classmate
Date
Page

Chapter - 5

Static Techniques:

Test Design is process of creating a set of inputs for given software that will provide a set of expected outputs.

The idea is to ensure that the system is working good enough and it can be released with as few problems as possible for the average user.

Categories of Test Design Techniques
1) Static Testing Techniques
2) Dynamic Technique.

Imp

1) Static Technique.
- Testing is carried out line by line in source code.
- It starts early in the lifecycle and so it is done during the verification process.
- It do not need computer as the testing of program is done without executing the program.
- It is white Box Testing
- It helps to determine:
  - the component available in class.
  - what is a function does
  - if there is error is in syntax (Syntatic error)
  - error in data types.
  - time bombs & event bombs

for example: Reviews, walkthrough, inspection etc.
- Static testing can start early in the lifecycle so early feedback on quality issues can be established.
- since, defects are found in early stage maintainance of defects costs relatively low.

- Rework takes less time & effort so more time is available for development.

Dynamic Testing Technique:

- Dynamic testing is carried out by actually executing programmed code with a given set of test cases.
- Dynamic testing takes place when the program itself is run either - using ((Stubs / drivers)) or execution from a ((debugger environment)) or (computer.))
- Dynamic testing may begin before the program is 100% complete or in order to test particular section of code and are applied to discrete functions or modules.
- It is a Black Box Technique.

- It helps to determine:
  - System crash
  - Non-functional requirements like of quality, performance, user friendlyness etc.

- If failure is encountered in Dynamic testing technique, then static testing is carried out to find the causes of failure.

Static Technique	Dynamic Technique
- Static testing is carried out line by line in source code without executing actual sw.	- Dynamic testing is carried out by executing actual programmed code with a given set of test cases
- It starts early in the lifestyle immediately after coding begin.	- It starts mostly after the coding phase is complete.
- It involves verification.	- It involves validation.
- It is whitebox testing.	- It is black box testing.
- It helps to determine the cause of failure.	- It helps to determine the presence of failure
- It checks the sanity of code, algorithm or document, syntax etc.	- It checks the response from the system.
- Static testing methodologies include reviews, walkthroughs, inspection etc.	- Dynamic testing methodologies include User Acceptance testing, load testing etc.
- Requires more number of meetings.	- Requires less number of meetings.
- cost of findings defects and fixing is less.	- cost of findings defects and fixing is high.

☆ Review Process:

Roles & Responsibilities involved during a review is :-

1) Manager
2) Moderator
3) Author
4) Reviewer
5) Scribe

1. Manager:
- They decide on execution of reviews.
- They schedule project and determine whether review project's objectives have been met.

2) Moderator:
- aka review leader
- performs entry check & define exit criteria
- coaches other team member (review member)
- schedule the meeting.
- leads the possible discussion and stores the data collected in the meeting.
- follows up the rework.

3) Author:
- Author is the person who wrote the document under review.
- Illuminates the unclear areas and understand the defects found in the meeting.
- Basic goal is to learn as much as possible without about improving the quality of document.

4) Reviewer:
- aka checkers or inspectors
- check the document for defects prior to the meeting.
- Manager can also be involved in the review.

5) Scribe:
- scribe is a separate person to do the logging of the defects found during the review.

formal review:
- It follows a formal process
- It is well structured and regulated.
- It consists of six main steps
  1) Planning
  2) Kick-off
  3) Preparation
  4) Review meeting
  5) Rework
  6) follow-up

1) Planning:
- The moderator performs the entry checks and defines the formal exit criteria.
- The entry criteria are to check that whether the document is ready to enter the formal review process or not.
- once the document is clear the entry checks, the moderator and author decides that which part of the document is to be reviewed.

2) Kick-off
- Here, a short introduction on the objective of the review and the document is provided to everyone in the meeting.
- Relationship of document under review with other documents are also explained.

3) Preparation
- The reviewers review the document individually using the related documents, procedures, rules and checklists provided.
- while reviewing, they identify defects and comment according to their understanding.
- All issues are logged in a logging form.

4) Review meeting:
- Author or scribe logs the identified defects from all the reviewers.
- If any issues needs discussion then the moderator will lead the discussion.
- At the end of the meeting, a decision on the document under review is made based on the formal exit criteria.

5) Rework:
- Rework is done by author if the number of defects found per page exceeds the certain level.

6) Follow-up:
- The moderator checks the re to ensure that the author has taken action on all known defects.

Types of review:

① Walkthrough:
- It is not a formal process
- It is led by the authors.
- Author guide the participants through the document according to his or her thought process to achieve a common misunderstanding and to gather feedback
- useful for people who are not familiar with s/w development process.
- useful for high level documents like requirement specification etc.

② Technical Review:
- It is less formal review.
- It is led by the trained moderator or technical expert.
- Defects are found by the experts who focus on the content of the document.

③ Inspection:
- It is most formal review type.
- led by trained moderator
- During inspection the documents are prepared and checked thoroughly by the reviewer before the meeting.
- The reviewers log the defects in the logging form.
- In meeting, the scribe logs all the defects obtained from all the reviewers.
- If any discussion is required for any defect then the moderator leads the discussion.
- Decision on the document is made based on formal exit criteria
- If more defects are found then certain range, then rework is done by the author.
- follow-up is done by the moderator to ensure that the author is working on the defects.

Success factors of review

Success of review depends on:
- How well the moderator is choosen
- How well the document is choosen
- How frequently the review is carried out.
- How trained the reviewers are
- How well the reviewers are managed by the moderator
- How well the rules of formal review are followed
- What additional tools are used.
- How well the reports are generated.

§ 3 Static analysis by Tools:

- Static analysis tools are generally used by developers as part of the development and component testing process.
- Here, instead of executing the code, itself, the tool is executed and the source code is provided as input data to the tool.
- These tools are mostly used by the developers
- They are the extension of compiler technology
- Static analysis can also be carried out in static ((analysis of requirements)) or static ((analysis of websites))

Importance of Static analysis tools

- To calculate metrics of code such as complexity or nesting levels.
- To enforce coding standards.
- To analyze structures and dependencies.
- Help in code understanding.
- To identify anomalies or defects in the code.

Static Testing:

* Test design is creating a set of inputs for given software that will provide a set of expected outputs.

* Idea is to ensure that the system is working good enough if it can be realised with as few problems as possible.

* 2 main categories:
  1. static technique
  2. Dynamic technique

Imp * static technique find causes of failure but dynamic technique finds failure.

- (ST) provides a great way to improve the quality of productivity of S/w development.
- it includes reviews and provides the overview of how they are connected.
- primary objective is to improve the quality of S/w products by helping authors to recognize and fix their defects early in SDLC.
- mostly manual testing is done or with tools but S/w is not executed.
- starts early in life cycle (in verification process)
- eg: walkthrough, inspection, audit, review
- early stage error detection can reduce rework thus saving resources.
- development productivity will increase
- types of defects that are easier to find during static testing are: deviation from

standard, missing requirements, design defects, non-maintainable codes, inconsistent interface specification.

Formal Review:

* formal review follows a formal process.
* It is well structured and regulated.
* It consists of 6 main steps/activities:
  1. Planning
  2. Kick-off
  3. Preparation
  4. Review meeting
  5. Rework
  6. follow-up

① Planning:

- moderator performs entry check and also defines formal exit criteria.
- entry check is done to ensure that reviewer's time is not wasted on artifact that is not ready for review.
- It checks whether the artifact is ready to enter review process or not.

② Kick-off

- goal is to give short introduction an objectives of review and documents to everyone in meeting.
- relationships between document under review and other documents are explained.

3) Preparation:

- Reviewers review the documents individually using the related documents procedures, rule and checklists
- each reviewer identifies defects, questions & comments on the document a role.
- issued are recorded using a logging form
- the success factors for a thorough preparation is the no. of page checked per hour also called on checking rate.

4. Meeting:

→ 3 phases

↳ 1) logging phase:
- identified defects are logged in a standard way page by page.
- done by author or scribe (log in game marks)

↳ 2) Discussion phase:
- moderater (chairman of discussion phase) takes care of the people issues and prevents discussion from getting too much personal or out of control.

↳ 3) Decision phase:
- at end of meeting, a decision on document under review has to be made by participants based on exit criteria

5) Rework:

- If no. of defects found per page exceeds certain level than document has to be reworked
- not every defects has to be reworked.

6) Follow up?
-> to make sure that the author has taken action on all known defects.
-> Responsibility of the moderator to ensure that the information is correct.

Informal Review:
- good is to keep the author & to improve the quality of document.
- " not documented.

Technical Review:
- goals: x technical concepts are used correctly
x access the value of technical concepts
x alternatives
x to have consistency in the use.
x informs participants about technical content.

Peoples involved in Review
1) Manager
2) Moderator
3) Author
4) Receiver
5) Scribe

Tools
<< compiler >>
↳ coding standard
↳ code metrics
↳ code structure
↳ control flow structure
↳ data flow structure
↳ data structure.

Test Design:
- is the act of creating and writing test suites
- test analysis gives us a generic idea for testing which covers quite a large range of possibilities
- to make a test case we need to be very specific.
- important aspect is that it checks that the system does what it is supposed to do.
- In this design, input and expected result should be clarified.
- ideally expected result should be clarified.
- ideally expected result should be predicted before the test is run.

test design techniques:
- helps us to select a good set of tests from the total no: of all possible tests for given system
- various test design techniques are good at its own point and weak at some other.
x ex: component testing is more likely to find coding logic defects than system design defects.

* Categories of (Testing Design Technique)
↳ static
↳ Dynamic -> specification based (black box behavioral technique)
-> structured based (white box)
-> experience based

#1 Specification Based:

① State transition (diagram):

finite state machine means the system can be in a finite no. of different states and the transitions from one state to another are determined by the rules of the machine

- a state transition model has 4 basic parts:
  1. states that the s/w may occupy.
  2. transitions from one state to another.
  3. Events that cause a transition.
  4. action that result from a transition.

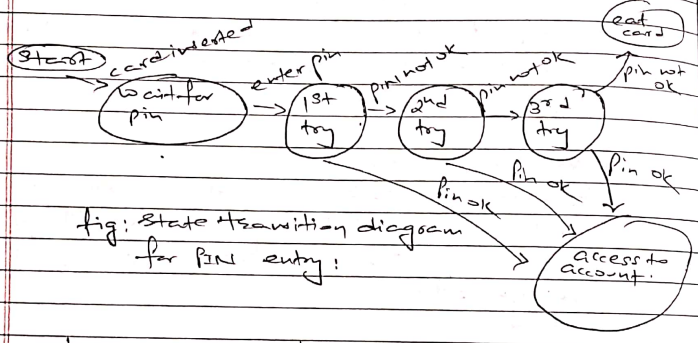

fig: State transition diagram for PIN entry:

Hence, we can see that in any given state, one event can cause only one action but that the same event (from a different state) may cause a different actions and a different end state.

② Use-case Testing:

- a technique that helps us identify test cases that exercise the whole system on a transactions by transaction basis from start to finish.
- each use-case describe the "interactions" the actor has with the system in order to achieve a specific task.
- use case are defined in terms of actor-not the system describing what the actor sees rather than what inputs the system expects.
- they often use language and terms of business rather than technical terms.
- It describes the process flows through a system.
- they serve as the foundation for developing test cases mostly at the system and acceptance testing levels.

③ Decision Table:

- a good way to deal with combination of things (input)
- also known as cause-effect table.
- Decision table provides a systematic way of stating complex business rule which is useful for developers as well as testers.

④ Testing combinations can be a challenge as the no. of combinations can be huge. Testing all combinations may be impractical & if not possible.

Conditions	Rule 1	Rule 2	Rule 3	Rule 4
Repayment amt entered				