Analysis and Design *of Enterprise Applications* in UML

Prof. T.H. Tse
Honorary Professor
Department of Computer Science
Email: thtse@cs.hku.hk
Web: hku.hk/thtse .

About the Instructor

Teaching

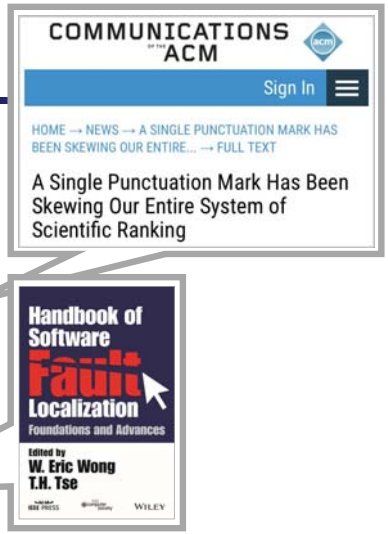
- ***Best teacher award***
- Teacher effectiveness ***94.4%*** in 2022
- Course effectiveness ***93.1%*** in 2022 .

About the Instructor

Teaching

Research

- ***Ranked internationally as no. 2*** among experts in metamorphic testing
- ***Grand champion*** of most influential paper award 2021
- ***New book*** 2023 .



About the Instructor

Teaching

Research

Administration

- ***Intermediary of $140,000,000 donations*** for The University of Hong Kong .

About the Instructor

Selected Past Students

- Associate Professor, City University of Hong Kong
- Associate Professor, Institute of Software, Chinese Academy of Sciences
- Associate Professor, Beihang University .

About the Instructor

Selected Past Students

- Associate Professor, City University of Hong Kong
  - Associate Professor, Institute of Software, Chinese Academy of Sciences .
- Associate Professor, Institute of Software, Chinese Academy of Sciences
- Associate Professor, Beihang University

About the Instructor

Selected Past Students

- Apple
- Alibaba
- Executive Director in Goldman Sachs
- Further study in Carnegie Mellon University
- Further study in University of California, Berkeley
- A+ in MSc Dissertation 2022 .

About the Tutor

Jolly Cheng mycheng@cs.hku.hk

- MSc(CompSc) with distinction
- Grade A in most courses
- ***Best Tutor Award*** in 2015 .

Important Links

- Moodle website for the course
  https://moodle.hku.hk/course/view.php?id=96555
  or https://www.cs.hku.hk/~c7201
- News and open discussions forum for the course
  - Via the Moodle website
- To apply for CS account
  https://intranet.cs.hku.hk/account/
- Technical problems support@cs.hku.hk .

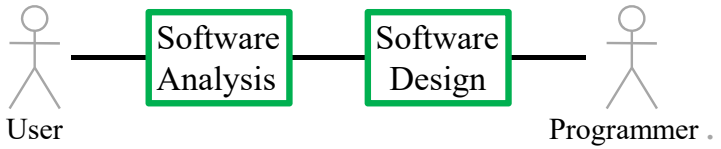
Open discussions forum is an important element of this course

Assessments

- 3 assignments 40%
- Examination 60% .

What Do You Learn from this Course?

We cover



.

What Do You Learn from this Course?

We cover

- Concepts
  - Object-orientedness
- Methodologies
  - Unified process
- Techniques
  - UML diagrams
- Tools
  - CASE tools .

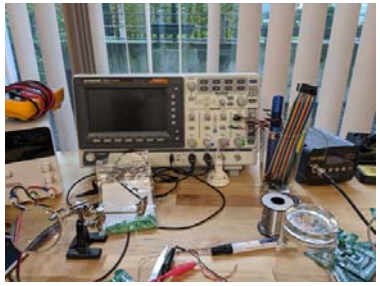
Prime concept in programming

Prime methodology in SE

Prime technique in SE

What Don't You Learn from this Course?

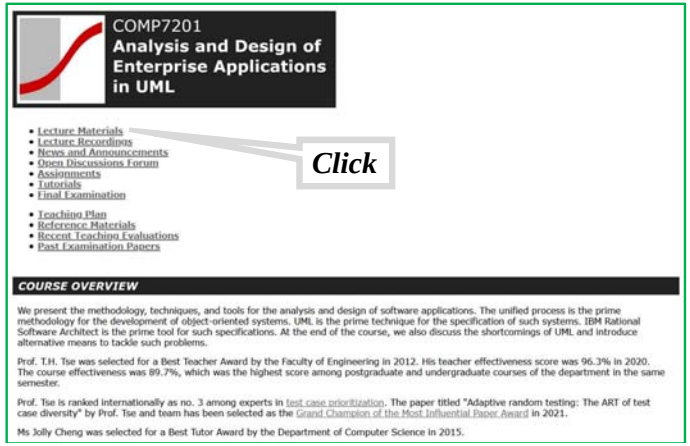
- ☹ Software engineering fundamentals?
- ☹ Systems analysis fundamentals?
- ☹ Object-oriented programming?
- ☹ Technician skills?



What Do You Learn from this Course?

- Will this course be taught at an abstract level?
  - It will be taught as a real-life practical course
- Will it be a theoretical course?
  - It will be a real-life practical course
- What programming language will be taught?
  - Programming is *not* part of the course
- How can the course be practical if it does not involve programming?
  - We learn from real-life examples …

We Learn from Real-Life Examples
Our Moodle Website

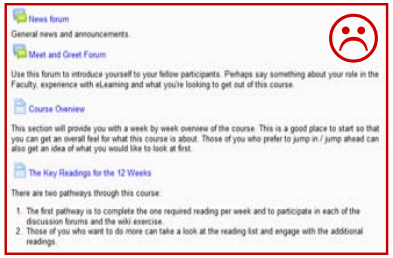


We Learn from Real-Life Examples
Our Moodle Website

We Learn from Real-Life Examples

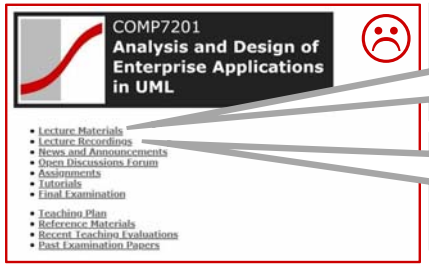
Our Moodle Website

- Demonstrates that a good design with ***interacting components*** and ***graphical user interface*** is more interesting



We Learn from Real-Life **Mistakes**

Our Moodle Website



Hidden hardcoded hyperlink
`http://www.cs.hku.hk/course/20XX/c7201/THTse/lectures.html`
changes every year

Hidden hardcoded hyperlink
`http://moodle.hku.hk/course/view.php?id=12345#section-2`
may change at any time

- Demonstrates that object-oriented design with good ***contractual interfaces*** (pre-agreed interfaces between interacting components) is necessary .

What Else Do You Learn from this Course?

- ☺ How to sell your project
- ☺ How to sell yourself .

Is This Course Easy?

No!

- This course is about real life
- Real life is challenging
- Especially if you are a team leader

Rather than following instructions, you need to understand concepts and make decisions

"Never, ever, let management promote you to an analyst, designer, or team lead. It will promote you out of programming" .

Is This Course Easy?

No! ☹

- Software analysis is user-driven

  - User requirements are challenging
  - We will ***deduct*** marks if you invent user requirements
  - In real life, your senior manager will not like it either .

Is This Course Easy?

No! ☹

- Software analysis is user-driven
- Software design is a creative process

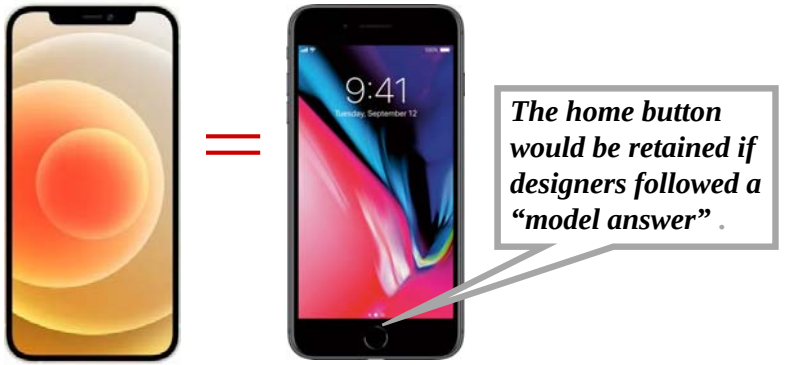
  - Software is not a product to be reproduced
  - It is a product to be created innovatively
  - ***Not*** part of the user requirements .

We Learn from Mistakes

Misconception in Software Design

What if Designs were not Creative?

iPhone 13 iPhone 8



The home button would be retained if designers followed a "model answer" .

Is This Course Easy?

No! ☹

- Software analysis is user-driven
- Software design is a creative process
- No model answer in real life

You must liberate yourself from
- Past examination papers
- Model answers .

Is This Course Easy?

No! ☹

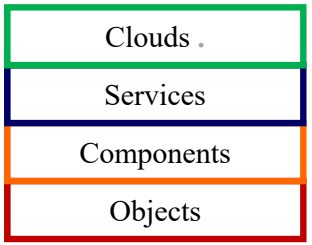
- Software analysis is user-driven
- Software design is a creative process
- No model answer in real life
- No theory in software engineering
- Step by step procedure proved to be problematic
- Learn from experience, with trial and error .

Is This Course Practical?

- Is it practical to draw so many diagrams?
  - Why don't we assign a programmer to code directly?
- Millions of lines of code in a typical system
- How many lines of code per day?
  - 10 lines per day 一日十行
- How many person years?
  - 300+
- Which is more practical: Direct programming or teamwork with well-controlled process? .

Is This Course Practical?

- Why object-oriented analysis and design?
  - Why not component-based development?
  - Why not service-oriented software development?
  - Why not cloud computing?
- Object-oriented analysis and design lay the foundation



Your Longer Term Expectations from this Course?

- ☹ An easy degree? — ***This course is not easy (by nature)***
- ☺ A good job? — ***Yes!***
- ☺ A good professional career? — ***Yes!***

Course Outline

Subject to change depending on your background, interests, and feedback .

- Introduction
- Fundamentals in relation to software engineering
- Fundamentals in relation to object-orientation
- Introducing the unified process
- Object-oriented analysis
- Practical guidelines for identifying classes and relationships
- Dynamic modelling with sequence diagrams
- Dynamic modelling with state machines
- Time and activity modelling
- Object-oriented design
- Introducing design patterns
- Introducing enterprise applications
- Shortcomings of UML
- Formal methods and beyond
- Modern Web design

Textbook?

Unfortunately, no single book can possibly cover OO design and programming in real depth .

Allen Holub

Advanced References

- *OMG Unified Modeling Language* (*OMG UML*) Version 2.5.1, Object Management Group, Needham, MA (2017), http://www.omg.org/spec/UML/2.5.1/PDF
- C.A.R. Hoare, *Communicating Sequential Processes*, Prentice-Hall (1985), http://www.usingcsp.com/cspbook.pdf

☺ Provides more breadth and depth to ***experienced*** students
☹ ***Not*** for students who cannot understand the course
☹ You should ask the professor .

Lectures

Lectures are lectures are lectures

- Slides ≠ notes
  - Interesting but not excessively detailed?
  - Detailed but not interesting?

Complaints from students who skip lectures

Complaints from students who attend lectures .

Lectures

Lectures are lectures are lectures

- Slides ≠ notes
  - Interesting but not excessively detailed?
  - Detailed but not interesting?

I will cater for students who attend lectures .

Lectures

Lectures are lectures are lectures

- Slides ≠ notes

What Fundamental is the Most Important in Software Engineering?

- Let us learn from successful enterprises

Lectures

Lectures are lectures are lectures

- Slides ≠ notes

An Interesting Exercise

- Mr George Soros
- Mrs Soros
- Me
- Mrs Tse ...

I will drive

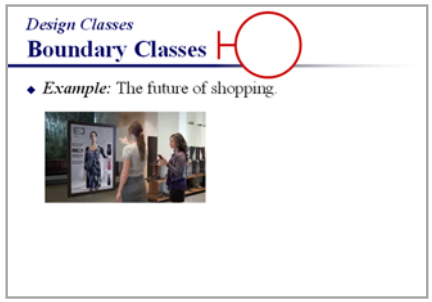
Who will sit next to me?

You are invited to ***analyse*** this problem .

Lectures

Lectures are lectures are lectures

- Slides ≠ notes



Lectures

Lectures are lectures are lectures

- Slides ≠ notes
- What I discuss may be more important than the slides
- Your feedback is also important .

Causes of Dull Lectures

Poker-face students .

One-way communication

Causes of Dull Lectures

- I dared not ask for your help
- A good student should not have problems
- A good student should always deliver

You should work with your supervisor as a ***team*** in order to excel .

Your Feedback is Important

I will

- target for the average students
- inspire the best students
- help the weaker students

You need to tell me during the course .

Feedback from Previous Students

Teacher Effectiveness

- 2022/23 **94.4**
- 2021/22 **93.2**
- 2020/21 **96.3**
- 2019/20 **83.3**
- 2018/19 **89.6**
- 2017/18 **86.0**
- 2016/17 **89.2**
- 2015/16 **95.2**
- 2014/15 **95.0**

- Professor tries to interact with the class a lot, which I like
- Unfortunately we had a rather shy class .

Feedback from Previous Students

- Prof. Tse has given me some very good directions, which affects my career directly
- This is the best course (real
- It is a really meaningful course!
- Rich and stimulate thinkin
- I really enjoy the course T
- Really intriguing
- I found your course challe
- I enjoyed this course very
- Very good in every aspect
- I think he is the best teach
- You are one of my most respectful teachers who taught me a lot of things that I always feel grateful about
- It was a pleasure learning from you this semester Thank you for the amazing lectures! ... I feel they had been my best class in my last 2 years here
- I was impressed by your expertise and experience among this domain
- Your lectures are the most interesting ones (in my opinion) in CS. And you are the first one who makes me feel software engineering is fun. Thanks for your teaching.
- your undergraduate course already learned so much not having met you earlier ...
- ncept! Good teacher!
- anks
- with good examples
- ledge to explain professional problems. It made us very easy to understand the key point of this course
- You are such a great professor
- You are really a very responsible and professional professor
- He is so good
- Your lessons are great and I appreciate them

Long Term:
Prof. Tse has given me some very good directions, which affects my career directly .

Feedback from Previous Students

- You are really a great teacher. You are the first teacher I've ever encountered throughout my undergraduate and postgraduate studies who would *personally* reply to *every* student's enquiry with patient and detail answers. Thank you very much
- Prof. Tse ans ions from students in an impressive and respo ich is of a great hel
- You can a few co
- Very wil
- Willing t
- A kind t may not
- Prof. Tse is a very committed teacher. He takes and answers questions from student in an impressive and responsible manner, which is of a great help in gaining full understanding of the material
- You are a nice teacher, I like you, and I admire you
- Up to now, the materials covered in the course are exactly those I am encountering, or the knowledge I need in my ... job, and I am sure you, no matter what jobs you are planning, will find useful in the future. Do pay some effort to this course
- My own opinion coincided with the evaluation result and I strongly feel that the course was taught pared with ars tructions e concern and in y moved by your hard work and admire you
- I am proud of being your student
- Prof. Tse has given me some very good directions, which affects my career directly
- I really appreciate your teaching method. It is totally different

Intermediate Term:
You are the first teacher I've ever encountered … who would ***personally*** reply to ***every*** student's enquiry with patient and detail answers .

Feedback from Previous Students

- Your course is very good, you give an penetrating analysis and you make it easy to understand
- Everyone who has ever taken your courses knows of your commitment to excellence in teaching
- Your slides are well-presented, concise and good
- Thank you for the quality of the lectures, as well as for the energy you put to teach the cours… effort you make to ensure the slides are … understand
- Excellent presentation and course mater…
- They told me several times that my presentation was very impressive — and this is because I learned a lot of presentation skills from you!
- Encourage students for questions
- I really appreciate your teaching
- The assignment series of this course is really great that it provides a complete example of applying the unified process, from analysis to design
- Our gratitude and respect for your effort in helping us to be a full-fledged and creative individual
- Indeed what you have taught me is very solid (not only research, but also about personality)
- Very interesting
- Very useful for daily jobs
- …od. Good
- …very very very helpful
- …ent professor
- Vi…
- Cool!
- Cool jokes
- Very humorous and very helpful!
- Full of passion
- Keep up the good work

Short Term:
Cool jokes .

Not All Comments are Positive

Always don't know what you are talking about

- Please ***interrupt*** me when you don't understand
- Do not wait until the course is over .

Not All Comments are Positive

Please focus on assignments and exam and the final grade, instead of asking students to "think and guess"

- This is an MSc course for professional career
- No model answer to guess from your real project manager .

Not All Comments are Consistent

I need more examples

The course gives us many examples

- Please ***interrupt*** me when you don't understand
- Do not wait until the course is over .

Not All Comments are Consistent

Too difficult: too much assumption on student background

As the teacher tries to balance student background, teaching material is a little bit too easy

Don't adjust the course difficulty

Too simple

It is ***your*** opinion that counts .

Some Comments are Consistent

3 hour class is too much; better break it into two classes of 1.5 hour each

The lecture length should be split into 2

Done .

Feedback from Us

Assignments

- Return grades in three weeks
- Overall comments in Moodle
- ***Further discussions*** in Moodle

Examination

- Overall comments in Moodle .

We Learn from Mistakes

- Learn also from other people's mistakes
- You do not need to pay the price of making these mistakes yourself .

We Learn from Mistakes

- I have learnt everything about Java
- What should I learn next?

In real life, programming is not everything .

We Learn from Mistakes

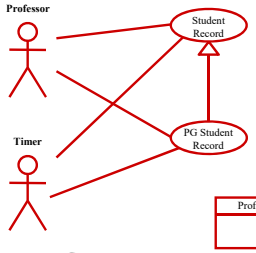
We Learn from Mistakes

What Is Multiplicity?

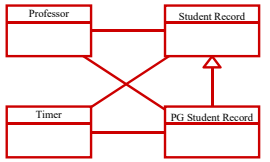
- Multiplicity is the number of instances one class relates to ONE instance of another class.
- For each association, there are two multiplicity decisions to make, one for each end of the association.
  - For each instance of Professor, many Course Offerings may be taught.
  - For each instance of Course Offering, there may be either one or zero Professor as the instructor.

IBM

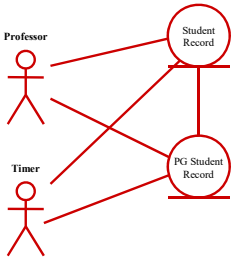
We Learn from Mistakes



Use Case Diagram

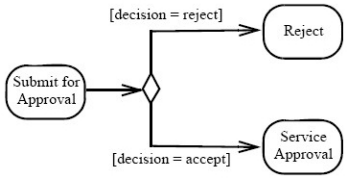


Class Diagram

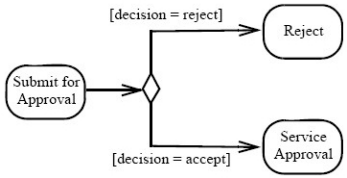


Communication Diagram

We Learn from Mistakes

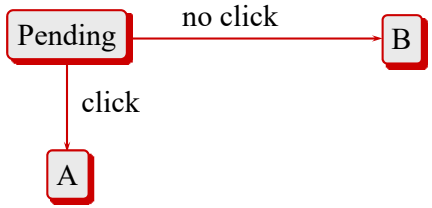


State Machine

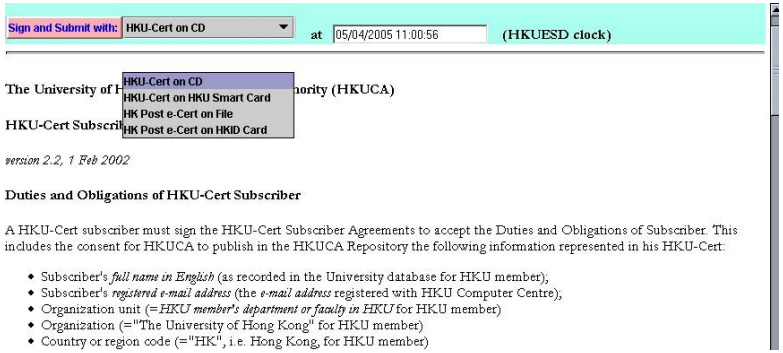


Activity Diagram

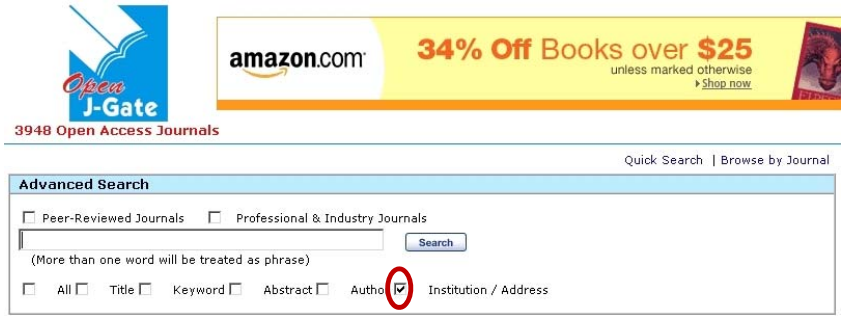
We Learn from Mistakes



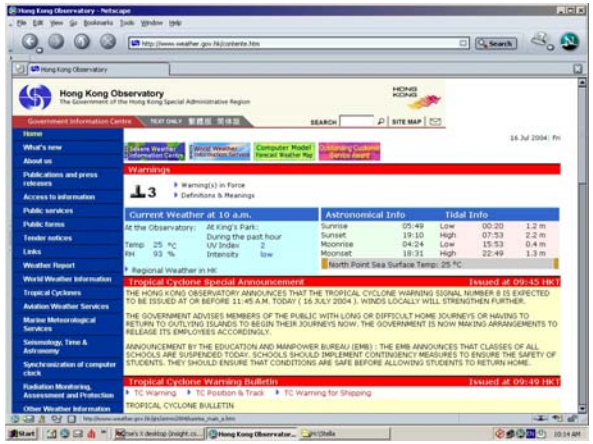
We Learn from Mistakes



We Learn from Mistakes



We Learn from Mistakes



We Learn from Mistakes

We Learn from Mistakes

THE UNIVERSITY OF HONG KONG

FACULTY OF ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE

CSIS0297 Introduction to Software Engineering

Date: May 22, 2007 Time: 9:30am-12:30pm

Candidates may use any calculator which fulfils the following criteria:
(a) it should be self-contained, silent, battery-operated and pocket-sized; (b) it should have numeral-display facilities only and should be used only for the purpose of calculation; (c) it should not have any printing device, alphanumeric keyboard, or graphic display; and (d) it should not contain any recorded data or program. It is the candidate's responsibility to ensure that the calculator operates satisfactorily and the candidate must record the name and type of the calculator on the front page of the examination scripts. Lists of permitted/prohibited calculators will not be made available to candidates for reference, and the onus will be on the candidate to ensure that the calculator used will not be in violation of the criteria listed above.

We Learn from Mistakes

- **23%** of all projects undertaken by internal information systems organizations are cancelled before completion
- **49%** of projects cost **189%** of their original estimates
- Only **28%** of projects are completed on time and within budget
- Projects completed have **42%** of the originally proposed features or functions .

Not Just Small Problems ...

Examples of Major Software Crisis

South China Morning Post

► Computer bug hits freight ► 10,000 bags left behind ► Officials 'very concerned'

Chaos forces cargo to Kai Tak

Not Just Small Problems ...

Examples of Major Software Crisis

- Hong Kong Airport is not alone
- Denver Airport Automated Baggage System
- American Airlines Sabre System
- Advanced Aviation System .

Denver Airport Automated Baggage System

- Designed for brand new international airport
- Comprised of 100 networked computers
- Planned to deliver 60,000 bags per hour from more than 12 gates
- Estimated cost US$200M
- Delayed the grand opening of the airport four times .

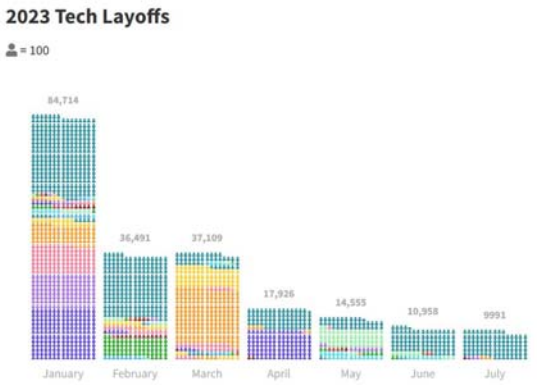
American Airlines Sabre System

- US$2B flight reservation system
- Shining example of a strategic information system to enable American to be one of the world's largest airlines
- An attempt to add hotel and car reservations to the system collapsed with a write-off of US$165M .

Advanced Aviation System

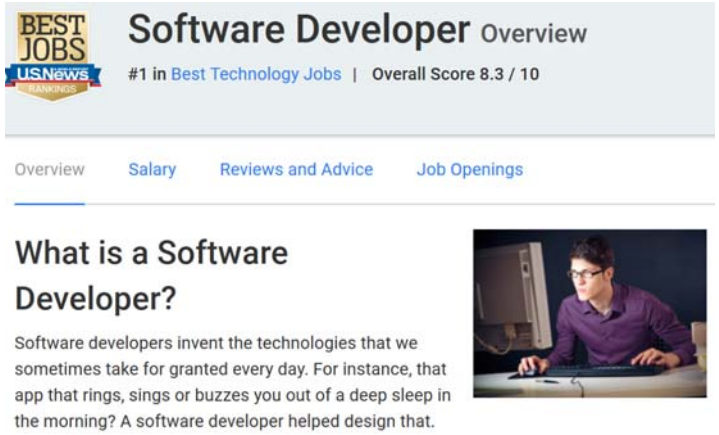
- FAA's next generation Air Traffic Control System was conservatively estimated to cost US$500 per line of code
  - Five times industry average for well-managed development process
- Center for Naval Analysis further found that the actual cost was US$900 per line
- "On average, every line of code developed needs to be rewritten once" .

No Wonder ...



On the Other Hand

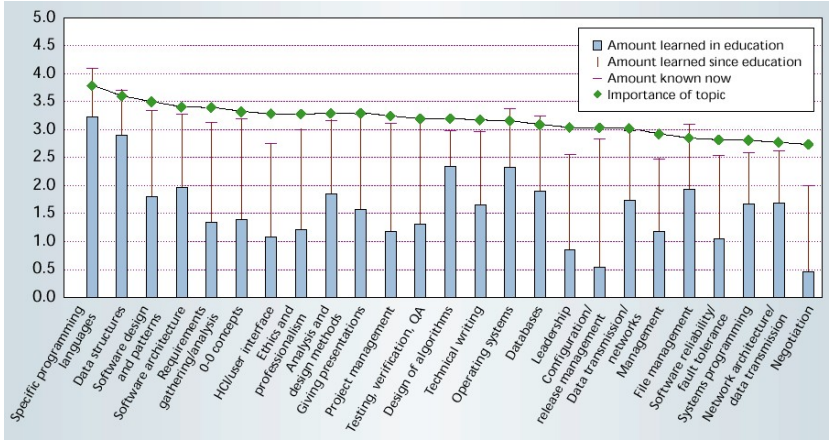
No. 1 in Best Technology Jobs



Mismatch?

25 Most Important Topics

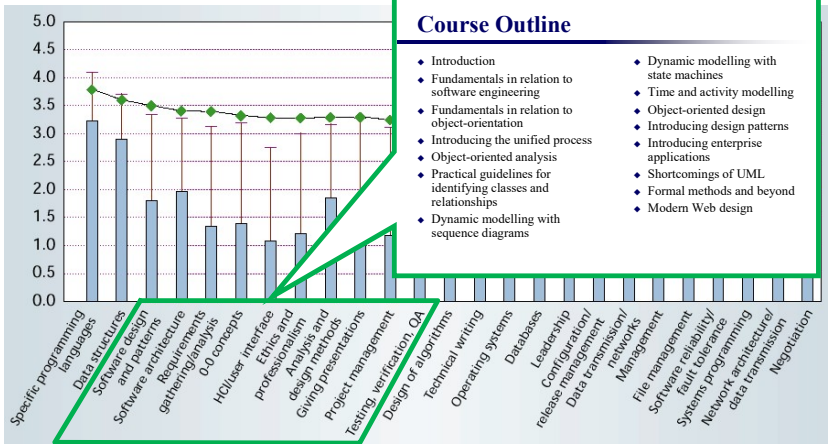
Ref: Lethbridge



Mismatch?

25 Most Important Topics

Ref: Lethbridge



Course Outline

- Introduction
- Fundamentals in relation to software engineering
- Fundamentals in relation to object-orientation
- Introducing the unified process
- Object-oriented analysis
- Practical guidelines for identifying classes and relationships
- Dynamic modelling with sequence diagrams
- Dynamic modelling with state machines
- Time and activity modelling
- Object-oriented design
- Introducing design patterns
- Introducing enterprise applications
- Shortcomings of UML
- Formal methods and beyond
- Modern Web design