeGrace and Stahl, "Wicked Problems, Righteous Solutions, A Catalogue of Modern Software Engineering Paradigms.", Prentice-Hall 1980

software's chronic crisis quiz
(InfoWorld 2002)

The typical software development project takes _____ months (1983).

Ans. 12-23 months

To be considered a large system, a system today must contain at least _____ lines of executable code (1985).

Ans. 50,000

For a medium-sized software system, how many lines of executable source code are typically produced per day per person (averaged over the entire period of development) (1981).

Ans. Less than 10!

The approximate number of errors found in every in every 1000 lines of executable source code during development of a software system is _____.

Ans. 50-60

Most errors found by users in software are the result of

- A. coding errors
- B. difficulties understanding the problem statement
- C. system integration errors
- D. errors in the design of the solution

Ans. B

What percentage of software development projects are judged successful (1995)?

What percentage are judged operational but less than successful?

Ans. 16%, 53%

Software defects cost the U.S. economy _____ per year (2002).

- a. \$100 million
- b. \$500 million
- c. \$1 billion
- d. \$60 billion

Ans. d

Is there hope?

Software engineering:
tools, techniques, and principles to
promote software quality

software engineering is an evolving field

Objectives of CS121

- Understand the problems
- Understand the tools, techniques, and principles that can help
- Practice

Major Topics

- Software Development
 - managing key processes
 - artifacts
- Software Design
 - principles
 - patterns
 - artifacts

Software Design & Development Practice

You'll develop three software projects in this class

- arcade game
- miniature golf game
- computer game of your design

Why games?

Games involve a range of problems that rarely show up in a single software project

- User interface design
- Computer graphics and sound
- Simulation and modeling
- Lots of mathematics
- Real-time
- Other possibilities: AI, networking, etc.

What about graphics

- You'll learn some basic OpenGL that will be more than adequate for your games.
- Yes ... you can do a few all-nighters and figure out some cool effects ... but that is not required ... that is not even recommended.
- This is not a graphics course. If you want to do cool effects take the graphics course.

Software Design & Development Practice

You'll develop three software projects in this class

- arcade game
 - focus: Software development cycle, processes (particularly requirements elicitation/analysis), artifacts of development process
- miniature golf game
 - focus: Software design process, design principles & patterns, communication, artifacts
- computer game of your design
 - focus: Putting it all together

Grades, Texts, etc.

see the course web page

My info

- My email z@cs.hmc.edu
- My office is 2341 Olin, X78360
- My schedule is available on my web page: www.cs.hmc.edu/~z