

Lecture 1

Introduction to Software Quality Assurance

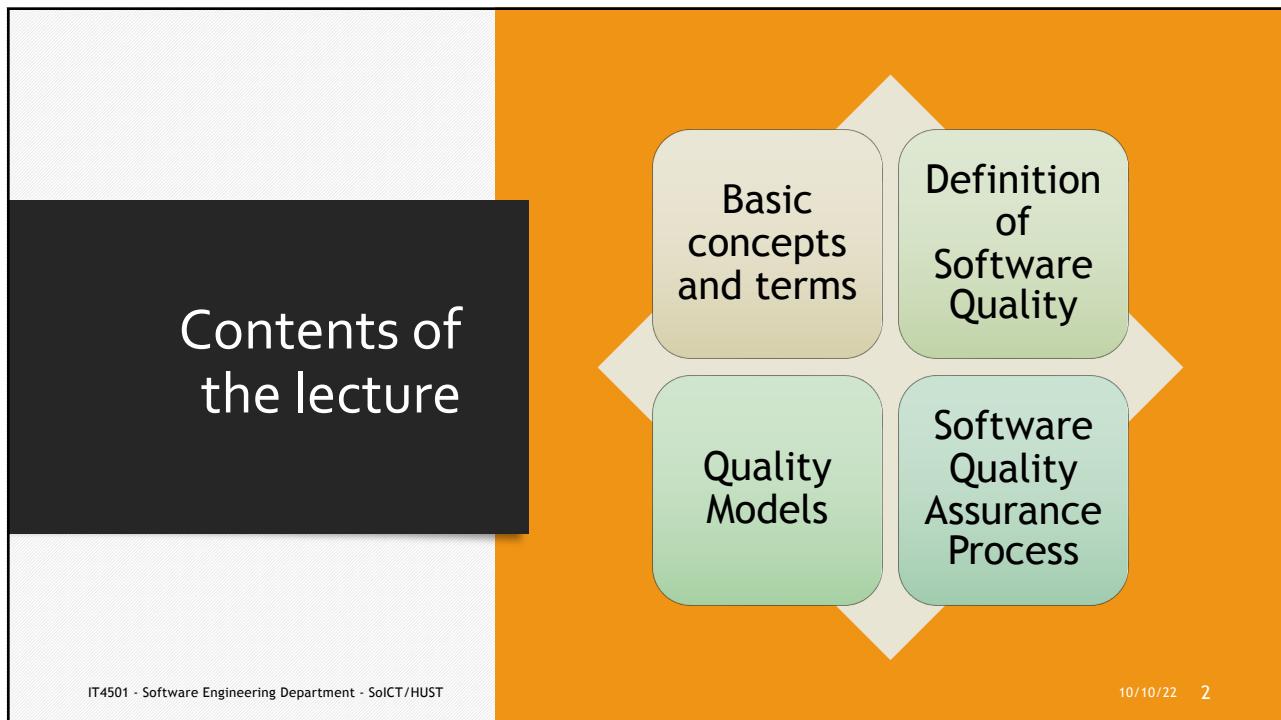
IT4501

Bùi Thị Mai Anh (anhbtm@soict.hust.edu.vn)

Bộ môn Công nghệ phần mềm
Viện công nghệ thông tin và truyền thông
Trường ĐH Bách Khoa Hà Nội

Giới thiệu về Đảm bảo chất lượng phần mềm

1



2

Basic concepts and terms Các khái niệm và thuật ngữ cơ bản

3

- Verification – Xác minh, kiểm tra
 - Verification là hoạt động đánh giá hệ thống phần mềm bằng cách xác định xem hệ thống có thỏa mãn các yêu cầu được nêu ra ban đầu hay không, được nhìn từ quan điểm của người phát triển
 - Sản phẩm ở đây có thể là một sản phẩm trung gian trong quá trình phát triển phần mềm như: tài liệu đặc tả, tài liệu thiết kế, mã nguồn chương trình,...
 - **Verify if the system is built right**
- Validation – Kiểm chứng
 - Validation là hoạt động xác nhận xem sản phẩm phần mềm có thỏa mãn tất cả các yêu cầu của người dùng hay không?
 - **Validate if we built a right system**

3

Failure– Fault – Error – Defect (Lỗi)

4

- Failure: một sự kiện xảy ra bất cứ khi nào hệ thống có một hành vi không tương ứng với các hành vi đặc tả của hệ thống
 - Failure – failed function caused by an error
- Error: là một trạng thái xảy ra lỗi của hệ thống gây ra bởi một Fault.
 - Faulty state
- Fault: thường là lí do dẫn tới xuất hiện Error trong hệ thống
 - Human error: mistake by engineer
 - Fault/Defect/Bug: các bất thường xảy ra trong phần mềm nói chung
- Debugging – Fault localizing (Sửa lỗi hay xác định lỗi)

4

Five views of Quality 5 quan điểm về chất lượng sản phẩm nói chung

5

- **Transcendental view (Quan điểm siêu việt về chất lượng)**: chất lượng là một thứ gì đó mà chúng ta có thể nhận ra nhưng lại không thể định nghĩa một cách cụ thể
- **User view**: Một phần mềm có chất lượng là khi các mục đích của phần mềm đều đạt được.
 - Khi đánh giá chất lượng theo quan điểm của người dùng, chúng ta cần phải trả lời câu hỏi: phần mềm có thỏa mãn các yêu cầu và mong đợi của người dùng hay không?
- **Manufacturing view**: chất lượng được hiểu là những gì tương thích với đặc tả
- **Product view**: chất lượng được xem là gắn liền với các đặc tính cố hữu của sản phẩm
- **Value-based view (Economic)**: chất lượng phụ thuộc vào số tiền mà khách hàng có thể bỏ ra để trả cho sản phẩm của mình

5

Software Quality Chất lượng phần mềm là gì?

6

Một cách chung nhất thì chất lượng của một sản phẩm được hiểu là khi sản phẩm đạt được tất cả các yêu cầu đề ra trong đặc tả của nó. Theo định nghĩa của IEEE thì chất lượng phần mềm gồm hai ý:

- Mức độ một hệ thống hay một thành phần hay một tiến trình đạt được các yêu cầu đặt ra
- Mức độ một hệ thống hay một thành phần hay một tiến trình đạt được những mong đợi của khách hàng hay người dùng hệ thống đó

Software quality is:

- (1)The degree to which a system, component or process meets specified requirements
- (2)The degree to which a system, component or process meets customer or user needs or expectations

(IEEE Std 610.12-1990)

6

Problems in Software Quality Assurance

7

- Vấn đề là luôn tồn tại căng thẳng giữa yêu cầu của người dùng (tính hiệu quả, độ tin cậy...) đối với sản phẩm phần mềm và yêu cầu về phát triển (dễ tài sử dụng, dễ bảo trì...)
 - Customer quality requirements (efficiency, reliability etc.)
 - Developer quality requirements (reusability, maintainability, etc.)
- WHY?
- Đặc tả của sản phẩm phần mềm không bao giờ đầy đủ ngay từ đầu và thường không nhất quán
- Để có được những tiêu chuẩn đánh giá chất lượng ngay từ đầu là rất khó

Software Quality Model Các mô hình chất lượng phần mềm

8

- Định nghĩa của IEEE không đủ để có thể áp dụng vào thực tế đánh giá chất lượng của các dự án phần mềm
- Chất lượng phần mềm cần được đặc tả bằng các mô hình chất lượng
- 2 cách tiếp cận chính khi xây dựng các mô hình chất lượng:
 - Mô hình chuẩn: McCall, ISO/IEC 9126, ISO/IEC 25000, Boehm, Dromey
 - Các mô hình mang tính thực dụng hơn: FURPS, GQM Approach

Factors – Criteria – Metrics Nhân tố – Tiêu chí – Độ đo

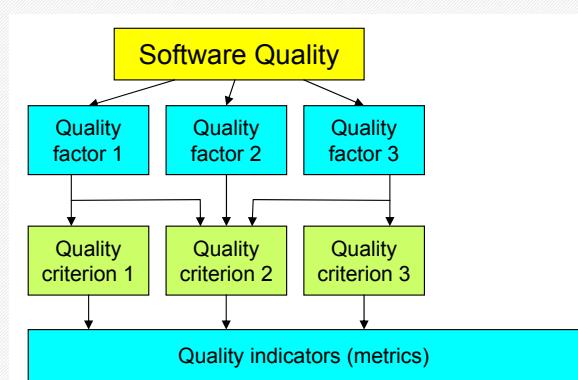
9

- Mỗi một mô hình chất lượng được xây dựng dựa trên:
 - Nhân tố:** Factors trong các mô hình chất lượng được sử dụng để mô tả góc nhìn từ phía bên ngoài đối với sản phẩm phần mềm hay góc nhìn từ quan điểm người dùng sản phẩm.
 - Các nhân tố này cho phép xác định các đặc tính chất lượng của phần mềm đứng từ quan điểm người dùng
 - Tiêu chí – Criteria:** trong các mô hình chất lượng được sử dụng để mô tả góc nhìn từ bên trong của sản phẩm phần mềm, hay góc nhìn về quan điểm chất lượng phần mềm từ người phát triển sản phẩm.
 - Các tiêu chí chất lượng cũng được coi như các đặc điểm được dùng để định nghĩa các nhân tố chất lượng.
- Độ đo – Metrics:** được định nghĩa và sử dụng để cung cấp một thước đo và phương pháp cụ thể để đo lường một cách định tính hoặc định lượng các tiêu chí chất lượng phần mềm

9

Factors – Criteria – Metrics in Software Quality Model

10



10

McCall's Quality Factors

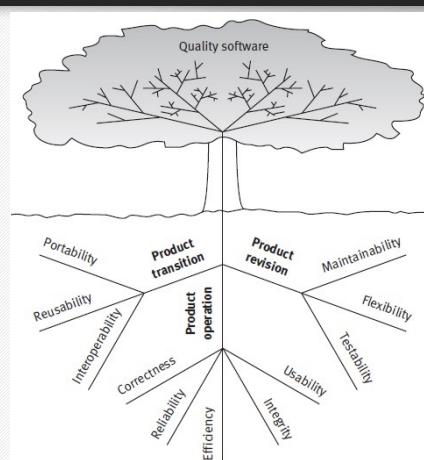
11

- McCall coi **nhân tố chất lượng** là một biểu hiện về đặc điểm hành vi của hệ thống
- 3 nhóm:
 - Nhóm vận hành sản phẩm - Product Operation
 - Nhóm đánh giá sản phẩm - Product Revision
 - Nhóm chuyển giao sản phẩm - Product Transition
- **11 nhân tố chất lượng**

11

McCall's Factor Model Tree Cây nhân tố chất lượng của mô hình McCall

12



Source: from Internet

12

11 quality factors of McCall's Model 11 nhân tố chất lượng của mô hình McCall

13

Quality Categories	Quality Factors	Broad Objectives
Product operation	Correctness	Does it do what the customer wants?
	Reliability	Does it do it accurately all of the time?
	Efficiency	Does it quickly solve the intended problem?
	Integrity	Is it secure?
	Usability	Can I run it?
Product revision	Maintainability	Can it be fixed?
	Testability	Can it be tested?
	Flexibility	Can it be changed?
Product transition	Portability	Can it be used on another machine?
	Reusability	Can parts of it be reused?
	Interoperability	Can it interface with another system?

Source: Software
Testing and Quality Assurance
Theory and Practice

13

McCall's Quality Criteria Các tiêu chí chất lượng của mô hình McCall

14

- A **quality criterion** is an attribute of a **quality factor** that is related to software development
 - Ex: Modularity is a quality criterion related to the quality factor maintainability
 - Traceability is a quality criterion related to the correctness of a system
- There are 23 quality criteria of McCall's Model

14

23 quality criteria of McCall

23 tiêu chí chất lượng đưa ra bởi McCall

15

Quality Criteria	Definition
Access audit (kiểm tra truy nhập)	Ease with which software and data can be checked for compliance with standards or other requirements
Access control (kiểm soát truy nhập)	Provisions for control and protection of the software and data
Accuracy (Tính chính xác)	Precision of computations and output
Communication commonality (Độ tương đồng giao tiếp)	Degree to which standard protocols and interfaces are used
Completeness (Tính đầy đủ)	Degree to which a full implementation of the required functionalities has been achieved

15

23 quality criteria of McCall

23 tiêu chí chất lượng đưa ra bởi McCall

16

Quality Criteria	Definition
Communicativeness (Sự sẵn sàng giao tiếp)	Ease with which inputs and outputs can be assimilated
Conciseness (Tính xúc tích)	Compactness of the source code, in terms of lines of code
Consistency (Tính nhất quán)	Use of uniform design and implementation techniques and notation throughout a project
Data commonality (Độ tương đồng dữ liệu)	Use of standard data representations
Error tolerance (Độ dung thứ lỗi)	Degree to which continuity of operation is ensured under adverse conditions
Execution efficiency	Run time efficiency of the software

16

23 quality criteria of McCall

23 tiêu chí chất lượng đưa ra bởi McCall

17

Quality Criteria	Definition
Expandability (Khả năng mở rộng)	Degree to which storage requirements or software functions can be expanded
Generality (Tính khái quát hoá)	Breadth of the potential application of software components
Hardware independence (Độc lập phần cứng)	Degree to which the software is dependent on the underlying hardware
Instrumentation (Thiết bị đo đạc)	Degree to which the software provides for measurement of its use or identification of errors
Modularity (Tính dày dặn)	Provision of highly independent modules
Operability (Khả năng vận hành)	Ease of operation of the software

17

23 quality criteria of McCall

23 tiêu chí chất lượng đưa ra bởi McCall

18

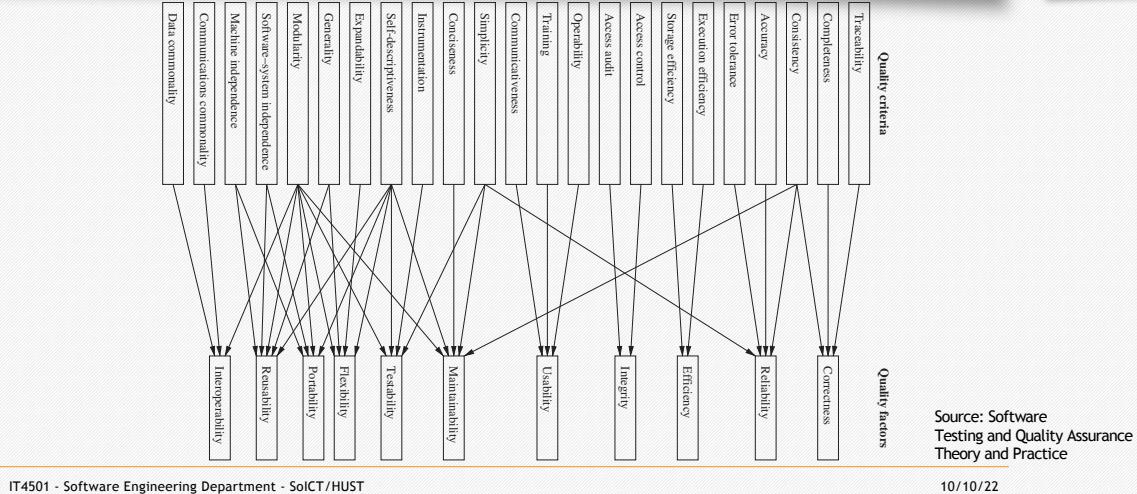
Quality Criteria	Definition
Self-documentation (Tự đặc tả tài liệu)	Provision of in-line documentation that explains implementation of components
Simplicity (Tính đơn giản)	Ease with which the software can be understood
Software system independence (Độc lập phần mềm)	Degree to which the software is independent of its software environment—nonstandard language constructs, operating system, libraries, database management system, etc.
Software efficiency (Hiệu năng phần mềm)	Run time storage requirements of the software
Traceability (Khả năng theo vết)	Ability to link software components to requirements
Training (Đào tạo)	Ease with which new users can use the system

18

Relationship between Quality factors and criteria

Mối quan hệ giữa nhân tố và tiêu chí chất lượng

19



19

Relationship between Quality factors and criteria

Mối quan hệ giữa nhân tố và tiêu chí chất lượng

20

Có hai đặc điểm về mối quan hệ cần lưu ý:

- Cải thiện một nhân tố chất lượng có thể làm ảnh hưởng tiêu cực đến các nhân tố chất lượng khác.
 - Ví dụ: Chúng ta muốn tăng khả năng kiểm thử được của một phần mềm thì tính hiệu quả của phần mềm có thể bị giảm. Bởi vì để làm cho code có thể kiểm thử được, lập trình viên không thể viết code một cách súc tích ngắn gọn do đó tính hiệu quả của code có thể bị giảm
 - Tương tự để tăng tính khả chuyển của code, lập trình viên cần viết code theo cách dễ hiểu do đó cũng ko thể viết theo cách súc tích ngắn gọn và sẽ làm giảm tính hiệu quả của code.
- Một số nhân tố chất lượng có thể ảnh hưởng tích cực đến các nhân tố chất lượng khác
 - Ví dụ: Tăng tính đúng đắn của hệ thống sẽ tăng độ tin cậy của hệ thống đó hoặc tăng tính kiểm thử được sẽ cải thiện khả năng bảo trì của hệ thống

20

Quality Metrics Độ đo chất lượng

21

- Các yếu tố chất lượng không thể đo đạc được trực tiếp
 - Làm thế nào để đo được mức độ kiểm thử được của hệ thống?
- Một độ đo chất lượng là một đơn vị đo cho phép đặc tả một số khía cạnh của một tiêu chí chất lượng. Các độ đo có thể được hình thành bằng cách:
 - Xây dựng một tập các câu hỏi liên quan đến các tiêu chí chất lượng tương ứng và đưa ra các câu trả lời "có" hoặc "không" cho mỗi câu hỏi
 - Tính % số câu trả lời có để đạt được một độ đo cụ thể cho tiêu chí chất lượng tương ứng
- Ví dụ: **self-descriptiveness – tính tự đặc tả được**
 - Liệu tất cả các tài liệu có được viết một cách rõ ràng và đơn giản để tất cả các chức năng, thuật toán, thủ tục đều dễ dàng hiểu được
 - Liệu thiết kế phía sau mỗi module có rõ ràng và dễ hiểu?

21

Boehm's Quality Model Mô hình chất lượng của Boehm

22

- Mô hình Boehm đơn giản hóa mô hình của McCall để có thể đánh giá một cách tự động và định lượng chất lượng của phần mềm.
- Trong mô hình này các yếu tố chất lượng được xếp vào 3 đặc tính chất lượng mức cao ví dụ: tính dễ sử dụng, tính bảo trì được và tính khả chuyển
- 7 yếu tố chất lượng khác được phân vào 3 nhóm này gọi là các đặc điểm chất lượng mức trung gian

22

7 quality factors of Boehm's Model

23

Factors	Criteria
Portability	Self-contentedness, device independence
Reliability	Self-contentedness, accuracy, completeness, robustness, integrity, consistency
Efficiency	Accountability, device efficiency, accessibility
Usability	Completeness
Testability (Human Engineering)	Accountability, communicativeness, self descriptiveness, structured
Understandability	Consistency, structured, conciseness
Modifiability (Flexibility)	Structured, expandability

23

Dromey's Quality Model

24

- Mô hình chất lượng của Dromey được đề nghị để đánh giá các pha phân tích yêu cầu, thiết kế và cài đặt của phát triển phần mềm.
- Các nhân tố chất lượng có liên quan đến việc cài đặt hệ thống phần mềm bao gồm:
 - Tính đúng đắn - Correctness
 - Cấu trúc bên trong - Internal
 - Tính ngữ cảnh - Contextual
 - Mức độ đặc tả - Descriptive

24

Dromey's quality factors and criteria Các nhân tố và tiêu chí chất lượng của mô hình Dromey

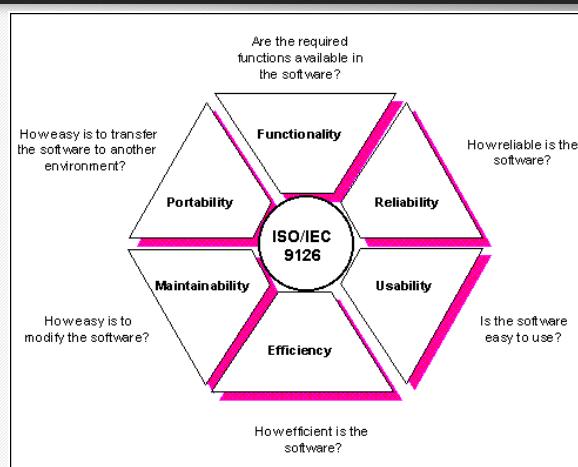
25

Factors	Criteria
Correctness	Functionality, Reliability
Internal	Maintainability, Efficiency, Reliability
Contextual	Maintainability, Reusability, Portability, Reliability
Descriptive	Maintainability, Efficiency, Reliability, Usability

25

ISO/IEC 9126 Quality Model Mô hình chất lượng ISO/IEC 9126

26

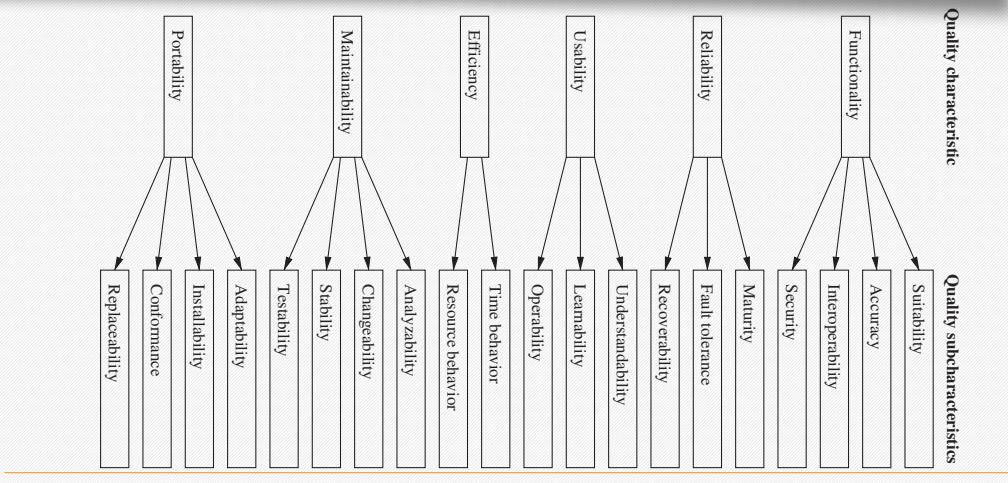


Source: from Internet

26

20 quality subcharacteristics of ISO/IEC 9126

27



IT4501 - Software Engineering Department - SolCT/HUST

10/10/22

27

Comparison of McCall and ISO/IEC 9126 So sánh hai mô hình (sự tương đồng)

28

- Các nhân tố chất lượng của mô hình McCall là các thuộc tính chất lượng trong mô hình ISO/IEC
- Tiêu chí chất lượng của McCall tương ứng với các thuộc tính con trong mô hình ISO/IEC
- Một số thuộc tính chung của cả hai mô hình
 - reliability
 - usability
 - efficiency
 - maintainability
 - portability

IT4501 - Software Engineering Department - SolCT/HUST

10/10/22

28

Comparison of McCall and ISO/IEC 9126 So sánh hai mô hình (sự khác nhau)

29

- Mô hình ISO 9126 nhấn mạnh các thuộc tính chất lượng có ý nghĩa đối với người dùng trong khi McCall model còn coi trọng các thuộc tính bên trong của sản phẩm
 - Ví dụ, reusability, người phát triển cố gắng thiết kế một phần mềm sao cho nhiều thành phần có thể tái sử dụng được, nhưng đặc tính chất lượng đó thì không hề được người dùng nhận biết được
- Trong McCall model, một tiêu chí chất lượng có thể ảnh hưởng đến nhiều nhân tố chất lượng trong khi trong mô hình ISO 9126, một thuộc tính chất lượng con chỉ ảnh hưởng duy nhất một thuộc tính chất lượng mức cao.

29

Hewlett Packard: F.U.R.P.S Mô hình FURPS của HP

30

- Mô hình FURPS của HP được ra đời từ các thống kê về dự án của HP năm 1987 được dùng để cải thiện chất lượng sản phẩm của HP
- Factors:
 - Functionality:** functions it performs, their generality and security
 - Usability:** consistency, documentation
 - Reliability:** frequency and severity of failure, accuracy of output
 - Performance:** response time, resource consumption
 - Supportability:** can it be extended, adapted, corrected?
- FURPS mang tính ứng dụng cao hơn và được áp dụng trong công nghiệp sản xuất phần mềm của một số công ty phần mềm lớn

30

Factors and Criteria of FURPS

31

Factors	Criteria
Functionality	Capability, security
Usability	Consistency, user documentation, training materials
Reliability	Frequency and security of failure, recoverability, predictability, accuracy, mean time between failure
Performance	Speed efficiency, availability, accuracy, throughput, response time, recovery time, resource usage
Supportability	Testability, extensibility, adaptability, maintainability, compatibility, configurability, serviceability, install ability, localizability

31

GQM: Goal-Question-Metric Mô hình GQM

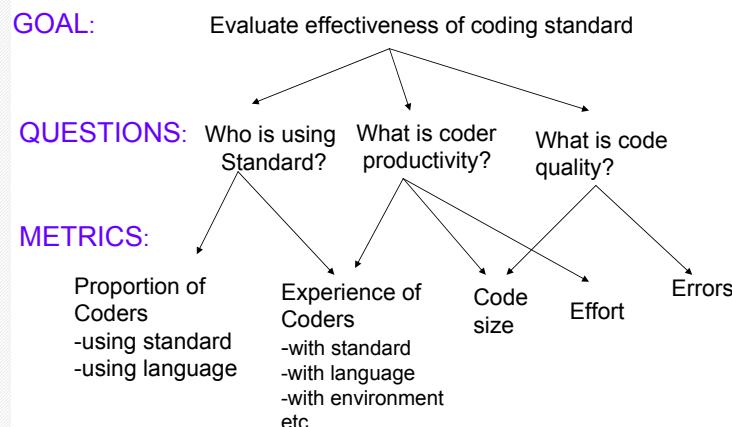
32

- Mô hình GQM được đưa ra hướng tới tính áp dụng cao vì nó được xây dựng dựa trên mục đích của việc kiểm tra giám sát chất lượng.
- Mô hình này gồm có 3 bước:
 - Liệt kê tất cả các mục tiêu chính của dự án phát triển hoặc bảo trì
 - Từ mỗi mục tiêu, cần đưa ra các danh sách câu hỏi có liên quan mà việc trả lời câu hỏi này sẽ xác định mục tiêu đó có đạt được hay không
 - Quyết định cái gì cần được đo đạc để trả lời các câu hỏi tương ứng với mục tiêu

32

Example of GQM

33



33

GQM Discussion Ưu điểm và nhược điểm của mô hình GQM

34

Mô hình GQM có các lợi điểm như sau:

- Vì mô hình này chỉ tập trung vào việc tạo ra các phép đo cần thiết để đánh giá được mục tiêu đề ra nên nó tương đối hiệu quả và ngắn gọn
- Nhiều phép đo khác nhau có thể cần thiết để trả lời được một câu hỏi đề ra
- Cùng một phép đo có thể được sử dụng để trả lời cho các câu hỏi khác nhau
- Mục tiêu chất lượng đề ra các mục đích cần thiết cho việc thu thập dữ liệu

34

GQM Discussion

Ưu điểm và nhược điểm của mô hình GQM

35

Mô hình GQM thì có những vấn đề sau:

- Mô hình này cần kết hợp các phép đo theo một cách nào đó để có thể trả lời cho các câu hỏi do đó, đôi khi việc áp dụng là không dễ dàng và đòi hỏi kinh nghiệm của người phát triển và người quản lý dự án
- Nhiều khi cần đến nhiều mô hình khác nhau để có thể diễn tả mối quan hệ giữa các phép đo.
- Mô hình này cần nhiều nỗ lực để có thể đưa ra được mục tiêu và các phép đo cụ thể
- So sánh với các mô hình chuẩn thì mô hình này có khả năng gây ra lỗi cao hơn

35

Quality Assurance Process

Quy trình đảm bảo chất lượng phần mềm

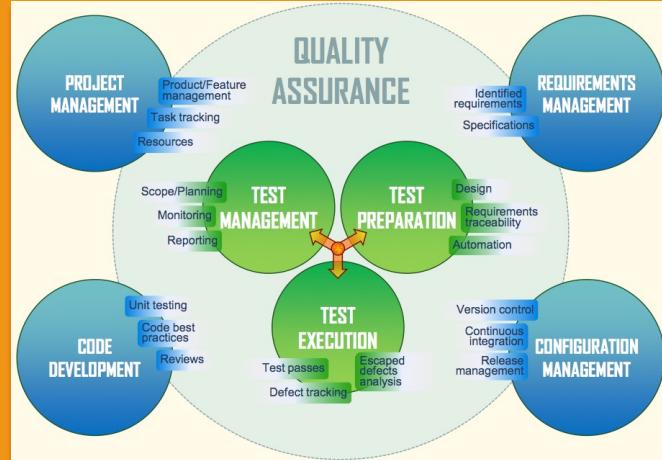
36

“Quality Assurance is any systematic process of checking to see whether a product or service being developed is meeting its specified requirements”

36

QA Process

- Quy trình đảm bảo phần mềm cần được tích hợp với tất cả các quy trình khác trong toàn bộ vòng đời phát triển phần mềm.
- Có 4 quy trình là:
 - quản lý dự án
 - quản lý yêu cầu
 - quản lý cấu hình
 - quản lý mã nguồn



QA Dependencies – Project Management Vai trò của QA trong quá trình quản lý dự án

38

Ngay tại bước lập kế hoạch cho dự án, quy trình đảm bảo chất lượng cần được khởi động:

- Xác định các mục tiêu và tiêu chí chất lượng cần quan tâm và kiểm chứng trong dự án
- Xác định các thủ tục kiểm thử và đảm bảo chất lượng cần được áp dụng
- Chỉ định trước các nguồn lực cần thiết (bao gồm con người, phần cứng, hệ thống cơ sở hạ tầng và môi trường cần thiết để thực hiện kiểm thử và đảm bảo chất lượng)

QA Dependencies - Requirements Management

Vai trò của QA trong quản lý yêu cầu phần mềm

39

Quy trình QA cần thực hiện các bước sau:

- Xác định tất cả các khía cạnh quan trọng của các chức năng của hệ thống và cố gắng để nhận được xác nhận từ phía khách hàng
- Xác định các hạn chế và ngoại lệ có thể có từ các yêu cầu của khách hàng
- Thiết lập tất cả các hiểu biết chung về dự án cho tất cả các thành viên của đội dự án, để đảm bảo đội dự án nắm được các vấn đề của dự án
- Cung cấp cơ sở cho các ước tính dự án, phát triển mã nguồn và các quy trình kiểm chứng
- Là một điểm khởi đầu cho việc giao tiếp hiệu quả và quản lý thay đổi một cách hiệu quả trong suốt quy trình phát triển sản phẩm

39

QA Dependencies - Configuration Management

QA đối với quy trình quản lý cấu hình

40

- Quản lý cấu hình phần mềm là bắt buộc trong mọi dự án phát triển phần mềm.
- Đảm bảo chất lượng phần mềm yêu cầu phải quản lý cấu hình một cách chặt chẽ và tuân theo các quy trình cụ thể.
 - Để đảm bảo cho các đội nhóm phát triển phân tán có thể cùng làm việc trên một mã nguồn
 - Đảm bảo cho baseline version có thể độc lập không bị ảnh hưởng bởi bất kỳ sự phá vỡ cấu trúc của một lỗi nào
 - Để biết được ở phiên bản mã nguồn nào thì lỗi xuất hiện lần đầu tiên và xác định được những thay đổi nào làm xuất hiện lỗi đó
 - Quản lý cấu hình tốt thì sẽ đảm bảo kiểm thử tự động thực hiện tốt nhất chức năng của nó ở mỗi và mọi giai đoạn mà version mã nguồn mới được ra đời

40

QA Dependencies - Code Development

Vai trò của QA đối với quy trình phát triển mã nguồn

41

Áp dụng quy trình vào phát triển mã nguồn sẽ làm tăng hiệu quả của quy trình đảm bảo chất lượng phần mềm.

- Phát triển mã nguồn theo một kiểu thống nhất sẽ làm tăng hiệu quả và làm cho dễ dàng hơn việc bảo trì sản phẩm mà không cần lệ thuộc cứng vào một hoặc một nhóm thành viên của dự án.
- Cho phép phát hiện lỗi sớm nhất và hiệu quả sửa lỗi là thấp nhất
- Cho phép thiết kế mã nguồn hướng kiểm thử
- Cho phép cục bộ hóa lỗi và giảm thiểu các tác động lên các thành phần phụ thuộc
- Chia sẻ tri thức giữa các thành viên trong đội dự án và cho phép cải tiến kỹ năng cá nhân của các thành viên

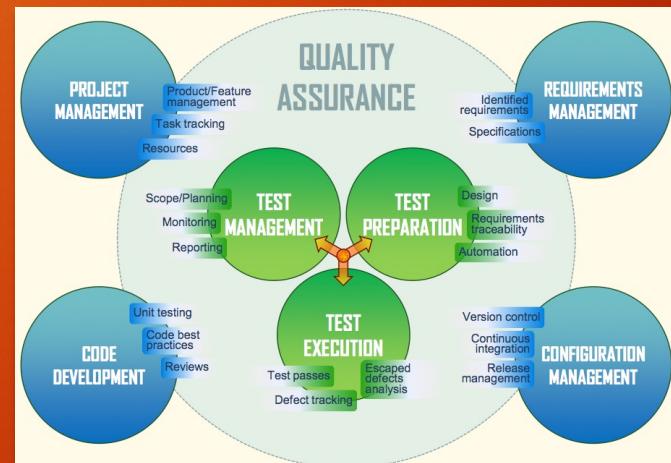
41

QA Core Components - Testing Process

Quy trình kiểm thử là nền tảng của quy trình QA

Kiểm thử có thể được thực hiện trong các pha riêng biệt hoặc các pha chồng chéo

- Kiểm thử ở nhiều mức độ khác nhau của sản phẩm: đơn vị, tính năng, tính năng tương tác, hệ thống
- Kiểm thử trong suốt quá trình bảo trì sản phẩm: khói, sanity, hồi quy (regression)
- Kiểm thử các khía cạnh đặc biệt: hiệu năng, hiệu suất, độ ổn định, khả năng tương tác, độ tuân thủ các chuẩn/quy định



42

Test Management

Quản lý kiểm thử trong quy trình QA

43

- Việc quản lý kiểm thử cần được thực hiện ngay từ giai đoạn đầu tiên và kéo dài đến tận khi dự án hoàn toàn kết thúc nhằm đạt được các mục tiêu về cải tiến chất lượng sản phẩm
- Các mục đích chính của quản lý kiểm thử:
 - Chiến lược và phạm vi kiểm thử - Xác định các chiến lược kiểm thử và phạm vi các chức năng của dự án
 - Các ràng buộc trong quá trình kiểm thử
 - Các thành phần, các tính năng hoặc các chức năng cần phải kiểm thử và không cần phải kiểm thử
 - Kiểm thử xem các mục nào thông qua và mục nào bị lỗi cần phải sửa chữa, thay đổi. Quy trình kiểm thử và các tiêu chí quay lại tiếp tục kiểm thử sau khi đã sửa lỗi
 - Các công việc kiểm thử và các yêu cầu về môi trường và triển khai cần thiết để tiến hành kiểm thử
 - Các trách nhiệm, phân công công việc và việc đào tạo cần thiết hoặc kết hợp với các đội nhóm phát triển khác trong quá trình kiểm thử
 - Lên kế hoạch kiểm thử cụ thể và xác định các rủi ro có thể có
 - Đánh giá bằng các độ đo chất lượng và viết báo cáo

43

Test Preparation

Chuẩn bị kiểm thử trong quy trình QA

44

- Mục tiêu của việc chuẩn bị kiểm thử là để xác định được các loại kiểm thử và môi trường kiểm thử cần thiết cho từng pha phát triển dự án theo kế hoạch kiểm thử
- Các hoạt động chính của chuẩn bị kiểm thử gồm có:
 - Phân tích các yêu cầu chức năng, các đặc tả và phạm vi của các chức năng
 - Phát triển ma trận theo vết yêu cầu (RTM – Requirement Traceability Matrix)
 - Thiết kế và phát triển các trường hợp kiểm thử cho tất cả các pha đã lên kế hoạch
 - Sử dụng các cách tiếp cận và các kĩ thuật để đảm bảo được một tập hợp các ca kiểm thử hiệu quả và tối ưu
 - Phát triển hoặc sử dụng thêm các công cụ để thực hiện kiểm thử tự động
 - Thiết lập các môi trường đặc biệt để có thể thực hiện kiểm thử phi chức năng

44

Test Execution

Thực thi kiểm thử trong quy trình QA

45

Các hoạt động thực thi kiểm thử thông thường:

- Chuẩn bị môi trường thực thi kiểm thử
- Thực thi các ca kiểm thử
- Phân tích các kết quả thực thi
- Thu thập và ghi lại các kết quả thực thi kiểm thử và ghi lại các lỗi
- Hỗ trợ đội phát triển để tái hiện lại các lỗi và phân tích các lỗi
- Kiểm chứng các ca kiểm thử đã được sửa chữa
- Phân tích các lỗi có thể bị bỏ sót
- Cập nhật các ca kiểm thử và môi trường kiểm thử theo kết quả của Kiểm thử thực thi và phân tích các lỗi có thể bỏ sót

45

Summary

Tóm lược lại nội dung bài học

46

- Principal terms of Quality Assurance
- IEEE's definition of Software Quality
- Quality models: McCall, Boehm, Dromey, ISO/IEC 9126, FURPS, GQM
- QA process and main activities of:
 - Test Preparation
 - Test Management
 - Test Execution

46