

Contents

Introduction	1
What is software engineering?	1
The ‘Art’ or Programming	1
What’s wrong with the ‘art’?	2
If you’re engineering it	2
Topics Covered in this Module	2
Activities Expected from You	2
Assignments - Details	2
Personal Software Process	3
Evidence Based Software Development	3
Your Process	4
Assignments	4

Introduction

- Dr. Vivek Nallur
- vivek.nallur@scss.tcd.ie
- [Website](#)

What is software engineering?

Engineering principles and practice that convert the art of programming into reliable software product/projects.

The ‘Art’ or Programming

Virtues of a good programmer (according to Larry Wall)

- Laziness
- Impatience
- Hubris

What's wrong with the 'art'?

- I'll design it as I go along
- I don't need to write down my ideas
- It'll be done when I'm done

If you're engineering it

- You need to know *what* to build
- You need to know *how* to build it
- You need to build it
- You need to *document* what you're doing
- You need to do it on time

Topics Covered in this Module

- Requirements
 - Elicitation, specification, validation
- Design
 - General concepts, key issues
- Architecture
 - Structures, styles, patterns
- Testing
 - Techniques, processes
- Software Engineering Processes

Activities Expected from You

- Listen in class (your exam will be directly based on discussion in class)
- Complete your assignments (every week)
- Reflect on your assignment feedback
- Reading list (Read these *before* you come to class)

Assignments - Details

- Account for 70% of your final grade
- Programming assignments
 - Well-defined input/output

- Submission via source-control
- Your code will be tested against test inputs

Personal Software Process

- Invented by Watts Humphrey at Software Engineering Institute
- Structured programming process intended to allow programmers to better understand and improve themselves
- Main components
 - Scripts
 - Tracking
 - Postmortem

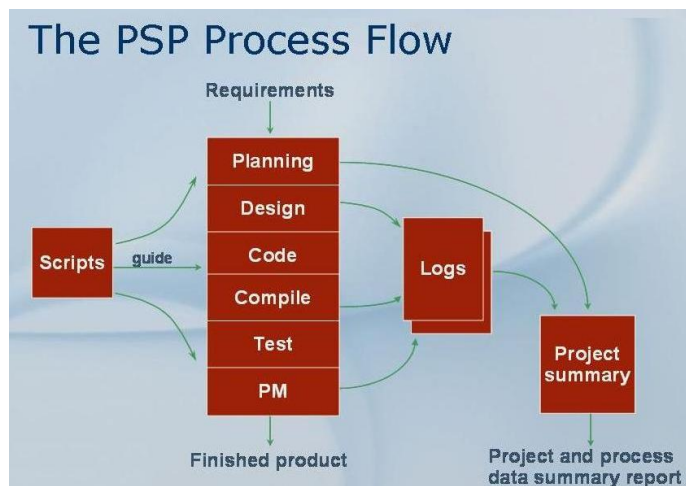


Figure 1: PSP Process Flow

Evidence Based Software Development

- PSP got one thing right: no evidence, no improvement
- So, for every phrase of the assignment, record *estimated* and *actual* time
- Calculate *velocity* of your own process
- So, you get a personalised how of how you code

Your Process

- Planning
- Designing
- Coding
- Testing

*Planning is **not** optional*

- Keep an Engineer's Log throughout your coding process

Assignments

- [Github](#)
- “Request Access”