

error detection

- analysis
- design
- desk checks
- Role play
- automated

Waterfall method:

analysys

Design

Coding

Testing

handover

+ clean code - bug

linear model

- easy to maintain
- output of each based on the next
- 

B DuP

#Big design upfront process

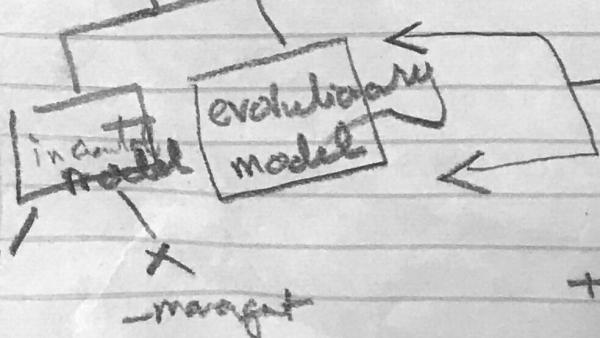
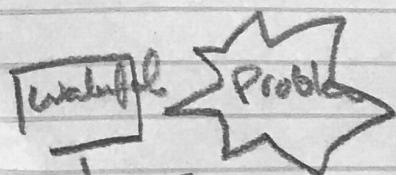
Waterfall method

- poor for risk management
- requires change
- customers only has product in the process
- 3 weeks

→ weak testing too late! time management

Analysys / analysys :-)

→ needs to have good cards



Small Simple steps

\* features

\*

+ Iterative model

Q> 3 weeks

Some but! Waterfall

whole Proj

agile

Test Driven Dev

Test Driven

OOD UML

Code Smells

Refactoring

Design patterns

C

username

U147480

P: Tuesday

Unified

Agile

Dev Ops

→ patterns

- Design patterns
  - + Single Responsibility Principle
  - + Open Closed

- Refactoring

→ Code Smells = something that is bad about your code

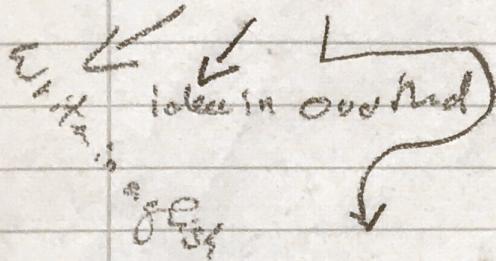
- + long method
- + hard to understand + update

- attendance 10% participation / 10%
  - week 11 annual exam 15%
  - mid term exam / week 6 / 10%
  - week 12 / Oral exam = 25%
- ↳ writing exam / 40%
- 

→ Exam  
→ Talk

How do we build Software:

Specification → Implementation



methodology

- 1 set of methods
2. rotation describing methods
3. process followed Dev