

Contents

A	3
acceptance testing (kiểm tra chấp nhận):	3
alpha release (bản phát hành alpha):	3
Ambiguity (Tính mơ hồ):	3
analysis workflow (Quy trình phân tích):	3
application domain (miền ứng dụng) :	3
architectural design:	4
B	4
beta release 86 (bản phát hành beta):	4
budget 82 (ngân sách - hay còn gọi cách khác là chi phí sẽ là bao nhiêu):	4
business case 79 ():	4
business model 89 (mô hình kinh doanh):	5
C	5
capability maturity model (CMM) 95 :	5
CMM lvl 3 khác j CMM lvl 2 ?	5
class 82 (lớp):	5
code artifact 83:	5
component 83 (thành phần):	6
concept exploration 79 (thăm dò ý tưởng)	6
construction phase 92 (Pha xây dựng)	6
contradiction 81 (mâu thuẫn)	6
core workflow 78 (quy trình cốt lõi)	6
cost 79 (chi phí)	7
COTS : (Commercial-off-the-shelf Software)	7
D	7
deadline 79 (thời hạn cuối cùng)	7
defined level (<i>Mức độ xác định</i>) 96	7
deliverable (tạm dịch: <i>Sản phẩm cung cấp</i>) 82	7
design workflow (<i>Pha/Quy trình thiết kế</i>) 82	7
detailed design (<i>Thiết kế chi tiết</i>) 82	8
domain (<i>Lĩnh vực</i>) 78	8
E	9
elaboration phase (<i>Pha thiết lập</i>) 91	9

I	9
implementation workflow (<i>Pha/Quy trình thực hiện</i>)	83
inception phase	89 (Pha đánh giá)
incompleteness	81 (tính không đầy đủ)
initial level	95 (cấp độ ban đầu)
integration testing	86 (thử nghiệm thích hợp)
International Organization for Standardization (ISO)	98
ISO 9000-3	98
ISO 9001	98
ISO/IEC 15504	99
K	11
key process area (KPA)	98
M	11
managed level	96
maturity	95
milestone	82
model 76 (Mô hình)	12
module	82
O	12
optimizing level 96 (Cấp độ tối ưu hoá)	12
P	12
product testing	86 (Thử nghiệm sản phẩm)
R	12
regression testing	87 (Thử nghiệm hồi quy)
reliability 79 (Độ tin cậy)	13
repeatable level 96 (Mức có khả năng lặp lại)	13
requirements workflow 78 (Quy trình yêu cầu)	13
retirement	88
S	14
specification document 81 (Tài liệu đặc tả):	14
SPICE	99
T	15
test workflow 84 (Quy trình kiểm tra)	15
traceability 84 (truy xuất nguồn gốc):	15

transition phase 92 (Pha chuyển tiếp):	15
U	15
Unified Modeling Language (Ngôn ngữ mô hình hóa thống nhất) (UML) 76:	15
Unified Process (tiến trình hợp nhất) 76	16
unit testing 85 (kiểm thử đơn vị):	17
W	17
workflow là gì? Khác gì với process?	17
Việc trích lớp và xây dựng các lớp là của pha thiết kế, tại sao người ta làm nó ngay trong pha phân tích?	17
Tại sao việc làm tài liệu đặc tả lại quan trọng đến vậy	17

A

acceptance testing (kiểm tra chấp nhận):

là phương pháp khi phần mềm được giao cho khách hàng, người kiểm tra nó trên phần cứng thực tế, sử dụng dữ liệu thực tế trái ngược với dữ liệu thử nghiệm

alpha release (bản phát hành alpha):

Alpha release là phiên bản được kiểm thử hoạt động chức năng thực tế hoặc giả lập do một số ít người dùng/khách hàng được chỉ định hoặc một nhóm test thực hiện tại nơi sản xuất phần mềm. Alpha release được thực hiện ngay sau kiểm thử hệ thống (Product Testing). Alpha release thường áp dụng cho các sản phẩm COTS (MS Office, Windows,...) là một hình thức kiểm thử chấp nhận nội bộ, trước khi phần mềm được tiến hành kiểm thử beta.

Ambiguity (Tính mơ hồ):

Tính mơ hồ, được hiểu là khả năng diễn giải các từ hoặc cụm từ theo những cách khác nhau, tạo thành một vấn đề nghiêm trọng trong phát triển phần mềm. Khi xem xét sự mơ hồ, điểm đầu tiên cần nhận mạnh là có nhiều dạng mơ hồ khác nhau. Các từ thường có nghĩa khác nhau.

analysis workflow (Quy trình phân tích):

Quy trình phân tích là quá trình kiểm tra quy trình làm việc của tổ chức, nói chung nhằm mục đích nâng cao hiệu quả hoạt động, phân tích, phân rã và mịn hóa yêu cầu của khách hàng. Nó xác định các lĩnh vực cải tiến quy trình như các nhiệm vụ hoặc quy trình dư thừa, bố trí nơi làm việc kém hiệu quả và các nút thắt cổ chai trong quy trình làm việc.

application domain (miền ứng dụng) :

tức là môi trường cụ thể mà sản phẩm phần mềm mục tiêu được vận hành. Mỗi miền ứng dụng có không gian địa chỉ ảo của riêng mình, phạm vi tài nguyên cho miền ứng dụng bằng cách sử dụng không gian địa chỉ đó.

architectural design:

thiết kế kiến trúc là việc phân rã sản phẩm thành các mô-đun. Thiết kế kiến trúc xác định các hệ thống con tạo nên một hệ thống và khuôn khổ để điều khiển và giao tiếp hệ thống con. Đầu ra của quá trình thiết kế này là một mô tả về kiến trúc phần mềm. Thiết kế kiến trúc là một giai đoạn đầu của quá trình thiết kế hệ thống. Nó thể hiện mối liên hệ giữa các quá trình đặc tả và thiết kế và thường được thực hiện song song với một số hoạt động đặc tả. Nó liên quan đến việc xác định các thành phần hệ thống chính và thông tin liên lạc của chúng.

B

beta release 86 (bản phát hành beta):

Bản phát hành alpha đã hiệu chỉnh được gọi là bản phát hành beta; nói chung, bản phát hành beta dự định gần với phiên bản cuối cùng. (Các điều khoản phát hành alpha và phát hành beta thường được áp dụng cho tất cả các loại sản phẩm phần mềm, không chỉ riêng phần mềm COTS.)

Được thực hiện sau alpha release, được phát hành tới một số nhóm khách hàng bên ngoài nhóm phát triển phần mềm để tăng phạm vi phản hồi từ người sử dụng tương lai lớn nhất. Beta release gần như phiên bản cuối cùng (final version)

tham khảo ngân hàng:

"beta release là bước được thực hiện sau alpha release, được phát hành tới một số nhóm khách hàng bên ngoài nhóm phát triển phần mềm để tăng phạm vi phản hồi từ người dùng. Đây gần như là bản cuối cùng."

budget 82 (ngân sách - hay còn gọi cách khác là chi phí sẽ là bao nhiêu):

là một trong 3 thành phần chính của kế hoạch khi các trích dẫn cụ thể đã được khách hàng chấp thuận, thì việc chuẩn bị kế hoạch quản lý dự án phần mềm sẽ bắt đầu. Ngân sách phát triển phần mềm thông minh dựa trên ước tính về số lượng đơn vị mà nhóm có thể hoàn thành trong mỗi sprint. Những điểm này sẽ giúp xác định khoảng thời gian cần thiết cho mỗi lần lặp lại và do đó, dự đoán vận tốc nhóm và chi phí tổng thể của dự án.

business case 79 ():

một khía cạnh quan trọng của phát triển phần mềm là trường hợp kinh doanh (**business case**) , một tài liệu chứng minh hiệu quả chi phí của sản phẩm mục tiêu. (Trên thực tế, "chi phí" không phải lúc nào cũng hoàn toàn là tài chính. Ví dụ, phần mềm quân sự thường được xây dựng vì các lý

do chiến lược hoặc chiến thuật. Ở đây, chi phí của phần mềm là thiệt hại tiềm tàng có thể phải chịu trong trường hợp không có vũ khí đã phát triển.)

(VD: Tại cuộc họp đầu tiên giữa khách hàng và nhà phát triển, khách hàng phác thảo sản phẩm. Theo quan điểm của các nhà phát triển, mô tả của khách hàng về sản phẩm mong muốn có thể không hợp lý, mâu thuẫn hoặc đơn giản là không thể đạt được -> Nhiệm vụ của các nhà phát triển ở giai đoạn này là xác định chính xác những gì khách hàng cần và tìm ra từ khách hàng những ràng buộc nào tồn tại.)

business model 89 (mô hình kinh doanh):

Khi các nhà phát triển đã hiểu đầy đủ về miền, bước thứ hai là xây dựng mô hình kinh doanh, tức là mô tả các quy trình kinh doanh của khách hàng. Nói cách khác, nhu cầu đầu tiên là hiểu chính miền và nhu cầu thứ hai là hiểu chính xác cách tổ chức khách hàng hoạt động trong miền đó.

C

capability maturity model (CMM) 95 :

Là các chiến lược có thể sử dụng để cải thiện quy trình phần mềm mà không phụ thuộc vào bất cứ một mô hình vòng đời phần mềm (life-cycle model) nào cả.

CMM được chia thành các loại: CMMs cho phần mềm (SW – CMM), quản lý nguồn nhân lực (P – CMM; P là viết tắt của “people”), hệ thống kỹ thuật (SE – CMM), Phát triển sản phẩm tích hợp (IPD – CMM) và mua lại phần mềm (SA – CMM).

- Các mức tăng trưởng trong mô hình CMM: 5 mức

+ Lv1 : Initial Level: không có công nghệ phần mềm được áp dụng ở đây. Mọi thứ đều thực hiện 1 cách rất cơ bản. Tuy nhiên do thiếu kinh nghiệm nên nhìn chung giá thành và thời gian sẽ bị đội lên.

+ Lv2: Repeatable Level: Ứng dụng những kỹ năng công nghệ phần mềm cơ bản vào công việc. Có sự quản lý về thời gian và giá thành cho dự án. Khi có sự cố nhờ có sự tính toán nên có thể kiểm soát được tình hình.

+ Lv3: Defined Level: Quy trình phần mềm đã có đầy đủ tài liệu. Kỹ thuật quản lý cũng như hỗ trợ kỹ thuật được định nghĩa rõ ràng. Và có sự cố gắng tối đa để cải thiện quy trình dự án. Trong level này, giới thiệu công nghệ mới chỉ làm cho công việc bị gián đoạn và k có lợi.

+ Lv4: (Managed Level) Từ mức này rất ít công ty có thể đạt đến. Đặt ra chất lượng và năng suất mục tiêu cho từng dự án. 2 yếu tố trên được xác định thường xuyên và thực hiện các biện pháp cần thiết để bảo đảm mục tiêu. Sử dụng quản lý chất lượng thống kê (Statistic Quality Control).

+ Lv5: (Optimizing Level): Tiếp tục cải thiện quy trình phần mềm. kỹ thuật quản lý quy trình và chất lượng thống kê (Statistical Quality & Process Control Techniques) được ứng dụng để

định hướng tổ chức. Kinh nghiệm cho dự án trước phục vụ cho dự án sau. Và có sự tiếp thu phản hồi ý kiến khách hàng, cải thiện năng suất & chất lượng.

CMM lvl 3 khác j CMM lvl 2 ?

Sự khác biệt quan trọng giữa mức độ trưởng thành 2 và mức độ trưởng thành 3 là phạm vi của các tiêu chuẩn, mô tả quy trình và thủ tục. Ở cấp độ trưởng thành 2, các tiêu chuẩn, mô tả quy trình và thủ tục có thể khá khác nhau trong từng trường hợp cụ thể của quy trình (ví dụ: trên một dự án cụ thể). Ở cấp độ trưởng thành 3, các tiêu chuẩn, mô tả quy trình và thủ tục cho một dự án được điều chỉnh từ bộ quy trình tiêu chuẩn của tổ chức để phù hợp với một dự án hoặc đơn vị tổ chức cụ thể. Bộ quy trình tiêu chuẩn của tổ chức bao gồm các quy trình được giải quyết ở mức độ chín muồi 2 và mức độ chín muồi 3. Kết quả là, các quá trình được thực hiện trong toàn tổ chức là nhất quán ngoại trừ những khác biệt được cho phép bởi các hướng dẫn điều chỉnh.

class 82 (lớp):

là một loại mô-đun cụ thể, là cơ sở của mô hình hướng đối tượng. Các lớp được trích xuất trong quy trình công việc phân tích và được thiết kế trong quy trình công việc thiết kế.

code artifact 83:

là mã tạo tác, được lập trình từ các lập trình viên riêng lẻ trong hệ thống con. Mọi chương trình đều được tạo thành từ các mã và mỗi quá trình được mã hóa tạo ra các hiện vật. Bao gồm các:

- Bộ thử nghiệm: Thử nghiệm được mã hóa để chạy trên chương trình nhằm đảm bảo một quy trình nhất định đang hoạt động
- Tập lệnh thiết lập: Cho phép chương trình chạy trên một máy nhất định
- Mã tuân thủ: Mã cuối cùng, được biên dịch, gửi đến người dùng
- Đầu ra thử nghiệm: Nhật ký và thông tin đến từ quá trình thử nghiệm

Mã hiện vật là không thể tránh khỏi, sản phẩm phụ quan trọng của quá trình mã hóa. Họ để một nhà phát triển kiểm tra bản in và phần bên ngoài của phần mềm của họ, cho phép họ hoàn thiện nó cho người dùng.

component 83 (thành phần):

là thuật ngữ mang ý nghĩa như là bộ phận, thành phần hoặc yếu tố cấu thành của một hệ thống hoặc phần mềm nào đó trong lĩnh vực IT. Trong lĩnh vực phần mềm, component có chức năng đặc biệt, được gọi ra từ các chương trình khác, được liên kết và sử dụng như một bộ phận của chương trình

Thành phần là một đối tượng phần mềm, nhằm tương tác với các thành phần khác, đóng gói chức năng nhất định hoặc một tập hợp các chức năng. Nó có một giao diện được xác định rõ ràng và tuân theo một hành vi được khuyến nghị chung cho tất cả các thành phần trong một kiến trúc.

concept exploration 79 (thăm dò ý tưởng)

là tìm hiểu về nhu cầu của khách hàng để từ đó tinh chỉnh, phân tích kỹ thuật của sản phẩm, thăm dò ý tưởng xác định các dự án có triển vọng và khả thi để phát triển, giúp đánh giá giải pháp thay thế hệ thống tốt nhất để thực hiện dựa trên chi phí và lợi ích.

Ngân hàng: (tìm hiểu về khái niệm) là khai phá những từ khóa của chuyên ngành khái niệm, tìm hiểu về nó, hiểu lĩnh vực ứng dụng của phần mềm

construction phase 92 (Pha xây dựng)

Trong giai đoạn xây dựng, ta thực hiện việc triển khai phần mềm của mình. Trong giai đoạn này, ta giảm thiểu rủi ro và loại bỏ nó. Tất cả các tính năng và thành phần được tích hợp vào một ứng dụng, thực hiện kiểm tra nghiêm ngặt và quá trình tối ưu hóa được thực hiện. Cùng lúc đó giảm thiểu chi phí phát triển và làm việc để nâng cao hiệu quả của nó. Giai đoạn xây dựng chủ yếu tập trung vào việc triển khai và thử nghiệm phần mềm.

contradiction 81 (mâu thuẫn)

Mâu thuẫn xảy ra khi có xung đột ngăn cản việc thực hiện đầy đủ các yêu cầu. Thường xuất hiện ở các tài liệu đặc tả.

- + **Thế nào là 1 bản thiết kế còn contradiction** : Bản thiết kế còn contradiction là bản thiết kế có sự mâu thuẫn dẫn đến phần mềm sau này có thể không hoạt động được ở chức năng đó. Ví dụ : Theo thiết kế của 1 hệ thống, van M17 sẽ được đóng khi áp suất vượt quá 35 psi. Tuy nhiên có một trạng thái khác được thiết kế là nếu áp suất vượt quá 35 psi thì còi sẽ báo động trong 30s sau đó van M17 mới tự động đóng. Do đó dẫn đến mâu thuẫn khiến hệ thống không chạy được chức năng đó.

core workflow 78 (quy trình cốt lõi)

là 5 quy trình cốt lõi của quy trình hợp nhất bao gồm: quy trình xác định yêu cầu, quy trình phân tích, quy trình thiết kế, quy trình triển khai phần mềm và quy trình kiểm tra

cost 79 (chi phí)

là chi phí mà khách hàng sẽ trả sau khi các thông số kỹ thuật đã được hoàn thiện. Ước tính chi phí trong kỹ thuật phần mềm thường liên quan đến chi tiêu tài chính cho nỗ lực phát triển và thử nghiệm phần mềm, điều này cũng có thể bao gồm việc xem xét các yêu cầu, bảo trì, đào tạo, quản lý và mua thêm thiết bị, máy chủ và phần mềm. Nhiều phương pháp đã được phát triển để ước tính chi phí phần mềm cho một dự án nhất định.

COTS : (Commercial-off-the-shelf Software)

Là sản phẩm phần mềm được nhà phát triển sản xuất bán với giá thành thấp, đáp ứng nhu cầu của rất nhiều người, thu lại lợi nhuận nhờ việc bán ra với số lượng lớn (vd : Microsoft..)

D

deadline 79 (thời hạn cuối cùng)

đây là thời gian mà khách hàng cần có sản phẩm phần mềm cho các hoạt động của công ty họ và bất kỳ sự chậm trễ nào trong cung cấp sản phẩm cũng sẽ ảnh hưởng cho công ty

defined level (Mức độ xác định) 96

CMM có 5 mức, trong đó defined level là mức 3, là mức có được định nghĩa.

Ở mức độ này:

- + Quy trình sản xuất phần mềm được ghi lại, có tài liệu đầy đủ.
- + Cả khía cạnh quản lý và kỹ thuật của quy trình đều được xác định rõ ràng.
- + Liên tục nỗ lực thực hiện để cải tiến quy trình và chất lượng sản phẩm nếu có thể; đồng thời cũng luôn xem xét lại sản phẩm để đảm bảo chất lượng.
- + Việc giới thiệu công nghệ mới là hợp lý để tăng chất lượng và năng suất hơn nữa. Ngược lại, “công nghệ cao” cũng làm cho việc điều khiển khủng hoảng mức 1 hỗn loạn hơn.

deliverable (tạm dịch: Sản phẩm cung cấp/Giao phẩm) 82

Là kết quả của một dự án mà khách hàng sẽ nhận được. Một deliverable có thể là báo cáo, tài liệu, một sản phẩm phần mềm hoặc bản nâng cấp,... Một deliverable có thể gồm nhiều deliverable nhỏ hơn.

Trong quy trình phân tích, khi các đặc tả đã được khách hàng chấp thuận, việc chuẩn bị kế hoạch quản lý dự án phần mềm sẽ bắt đầu. Các thành phần chính của kế hoạch là *deliverable (khách hàng sẽ nhận được gì)*, các mốc quan trọng (khi khách hàng nhận được chúng) và ngân sách (chi phí sẽ là bao nhiêu).

Bonus: So sánh deliverables (giao phẩm) - milestone (cột mốc)

- + Một số deliverables phụ thuộc vào các deliverables khác cần được hoàn thành trước; điều này là phổ biến trong các dự án có nhiều cột mốc (milestone) liên tiếp.
- + Một deliverable khác với một cột mốc dự án ở chỗ một cột mốc là thước đo tiến độ hướng tới đầu ra, trong khi deliverable là đầu ra được giao cho khách hàng
- + Đối với một dự án điển hình, một cột mốc có thể là sự hoàn thành của một thiết kế sản phẩm, trong khi deliverable có thể là sơ đồ kỹ thuật hoặc báo cáo thiết kế chi tiết của sản phẩm.

design workflow (Pha/Quy trình thiết kế) 82

Là 1 trong 5 quy trình làm việc cốt lõi, bao gồm: quy trình yêu cầu, quy trình phân tích, quy trình thiết kế, quy trình thực hiện và quy trình kiểm tra.

Mục đích của quy trình thiết kế là tinh chỉnh, mô hình hóa các tạo tác của pha phân tích cho đến khi tài liệu ở dạng mà các lập trình viên có thể thực hiện code trên ngôn ngữ lập trình.

(Theo Slide, trang 47)

- Các vấn đề xem xét:

- + Chọn ngôn ngữ lập trình
- + Tính sử dụng lại (reusability)
- + Tính thiết kế mở (open-design)
- + Tính khả chuyển (portability)

- Phương pháp:

- + Trích các lớp
- + Xác định quan hệ giữa các lớp (thiết kế kiến trúc)
- + Thiết kế các thuộc tính và phương thức (method) cho mỗi lớp (thiết kế chi tiết)

- Kết quả cần đạt được:

- + Bản mẫu các lớp, thuộc tính và phương thức
- + Thuật toán xử lý trong các phương thức để có thể cài đặt được ngay

detailed design (Thiết kế chi tiết) 82

Là bản thiết kế chi tiết các thuộc tính và phương thức cho mỗi lớp.

Trong quy trình thiết kế, đầu tiên là thiết kế kiến trúc, trong đó sản phẩm được chia thành các thành phần, được gọi là các module. Sau đó, mỗi module được thiết kế, thủ tục này được gọi là detailed design (thiết kế chi tiết). Bản thiết kế chi tiết rất quan trọng sau đó vì nó cung cấp thông tin để lập trình viên thực hiện mã.

domain (Lĩnh vực) 78

Là cách gọi tắt của application domain, tức là lĩnh vực ứng dụng.

Trong quy trình yêu cầu, nhiệm vụ đầu tiên của nhóm phát triển là có được sự hiểu biết cơ bản về lĩnh vực ứng dụng, tức là môi trường cụ thể mà sản phẩm phần mềm mục tiêu được vận hành, nắm được các khái niệm chuyên ngành trong lĩnh vực tương ứng.

Lĩnh vực có thể là ngân hàng, sản xuất ô tô hoặc vật lý hạt nhân,...

**tự lấy ví dụ : Ví dụ như nếu làm phần mềm về quán ăn thì cần hiểu các khái niệm cơ bản về lĩnh vực này, cách nó vận hành buôn bán như thế nào, thanh toán ra sao, các dịch vụ đi kèm,...*

**Hoàng Dương note: phần này dịch thô thì nghĩa là Miền, nhưng đọc hiểu và xem trong slide thì có vẻ trong phần này hiểu là một lĩnh vực nào đó.*

E

elaboration phase (Pha thiết lập) 91

là Pha thứ 2 của Unified Process.

(Theo Slide, trang 61)

- Mục tiêu:

- + Mịn hóa các kết quả sau pha inception và requirement
- + Phân tích rủi ro theo mức độ nghiêm trọng
- + Mịn hóa bản phân tích kinh doanh có trong pha inception
- + Xem xét lại SPMP

- Phương pháp: Sử dụng các kĩ thuật và phương pháp trong pha inception và requirement

I

implementation workflow (Pha/Quy trình thực hiện) 83

Là 1 trong 5 quy trình làm việc cốt lõi, bao gồm: quy trình yêu cầu, quy trình phân tích, quy trình thiết kế, quy trình thực hiện và quy trình kiểm tra.

(Theo sách, trang 83)

Mục tiêu của quy trình thực hiện là thực hiện sản phẩm phần mềm mục tiêu bằng (các) ngôn ngữ triển khai đã chọn theo kết quả pha thiết kế.

- Phương pháp: Cài đặt và Tích hợp theo Class, Module

Một sản phẩm phần mềm nhỏ đôi khi được nhà thiết kế cố vấn. Ngược lại, một sản phẩm phần mềm lớn được phân chia thành các hệ thống con nhỏ hơn, sau đó được thực hiện song song bởi các nhóm viết mã.

Khi đến lượt, các hệ thống con bao gồm các component (thành phần) hoặc các code artifact (mã tạo tác) được thực hiện bởi một lập trình viên riêng lẻ.

inception phase 89 (Pha đánh giá)

Là giai đoạn đầu tiên xác định xem liệu việc phát triển sản phẩm phần mềm mục tiêu có đáng giá hay không. Nói cách khác, mục tiêu chính của giai đoạn này là xác định xem sản phẩm phần mềm được đề xuất có khả thi về mặt kinh tế hay không.

incompleteness 81 (tính không đầy đủ)

Tính không đầy đủ là một vấn đề về các thông số kỹ thuật; nghĩa là, một số dữ kiện hoặc yêu cầu có liên quan có thể bị bỏ sót. Ví dụ, tài liệu đặc tả có thể không nêu rõ những hành động nào sẽ được thực hiện nếu dữ liệu đầu vào có lỗi.

initial level 95 (cấp độ ban đầu)

Là mức độ đầu tiên của CMM. Ở mức độ này quá trình hoàn thiện sản phẩm chưa có được sự quản lý sáng suốt hợp lý cũng như khả năng lên kế hoạch vẫn còn kém. Vì vậy, hầu hết các sản phẩm được sản xuất đều vượt quá tài chính và thời gian dự kiến của công ty. Ở giai đoạn này thì quá trình sản xuất phụ thuộc hoàn toàn vào nhân sự cty, chỉ 1 thay đổi trong nhân sự cũng dẫn đến sự thay đổi trong quá trình sx.

integration testing 86 (thử nghiệm thích hợp)

Mục đích của thử nghiệm tích hợp này là để kiểm tra xem các thành phần có kết hợp chính xác với nhau để đạt được sản phẩm đáp ứng các thông số kỹ thuật của nó hay không. Trong quá trình kiểm tra tích hợp, phải đặc biệt chú ý đến việc kiểm tra các giao diện thành phần. Điều quan trọng là số lượng, thứ tự và loại đối số chính thức phải khớp với số lượng, thứ tự và loại đối số thực tế. Việc kiểm tra kiểu mạnh này [van Wijngaarden et al., 1975] được trình biên dịch và trình liên kết thực hiện tốt nhất. Tuy nhiên, nhiều ngôn ngữ không phải kiểu mạnh. Khi một ngôn ngữ như vậy được sử dụng, các thành viên của nhóm SQA phải kiểm tra các giao diện.

International Organization for Standardization (ISO) 98

Là cơ quan thiết lập các tiêu chuẩn quốc tế, đưa ra các tiêu chuẩn thương mại và công nghiệp được áp dụng trên toàn thế giới

ISO 9000-3 98

là phần hướng dẫn được ISO phát hành để hỗ trợ trong việc áp dụng ISO 9000-3 vào phần mềm. Hướng dẫn ISO 9000-3 được sử dụng như một hướng dẫn cho việc áp dụng Tiêu chuẩn ISO 9001 vào việc phát triển, cung cấp và bảo trì phần mềm. Tiêu chuẩn và Hướng dẫn đang được sử dụng trong toàn ngành công nghiệp phần mềm như một tiêu chuẩn cho chất lượng quy trình và sản phẩm phần mềm. Chưa có một số tổ chức phần mềm đáp ứng chất lượng quy trình cần thiết để đáp ứng Tiêu chuẩn, nhưng ngành công nghiệp sẽ ngày càng tìm cách đảm bảo rằng các tổ chức phần mềm được chứng nhận theo Tiêu chuẩn.

ISO 9001 98

ISO 9001 là một tiêu chuẩn về hệ thống quản lý chất lượng, do Tổ chức Tiêu chuẩn hóa Quốc tế phát triển và ban hành được sử dụng và công nhận trên toàn thế giới. Nó xác định một tập hợp các yêu cầu quản lý chất lượng như (Bối cảnh của tổ chức, Sự lãnh đạo, Hoạch định, Hỗ trợ, Thực hiện, Đánh giá kết quả thực hiện.)

ISO/IEC 15504 99

ISO / IEC 15504 là một sáng kiến cải tiến quy trình quốc tế, giống như ISO 9000. Sáng kiến này trước đây được gọi là SPICE, một từ viết tắt được hình thành từ Software Process Improvement Capability dEtermination. Hơn 40 quốc gia đã đóng góp tích cực vào nỗ lực SPICE. SPICE do Bộ Quốc phòng Anh (MOD) khởi xướng với mục đích lâu dài là thiết lập SPICE như một tiêu chuẩn quốc tế (MOD là đối tác của Vương quốc Anh với Bộ Quốc phòng Hoa Kỳ, đơn vị khởi xướng CMM). Phiên bản đầu tiên của SPICE được hoàn thành vào năm 1995. Vào tháng 7 năm 1997, sáng kiến SPICE đã được một ủy ban chung của Tổ chức Tiêu chuẩn hóa Quốc tế và Ủy ban Kỹ thuật Điện Quốc tế tiếp quản. Vì lý do này, tên của sáng kiến đã được đổi từ SPICE thành ISO / IEC 15504, gọi tắt là 15504.

K

key process area (KPA) 98

Trong capability maturity model(CMM) mỗi cấp độ trưởng thành (ngoại trừ Cấp độ 1) bao gồm một số lĩnh vực quy trình chính. Mỗi Lĩnh vực Quy trình Chính (KPA) xác định một nhóm các hoạt động liên quan mà khi được thực hiện chung sẽ đạt được một tập hợp các mục tiêu đã xác định được coi là cần thiết để nâng cao năng lực của quy trình cho mức độ trưởng thành cụ thể đó. Các khu vực quy trình chính được tổ chức theo các đặc điểm chung. Những đặc điểm chung này đề cập đến việc thực hiện và thể chế hóa các mục tiêu đã xác định và được tạo thành từ các thực tiễn chính. Các thực hành chính mô tả các hoạt động hoặc cơ sở hạ tầng phải được thực hiện hoặc đẩy nhanh tiến độ để đạt được các mục tiêu đã xác định.

M

managed level 96

Là mức độ 4 của CMM. Ở mức độ này thì chất lượng và quy trình sản xuất được giám sát nghiêm ngặt. Cùng với đó, việc điều chỉnh sản phẩm theo kết quả thống kê luôn được thực hiện liên tục

maturity 95

thước đo mức độ hoàn thiện, phát triển của 1 quá trình hoàn thiện phần mềm

milestone 82

Milestone là "Cột mốc" quan trọng của dự án. Các mốc này là những chỉ dấu bắt đầu hoặc kết thúc một dự án hoặc đánh dấu sự hoàn thành của một giai đoạn công việc chính. Nếu một Milestone tập trung vào những điểm quan trọng trong tiến trình của dự án. Bạn sẽ thấy nó rất hữu hiệu trong tiến trình dự án. Nó có nhiệm vụ chia nhỏ một dự án lớn thành các phần có thể quản lý được và giúp quá trình quản lý dự án "dễ ăn" hơn

model 76 (Mô hình)

Mô hình là một tập hợp các sơ đồ ngôn ngữ mô hình hóa thống nhất đại diện cho một hoặc nhiều khía cạnh của sản phẩm phần mềm sẽ được phát triển. Có nghĩa là ngôn ngữ mô hình hóa thống nhất là công cụ được sử dụng để đại diện (mô hình hóa) sản phẩm phần mềm mục tiêu

module 82

Mô đun là các đơn vị nhỏ được cấu thành trong tổng thể, nhằm tạo nên một tổng thể thống nhất, liên kết và bao trọn đa chức năng.

Các nhà thiết kế phân tách sản phẩm thành các mô-đun, các đoạn mã độc lập với các giao diện được xác định rõ ràng với phần còn lại của sản phẩm. Giao diện của mỗi mô-đun (nghĩa là, các đối số được chuyển đến mô-đun và các đối số được trả về bởi mô-đun) phải được xác định chi tiết. Ví dụ, một mô-đun có thể đo mực nước trong lò phản ứng hạt nhân và gây ra âm thanh báo động nếu mức quá thấp. Một mô-đun trong sản phẩm điện tử hàng không có thể lấy hai hoặc nhiều tập hợp tọa độ của tên lửa đối phương đang lao tới, tính toán quỹ đạo của nó và gọi một mô-đun khác để thông báo cho phi công về hành động né tránh có thể xảy ra.

O

optimizing level 96 (Cấp độ tối ưu hoá)

Là mức cuối cùng của CMM. Ở mức này chất lượng sản phẩm liên tục được cải thiện, ghi nhận phản hồi và kinh nghiệm có được từ sản phẩm trước để cải tiến những sản phẩm tiếp theo

P

product testing 86 (Thử nghiệm sản phẩm)

Là Thử nghiệm đối với một sản phẩm mới để kiểm tra, chẳng hạn như sản phẩm đó có an toàn và hoạt động chính xác hay không

Ngay sau khi quá trình thử nghiệm sản phẩm hoàn tất, các phiên bản của sản phẩm hoàn chỉnh sẽ được cung cấp cho một số khách hàng có thể có trong tương lai được chọn để thử nghiệm tại chỗ. Phiên bản đầu tiên như vậy được gọi là bản phát hành alpha. Bản phát hành alpha đã hiệu chỉnh được gọi là bản phát hành beta; nói chung, bản phát hành beta được thiết kế để áp dụng cho phiên bản cuối cùng

R

regression testing 87 (Thử nghiệm hồi quy)

- Là quy trình kiểm tra sản phẩm được thử nghiệm dựa trên các trường hợp thử nghiệm trước đó để đảm bảo rằng chức năng của phần còn lại của sản phẩm không bị ảnh hưởng
- Khi một khi lập trình viên đã xác định rằng các thay đổi mong muốn đã được thực hiện, sản phẩm phải được thử nghiệm dựa trên các trường hợp thử nghiệm trước đó để đảm bảo rằng chức năng của phần còn lại của sản phẩm không bị ảnh hưởng. Thủ tục này được gọi là kiểm tra hồi quy

reliability 79 (Độ tin cậy)

- Độ tin cậy là một thuộc tính của bất kỳ thành phần nào liên quan đến máy tính (ví dụ: phần mềm hoặc phần cứng hoặc mạng) hoạt động nhất quán theo các thông số kỹ thuật của nó.
- Ví dụ: sản phẩm phải hoạt động 99% thời gian hoặc thời gian trung bình giữa các lần hỏng hóc phải ít nhất 4 tháng
-

repeatable level 96(Mức có khả năng lặp lại)

là mức thứ 2 trong 5 mức của CMN(bộ các tiêu chuẩn đánh giá chiến lược hoàn thiện tiến trình phần mềm).Ở mức độ này:

- Các quyết định quản lý dựa vào các dự án tương tự trước đó
- Có phương pháp đo các tiêu chí
- Kết quả dự án này có thể được dùng để ước lượng chi phí và thời gian cho các dự án tiếp theo
- Khi có lỗi xảy ra, việc khắc phục lỗi được thực hiện ngay

requirements workflow 78 (Quy trình yêu cầu)

Là 1 trong 5 quy trình của mô hình vòng đời lặp và tăng trưởng.

1.Mục đích:

- Xác định rõ cái khách hàng cần, chứ không phải là cái khách hàng muốn.

2.Phương pháp của quy trình:

Xác định rõ lĩnh vực ứng dụng của phần mềm:

- Làm rõ các khái niệm chuyên ngành trong lĩnh vực tương ứng

Xây dựng mô hình nghiệp vụ của khách hàng:

- Làm việc với chuyên gia nghiệp vụ
- Sử dụng công cụ UML
- Đánh giá tính khả thi

3.Kết quả cần đạt được:

Thời hạn giao sản phẩm (deadline)

- Độ tin cậy (reliability)
- Chi phí (cost)

Ngoài ra còn phải thống nhất thêm một số yêu cầu khác như: tính di động , thời gian phản hồi, khả năng chạy song song của phần mềm

retirement 88

Là giai đoạn cuối của vòng đời một phần mềm, giai đoạn này đến khi việc việc bảo trì sau giao hàng không còn đem lại hiệu quả do các lí do sau:

- Có những đề xuất thay đổi ảnh hưởng lớn tới mức khiến thiết kế tổng thể bị thay đổi.Do đó, việc thiết kế và lập trình lại sẽ tốn ít kinh phí hơn.
- Có quá nhiều thay đổi trong thiết kế gốc vô tình khiến cho các phần của sản phẩm bị phụ thuộc lẫn nhau khiến cho ngay cả thay đổi nhỏ hiện giờ có thể gây ảnh hưởng lớn đến sản phẩm.
- Tài liệu có thể không được cập nhật đầy đủ, do vậy làm tăng khả năng gây ra lỗi hồi quy tới phần mở rộng của sản phẩm do vậy việc làm lại sẽ an toàn hơn việc bảo trì.
- Phần cứng (hoặc hệ điều hành) mà sản phẩm chạy trên đó bị thay thế, do vậy sẽ tiết kiệm nếu làm lại từ đầu hơn là chỉnh sửa lại.

S

specification document 81(Tài liệu đặc tả):

Tài liệu đặc tả là một hợp đồng giữa khách hàng và nhà phát triển. Nó chỉ định chính xác những gì sản phẩm phải làm và những ràng buộc đối với sản phẩm

SPICE 99

SPICE là tên gọi trước đây của ISO/IEC 504 là một sáng kiến cải tiến quy trình quốc tế giống như ISO 9000. SPICE là viết tắt của Software Process Improvement Capability dEtermination (Xác định khả năng và cải tiến quy trình phần mềm) . Hơn 40 quốc gia đã đóng góp tích cực vào SPICE. SPICE do Bộ Quốc phòng Anh (MOD) khởi xướng với mục đích lâu dài là thiết lập SPICE như một tiêu chuẩn quốc tế (MOD là đối tác của Vương quốc Anh với Bộ Quốc phòng Hoa Kỳ, đơn vị khởi

xương CMM). Phiên bản đầu tiên của SPICE được hoàn thành vào năm 1995. Vào tháng 7 năm 1997, sáng kiến SPICE đã được một ủy ban chung của Tổ chức Tiêu chuẩn hóa Quốc tế và Ủy ban Kỹ thuật Điện Quốc tế tiếp quản. Vì lý do này, tên của sáng kiến đã được đổi từ SPICE thành ISO / IEC 15504, gọi tắt là 15504.

T

test workflow 84 (Quy trình kiểm tra)

là quy trình các chuyên gia phần mềm phải kiểm tra và thử nghiệm lại từng hiện vật mà họ phát triển hoặc duy trì. Sau đó được chuyển giao cho nhóm đảm bảo chất lượng phần mềm để kiểm tra độc lập. Đây cũ quy trình diễn ra song song với các quy trình khác từ lúc bắt đầu tới lúc kết thúc, có 2 khía cạnh chính của kiểm thử:

- Mọi nhà phát triển và bảo trì phải đảm bảo rằng công việc của họ làm là chính xác. Do đó, các chuyên gia về phần mềm phải kiểm tra hoặc kiểm tra mọi artifact mà họ phát triển hoặc bảo trì
- Một khi họ chắc chắn rằng artifact của họ đã hoạt động tốt, nó sẽ được chuyển tới nhóm đảm bảo chất lượng phần mềm để kiểm thử độc lập.

Mục đích chính của quy trình kiểm thử là đảm bảo tất cả các artifact hoạt động tốt và không gây ra ảnh hưởng nào nghiêm trọng tới các artifact liên quan đến nó.

traceability 84 (truy xuất nguồn gốc):

Quá trình xem xét tất cả các test cases được xác định cho bất kỳ yêu cầu nào được gọi là Truy xuất nguồn gốc. Là đặt tính phải có để các tạo tác yêu cầu có thể kiểm tra được trong vòng đời của sản phẩm phần mềm. Truy xuất nguồn gốc cho phép xác định các yêu cầu nào sinh ra nhiều lỗi nhất trong quá trình kiểm thử.

transition phase 92 (Pha chuyển tiếp):

Mục tiêu: - Đảm bảo tất cả các yêu cầu của khách hàng đã được thực hiện một cách đúng đắn

- Các lỗi đã được sửa
- Các tài liệu hướng dẫn sử dụng đã hoàn chỉnh

U

Unified Modeling Language (Ngôn ngữ mô hình hóa thống nhất) (UML) 76:

là một ngôn ngữ mô hình gồm các ký hiệu đồ họa mà các phương pháp hướng đối tượng sử dụng để thiết kế các hệ thống thông tin một cách nhanh chóng.

Cách xây dựng các mô hình trong UML phù hợp mô tả các hệ thống thông tin cả về cấu trúc cũng như hoạt động. Cách tiếp cận theo mô hình của UML giúp ích rất nhiều cho những người thiết kế

và thực hiện hệ thống thông tin cũng như những người sử dụng nó; tạo nên một cái nhìn bao quát và đầy đủ về hệ thống thông tin dự định xây dựng. Cách nhìn bao quát này giúp nắm bắt trọn vẹn các yêu cầu của người dùng; phục vụ từ giai đoạn phân tích đến việc thiết kế, thẩm định và kiểm tra sản phẩm ứng dụng công nghệ thông tin. Các mô hình hướng đối tượng được lập cũng là cơ sở cho việc ứng dụng các chương trình tự động sinh mã trong các ngôn ngữ lập trình hướng đối tượng, chẳng hạn như ngôn ngữ C++, Java,... Phương pháp mô hình này rất hữu dụng trong [lập trình hướng đối tượng](#). Các mô hình được sử dụng bao gồm Mô hình đối tượng (mô hình tĩnh) và Mô hình động.

Unified Process (tiền trình hợp nhất) 76

- Gồm 4 pha:
 - Đánh giá
 - Thiết lập
 - Xây dựng
 - Chuyển tiếp

Mỗi bước thực hiện tương ứng với:

- 1 trong 5 workflows
- 1 trong 4 pha

Quan hệ:

- Workflow tương ứng với cách nhìn kỹ thuật
- Pha tương ứng cách nhìn nghiệp vụ

Tại sao mỗi bước phải có hai cách nhìn khác nhau?

Phân tích kinh doanh:

- Giá phát triển có mang tính kinh tế?
- Bao lâu sẽ quay vòng vốn?
- Nếu từ bỏ dự án thì chi phí hết bao nhiêu?
- Nếu sản phẩm dạng COTS, có cần có chiến dịch tiếp thị sản phẩm?
- Sản phẩm có thể giao đúng hạn không?
- Thiệt hại gì nếu giao sản phẩm cho khách hàng trễ hạn

Phân tích rủi ro:

- Liệu team có đủ kinh nghiệm cần thiết?
- Có cần phần cứng mới cho sản phẩm?
- Nếu có, thì thiệt hại gì nếu người ta giao phần cứng trễ hạn?

- Trong trường hợp đó, có nên đặt hàng một nhà cung cấp phần cứng khác để dự phòng không?
- Có cần công cụ hỗ trợ nào không?
- Nếu có, liệu chúng có sẵn hay không, hay có cần toàn bộ chức năng của nó hay không?

unit testing 85 (kiểm thử đơn vị):

là phương pháp được thực hiện khi mỗi thành phần được kiểm tra trong khi nó đang được triển khai (kiểm tra tại bàn); và sau khi nó đã được triển khai, nó sẽ được chạy dựa trên các trường hợp thử nghiệm. Việc kiểm tra không chính thức này được thực hiện bởi lập trình viên. Sau đó, nhóm đảm bảo chất lượng kiểm tra thành phần một cách phương pháp. Kiểm thử đơn vị được thực hiện trong quá trình phát triển ứng dụng. Mục tiêu của Kiểm thử đơn vị là cô lập một phần code và xác minh tính chính xác của đơn vị đó.

W

workflow là gì? Khác gì với process?

Định nghĩa đơn giản nhất của workflow: **là các định nghĩa của các qui trình đã chuẩn hóa.** Và khi mình viết các module cho từng công việc, workflow là 1 chuỗi công việc phải làm.

Là 1 cách thực hiện cụ thể của process

Quá trình (process)

- + bao gồm nhiều quy trình trong việc tạo ra 1 phần mềm
- + có thể có nhiều mô hình khác nhau trong 1 process
- + Process có thể sử dụng mô hình workflow để làm ra phần mềm

Việc trích lớp và xây dựng các lớp là của pha thiết kế, tại sao người ta làm nó ngay trong pha phân tích?

Là do bước đầu tiên trong quá trình phân tích là xác định các lớp. Bởi vì mỗi lớp là một kiểu của mô-đun, việc phân tích mô-đun được thực hiện trong suốt quá trình phân tích.

Tại sao việc làm tài liệu đặc tả lại quan trọng đến vậy

SRS là tài liệu đặc tả vô cùng quan trọng trong quá trình phát triển phần mềm, nó có vai trò:

- Giúp cho các bên thứ ba - stakeholders đều hiểu được hệ thống theo cùng một hướng, tránh trường hợp mỗi người một ý.
- Giúp cho đội phát triển xây dựng hệ thống một cách chính xác, đặc tả được các tính năng, không đi lạc hướng so với yêu cầu của khách hàng.
- SRS giúp nhà kiểm thử hệ thống đọc hiểu từ đó xây dựng nên kịch bản kiểm thử chi tiết nhất.
- Giúp cho việc bảo trì hệ thống và cải tiến những chức năng của hệ thống một cách nhanh chóng và dễ dàng.