

Contemp Comp 4780

Process dev software is not based on best approach

methodology

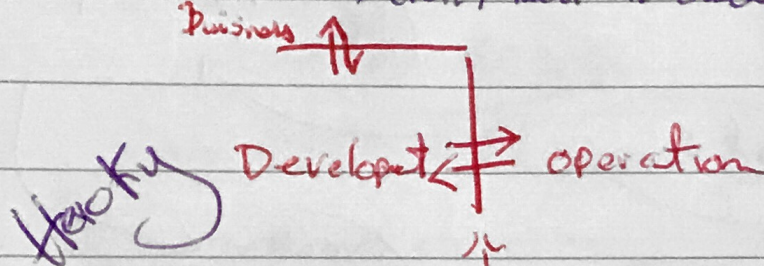
meta model: denoted in history

OOP is top

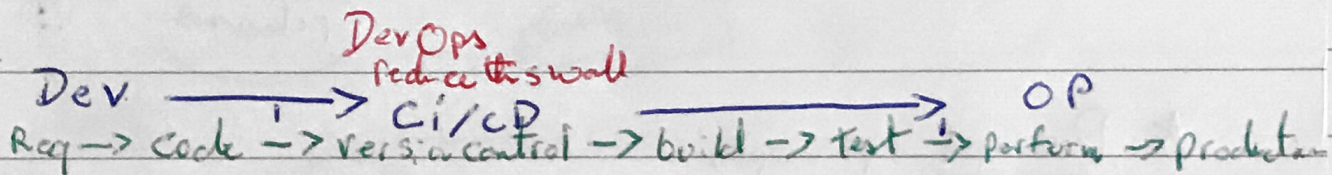
Dev ops

- server
- performance
- service
- Run
- monitor
- failures
- server attacks

don't need to check files in depth



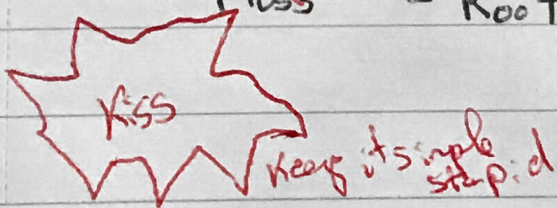
- continuous integration
- continuous Delivery



Refactoring: change design without changing design

- Floss

- Root canal / design Depth: functionality without Refactoring



extreme programming (XP): program being controlled by Developer

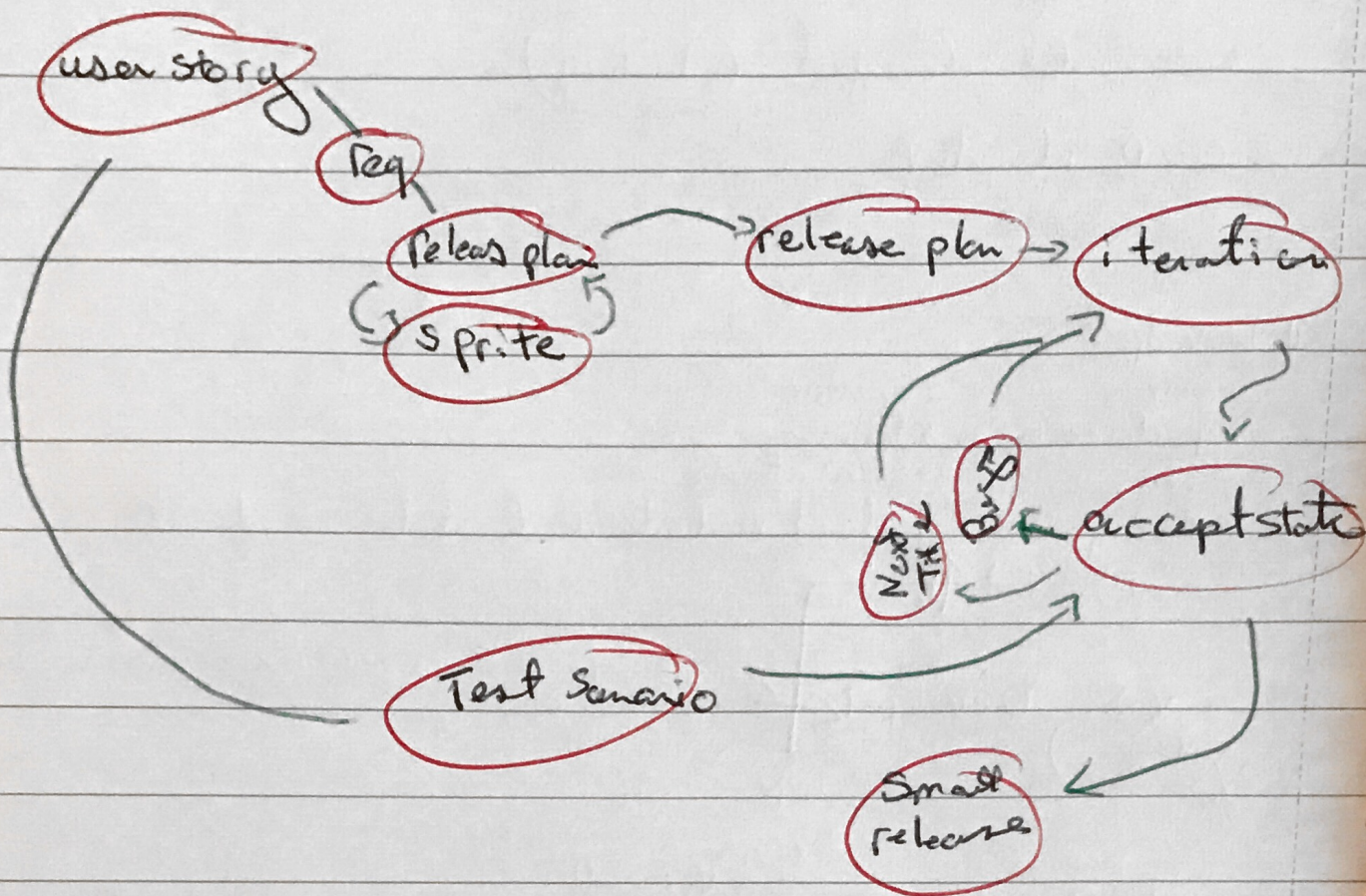
- early testing
- rapid prototyping
- small environment

Key features: another environment - testing - pay as go

Code & fix

- not testing
- not refactoring

XP



* Dev ops : principles

- + Continuous integration / Delivery
- + Micro service arch: API / HTTP
- + Infrastructure code
- + monitoring & logging

Term

Source: git, bit bucket

build: gradle, Maven, Ant

Containers: Docker, Kubernetes

continuous integration: Jenkins, Bamboo

infrastructure as code: puppet, ansible, chef

monitoring: nagios, prometheus

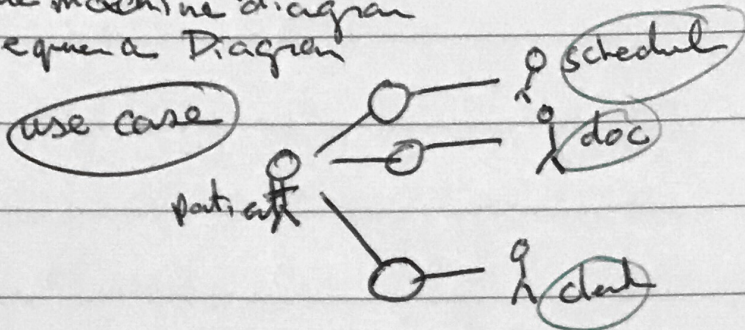
+ UML

unified modeling lang

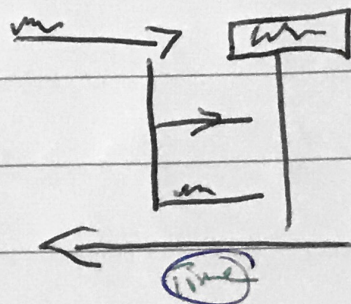
14 diagram types

we look @ 4

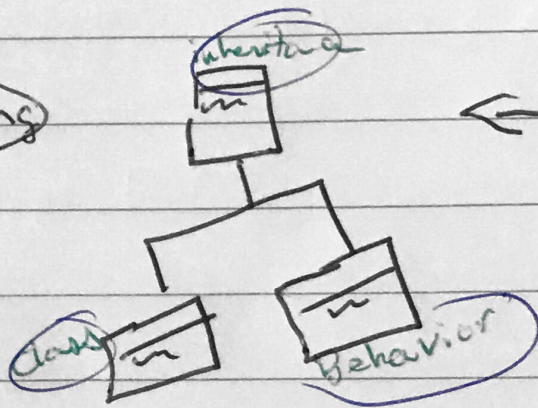
- class diagram
- use case diagram
- state machine diagram
- sequence diagram



Sequence Diagram



class diagram



State

