MOD-1

1) Principles of s/w delivery

-> create a repeatable, reliable process for releasing s/w

-> If it hurts, do it more frequently & bring the pain fwd

-> Build quality in

-> Done means released

-> Everybody is responsible for the delivery process

-> Continous Improvement

2) Configuration Management

-> Using version control

-> Mnging dependencies

-> Mnging s/w configuration

a) Using Version Control

-> keep absolutely everything in version control

-> check in regularly to trunk

-> Use meaningful commit messages

b) Mnging s/w configuration

-> Configuration & flexibility

-> Mnging appln configuration

-> Accessing configuration

-> Modeling configuration

⇒ Principles of mnging appln configuration

→ Consider where in your appln's lifecycle it makes sense to inject a particular piece of configuration

→ keep available configuration options

→ Use clear naming conventions

→ Ensure that conf. info. is modular

→ Ensure that you have tests for ur configuration

⇒ CI

→ prerequisites for CI

→ Using CI s/w

→ Using CI s/w - essential practices

→ Distributed version control system

⇒ prerequisites of CI

→ check in regularly

→ Benefits of check-in regularly

→ create a comprehensive automated test suite

→ Unit test

→ Component ,,

→ Acceptance ,,

→ keep the build & test process short

→ The first thing to consider is making your test run faster

→ Mnging ur development workspace

⇒ Using CI S/w

→ Basic operation

→ Don't check in on a broken build

⇒ Using CI s/w - essential practices

→ Always run all commit test locally before committing/ get your CI server to do it for you.

→ wait for commit tests to pass before moving on

→ Never go home on a broken build

-> Always be prepared to revert to the previous revision

-> Time box fixing before reverting

-> Don't comment out failing tests

-> Take responsibility for all breakages that result from your changes

-> TDD

=> Implementing a testing strategy

=> Business facing tests that support the development process.

-> Happy path

-> Alternate path

-> Sad path

-> Automating Acceptance test

-> Regression test

-> A good rule of thumb

=> Technology facing tests that support the development process

-> Unit Tests

-> Component test

-> Deployment test

=> Business facing that critique the prjct

=> ~~Business~~ -> Showcase

-> Exploratory testing

-> Usability testing

=> Test Doubles

Dummy

Fake objects

Stubs

Spies

Mocks