

## Testing of EA

- not compromise quality
- what → need to test, kind of test, condition to enter, exit
- How → slow to test, How much before rollout, How long each test
- where → to installed & test
- when → to start testing, end testing
- who → who are involved

## Types and methods of testing

- what
- functional → as per regn, test screen flow  
dataflow, business logic  
use & sys vali
  - nonfunctional → check quality attr, security,  
Performance, maintainability, reliability  
avail, scalability

How

- white box - check inter structure  
flow of dat, control, Branching

- black box - without understand internal

keep internals in mind

- stru & testing from external  
unit/component/app - to be tested

Gray box

but testing hybrid form white + black  
black box

## Testing levels

- several levels before rolling out
  - How much, when to last
- 

unit testing - ~~It test~~ test std functional / code  
first level (during construction)

Integration testing -

- grouped together consistently
- check interaction among units

System testing

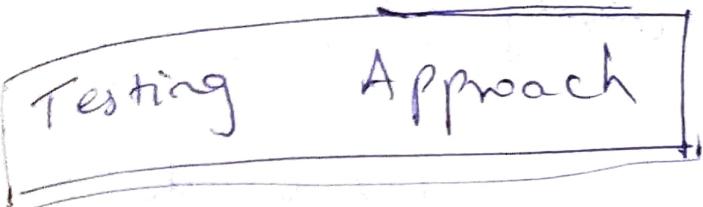
- test entire system
- different i/p, check diff functionalities
- check functional & non-functional

Acceptance testing

- end of testing phase
- before rollout
- test cases - end user req
- Success: completion of this testing  
(accepted by client)

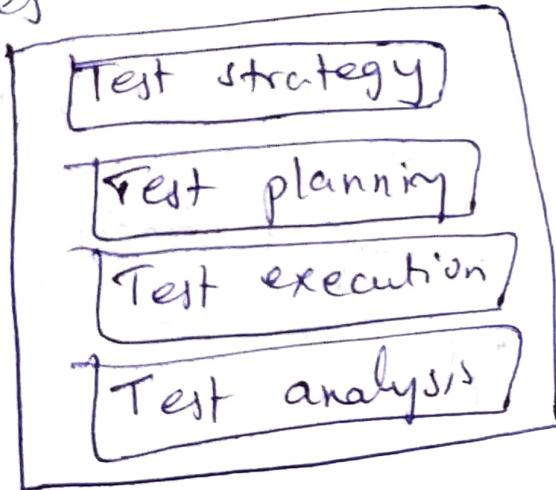
## Production testing

last level, Application in production environment.



- Popular model of testing : V-model
- testing spread across entire life cycle

Four stages



## Test strategy

- overall approach to test
- consider environment, type of testing, testing team, effort, cost, timeline of testing, defect tracking, tools to be used
- test planning, when to end the testing

## Test planning

- plan to execute
- good test plan → find gaps in req
- overall testing activity (test cases & procedures)

## Test execution

- kept document, expected results

dev team: unit & integration team.

Sys testing = large, EA

results are logges

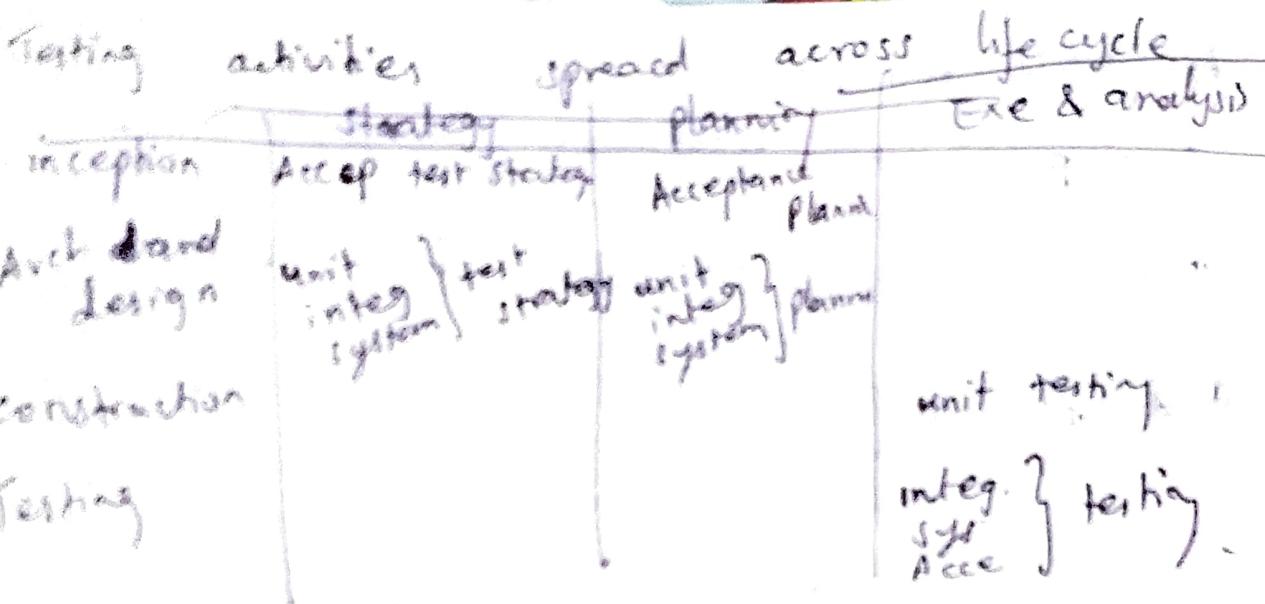
all integration points also need to be checked

- \* Defects entered in defect tracking system and passed on app dev. team

## Test analysis

defects captured ⇒ analyzed for complexity, type, severity, impact etc

- corrective actions are taken based on defect analysis.



### automated Regression testing tools

- traceability tools
- code coverage tools
- performance
- load testing tools
- test case generation tools

- defect tracking tools
- static analysis tools
- compliance checking tools
- test data generation tools

### Enterprise Application Environment in lifecycle

Development

Test

Quality assurance

Staging

Production

Testing level

production

acceptance

System test

Integration

Unit

Environment

Production

Staging

Quality Ass.

Test

development

## Application environment

Development  $\Rightarrow$  high access low budget and low security

Test  $\Rightarrow$  perform integration activity & integration testing

Quality Assurance (QA) high access

used for QA activity by QA team

Most non fun unit tested.

Staging - mimics production env.  
check all aspects except security  
used to train end users  
phy. config are tested & ready for env.

Production - env. where final EA resides  
high security, high access control

assume

Loms process a peak - 60

Concurr trans/sec. = 40

Support also ~~concurr~~ concurrent users session

each app a day 1000

internet facing appli avail neg 29.99

## LoMS Environment (non function)

- ① Concurrency
- ② capacity
- ③ security
- ④ Scalability
- ⑤ Availability

## Integration Testing

- ① After unit testing do I
- ② test  $\Rightarrow$  after combined together
- ③ testing team use s/w construction map.
- ④ gray box testing

## Same Integration test case

### Test case

Enter loan appli  
detail form & submit

Test case	Components under test	Expected result	Result
Enter loan appli detail form & submit	Initiateform.jsp Save/Initiate Loan details Action class	Save initiated Loan detail action should be invoked	pass/fail

## System Testing

- blackbox testing after integrated
  - check fun & non fun req
- ① Performance testing
  - ② Penetration testing
  - ③ Usability
  - ④ Globalization
  - ⑤ Interface testing

## Performance Testing

— Validate responsiveness of the system

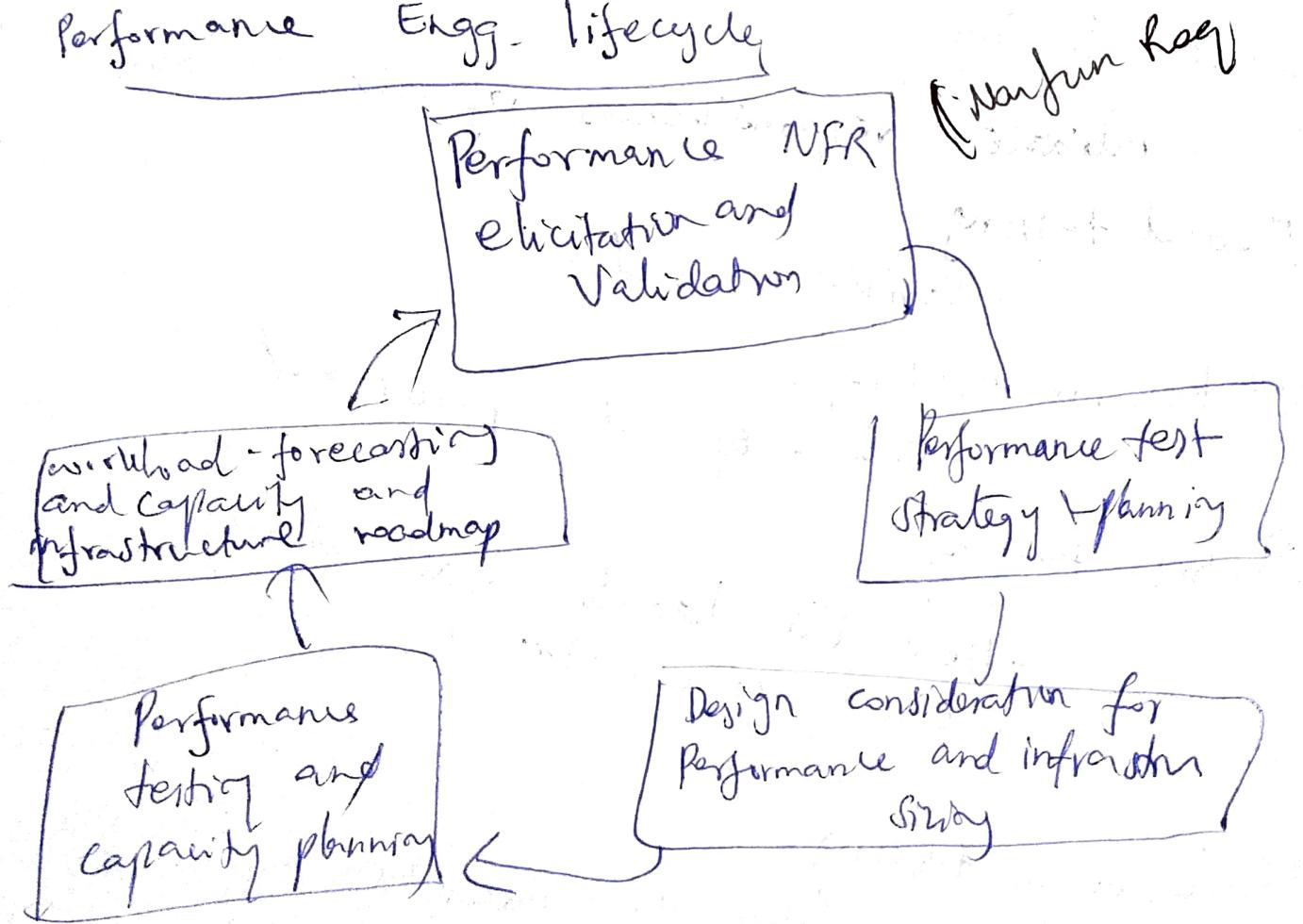
- ① Load testing
  - Variety of loads
- ② Volume testing
  - Volume of data processed
- ③ Stress testing
  - How EA handle bad during unusual circumstances
- ④ Endurance testing
  - Problem related to ~~new~~ memory leaks.

## Perfor Test

### obj:

- ① Determine EA @ given user load & Data Volume  
Load & Volume testing
- ② Determine breakpt of system under  
mounting load - stress testing
- ③ determining the performance of  
bottlenecks - ~~etc~~ endurance
- ④ Benchmarking the performance

## Performance Engg. lifecycle



## Performance NFR metrics

Max No of concurrent users : 250

actual

25

Peak throughput (hit/sec) : 60

69

Average throughput : 40

54.5

max refresh(in sec)

: 5

16.9

No. of loan app / day : 1000

25000

## using Jmeter

To optimize the performance - activities

① Application code analy

② DB optimization

③ Infrastructure Resource and capacity optimization

↓  
new latency rate  
Collision rate , memory & processing capacity

## Penetration Testing

OWASP

proxy

paros

① black box testing

② security perspective

Reviewing  
Req to handle  
app sec  
future attack  
vector

Penetration  
testing

Security NFR  
elicitation and  
Validation

Security test  
strategy and planing  
and threat modelling

Security code  
review - static  
code analysis

Design consideration  
for security

## Test case

input in  
username field  
~~scott!~~

system alert  
user regarding bad  
input

Pass/Fail

Crawl the tool like  
~~web~~ scarab

to  
get user  
authentication

Pass/Fail

~~SQL inject~~

## usability Testing

- black box testing
- Ensure easy to understand and use.
- performed by end users

Customer  $\Rightarrow$  need self guided to fill form

bank Rep  $\Rightarrow$  easy navigation

If global bank  $\Rightarrow$  need internationalization

## test case

$\Rightarrow$  based on user's behavior & nature of use.

<u>Test Case</u>	<u>Expected result</u>	<u>result</u>
① loan initiation need self guided	easy to use available whenever needed	Pass/Fail
② Loan approval simple workflow from initial to approval	workflow complete max: 3 steps	Pass/Fail
③ Globalization multilingual support	display current local lang.	Pass/Fail
④ Help adequate help	help, tips	Pass/Fail
⑤ Personalization display info relevant to customer	display to customer	Pass/Fail

## Globalization Testing

- Enterprises reaching out - globe
- user need customers friendliness.
- manual testing to validate
  - check localized based on the locale w/o affect design.

glb test term =  $\frac{I18N}{\text{locatn Inter}} + \frac{L10N}{\text{locav}}$

- testing done by test engineers and business analyst

✓ Language translation.  
time zones, date, format  
Currency symbols.

### Test case

login bank repre  
select locale  
sort cust name

Expected Result  
sort order  
w.r.t. locale

Pass/fail

login customer  
select locale  
check currency  
form

based on  
locale

Pass/fail

login as customer  
select - locale  
~~check~~ translations  
on customer page

correctly  
translated

pass/fail

### Interface Testing

Validate incoming and outgoing interface  
Inter system testing

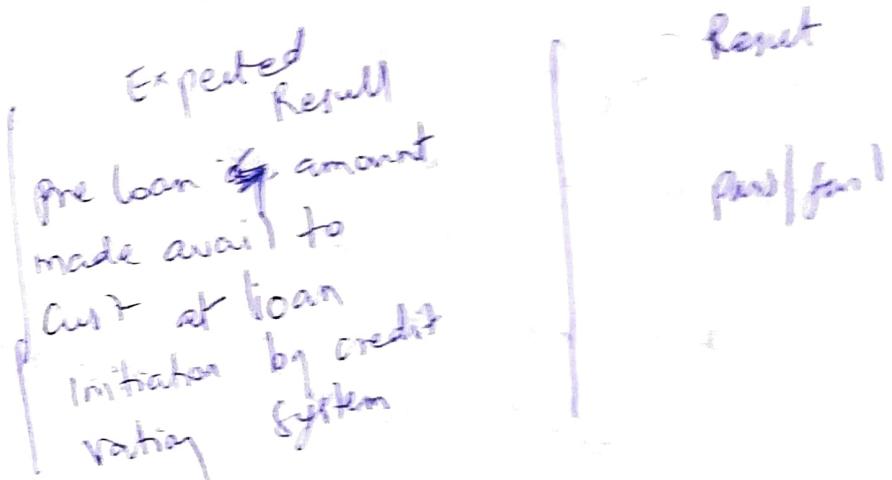
- Validate the existence of  
appli under test with other EA

## obj

- nature of communication interfaces
- req-res handling of B2B
- data formats exchanged through interfaces
- error in syn - interfaces

## Test case

login: customer  
check the pre  
approval loan amt  
at the time loan  
initiat



## SoapUI

### User acceptance testing

- black box testing -
- performed by end users

user acceptance testing are successful  
completion of unit, integration and system testing

\* bug identified  $\Rightarrow$  documented as "known issues"

## Obj Accept Test to Validate

- 1) Stated function in the sys. req Specie document
- 2) Part of non fun req like usability & response time
- 3) The access control of module of appl
- 4) Complete business processes
- 5) transfers data from outgoing & incoming interfaces
- 6) batch processes
- 7) The report format & contents'

Test case	Expected Result	Result
① login as existing Cust. and check pre approved loan amt	Pre approved loan amt should be displayed on credit rating of cust	P/F
2) login : exist cust check font, colors.	correct	P/F
③ login : Bank rep upload signed loan appl	format should be appropriate as per <del>bank</del> bank guidance	P/F

## Rolling out, EA

- ① making appli → live on production envi.  
for end user

- ② EA appli handed over to maintenance team

strategies to rollout

### Brand new rollout

- new application
- easiest to manage

### Parallel switch

- new appli env in parallel
- new env is verified ; appli traffic switched from old to new
- old kept standby by predefined period

### Rolling rollout

- rollout in a clustered & highly managed env.
- a node of cluster = offline. - upgrade verify - brought online status

challenge) → trans integrity and data continuity

### Phased rollout

- slower variant rolling rollout strategy
- based on particular region to fit

up-down-up rollout

- start with bringing everything down
- system is upgraded, verified and brought up again.
- cost is enormous, lot of upfront

Planning

• after rollout & strategy

Prep for smooth app. rollout.

① Rollout plan

- seq of rollout related events
- schedule

② Communication plan

- describe who get involved
- at what stage

③ certification plan

handshake protocol

release notes

scripts } config file } should all be in place

④ Plan B

if things not go well

in terms of money

brand image

- ⑤ Identify best time and duration for launching the rollout
- ⑥ depending on criticality of rollout, a dress rehearsal can be done before rollout

## Steps to Rollout

- ① Make sure all comm. sys. like emails, pagers, cellphones are working.
- ② Identify and verify the binaries, config files, Scripts and artifacts to be used in app deployment
- ③ Run backups and setup recovery points
- ④ Turn off monitoring, alerting, & alarm notification systems - increase queue depths and cache values and rollback logs for
- ⑤ Turn on auditing and trace purpose recovery
- ⑥ Take system offline either one node at a time (or) whole system & initiate the rollout

⑦ Initiate verification steps and certify install. It could be automated scripts use tests.

⑧ Turn on monitoring, alerting & alarm notif sys

⑨ bring system online either one node at a time or a whole system

⑩ monitor system parameters for system health

⑪ Turn off auditing and rollback logs, if everything checks out. This can be kept going for a brief period.

⑫ Verify and certify the rollout