**I TỔNG QUAN VỀ ĐỘ ĐO**

1. ***Khái niệm độ đo***

***Khái niệm độ đo***: là chỉ số đăc trưng cho một khía cạnh nào đó.Trong công nghệ phần mềm có độ đo của phần mềm(software metric/ software measure).độ đo của dự án (project metric) và độ đo của quy trình phần mềm(process metric).

Độ đo chia ra làm hai loại:độ đo trực tiếp và độ đo gián tiếp

* Độ đo trực tiếp là độ đo có thể tính đếm trực tiếp thông qua các độ đo khác (VD:độ đo LOC-lines of code),
* Độ đo gián tiếp là các độ đo tín qua các độ đo khác

Dự án cũng có độ đo,chi phí cho dự án,năng suất của dự án.Quy trình phần mềm cũng có độ đo,chẳng hạn tỉ lệ chi phí trung bình cho mỗi giai đoạn phát triền phần mềm đối với quy trình thác nước.

1. ***Mục tiêu đo lường phần mềm***

Mục tiêu chính của đo lường chất lượng phần mềm

* Để thuận tiện cho việc điều khiển quản lý cũng như lập kế hoạch và thực thi của sự can thiệp quản lý thích hợp. Đạt được mục đích này dựa trên sự tính toán của độ đo đối với:
* Độ chênh lệch giữa sự thực thi chức năng (chất lượng) thực tế từ sự thực thi đã lập kế hoạch.
* Đô chênh lệch của thực hiện ngân sách và thời gian biểu thực tế từ sự thực thi đã lập kế hoạch.
* Để xác định trạng thái (situation) yêu cầu hoặc cho phép cải tiến quy trình phát triển hay bảo trì dưới dạng các hoạt động ngăn ngừa và sửa đổi được đưa ra trong suốt tổ chức. Đạt mục tiêu này dựa trên: Tích lũy thông tin về độ đo đối với sự thực hiện của cả nhóm, đơn vị,..

Việc so sánh sẽ cung cấp thực tế cơ bản cho ứng dụng quản lý của độ đo và sự tiến bộ của SQA nói chung. Độ đo được sử dụng cho việc so sánh dữ liệu thực thi với chỉ dẫn (indicator), giá trị định lượng như sau:

* Xác định các chuẩn chất lượng phần mềm.
* Tập hợp mục tiêu chất lượng cho tổ chức và cá nhân.
* Thành tựu chất lượng trong năm trước.
* Thành tựu chất lượng của dự án trước.
* Mức độ chất lượng trung bình đạt được bởi các nhóm khác áp dụng cùng công cụ phát triển trong môi trường phát triền giống nhau.
* Thành tựu chất lượng trung bình của tổ chức.

Thực tiễn công nghiệp cho sự thỏa mãn yêu cầu chất lượng.

1. ***Các tiêu chí đánh giá chất lượng một số loại phần mềm***

Phần mềm có thể phân thành nhiều loại theo nhiều tiêu chí khác nhau tùy theo mục đích, tuỳ theo lĩnh vực ứng dụng: phần mềm kinh doanh, phần mềm giáo dục, phần mềm giải trí, phần mềm tài chính kế toán, phần mềm an ninh và bảo mật … Mỗi loại phần mềm đƣợc xây dựng các tiêu chí đánh giá riêng biệt.

*Các tiêu chí đánh giá phần mềm giáo dục*

* Cài đặt và thiết lập cấu hình: Hướng dẫn rõ ràng, chính xác cho việc cài đặt cũng như việc thiết lập cấu hình. Thời gian cài đặt phần mềm hợp lý, chấp nhận được. Nơi cài đặt đã có đầy đủ phần cứng và phần mềm theo yêu cầu.
* Tiện lợi và thiên thiện với người dùng: Tài liệu hướng dẫn sử dụng rõ ràng, chính xác. Yêu cầu về kỹ năng Công nghệ thông tin phù hợp với người sử dụng. Nội dung không có lỗi. Giao diện phần mềm thiết kế thân thiện, dễ sử dụng. Sử dụng văn phạm và chính tả chính xác.
* Phù hợp, hữu ích và linh hoạt: Phần mềm đáp ứng đƣợc mục tiêu giáo dục. Nội dung và nhiệm vụ phù hợp với học sinh. Phương pháp luận của phần mềm phù hợp với phương pháp giảng dạy của giáo viên, tình hình thực tế. Nội dung của phần mềm được tổ chức chặt chẽ. Phần mềm xây dựng dựa trên nội dung kiến thức. Nội dung luôn được cập nhật mới, có mục đích rõ ràng, chính xác.
* Thi hành: Phần mềm chạy không bị lỗi. Hình ảnh, âm thanh, video hiển thị nhanh, rõ nét, chính xác trong khoảng thời gian hợp lý. Các liên kết, nút lệnh, đường dẫn di chuyển chính xác. Phần mềm có các lựa chọn in ấn, sao chép, nhập, xuất dữ liệu … Phần mềm hỗ trợ font tiếng Việt và các ký tự đặc biệt.
* Phản hồi cho học sinh: Học sinh có thể lựa chọn hoặc bỏ qua một số phần trong chương trình. Học sinh có thể xem phần trợ giúp hoặc hướng dẫn. Học sinh nhận được phản hồi cho câu hỏi của mình, cũng như xem các kiến thức liên quan.
* Lợi ích chung: Phần mềm thúc đẩy việc học tập của học sinh. Phần mềm duy trì sự quan tâm tới việc tìm hiểu kiến thức của người sử dụng. Phần mềm tiết kiệm thời gian trong giờ học. Phần mềm cho phép theo dõi sự tiến bộ của học sinh, ngƣời sử dụng và là nơi lưu trữ thông tin.
* Hỗ trợ kỹ thuật: Đƣợc cung cấp các tài liệu hỗ trợ. Sẵn sàng đáp ứng các hỗ trợ về mặt kỹ thuật. Công cụ hỗ trợ toàn diện có sẵn. Có thể truy cập vào phần trợ giúp khi đang sử dụng chương trình.
* Chi phí: So với các chương trình tƣơng tự chi phí bản quyền hợp lý. Phần mềm đƣợc nâng cấp và hỗ trợ miễn phí.

*Tiêu chí đánh giá phần mềm an ninh bảo mật*

* Dễ cài đặt và sử dụng: Chương trình phần mềm phải dễ và cài đặt nhanh chóng. Dễ hiểu, dễ sử dụng cho người dùng dù người đó có là chuyên gia về máy tính hay không. Khi thay đổi các thiết lập chương trình không làm khó khăn người sử dụng. Chương trình phần mềm chỉ cần cài đặt một lần trên hệ thống sau đó nó tự làm công việc của mình (phần mềm diệt virus).
* Phát hiện và cảnh báo nguy hiểm: Phần mềm tốt, hiệu quả luôn phát hiện, cảnh báo và ngăn chặn nguy hiểm, sự lây nhiễm một cách nhanh chóng trong hệ thống, hay trong các ứng dụng … Phần mềm an ninh, bảo mật được coi là đáng tin cậy thông qua việc nó phát hiện và ngăn chặn các file nhiễm virus. Tốc độ thực hiện cũng rất quan trọng vì nó có thể can thiệp và làm chậm những ứng dụng đang được thực thi trên máy.
* Cô lập và tiêu diệt những file bị lây nhiễm một cách hiệu quả: Việc xác định các file nguy hiểm cũng như bị lây nhiễm nhanh chóng thông qua việc xóa hay cách ly chúng một cách hợp lý - điều đó ngăn chặn nguy hiểm và sự lây lan trong hệ thống hoặc trong mạng.
* Nhiều tính năng: Điều này làm khả năng an ninh, bảo mật của chương trình phần mềm hoàn thiện hơn. Phần mềm tốt khả năng xử lý, bảo vệ nâng cao … nhiều máy tính được bảo vệ an toàn hơn.
* Báo cáo: Chương trình đưa ra thông báo ngay khi phát hiện tác nhân gây hại cũng như những lựa chọn xử lý hợp lý cho người sử dụng.

**II ĐỘ ĐO CHẤT LƯỢNG PHẦN MỀM**

1. ***Khái niệm độ đo phần mềm***

Độ đo phần mềm là kỹ thuật cơ sở phương pháp ước lượng. Độ đo phần mềm là một kiểu độ đo liên quan đến hệ thống phần mềm. Xây dựng hệ thống độ đo phần mềm có 2 hướng, thứ nhất, thu thập dữ liệu, phân tích và tìm sự tương quan trong kho dữ liệu đó. Thứ hai, tìm các độ đo, tổng hợp và thực hiện báo cáo trên độ đo đã có. Có hai lớp độ đo là độ đo thuộc tính và độ đo dự đoán (liên quan tới quá trình sản xuất sản phẩm và thuộc tính của sản phẩm).

Hai cách khác nhưng được định nghĩa bổ sung(IEEE, 1990) miêu tả độ đo chất lượng phần mềm như một phạm trù của công cụ SQA.

(1) Một sự đo lường định lượng mức độ để một khoản mục có một thuộc tính chất lượng đã quy định.

(2) Đầu vào của một chức năng là dữ liệu phần mềm và đầu ra của nó là một giá trị số duy nhất có thể được giải thích như mức độ để các phần mềm có một thuộc tính chất lượng đã quy định.

Tức là, định nghĩa thứ hai đề cập đến quá trình đưa ra các độ đo chất lượng trong khi định nghĩa đầu tiên đề cập đến kết quả của quá trình nói trên.

Thông thường người ta tin rằng độ đo chất lượng phần mềm nên chứa trong phần mềm, như trong các ngành công nghiệp khác, giữa các công cụ cơ bản được dùng để giúp đỡ sự quản lý trong ba lĩnh vực cơ bản sau: *điều khiển phát triển và bảo trì phần mềm, hỗ trợ đưa ra quyết định, khởi tạo các hoạt động hiệu chỉnh.* Phân tích thống kê các dữ liệu độ đo được mong đợi để xác định (sinh động và có ý nghĩa thống kê) những thay đổi đã được khởi tạo như một kết quả của ứng dụng của công cụ phát triển mới, thủ tục thay đổi và những sự can thiệp khác.

Phạm vị của độ đo chất lượng phần mềm đã được mở rộng đáng kể qua một vài thập niên trước đây. Chúng ta sẽ xem xét lại một số việc bảo trì và phát triền phần mềm thích hợp nhất đưa ra ở trọng tâm của chương này. Độ đo như là một công cụ đảm bảo chất lượng phần mềm, không may, không được ứng dụng ở mức độ đầy đủ trong công nghiệp phần mềm, cũng không được cung cấp lợi ích ở mức độ mong đợi. Chỉ một phần nhỏ của tổ chức phát triển phần mềm áp ụng độ đo chất lượng phần mềm một cách có hệ thống.

1. ***Các loại độ đo phần mềm***

Có 2 loại độ đo:

* Độ đo trực tiếp: là độ đo có thể tính trực tiếp không thông qua các độ đo khác (LOC- lines of code)
* Độ đo gián tiếp: là các độ đo tính thông qua các độ đo khác (ví dụ tỉ lệ lỗi = số lỗi / số dòng mã nguồn.

Đo kích thước phần mềm chỉ làm 2 loại:Đo qua dòng lệnh(LOC-lines of code) và qua điểm chức năng (FT-Functionnal Points)

* *Qua dòng lệnh*(LOC-lines of code): **Dòng nguồn mã** **(SLOC),** còn được gọi là **dòng mã** **(LOC),** là một độ đo phần mềm dùng để đo kích thước của một [chương trình máy tính](https://translate.googleusercontent.com/translate_c?depth=1&hl=vi&prev=search&rurl=translate.google.com.vn&sl=en&u=https://en.wikipedia.org/wiki/Computer_program&usg=ALkJrhiEIt8PQcDmS3zdDwmqdsMftb7E6g) bằng cách đếm số dòng trong văn bản của các chương trình mã nguồn . SLOC thường được sử dụng để dự đoán những nỗ lực đó sẽ được yêu cầu để phát triển một chương trình, cũng như để ước tính naăng suất lập trình hoặc baảo trì khi phần mềm được sản xuất.

Đo kích thước phần mềm chỉ làm 2 loại:Đo qua dòng lệnh(LOC-lines of code) và qua điểm chức năng (FT-Functionnal Points)

* *Qua dòng lệnh*(LOC-lines of code): **Dòng nguồn mã** **(SLOC),** còn được gọi là **dòng mã** **(LOC),** là một độ đo phần mềm dùng để đo kích thước của một [chương trình máy tính](https://translate.googleusercontent.com/translate_c?depth=1&hl=vi&prev=search&rurl=translate.google.com.vn&sl=en&u=https://en.wikipedia.org/wiki/Computer_program&usg=ALkJrhiEIt8PQcDmS3zdDwmqdsMftb7E6g) bằng cách đếm số dòng trong văn bản của các chương trình mã nguồn, độ đo này chỉ có thể chính xác sau khi dự án đã kết thúc. Tuy nhiên bằng kinh nghiệm hoặc bằng thống kê tương tự có thể ước lượng đuoạc khối lượng mã nguồn của một phần mềm trước khi kết thức dự án.

SLOC thường được sử dụng để dự đoán những nỗ lực đó sẽ được yêu cầu để phát triển một chương trình, cũng như để ước tính năng suất lập trình hoặc bảo trì khi phần mềm được sản xuất.

* *Qua điểm chức năng* (FT-Functionnal Points) Tổng hợp các đặc trưng của module
  + Input
  + Output
  + Interface
  + Files

Đặt trọng số cho các đặc trưng

* Trọng số phụ thuộc vào ngữ cảnh (dự án) cụ thể

− độ phức tạp của bài toán

− Các yêu cầu về chất lượng, hiệu năng

− Kích thước của dữ liệu sử dụng

Điểm chức năng FP

* FP = a1I + a2O + a3E + a4L + a5F

− I : số Input

− O: số Output

− E: số yêu cầu

− L: số tệp truy cập

− F: số giao diện ngoại lai (devices, systems)

VD: FP = 4I + 5O + 4E + 10L + 7F

* Hàm: Tính ước số chung lớn nhất của hai số nguyên

− Input I = 2

− Output O = 1

− Yêu cầu E = 1 -> Điểm chức năng FP = 17

Độ đo về chất lượng dựa trên thống kê

* Độ tin cậy: MTBF – Mean Time Between Failures
  + thời gian chạy liên tục không có lỗi
* Thời gian khôi phục hệ thống
* MTTR
* Mean Time To Recover
* Tính sẵn có:

1. ***Các độ đo tiến trình***
   1. ***Độ đo chất lượng quy trình phần mềm***

*Độ đo chất lượng quy trình phần mềm bao gồm:*

* Độ đo mật độ lỗi:

|  |  |  |
| --- | --- | --- |
| Code | Tên | Công thức tính |
| CED | Mật độ lỗi code | CED = |
| DED | Mật độ lỗi phát triển | DED = |
| WCED | Mật độ lỗi code đã đánh | WCED = |
| WDED | Mật độ lỗi phát triển đã đánh trọng số | WDED = |
| WCEF | Lỗi code đã đánh trọng số cho mỗi điểm chức năng | WCEF = |
| WDEF | Lỗi phát triển đánh trọng số mỗi điểm chức năng | WDED = |

Từ khóa:

NCE= số lỗi code được phát hiện trong code phần mềm bởi quá trình test và duyệt. Dữ liệu cho phép đo này được chọn lọc từ các báo cáo test và duyệt code.

KLOC= hàng nghìn dòng code

NDE= tổng số lỗi phát triển (thiết kế và code) được phát hiện trong quy trình phát triển phần mềm. Dữ liệu cho phép đo này được tìm ra trong rất nhiều báo cáo test rà soát code và thiết kế.

WCE= lỗi code đánh trọng số được phát hiện. Nguồn dữ liệu cho độ đo này là tương tự như nguồn của NCE.

WDE= tổng lỗi phát triển (thiết kế vào code) đánh trọng số được phát hiện trong phát triển của phần mềm. Các nguồn dữ liệu cho độ đo này giống với NDE.

NFP: số điểm chức năng được yêu cầu cho phát triển phần mềm. Nguồn gốc số điểm chức năng là sự khảo sát chuyên nghiệp của phần mềm liên quan.

* Độ đo tính nghiêm trọng của lỗi

|  |  |  |
| --- | --- | --- |
| Code | Tên | Công thức tính |
| ASCE | Trung bình nghiêm trọng của lỗi code | ASCE = |
| ASDE | Trung bình nghiêm trọng của lỗi phát triển | ASDE = |

* 1. ***Độ đo thời gian biểu của quy trình phần mềm***

|  |  |  |
| --- | --- | --- |
| Code | Tên | Công thức tính |
| TTO | Tuân theo thời gian biểu | TTO = |
| ADMC | Trung bình trễ hoàn thành mốc quan trọng | ADMC = |

**Bảng Các độ đo thời gian biểu tiến trình phần mềm**

Từ khóa:

MSOT= Những mốc quan trọng(milestones) đã hoàn thành đúng thời gian

MS= tổng số mốc quan trọng

TCDAM= tổng các trễ hoàn thành (hàng ngày, hàng tuần…) cho tất cả mốc quan trọng.

Để tính toán phép đo này, các trễ đã báo cáo cho tất cả các mốc quan trọng liên quan được tổng kết lại. Các mốc quan trọng hoàn thành đúng thời gian hoặc trước lịch biểu được xem là trễ ‘O’. Một số chuyên gia xem các mốc quan trọng hoàn thành trước lịch biểu như các trễ “âm”. Chúng được xét để cân bằng hiệu quả (hiệu lực) của các trễ “dương”. Trong trường hợp này, giá trị của ADMC có thể thấp hơn giá trị đạt được theo độ đo đề nghị ban đầu.

***3.3 Độ đo tính hiệu quả loại bỏ lỗi***

|  |  |  |
| --- | --- | --- |
| Code | Tên | Công thức tính |
| DERE | Hiệu quả loại bỏ lỗi phát triển | DERE = |
| DWERE | Hiệu quả loại bỏ lỗi phát triển có đánh trọng số | DWERE = |

Từ khóa:

NYF: số thất bại(failures) phần mềm được phát hiện trong suốt 1 năm bảo trì dịch vụ.

WYF: số thất bại phần mềm đã đánh trọng số trong suốt 1 năm bảo trì dịch vụ.

***3.4 Độ đo hiệu suất quy trình phần mềm***

*4 độ đo năng suất tiến trình, trực tiếp và gián tiếp*

|  |  |  |
| --- | --- | --- |
| Code | Tên | Công thức tính |
| DevP | năng suất phát triển | DevP = DevH/KLOC |
| FDevP | năng suất phát triển điểm chức năng | FdevP = DevH/KLOC |
| Cre | tái sử dụng mã | Cre = ReKLOC/KLOC |
| DocRe | tái sử dụng tài liệu | DocRe = ReDoc/NDoc |

Trong đó:

DevH= tổng thời gian làm việc dành cho phát triển của hệ thống phần mềm.

ReKLOC = số dòng lệnh sử dụng lại trong hàng nghìn dòng code

ReDoc = số trang sử dụng lại của tài liệu

NDoc= số trang của tài liệu

1. ***Các độ đo sản phẩm***

Độ đo sản phẩm (product metrics) bao gồm:

- Độ đo chất lượng HD (Help Desk)

- Độ đo năng suất và hiệu quả HD

- Độ đo chất lượng bảo trì sửa chữa

- Độ đo năng suất và hiệu quả bảo trì sửa chữa phần mềm

***Độ đo chất lương HD***

Các kiểu của độ đo chất lượng HD được thảo luận ở đây đề cập đến:

* Độ đo mật độ cuộc gọi HD- phạm vi của yêu cầu khách hàng cho dịch vụ HD được đo bởi số cuộc gọi.
* Độ đo tính nghiêm trọng của HD được tăng lên.
* Độ đo thành công HD - Mức độ của sự thành công trong sự phản ứng lại tới những cuộc gọi này. Thành công đạt được bằng sự hoàn thành dịch vụ yêu cầu trong thời gian nhất định của hợp đồng dịch vụ.

***Độ đo năng suất và hiệu quả HD***

Độ đo năng suất liên quan tới tổng tài nguyên đầu tư trong suốt một giai đoạn cụ thể, trong khi độ đo hiệu quả liên quan tới tài nguyên đầu tư trong đáp ứng lại tới 1 cuộc gọi HD.

***Độ đo năng suất HD*** Độ đo năng suất HD làm cho việc sử dụng của phép đo KLMC dễ dàng áp dụng cho kích thước hệ thống phần mềm bảo trì hoặc theo ước lượng điểm chức năng của hệ thống phần mềm. Hai độ đo năng suất HD được miêu tả :

|  |  |  |
| --- | --- | --- |
| Code | Tên | Công thức tính |
| HDP | Năng suất HD | HDP = HDYH/KLMC |
| FHDP | Năng suất điểm chức năng HD | FHDP = HDYH/NMFP |

Trong đó:

* HDYH = Tổng thời gian làm việc hàng năm đầu tư trong dịch vụ HD của hệ thống phần mềm.
* KLMC và NMFP được địng nghĩa ở phần trước.

***Độ đo hiệu quả***

HD Độ đo trong nhóm này liên quan tới tài nguyên được đầu tư trong đáp ứng lại cuộc gọi HD của khách hàng. Một độ đo thông dụng đựơc miêu tả ở đây, hiệu quả HD (HDE):

HDE = HDYH/NHYC

Độ đo bảo trì sửa chữa phần mềm xử lý hàng loạt khía cạnh chất lượng của dịch vụ bảo trì. Một sự khác biệt cần thiết giữa thất bại hệ thống phần mềm được giải quyết bởi nhóm bảo trì và thất bại do dịch vụ bảo trì liên quan tới trường hợp bảo trì không cung cấp một sự sửa chữa thõa mãn những tiêu chuẩn hoặc yêu cầu hợp đồng. Do đó, độ đo bảo trì phần mềm được phân loại như sau:

* Độ đo mật độ thất bại hệ thống phần mềm: Giải quyết phạm vi của yêu cầu tới bảo trì sửa chữa, dựa trên bản ghi của các thất bại được xác định trong hoạt động bình thường của hệ thống phần mềm.
* Độ đo nghiêm trọng thất bại hệ thống phần mềm: Giải quyết sự nghiêm trọng của thất bại hệ thống phần mềm quan tâm bởi nhóm bảo trì sửa chữa.
* Độ đo thất bại của dịch vụ bảo trì: Giải quyết trường hợp mà dịch vụ bảo trì không thể hoàn thành sửa chữa thất bại đúng thời gian hoặc việc sữa lỗi thất bại.

Độ đo sự sẵn sàng hệ thống phần mềm: Giải quyết phạm vi của sự xáo trộn được gây ra tới khách hàng như thực hiện bởi một giai đoạn thời gian nơi dịch vụ của hệ thống phần mềm không sẵn sàng hoặc chỉ sẵn sàng một phần

1. ***Thực hiện đo chất lượng phần mềm***

Để thực hiện đo chất lượng phần mềm cần:

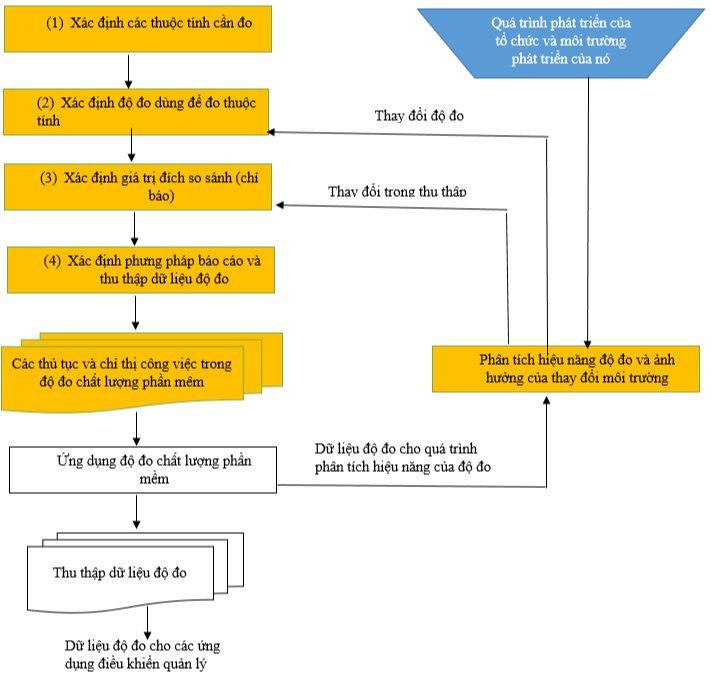
- Xác định các độ đo chất lượng phần mềm

- Áp dụng thông thường bởi đơn vị…

- Phân tích thống kê dữ liệu độ đo đã thu thập được.

- Các hành động tiếp theo bao gồm:

* Các thay đổi trong tổ chức và các phương pháp của các đơn vị bảo trì và phát triển phần mềm và/hoặc bất kỳ cá nhân nào thu thập dữ liệu độ đo.
* Thay đổi về độ đo và thu thập dữ liệu độ đo.
* Áp dụng dữ liệu và phân tích dữ liệu nhằm lên kế hoạch cho các hoạt động sửa lỗi cho tất cả các đơn vị liên quan.



1. ***Giới hạn và các vấn đề trong áp dụng tính giá các độ đo chất lượng phần mềm***

*Những giới hạn của các độ đo phần mềm*

* Ứng dụng của độ đo chất lượng có nhiều cản trở. Những cản trở này có thể nhóm theo sau:
* Ngân sách ràng bụôc trong chỉ định tài nguyên cần thiết (nhân sự, tài chính, vv..) cho phát triển của hệ thống độ đo chất lượng và ứng dụng bình thường của nó. Yếu tố con người, đặc biệt là sự chống lại của người lao động để ước lượng những hoạt động này.
* Không chắc chắn về giá trị của dữ liệu, đã ăn sâu vào các bản ghi thiên vị và thành kiến.

*Các vấn đề trong áp dụng tính giá các độ đo chất lượng phần mềm*

Ứng dụng của mô hình chi phí chất lượng phần mềm thường đi cùng với các vấn đề cần phải giải quyết ở bất cứ ngành công nghiệp nào. Những vấn đề này tác động lên độ chính xác và đầy đủ của dữ liệu chi phí chất lượng do:

* Sự thiếu chính xác và thiếu đầy đủ của việc xác định và phân loại các chi phí chất lượng.
* Các báo cáo cẩu thả của các thành viên nhóm và người ngoài nhóm. o Việc báo cáo lệch lạc các chi phí phần mềm.
* Các bản ghi lệch lạc về các chi phí do thất bại bên ngoài, xuất phát gián tiếp từ các khoản đền bù được ngụy trang cho khách hàng (ví dụ: hạ giá các dịch vụ tương lại, cung cấp các dịch vụ miễn phí…). Chi phí cho các thất bại cần đền bù đó không được ghi nhận như là chi phí do thất bại bên ngoài.

Những vấn đề trên đây nảy sinh trong ngữ cảnh công nghiệp phần mềm nhưng cũng có những vấn đề khác nữa. Một số trong chúng chỉ có trong sản xuất phần mềm mà thôi. Chúng ta sẽ tập trung vào các vấn đề gặp phải khi ghi nhận các chi phí chuẩn bị và điều khiển công việc quản lí cùng các chi phí do thất bại trong quản lí bởi vì những khoản mục này ảnh hưởng tới chi phí và tính dễ hiểu của tổng chi phí cho đảm bảo chất lượng phần mềm, đặc biệt là khi áp dụng mô hình chi phí mở rộng.

Các vấn đề nảy sinh khi thu thập dữ liệu trong chuẩn bị và điều khiển việc quản lí chi phí bao gồm:

Xác định trách nhiệm khi có thất bại trong thực hiện lịch biểu. Những chi phí này có thể gán cho khách hàng (trong trường hợp khác hàng được yêu cầu bồi thường cho nhà thầu), nhóm phát triển (được xem như chi phí do thất bại bên ngoài) hay ban quản lí (được xem như chi phí cho thất bại trong quản lí). Chi phí cho thất bại lịch biểu thường được cân nhắc cho một giai đoạn đủ dài vì các nguyên nhân trực tiếp hay các đóng góp của từng thành phần trong dự án dẫn tới thất bại khó được xác định.

**III PHƯƠNG PHÁP ĐÁNH GIÁ PHẦN MỀM**

1. ***Các giai đoạn tiến hành đánh giá phần mềm***

Trong thực tế thường dùng hai cách để đánh giá phần mềm, đó là đánh giá phần mềm trong quá trình phát triển phần mềm, và đánh giá phần mềm trong quá trình sử dụng. Đánh giá phần mềm trong quá trình phát triển tương đương với quá trình chạy thử phần mềm trước khi đưa ra ngoài thị trường bằng cách lấy ý kiến của các lập trình viên, các đối tượng sử dụng thử phần mềm đó. Còn đánh giá phần mềm trong quá trình sử dụng là khi đã đưa phần mềm tới người sử dụng, có thể thu thập số liệu từ những người trực tiếp sử dụng phần mềm đó. Phần mềm có thể phân thành nhiều loại theo nhiều tiêu chí khác nhau tùy theo mục đích.

Dựa vào phương thức hoạt động có thể chia phần mềm thành ba loại chính:

* Phần mềm hệ thống là chƣơng trình tổ chức việc quản lý, điều phối các bộ phận, chức năng của máy tính, dùng để vận hành máy tính, và các phần cứng máy tính, điều khiển máy tính và tạo môi trường hoạt động cho các phần mềm khác. Ví dụ như hệ điều hành máy tính Windows XP, Linux, Unix, các thư viện động (còn gọi là thư viện liên kết động – dynamic linked library – DLL) của hệ điều hành, các trình điều khiển (driver), phần sụn (firmware) và BIOS. Đây là các loại phần mềm mà hệ điều hành liên lạc với chúng để điều khiển và quản lý các thiết bị phần cứng.
* Phần mềm ứng dụng các chương trình yêu cầu những ứng dụng cụ thể, để người sử dụng có thể hoàn thành một hay nhiều công việc. Phần mềm ứng dụng được chia thành nhiều loại khác nhau:

- Phần mềm văn phòng (Offices): Microsoft Office, Vietkey, Unikey, Adobe Reader, Solid Converter …

- Phần mềm đa phương tiện (Multimedia): Window Media, KMPlayer, Gom, các chương trình convert …

- Phần mềm đồ họa (Graphics): Photoshop, Corel, AutoCad …

- Phần mềm tiện ích Internet: Internet Explorer, FireFox, Yahoo Messenger, Skype, các chương trình hỗ trợ dowload, tăng tốc truy cập internet …

- Phần mềm bảo mật (Security): các chương trình về Antivirus, Firewall, Deepfreez …

- Phần mềm trò chơi (Game): tất cả các games.

- Phần mềm giáo dục: Violet …

- Phần mềm cơ sở dữ liệu: SQL server, Access, Foxpro …

* Phần mềm chuyển dịch mã bao gồm trình biên dịch và trình thông dịch. Loại chương trình này sẽ đọc các câu lệnh từ mã nguồn được viết bởi các lập trình viên bằng một ngôn ngữ lập trình và dịch nó sang dạng ngôn ngữ máy hay dịch nó sang một dạng khác như là tập tin đối tượng (object file) và các tập tin thư viện (library file) mà các phần mềm khác (như hệ điều hành chẳng hạn) có thể hiểu để vận hành máy tính thực thi các lệnh.

Dựa vào khả năng ứng dụng có thể chia phần mềm thành hai loại chính:

* Những phần mềm không phụ thuộc, bất kỳ khách hàng nào cũng có thể mua để sử dụng. Ví dụ, phần mềm về cơ sở dữ liệu như Oracle, đồ họa như Photoshop, soạn thảo và xử lý văn bản … Ưu điểm: là phần mềm có khả năng ứng dụng rộng rãi cho nhiều nhóm ngƣời sử dụng. Khuyết điểm: thiếu tính tùy biến.
* Những phần mềm được viết theo đơn đặt hàng hay hợp đồng của một khách hàng cụ thể nào đó (công ty, bệnh viện, trường học …). Ví dụ, phần mềm điều khiển, phần mềm hỗ trợ bán hàng … Ưu điểm: có tính uyển chuyển, tùy biến cao để đáp ứng nhu cầu của một nhóm người sử dụng nào đó. Khuyết điểm: thông thường đây là những phần mềm ứng dụng trong phạm vi hẹp.

Dựa vào nhu cầu của ngƣời dùng có thể chia phần mềm làm bảy nhóm chính.

Nhóm các hệ điều hành: gồm các chương trình quản lý ổ đĩa, chương trình quản lý các tệp, quản lý thư viện, quản trị mạng, quản lý các chương trình dịch.

Nhóm chương trình dịch: mỗi một ngôn ngữ có một chương trình dịch riêng.

Nhóm các chương trình ứng dụng: gồm có những chương trình soạn thảo văn bản, chương trình xử lý bảng tính điện tử, các chương trình đồ họa, chương trình tạo giao diện thân thiện giữa người dùng và hệ điều hành, các chương trình mở rộng các chức năng tệp. Nhóm các tiện ích và trò chơi: chương trình tìm và diệt virus, các trò chơi.

Nhóm các hệ quản trị cơ sở dữ liệu.

Nhóm các chƣơng trình ứng dụng có tính hệ thống.

Nhóm các chƣơng trình xử lý dữ liệu đa năng: chương trình phục vụ tính toán cấp cơ sở, hệ chuyên gia, hệ tương tự, hệ mô phỏng, hệ tự động thiết kế, dạy học và tự học. Chương trình xử lý, nhận dạng, phân tích, tổng hợp tiếng nói, hình ảnh. Tất cả những chương trình điều khiển quy trình công nghiệp. Để đánh giá được phần mềm theo tiêu chuẩn và theo đúng quy trình, phải đánh giá phần mềm theo từng chủng loại vì tính chất của mỗi loại phần mềm có những đặc trưng riêng. Trong phạm vi đề tài trình bày quy trình đánh giá sản phẩm phần mềm trong quá trình sử dụng, cụ thể là nhóm phần mềm ứng dụng, thực hiện theo 4 bước mô tả trong hình sau.



Bước 1: Thiết lập các yêu cầu đánh giá

Xác lập mục đích đánh giá: Mục đích của việc xác định yêu cầu đánh giá là để miêu tả đối tượng cần đánh giá. Đối tượng của sản phẩm phần mềm là mục đích sử dụng và các rủi ro trong quá trình sử dụng.

Xác định loại sản phẩm: Phân tích các yêu cầu đánh giá bao gồm xác định các yêu cầu, phạm vi hoạt động từ phía khách hàng, cung cấp cho đối tượng yêu cầu đánh giá và mô tả yêu cầu đánh giá từ phía tổ chức thực hiện việc đánh giá, thống nhất các yêu cầu đánh giá. Phạm vi ứng dụng của sản phẩm phải đƣợc xác định song song với việc mô tả mục đích sử dụng. Yếu tố cần đánh giá là độ an toàn, tính bảo mật hay giao diện cũng được xét đến.

Xây dựng mô hình chất lượng: Yêu cầu đánh giá phải mô tả về miền ứng dụng của sản phẩm và phải theo một mô hình chất lượng. Yêu cầu đánh giá sẽ bao gồm một danh sách các yêu cầu chất lượng tham chiếu. Mỗi yêu cầu cần đánh giá cần cung cấp các tham số kỹ thuật của sản phẩm phần mềm cần đánh giá. Các tham số kỹ thuật phải tham chiếu đến chuẩn công nghệ phần mềm. Yêu cầu đánh giá phải được hai bên khách hàng và tổ chức thực hiện đánh giá phê chuẩn. Yêu cầu đánh giá phải được trình bày trong báo cáo đánh giá và trong hồ sơ đánh giá.

Bước 2: Xác lập cơ chế đánh giá

Xác lập các đại lượng và độ đo: Độ đo gồm có độ đo hướng ngoại, độ đo hướng nội và độ đo chất lượng sử dụng. Để chọn phương pháp đo đặc tả đánh giá nên xác định: Tiêu chí của sản phẩm cần đánh giá; Phương pháp đo chất lượng ngoài định lượng; Các modul đánh giá được đóng gói; Danh sách kiểm tra cần trả lời bằng đánh giá; Danh sách các ví dụ có thể giúp ích trong việc trả lời các câu hỏi; Các test case sử dụng; Dữ liệu cần thu thập và phân tích, cùng định dạng của dữ liệu đó; Phương pháp để đánh giá kết quả đánh giá; Phương pháp phù hợp để phân loại kết quả đánh giá được để cho phép lựa chọn sản phẩm phù hợp giữa các sản phẩm cùng chức năng; Phương pháp so sánh nhiều hơn một sản phẩm phần mềm. Phương pháp này có thể dựa theo mức ưu tiên của các tiêu chí chất lượng.

Thiết lập mức đo chuẩn: bao gồm:

- Xây dựng một tiêu chuẩn chính thức cho một phương pháp đo áp dụng cho sản phẩm hay một tập các thành phần, đi kèm bản hướng dẫn ghi lại kết quả đo trong báo cáo đánh giá.

- Tham chiếu tới bản kê trong yêu cầu sản phẩm phần mềm cần đánh giá và tiêu chuẩn thực hiện phần mềm được sử dụng.

- Tham chiếu tới báo cáo trong tiêu chuẩn.

Thiết lập các tiêu chí đánh giá: Tổ chức thực hiện đánh giá xác nhận chuẩn đánh giá khi xem xét các yêu cầu đánh giá. Tổ chức này kiểm tra liệu các thành phần được liệt kê trong bản mô tả sản phẩm có cung cấp thông tin đầy đủ để thực hiện đánh giá theo yêu cầu. Đồng thời xác minh liệu phương pháp đánh giá có khả năng đáp ứng yêu cầu đánh giá.Thông thường các tiêu chí đánh giá sẽ sử dụng các tiêu chí được mô tả trong tiêu chuẩn ISO-9126, bao gồm:

* Tiêu chí chức năng (Functionality)
* Tiêu chí độ tin cậy (Reliability)
* Tiêu chí khả dụ ng (Usability)
* Tiêu chí hiệu quả (Effictiency)
* Tiêu chí bảo trì được (Maintainability)
* Tiêu chí khả chuyển (Portability)

Bước 3: Thiết kế kế hoạch đánh giá sản phẩm phần mềm

Việc thiết kế đánh giá cung cấp tư liệu được sử dụng bởi tổ chức được thực hiện đánh giá để áp dụng phương pháp đo trong một tiêu chuẩn đánh giá. Tổ chức thực hiện đánh giá đưa phương án sử dụng nguồn tài liệu cho một đánh giá cụ thể cũng như cách thức áp dụng nguồn tài liệu khi thực hiện. Quá trình thiết kế phương án đánh giá bao gồm: Văn bản hóa phương pháp và quá trình thiết kế dự thảo đánh giá; Đánh giá lại bản dự thảo; Lên kế hoạch thực hiện đánh giá khi xem xét nguồn tài liệu đã có.

Bước 4: Thực hiện đánh giá

Mục đích của việc thực hiện đánh giá là để đạt được các kết quả từ việc thực thi các hoạt động nhằm đo và kiểm tra sản phẩm phần mềm theo yêu cầu đánh giá như đã được mô tả trong các đặc tả và như được lên kế hoạch trong các kế hoạch đánh giá.

Để thực hiện đánh giá, ngƣời đánh giá phải thực hiện lần lƣợt các công việc chính là:

- Thực hiện đo

- So sánh với tiêu chí đánh giá

Bộ tiêu chuẩn theo ISO có thể sử dụng đánh giá cho nhiều sản phẩm phần mềm khác nhau, nhưng bởi mỗi chủng loại sản phẩm phần mềm có đặc trƣng riêng, tiêu chí này là quan trọng đối với loại phần mềm này nhưng lại không quan trọng đối với loại phần mềm khác. Vì vậy, với mỗi loại phần mềm sẽ đề xuất hệ số cho từng tiêu chí. Đánh giá kết quả thu được dựa vào hệ số chất lượng, được tính bởi công thức:

QF = TQVS / ITQVS

Trong đó:

TQVS – Tổng giá trị chất lượng phần mềm được đánh giá.

ITQVS - Tổng giá trị chất lượng phần mềm trong trường hợp lý tưởng. Việc phân cấp phần mềm được dựa trên giá trị của hệ số chất lượng (QF)

|  |  |  |
| --- | --- | --- |
| STT | Hệ số chất lượng (QF) | Chất lượng |
| 1 | 0.00 ≤ QF ≤ 0.40 | Kém |
| 2 | 0.41 ≤ QF ≤ 0.65 | Đạt |
| 3 | 0.66 ≤ QF ≤ 0.85 | Tốt |
| 4 | 0.86 ≤ QF ≤ 1.00 | Xuất Sắc |

Sau khi thực hiện đánh giá cần xây dựng biên bản đánh giá nhằm mục đích thiết lập một mức độ tin tưởng hợp lý để sản phẩm phần mềm có thể đáp ứng được các yêu cầu đánh giá, xác định và đặc tả những thiếu sót đối với các yêu cầu đánh giá và những đánh giá cần thêm để xác định phạm vi của thiếu sót trên. Nội dung của biên bản đánh giá phải thể hiện được việc thực thi các đánh giá theo các thủ tục mô tả trong kế hoạch, các bước trong thủ tục đánh giá, kết quả đánh giá. Tiếp theo, biên bản báo cáo phác thảo phải được chuyển tới người yêu cầu đánh giá.

1. ***Đánh giá phần mềm về giao diện***

Giao diện người dùng là phương tiện giao tiếp giữa người dùng và các chương trình phần mềm. Khi hệ thống tin học hóa ngày càng cần thiết trong nhiều lĩnh vực đời sống thì vấn đề thiết kế giao diện càng ngày trở nên quan trọng trong việc phát triển phần mềm. Để đánh giá phần mềm giao diện, có thể sử dụng các tiêu trí:

Về tính tương tác chung:

* Giao diện thiết kế rõ ràng, dễ hiểu và dễ sử dụng, bố cục hợp lý.
* Giao diện phải nhất quán, cho thông tin phản hồi có ý nghĩa, yêu cầu xác nhận những tác vụ mang tính phá hoại(xóa file, account).
* Cho phép dễ dàng lần ngược nhiều hành động(hỗ trợ Undo, Redo).
* Tìm kiếm tính hiệu quả trong đối thoại, vận động và ý nghĩa, phân loại các hoạt động theo chức năng và tổ chức màn hình hài hòa theo vùng, cung cấp tiện nghi trợ giúp làm ngữ cảnh.
* Đồng nhất (menu, lệch, hiển thị....) các thành phần điều khiển giao diện được thể hiện rõ nét giúp người sử dụng kiểm soát nhanh chóng.
* Thời gian hệ thống đáp ứng: không quá lâu.
* Hạn chế thông tin phải ghi nhớ giữa hai tác vụ liên tiếp.
* Tối ưu trong trình bày hội thoại và di chuyển mouse.
* Chấp nhận lỗi từ phía người sử dụng.
* Cung cấp trợ giúp trực tuyến.
* Dùng động từ đơn giản và ngắn gọn để đặt tên lệnh.

Về hiển thị thông tin:

* Giao diện chỉ hiển thị thông tin có liên quan đến ngữ cảnh hiện tại.
* Dùng định dạng trình bày cho phép người sử dụng tiếp thu nhanh chóng thông tin.
* Dùng nhãn nhất quán, cách viết tắt chuẩn, dùng tên, từ viết tắt và màu sắc gợi nhớ.
* Đưa ra các thông báo lỗi có ý nghĩa: thể hiện ở thông báo nên mô tả vấn đề mà người dùng có thể hiểu được, thông báo nên đưa ra những lời khuyên có tính xây dựng để khôi phục lỗi, thông báo nên chỉ ra bất kỳ hậu quả lỗi tiêu cực nào để người dùng có thể kiểm tra. Thông báo nên đi kèm với tín hiệu nghe thấy được.
* Các thông báo phản hồi và thông báo lỗi được trình bày phù hợp với nội dung của lỗi hay thông báo cần phản hồi tới người sử dụng. Có các chức năng cần hoặc thông báo gợi ý cần thiết cho người sử dụng lựa chọn trường hợp giải quyết hợp lý.
* Dùng chữ hoa, chữ thường, thụt cấp và gộp nhóm văn bản để trợ giúp cho việc hiểu.
* Chỉ hiển thị thông tin phù hợp với ngữ cảnh hiện tại.
* Cho phép tương tác trực quan.
* Hiển thị dữ liệu ở nhiều dạng khác nhau trong cửa sổ.
* Thiết lập biểu diễn tương tự.
* Sử dụng không gian màn hình một cách tối ưu.

Về việc vào dữ liệu:

* Tối thiểu số hành động đưa vào mà người sử dụng thực hiện.
* Duy trì sự nhất quán giữa việc hiển thị thông tin và việc vào dữ liệu.
* Để người dùng kiểm soát luồng tương tác.
* Cung cấp trợ giúp cho mọi hành động đưa vào, trả lời câu hỏi: trợ giúp có sẵn với mọi chức năng vào mọi lúc? Người sử dụng sẽ yêu cầu trợ giúp thế nào?
* Cách thức trình bày trợ giúp? Người dụng sẽ trở về tương tác thông thường như thế nào? Thông tin trợ giúp được cấu trúc ra sao?
* Hạn chế input trực tiếp (có thể lựa chọn từ một số dữ liệu có sẵn).
* Đồng nhất thông tin Input và hiển thị.
* Cho phép tùy biến input.
* Không cho phép chức năng không thích hợp cho ngữ cảnh hiện tại.
* Cho phép input ở nhiều dạng khác nhau.
* Để người dùng kiểm soát ở nhiều dạng khác nhau.
* Để người dùng kiểm soát dòng sự kiện tương tác.
* Tự động tính các giá trị input cho người nếu có thể.

1. ***Đánh giá phần mềm về chức năng***

Chức năng là tiêu chí quan trọng nhất để đánh giá và lựa chọn phần mềm. Các chức năng phải phù hợp với từng loại phần mềm cụ thể. Với những phần mềm nổi tiếng, không có nghĩa nhà cung cấp có các chức năng cụ thể đáp ứng yêu cầu của người sử dụng. Khi thiết kế chức năng cho phần mềm, người thiết kế thường đáp ứng các tiêu chí: Linh hoạt đối với những yêu cầu thay đổi không định trước; Dễ thử nghiệm; Tính sáng sủa, dễ đọc; Có quan hệ chặt chẽ giữa thiết kế và yêu cầu … Ta có thể đánh giá phần mềm theo một số tiêu chí sau:

- Sự hiện diện của các chức năng, tất cả các chức năng được đề cập trong tư liệu phải được chạy theo một dạng được trình bày trong tư liệu, kể cả tư liệu người dùng.

- Tính chính xác. Tất cả các chương trình và dữ liệu phải phù hợp với phần trình bày trong phần mô tả sản phẩm và tư liệu người sử dụng. Các chức năng phải thể hiện rõ nhiệm vụ của công việc.

- Các ứng dụng con có khả năng thực thi chính xác các nhiệm vụ.

- Có các thông báo tới người sử dụng về tình hình xảy ra lỗi chủ quan hoặc khách quan, có những gợi ý cần thiết cho phép ngƣời sử dụng tự giải quyết các lỗi.

- Các điều kiện biên trong các trường hợp được giới hạn đúng với dữ liệu của trường đó.

- Có khả năng kiểm soát được những thay đổi dữ liệu các trường từ phía người sử dụng.

- Các chức năng điều hướng liên kết hợp lý, cho phép người sử dụng có khả năng kiểm soát và khai thác phần mềm một cách hiệu quả nhất.

- Tính nhất quán.

1. ***Đánh giá phần mềm về tiện ích***

Các loại phần mềm có nhiều loại tiện ích khác nhau, có thể dựa vào một số tiêu chí sau để đánh giá tiện ích:

- Hỗ trợ nhiều người sử dụng cùng một lúc, việc phân quyền nhóm người sử dụng linh hoạt (qua mạng cục bộ LAN)

- Quản lý truy xuất thông tin theo từng nhóm công việc một cách linh hoạt, hiệu quả nhanh chóng.

- Hệ thống báo cáo đa dạng, dễ dàng xuất ra Word, Excel, pdf, html …

- Quá trình tìm kiếm tài liệu nhanh chóng, tiện lợi.

- Tự động cập nhật cấu hình qua Internet.

- Dễ dàng kết nối với các thiết bị ngoại vi.

- Tiện ích (Add-on) cho phép thêm các tính năng vốn không có trong chương trình chuẩn. Dễ dàng cài đặt cũng như gỡ bỏ hoặc vô hiệu hóa.

- Hệ thống có tính năng tìm kiếm thông tin hay không?

- Hệ thống luôn phải hỗ trợ một giao diện, tài liệu trợ giúp người sử dụng trong quá trình sử dụng phần mềm.

- Trợ giúp được trình bày rõ ràng, dễ hiểu cho phép người sử dụng có khả năng xem nội dung hướng dẫn và hiểu được nội dung, mục đích và cơ chế hoạt động của phần mềm.

1. Đánh giá phần mềm về bảo mật, an toàn

Các sản phẩm phần mềm ngày nay bị đe dọa bảo mật với nhiều hệ máy tính Windows, trong đó có thư rác, phising và nhiều website độc hại khác. Mỗi loại phần mềm có những mức độ bảo mật khác nhau, có thể đánh giá sự bảo mật an toàn của sản phẩm phần mềm thông qua một số tiêu chí sau:

- Phần mềm phải có những động cơ riêng biệt giúp cảnh báo về những cấu hình trình duyệt không an toàn và các lỗ hổng bảo mật tiềm tàng khác, là khả năng của sản phẩm phần mềm bảo vệ các thông tin, dữ liệu khi bị xâm phạm bất hợp pháp.

- Bảo mật nội dung dữ liệu: Các dữ liệu riêng tư, quan trọng phải được mã hóa trước khi truyền trên đường truyền.

- Các giao thức kèm theo phải được bảo mật tối đa để đảm bảo thông tin không bị rò rỉ qua các lỗ hổng của giao thức. - Có cơ chế backup sao lưu dữ liệu quan trọng trong máy tính tránh bị mất dữ liệu và hiểm họa do virut gây.

- Hệ thống có tính bảo mật là hệ thống mà thông tin của nó không cho phép những người không có quyền được xem.

- Hệ thống phải luôn có tính nhất quán dữ liệu. Phải ngăn ngừa sự thay đổi một thông tin nào đó dù là vô tình hay cố ý (không được phép).