

EECE 310: Software Engineering

Ali Mesbah

@amesbah

Today's Objectives

- Getting to know each other
- What is this course about?
- How to successfully pass this course

Contact Info

Lecturer:

Dr. Ali Mesbah

Email:

amesbah@ece.ubc.ca

Twitter:

@amesbah

Website:

<http://www.ece.ubc.ca/~amesbah/>

Office:

KAISER 4044

Office hours: Thursdays 14:30-15:30 (or by
appt.)

Software AnaLysis and Testing (**SALT**)

<http://salt.ece.ubc.ca>

Group size:

- 6 full-time PhD, 6 MASC,
and 2 MEng students

Research:

- Software engineering
- Software testing
- Web and mobile engineering

Teaching:

- Software engineering
- Software testing
- Empirical software
engineering research

Partners:

- Close collaboration with ~12
different companies

SALT Lab Mission Statement

*"novel and automated techniques
for supporting software
dependability and evolution*

SALT Lab

Program Comprehension

Fault Localization

Debugging

Test Automation

Research Agenda



OptimusQA
Your Trusted QA Partner



Microsoft®



Google™

quickmobile



LOCKHEED MARTIN



FusionPipe
SOFTWARE SOLUTIONS
Fueling the Mobile Internet Revolution
Solutions for mobile web, mobile commerce, and mobile advertising

You are...?

TAs

- Keheliya Gallaba, kgallaba@ece.ubc.ca
- Dawood Al-Masslawi, masslawi@mail.ubc.ca
- Saba Alimadadi, saba@ece.ubc.ca

Resources

- Connect
 - <http://elearning.ubc.ca/connect/>
- Piazza (discussions)
 - <https://piazza.com/ubc.ca/winterterm12014/eece310>
 - Enroll!
- GitHub (Lab assignments)
 - <https://github.com/UBC-EECE310>
 - Create an account on GitHub if you don't have one!
 - Submit your account ASAP (see Connect -> Announcements)

Connect

- Syllabus (read it!)
- Lecture Notes
 - After each lecture
- Reading material (articles)
- Lab assignments

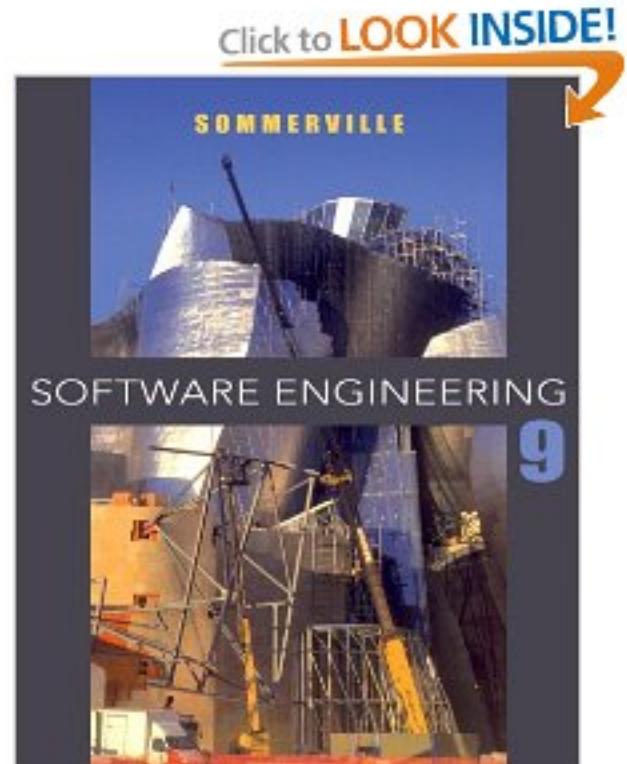
Check Connect regularly for announcements and assignments!

Textbook

Software Engineering

9th Edition, Ian Sommerville

Recommended,
but not *required*



SE books you must have on your bookshelf

- **Clean Code: A Handbook of Agile Software Craftsmanship**
- **Agile software development: principles patterns and practices**
- **Design Patterns: Elements of Reusable Object-Oriented Software**
- **Code Complete: A Practical Handbook of Software Construction**
- **Refactoring: Improving the Design of Existing Code**
- **Working Effectively with Legacy Code**
- **The Pragmatic Programmer: From Journeyman to Master**

Assessment

- Participation 5%
- Presentations 10%
- Midterm 15%
 - Oct. 15th, in-class
- Lab work 30%
- Final exam 40%

Must pass the *exam* and *project* to pass the course!

Active Participation

Good participation:

- Class attendance
- Asking good questions
 - in-class and on **Piazza**
- Answering questions

Bad participation:

- Absence, indifference, sleeping, checking social media, playing games in class!

All Unread Updated Unresolved Hidden

Pinned

This is a sample question.

Questions

How do I ask a question?

How do I answer a question?

How do I search for a question or answer?

What is a followup discussion?

Who marks a followup discussion resolved?

Why the name Piazza?

What does an unanswered question look like?

How do I use LaTeX?

How do I format code?

Can I ask a question anonymously?

Can I answer or edit anonymously?

Is anonymous really anonymous?

How do I attach a file?

Can I embed images?

Can I post announcements?

Can I post so only instructors see my question?

How do I tag my question?

Can I view edit history?

How do I hide a question?

question. 255 Views, 2 Follows

Actions



How do I ask a question?

Last updated by Anonymous 2 months ago

Good question! Follow question

Instructors' response.

Actions



Improving the Instructors' Response

Build on this response with your fellow instructors by clicking "Edit Answer" below.

Last updated by Piazza Team 2 months ago

Edit Answer Good Answer! Ask a Followup »

followup discussions.



Starting a Followup Discussion

Have questions or comments after reading the above? Start a followup discussion here to get clarification.

Still Confused? Ask New Followup

Average Response Time: Special Mentions:

N/A

There are no special mentions at this time.

Online Now | This W...

46 | 624

Lab work

- All labs are in **MCLD 348**,
 - See UBC course calendar to find your lab section/hour.
- Working in teams (randomly picked by TAs)
 - Instructor/TA evaluations
 - Peer-evaluations

More details will follow soon!

Lab Assignments

- 5 assignments
- Based on a Java game called JPacman
- You will learn/use: Scrum, UML, Git/GitHub, Java, Maven, Eclipse, JUnit, etc.

Lab Project

- You select and contribute to a real-world open source software on GitHub
- Examples of contributions:
 - Fixing a reported bug
 - Adding a requested feature
 - Documenting the code
 - Cleaning up the code (refactoring)
 - etc
- You have to send your contributions as a ***pull request*** to the project on GitHub
- Write a **report** and submit via Connect
- **Bonus:** if your pull-request is accepted you get extra points!

Lecture Style

- This will vary!
- Sometimes the lectures will be very typical, where I talk on a certain topic
- Sometimes mini-lectures plus in-class exercises -- here's where participation comes in!
- Maybe guest speakers!
- Presentations by you on hot SE related topics and tools

Student presentations

- Each team will present only once
- Topic should be related to software engineering
- Submit 3 topics you find interesting.
- We'll pick one and allocate a timeslot

Questions?

What is Software Engineering?

- People working *together* to create a *dependable* software system that satisfies the client's *requirements*.
- This involves facing technical and social (interpersonal) challenges!

SE vs typical class projects

	Class project	Industry project
Time	1-2 months	Years
People	1-2	10 - 100s
LOC	100s	Millions
Capital at stake	0 (only a grade)	Millions of \$

What do you want to learn in this course?

Write down in bullet form, and **hand it in**.

No class on Friday (5 Sep)

- See you on Monday!