Personal Software Engineering

OVERVIEW

What we want you to learn in SWEN-250

Learning the foundations of software

Using the basic tools (Command line, compilers, version control …)

Writing ‘clean’ code

How in read/ interpret instructions (requirements)

Managing your time (your ‘personal’ software process)

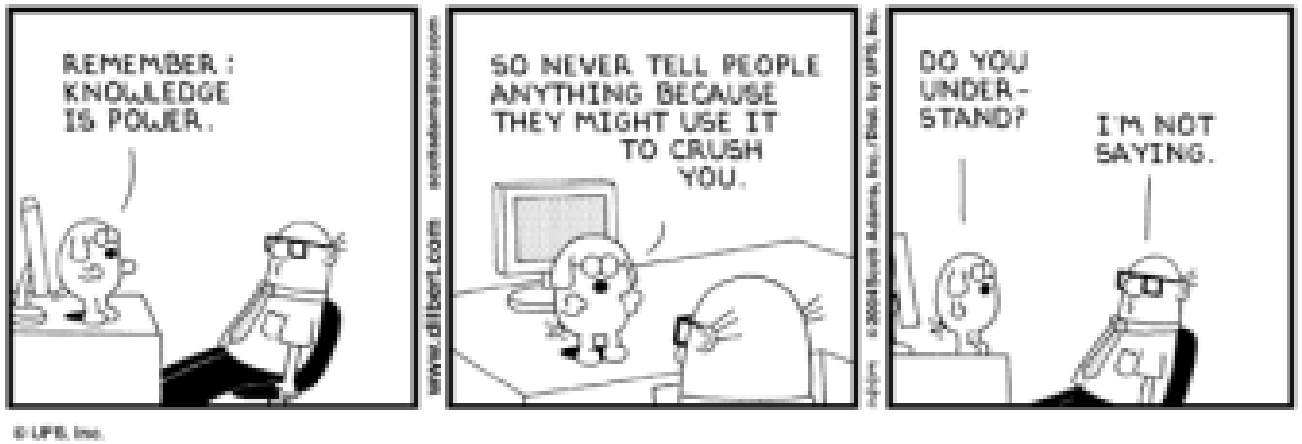
Being an engineer (not just a coder)

You will learn all this using Unix, C, C++, git …

Other details…

- Always (double)check submission has been made by checking GitLab web interface.
- Deadlines are stated on the due dates page.
- Let’s get to know each other and have fun!

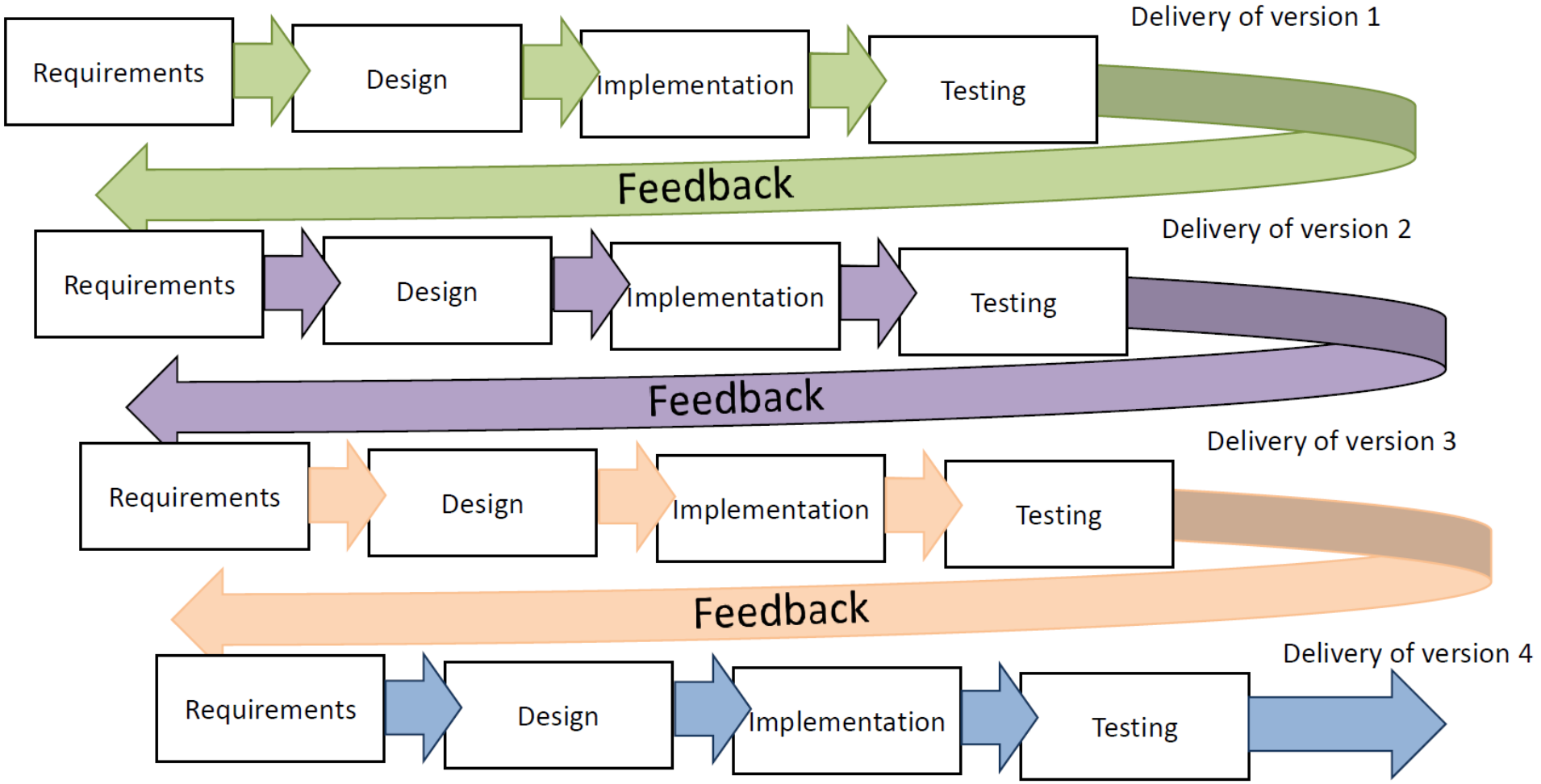
To share or not to share …



Programs versus Software Products

Programs	Software Products
Usually small in size	Large
Author himself is sole user	Large number of users
Single developer	Team of developers
Lacks proper user interface	Well-designed interface
Lacks proper documentation	Well documented & usermanual prepared
Ad hoc development	Systematic development

Incremental Development



Defining a Personal Process

- What is a Process?
- Why Personal?
- 5W's + H

Why What

Who When

Where How

- People + Process + Product + Technology

Defining a Personal Process (cont)

- Tools + Personal Habits
- Continuous and Incremental integration
- Reflection and Improvement
- Metrics for success

SE Accounts

- Not your RIT main account
- Can be same password.. but should it?
- Your department resources

Linux Environment Intro

- ~~nitron.se.rit.edu~~ hamilton.se.r.it.edu

SE Dept. main shared server
  - All SE students have an account (not the same as your RIT 'main' account)

- Z:/ → *Shared/ global drive. No matter where you login on the SE computers, you will see this mapped*

- ssh (secure shell)
  - See instructions on setting up and using

- Basic Linux commands: https://linuxjourney.com/lesson/the-shell

- Git tutorial: https://www.youtube.com/watch?v=USjZcfj8yxE

How you do your work …

On your desktop …

- Open a terminal (e.g. Windows Terminal … or powershell or mac terminal)
- Connect to hamilton (see instructions for hamilton, ssh)
- Navigate to your repo local folder (see instructions for gitlab)
- Edit/ compile/ test your code (all command line!)
- Push your code to gitlab (also command line)
- Repeat (often)