

→ Software Quality Engineering

final :

Part III

① Defect prevention:

making sure that the defects do not happen in the first place or the defects are removed/reduced before making their way into the software product.

② Defect Prevention strategies:

1) Error Blocking: identifying common error and blocking them to prevent fault injection.

→ following well defined processes, standards and appropriate tool can block identified common errors.

2) Error Source Removal: identify common error sources and removing them.

→ Can remove these identified sources by education and training, improvements initiatives.

Software Inspection :- (QA Activity)

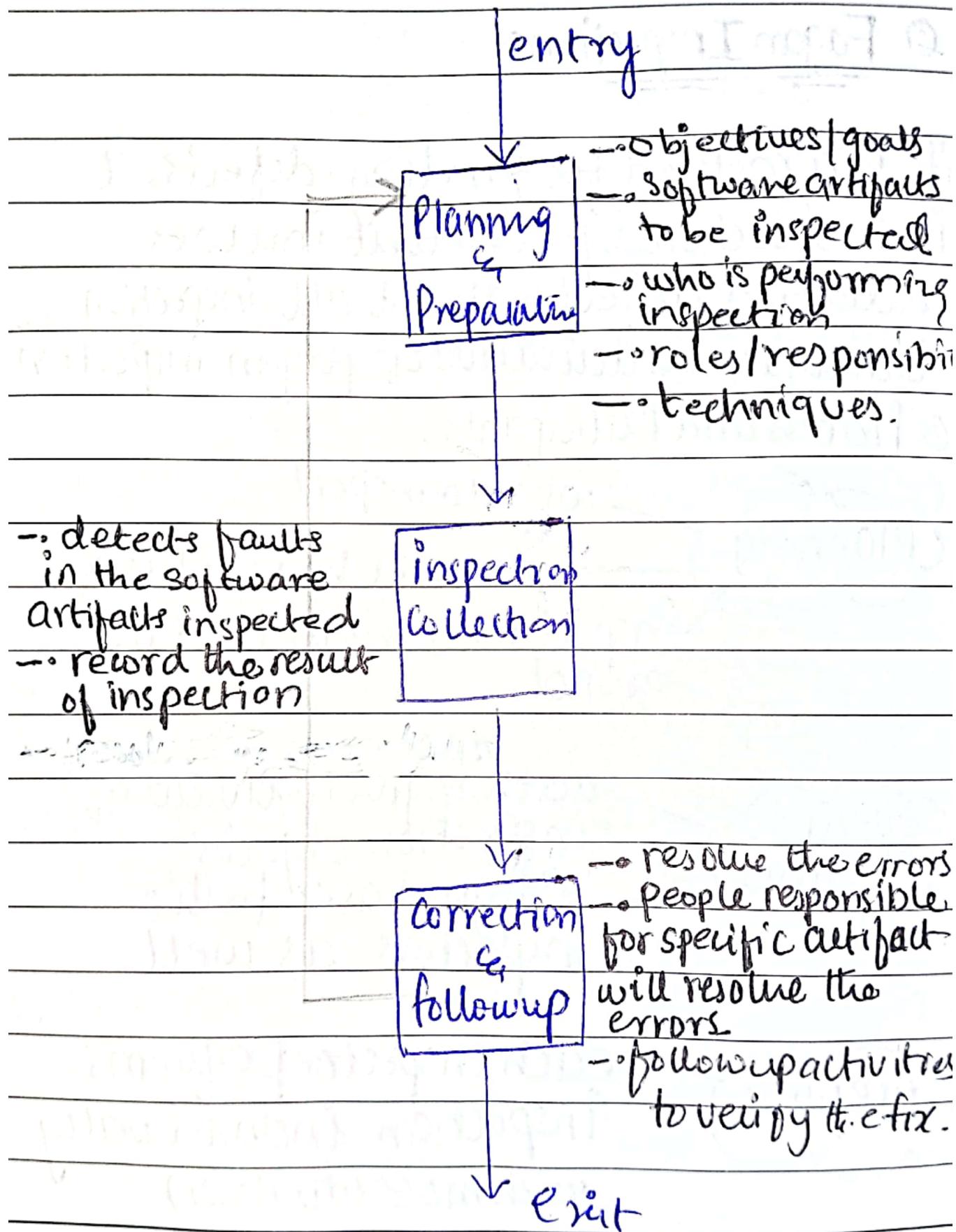
- directly detects and corrects problems/errors without executing the software.
- Inspection can be done on different software artifacts and deliverable.
- deals with finding software defects.
- defects can be fixed easily through follow up activities.
- Various software artifacts available late in the development can be inspected but not tested, e.g.: product release, support plans.
- anything tangible can be inspected.

Degree Of Formality :

there are different degrees of formality ranging from informal reviews to very formal inspection techniques.

Generic inspection process

Date:



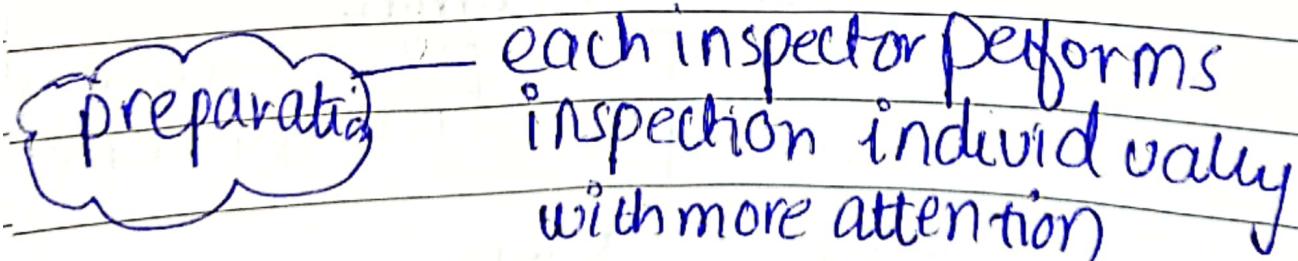
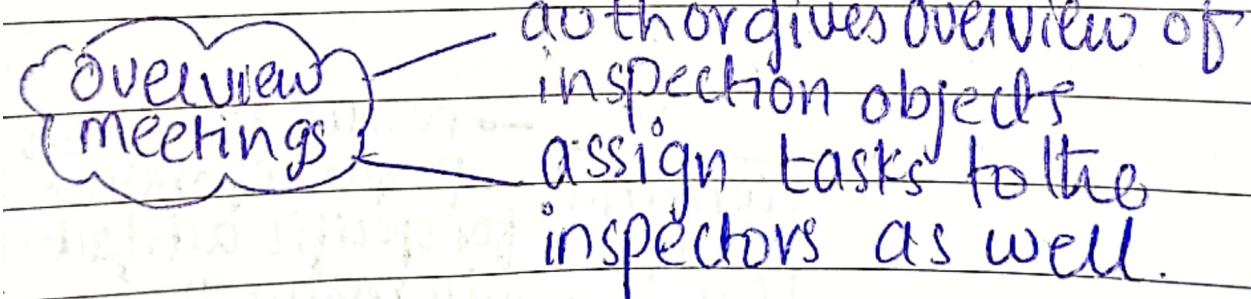
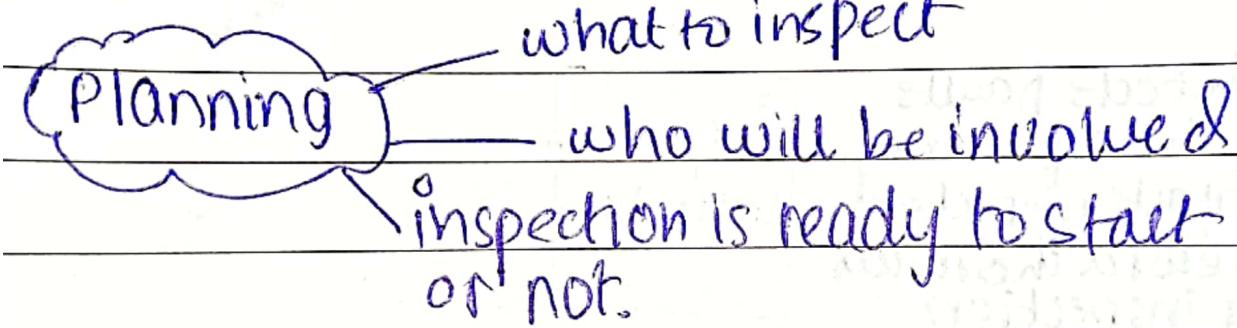
Types of Inspection :-

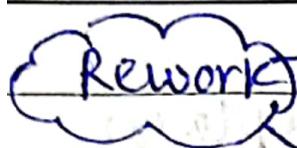
Date:

① Fagan Inspection :-

it is a method for finding defects. it is a structured process that involves detecting defects. almost all inspection techniques are derivative of fagan inspection

② Process and Participants :





author will fix the identified problems

author will provide other responses.



closing the inspection process by final validation

Ingenueic inspection maps In fagan Inspection

1) Planning & preparation

i) Planning

ii) Overview

iii) Preparation

2) inspection & Collection

(i) Inspection

3) Correction & followup

(i) Correction

(ii) followup.

Inspection Team

involves 4 people in
the teams

Potential inspectors are
identified in stage I

An ideal mix would include
people with different roles to
bring diverse view & perspective.

Inspection
work

→ overall coverage and area of focus
should be considered during assignment
different inspectors will
be assigned different pieces
so no duplicate inspection
effort

Critical pieces required attention
of more than one

focus on defect detection not
on resolution

inspection meeting last two hours
in which one inspector is identified as
a leader and oversees the meeting
and ensures the purpose of it. (defect
identification & consolidation)

Gill Inspection

- ① an additional step "Process brainstorming" is added right after Fagan inspection.
- ② input to the inspection process could be Product document, rules, checklists, source documents, any technical documents even diagrams can be inspected.
- ③ The output from inspection is:
 - inspected & corrected document
 - Change request
 - suggest process improve
- ④ The inspection process forms a feedback loop.

Fagan	GILB
1- Planning	1- Planning
2- Overview	2- Kickoff
3- Preparation	3- Individual Checking
4- Inspection	4- Logging meeting
5- Rework	5- a- edit b- process brainstorming
6- followup	6- Edit audit
<u>Phased Inspection :</u>	

Overall inspection is divided into multiple phases with each focusing on a specific area or a specific class of problem.

Review your own code or document for finding errors.

Desk check

inspection of technical documents produced by oneself

it focuses on logical & conceptual problems, to make effective use of valuable time.

Review

it also focuses on logical & conceptual Problem.

inspection of technical document produced by someone else

desk checks are complement by review, because in review people from different background will be reviewing your document it will bring diverse view to inspect.

A special form of review

a more organized
review applied to
design & code.

meetings for
walkthrough

walkthrough

defect detection
is not the focus.

meeting
requires less
preparation

designer or code owner

will lead the meeting
and explain his code
and design.

Inspection Limitations

- difficulties in dealing with dynamic & complex interactions
- difficulties with task automation

Formal Verification

to verify the correctness or absence of faults in some given program code or design against its formal specification.

both formal verification and formal specifications are formal methods.

Use of Formal Methods:

1) Constructing formal Specification

2) Performing formal Verification

Defect Containment :

Defect Containment is about minimizing the defect in software.

① Defect Containment techniques :-

① Fault tolerance techniques :-

the ability of a system to continue operating without interruption when one or more of its components fail

② Failure containment techniques :-

reduces the impact of system failure so it can be avoided or minimized during execution of program/system.

Fault Tolerance :-

is the process of working of a system in a proper way inspite of the occurrence of the failure.

- ① N-version programming
- ② Recovery Blocks.
- ③ Checkpointing & Rollback Recovery.

Effectiveness Compaision :-

ability to provide additional info for improvement

defect they are dealing with

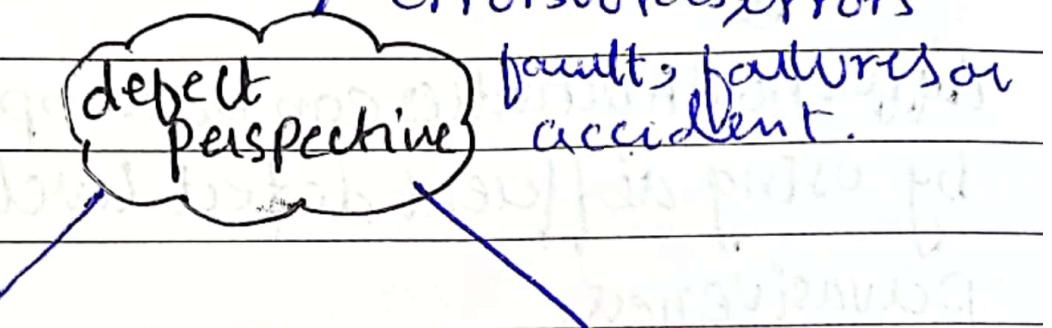
QA alternative compaison

suitability to different defect levels

Problem they are good at addressing

effective comparison based on :-

QA alternatives can be compared by whether they are dealing with errorsources errors

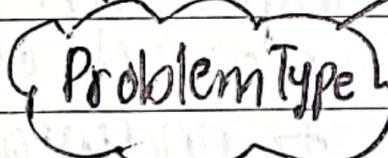


Detection or observation of specific problems

followup action that deals with observed or detected problem

comparison with different perspective of defects, error sources, types of fault etc

different QA alternat might be effective for different types of problem



Effective Comparison based on:

defect level &
Pervasiveness

different QA activities can be compared
by using different defect levels or
pervasiveness.

Result Interpretation

ease of result interpretation
plays an important role. in an activity

Result interpretation is harder for formal verification, fault tolerance, failure confinement as compare to testing & inspection

Testing Vs Inspection

Program can be tested after the execution.

all executable programs can be tested.

testing doesn't support defect removal in early phases of development.

testing involves dynamic examination.

testing is better in detecting problems in interaction of multiple component.

testing results are hard to interpret.

Inspection detects & correct problem without execution.

all kinds of software artifacts can be inspected.

Inspection can remove defects in early phases of development.

Inspection involves static examination.

inspection cannot work properly in finding error in interaction of multiple components.

Inspection results are easy to interpret.

Fault tolerance Vs Fault Containment

- | | |
|--|--|
| ① Ability of system to continue operating without interrupt when one or more of its components fails | Ability to allow the system to limit the impact of manifested faults to predefined boundaries. |
| ② It is necessary for systems that are used to protect people's safety. | ③ It is necessary to prevent faults from spreading in all types of systems. |