Quality

- Quality is always a fundamental element of every engineering process & it is very difficult to achieve & sustain.
- To maintain the quality in s/w engineering, the values and the practices should be followed both in the product as well as process.
- So what is process quality: showing the transparency & tightness

Ex: In agile, short iterations will increase better quality by control & fast response.

- What is product quality: The practices like Pair programming, code inspection, unit testing, fast identification of integration problems will improve the product quality.

{Compare Agile with others}

The Agile Approach to Quality Assurance

- In other approaches, the task is divided & allocated to several different teams according to their functionality. Where as in Agile approach, there is no passing on of responsibility & all the team members are equally responsible for the s/w quality.

Scanned with CamScanner

- As software is an intangible product, we need a different development process to maintain the quality.
- Quality : refers to both product & process
- Quality Assurance (QA) : is associated with a specific stage of some development process. This is carried by QA people [These people are not developers. They just check the quality.]
- Agile (ASD) Software Development approach with respect to Quality :

"Quality may refer to the activities/work product. All check-in code must pass unit tests at 100% at all times. Sometimes quality is just a numerical value but sometimes it's a fuzzy value."

- Quality w.r.t ASD refers to the entire team during the entire process of software Development it measures the code as well as the activities performed during the development process, both in quantitative and qualitative manner.
- The term Quality Assurance, in Agile S/w Development does not appear in a specific stage of the development process, it is for all the time during the entire process.

Scanned with CamScanner

Quality-Related Aspect	Agile Approach	other Approaches
Who is responsible for s/w Quality?	All Development Team Members	QA Team [Quality Assurance]
When are Quality related Topics addressed?	All the Time	At QA/Testing Stage only.
Quality related activities status	Always active Just like other Activities of agile.	Low
Work Sytle	Collaborative with all the Team Members	Developers & QA Team may have conflicting Issues.

① Process Quality

- In order to maintain good quality in s/w process we need exhibit transparency & tightness. These 2 are the main characteristics of high Quality.
- To Maintain Transparency:

@ All the ~~planned~~ Planning sessions are performed when all the people [developers, Customers, testers, System Analysts, etc] of development process are present All of them should participate in release planning & short iteration planning. This will make them to know the project subject and the features involved in it.

Scanned with CamScanner

② High level communication should be present among the team members inorder to decrease misunderstandings and make them clear.

③ Obey | Adhere to customer requirements which are heard by all teammates.

④ Procen Measures should be available all the time to all the people involved in the Develop-ment process..

– To Maintain Tightness:

① One Busines Day is allocated for every 2 weeks to present the work accomplished in previous iteration, reflective thinking & planning for next 9 Development Days. This shows the tight rhythm of the project.

② Tightness controls project by revealing and dealing with unexpected events at early stages.

③ All the reports related to various measures of Development taskes are maintained tightly.

Scanned with CamScanner

② Product Quality

(i) Refactoring:

- The technique of altering the code internally so that the external features do not change.
- By refactoring, code quality can be improved.
- It provides a simple and clear design which is easy to maintain and simplifies future development extensions.

(ii) Pair Programming:

- Instead of one person working on same code, it is better if 2 persons work on same code
- A technique in which two persons sit and work at one workstation is called pair programming
- This is code inspection technique which provides information in 2 levels of abstraction.
- This multi-level examination (2 persons) increases the code quality.

(iii) Acceptance Tests:

- Product quality is also increased by definition of acceptance tests.
- When all the requirements of the customer are met then acceptance Tests are passed.

Scanned with CamScanner

- The acceptance tests will increase the ~~team~~ teammates confidence with respect to correctness in code.

(iv) Test Driven Development (TDD):

- TDD is a technique that enables a step-by-step development of a speific functionality together with its unit test.
- It is a programming technique which aims to provide clean, fault-free code.
- First we write a test case that fails and then we write a possible code to pass the test succesfully.

Test Driven Development (TDD) ***

- TDD is a programming technique which aims to provide clean and fault-free code.
- In TDD, we first write a testcase that fails, and then we write a simple possible code to pass the testcase succesfully.
- TDD implies that new code is added only if an automated test has failed.
- TDD guidline is
  - RED - writing a simple test that fails.

Scanned with CamScanner

- GREEN - write code that passes the test.
- REFACTOR - code quality should be improved without adding / changing functionality.

- With this implementation of TDD, a high quality code will be generated.

How TDD Help to overcome problems in Testing

1. Not enough time to test:

- Initially code is written and later testing will be done. So sometimes because of more time in the development phase, testing phase will get less time to test.

2. Provides negative feedback:

- Testing means to find bugs in the own work (agile). When you find an error in your own work, this may create some negativeness in you. (who would enjoy it?)

3. Responsibility for testing is transferred:

- Some times during the development process, a bug may be fixed by other developer other than who actually developed it. So who is the actual responsibility of that bug

Ex: Bug1 is found in the code developed by person A, But the Bug may be fixed by Person B.

Scanned with CamScanner

4. Testing is a low status job:

- Generally testing is done at the end of the production line (i.e before production)
- Tester should have a great skillset to find bugs in the code written by experts. For this, the testers will struggle to win and gain respect.
- This lack of the status and support make's testers job more difficult and time consuming.

5. Testing is hard to manage:

- Testing slow down the development process because, in the middle the developer should also fix the bugs.
- TDD turn development and testing into a controlled processes.

6. Testing is hard:

- Testing is a difficult task which is from cognitive perspective.
- See a simple failed testcase, fix it and get it passed.
- TDD improves one understanding on what should be developed because, test will be written before we write the code.

Scanned with CamScanner

Advantages & Disadvantages of TDD

↳ See in PPT / text book.
6.5.3.

Measured TDD

- Measured TDD aims to improve the performance of TDD by incorporating measures & control elements into the TDD process itself.
- You are following TDD Technique. Are you successfully? upto what extent are you successfull? How you measure your success?
- Size : No. of lines of code
- Complexity: Calculating cyclomatic complexity.
- Measured TDD has the added value of measuring while developing.

Quality in Learning Environments

1. Size & complexity Measures :

- Example: 19 different functions are developed using measured TDD process.
- Generally one function may have many no. of TDD steps. In that case these 19 functions have around 75 TDD steps (Assume)

Scanned with CamScanner

- One single line of code may be inspected by several test cases; that means there will be more lines of test than lines of code.
- Ex. 19 functions may have (800) lines of code which may have (1582) lines of tests.

② Teaching & Learning Principles:

- A quality work product attributes are:
  - functionality
  - usability
  - Reliability
  - Safety
  - Security
  - Maintainability
  - Portability
  - Efficiency
  - Performance
  - Availability.
- All the S/w Engineers ensure that their products meet the highest professional standards posible.
- For this purpose, the concept of quality should be taught as part of software Engineering (SE) Eduction.

Scanned with CamScanner

- The quality related issues should be integrated & intertwined in all learned topics

① Let learners experience the S/w Development approach:

- Since quality is a complex concept, it is a gradually learning process based on learner's experience.

② Elicit communication:

- Quality is a Multifaced concepts. So first you have to learn them, then you should also teach them to your team members. This increases the further communication.

Scanned with CamScanner