



Software Engineering

INTRODUCTION

Class Rules

No compromise on Classroom discipline

- ▶ Bring your own pen and a register/copy/paper.
- ▶ No Food, Tea, coffee.
- ▶ No Unnecessary IN/OUT from Lecture
- ▶ Mobile phones shall be OFF/SILENT
 - ▶ RS 1000/- Fine.
- ▶ No gossips/talks during Lecture.

- ▶ Please leave the Lecture Room after your attendance if Something bothers you more important than Lecture.

Marks Distribution

Assessment Type	Weight
Quizzes	7.5
Assignments/Deliv.	7.5
Class Participation	5
Project	10
Sessional Exams 1	15
Sessional Exams 2	15
Final Exam	40
Total	<u>100</u>

How do you kill this monster?

“There is no silver bullet”

- A good Software Engineer knows several methods, tools, and techniques:
- How does it work?
- What are the benefits?
- What are the drawbacks?
- Is it good for my situation?



Fred Brooks, 1986

"No Silver Bullet — Essence and Accidents of Software Engineering", *Proceedings of the IFIP Tenth World Computing Conference*.

- By the end of this course, you will--
 - Be able to define and describe the phases of the software engineering lifecycle.
 - Be able to explain the role of key processes and technologies in modern software development.
 - Be able to productively apply instances of major tools used in elementary SE tasks.
 - Design and implement a portfolio-worthy software engineering project in a small team environment that can be showcased to recruiters.

Software Engineering is about People

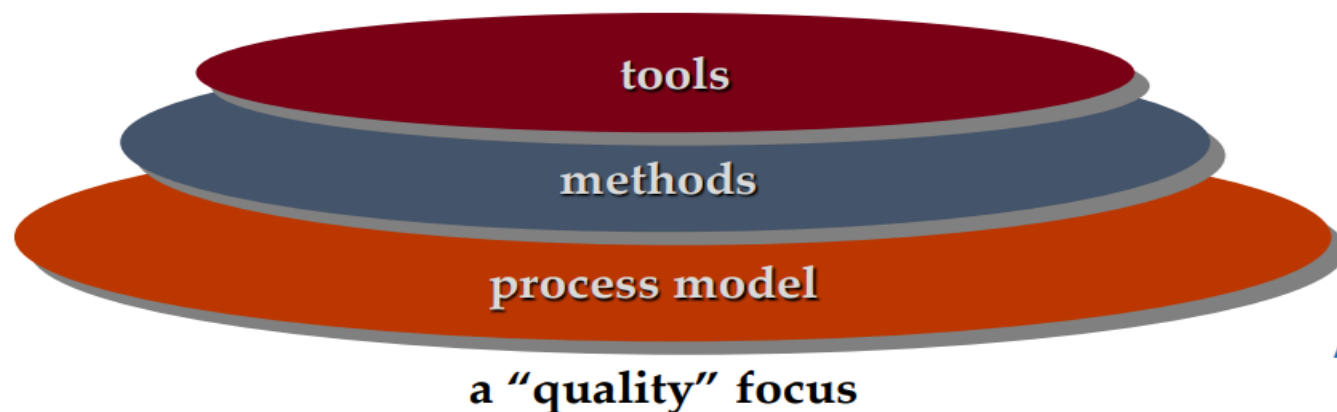
“Any fool can write code that a computer can understand. Good programmers write code that humans can understand”

- Martin Fowler



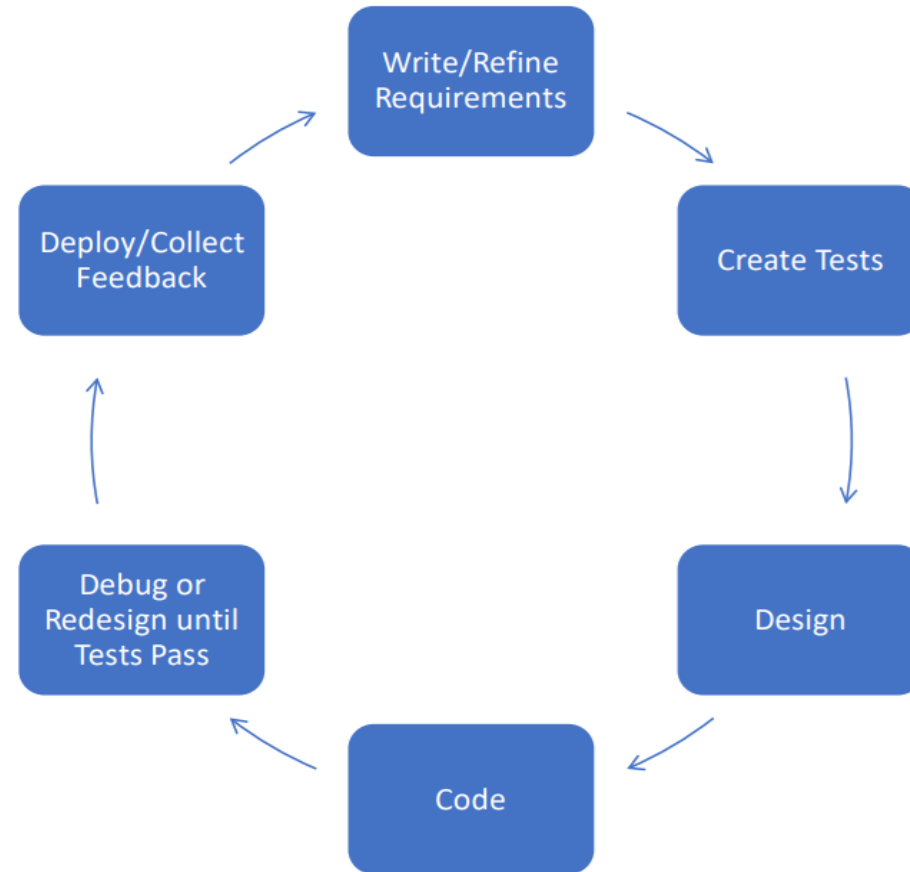
What is software engineering?

- Software Engineering refers to the tools and processes that we use to
 - design,
 - construct, and
 - maintain programs
 - over time.



Applying “engineering” to Software!

Software Engineering encompasses the entire software development life cycle



But this raises many questions

- How big is each cycle?
 - In code to be written?
 - In time?
 - In person-power?
- Can you have multiple cycles going at once?
- What artifacts need to be produced at the end of each stage?
 - Need to prepare for the next time through the cycle.
 - Need to document what was done, so that others can build on your work.

The answers depend on many factors

- Depends on things like:
 - the size of the team
 - the size of the product
 - the longevity of the product
- There's no one "right" way; there are always tradeoffs.
- But there are best practices, which we will expect you to follow.

Course Topics

You will learn about

- ▶ Software Engineering Fundamentals
- ▶ Software process models
 - ▶ Traditional (Waterfall)
 - ▶ Non-Traditional (Scrum)
- ▶ Software process improvement models
- ▶ Software Requirement Engineering
- ▶ Software Analysis and Design
- ▶ Software Architecture
- ▶ Software Testing
- ▶ Software Project Management
 - ▶ Cost estimation
 - ▶ Time estimation

Other Goal



TEAMWORK



GOOD
COMMUNICATION



PROACTIVE
APPROACH



ACADEMIC
HONESTY

Course Books

- ▶ Software Engineering, Ninth Edition, 2010.
Sommerville, Ian Addison Wesley

Reference Material

- ▶ Software Engineering: A Practitioner's Approach, Pressman, R.S. & Maxim B., 8th Edition (2015), McGraw-Hill.
- ▶ SE and Testing, b. B. Agarwal s. P. Tayal m. Gupta, Jones and Bartlett Publishers.



Software
Engineering

What is Software Engineering?

- ▶ As defined in IEEE Standard 610.12:
 - ▶ *The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.*

CLOs

At the end of this course
student should be able to:

1. Select an appropriate software development process for a software project
2. Develop a model of requirements for a software system
3. Design architecture of a software system by choosing the most appropriate architecture styles
4. Design test cases for a software system
5. Construct reasonable sized software in team setting

