

SE is Engineering

- Making things that work - practical
- Use of models, standard designs, methods, tools
- Constraints: time, money, organisation
- Managing people, communication

Software Engineering

8

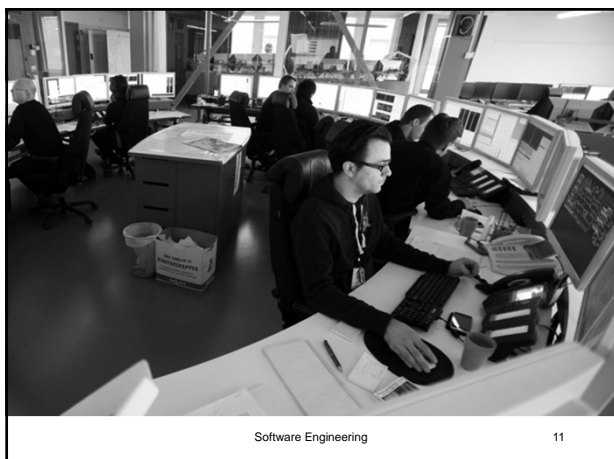


...but not always like Engineering

- Every project is mostly new
- Software is "invisible",
- perceived to be adaptable (rewrite code vs. rebuild a bridge)
- lacks physical boundaries - gets complex
- Usability - "getting it right"
- Legacy systems

Software Engineering

10



Software Engineering

11

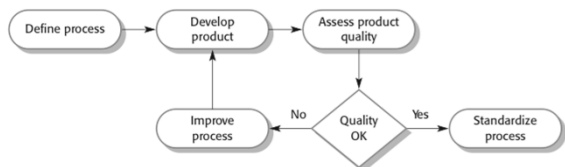
Process

- Process
- what really happens ... too messy
- Process model
- abstraction, common themes
- Method (or: Process)
- what *should* happen
Every company has one! (ADM, PROPS)

Software Engineering

12

Figure 24.3 Process-based quality



Product Quality

- Acceptable: usable, learnable, compatible
- Efficient: response time, memory use
- Dependable: safe, reliable, secure
- Maintainable: documented, structured

[Fig. 1.2, 24.2]

Process (method) Quality

- People actually follow it:
 - acceptable, usable, learnable
- It delivers:
 - efficient, in time, acceptable product quality
- Manageable
 - visible, robust to problems
- Supportable
 - documented, structured

The ingredients

Requirements

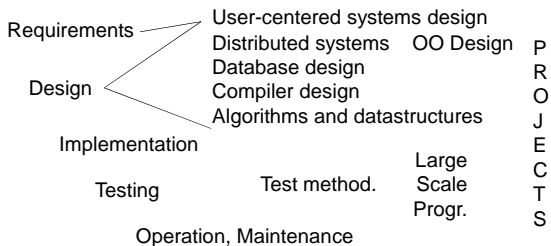
Design

Implementation

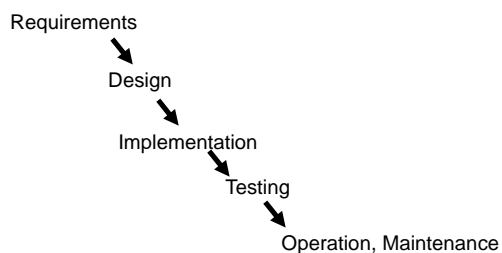
Testing

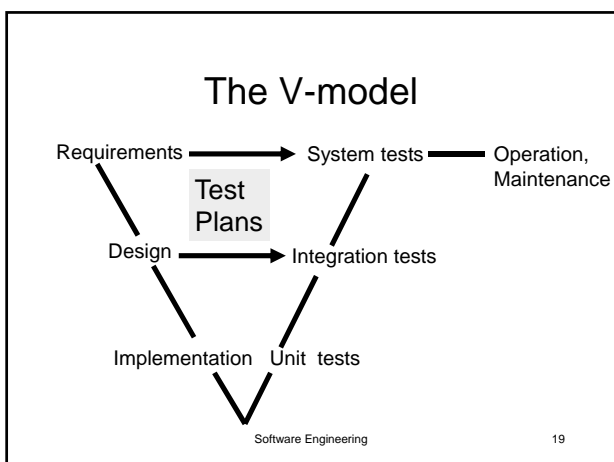
Operation, Maintenance

Some related courses



The waterfall model





Validation

- Will the product satisfy the customer needs?
- Are we building the right product?

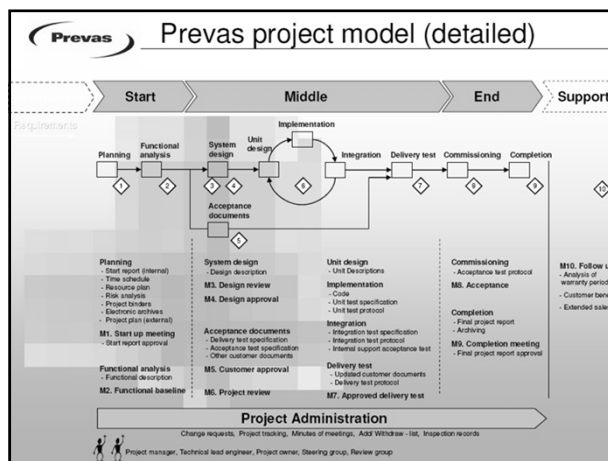
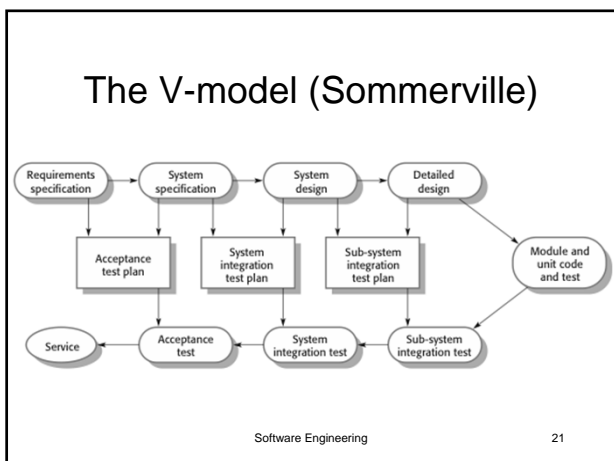
↻

Verification

- Do we satisfy the requirements?
- Are we building the product right?

↻

Software Engineering 20



Outline of the course

- The V-model as a roadmap
- Dependable systems
- Project management
- Ethics

Software Engineering 23

