Mô hình hóa tiến trình phát triển phần mềm với software prototyping và Mô hình phân tích động thái

I. Mô hình hóa tiến trình phát triển phần mềm

1. Tiến trình phát triển phần mềm - Mô hình software prototyping

Một tiến trình phát triển phần mềm là một tập của các hoạt động cần thiết để chuyển các yêu cầu người dùng thành một hệ thống phần mềm đáp ứng được các yêu cầu đặt ra.

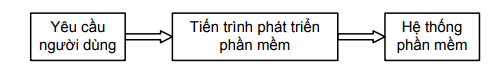


Figure 1: Tiến trình phát triển phần mềm cơ bản

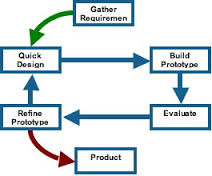
Mô hình tiến trình phát triển phần mềm là sự mô tả tiến trình một cách đơn giản khi xem xét nó từ một cách nhìn cụ thể.

**Sofware Prototying** là một mô hình tiến trình phát triển phần mềm liên quan đến việc xây dựng một prototype của hệ thống, phần mềm hoặc một phần của hệ thống, phần mềm dùng để kiểm tra các chức năng, yêu cầu mà khách hành đưa ra và tính khả thi mà không đi sâu vào tính chính xác logic của hệ thống hay phần mềm đó.

Sofware Prototying đã trở thành một mô hình rất phổ biến và hữu dụng trong thiết kế hệ thống hay phần mềm bởi nó cho phép các nhà phát triển có thể hiểu và hình dung được các yêu cầu của hệ thống, phần mềm ngay từ giai đoạn rất sớm của quá trình phát triển.

Sofware Prototyping rất hữu dụng khi nhà phát triển không chắc chắn về các yêu cầu, thuật toán, tính hiệu quả hay tính khả thi của hệ thống, phần mềm.

2. Các giai đoạn thực hiện



Giai đoạn 1: Xác địch mục tiêu, yêu cầu cơ bản Basic Requirement Identification

* Liên quan đến việc hiểu được những yêu cầu cơ bản nhất của phần mềm đặc biệt là UI (user interface). Những bước thiết kế dữ liệu bên trong và các khía cạnh như hiệu năng, bảo mật có thể hoàn tuàn được bỏ qua tại bước này.

Giai đoạn 2: Thiết kế prototype (Developing the initial Prototype)

* Việc thiết kế prototype được thực hiện ở giai đoạn này. Các tính năng căn bản nhất của phần mềm được thể hiện và UI được cài đặt.
* Để giảm thiểu chi phí prototype ban đầu, nhà phát triển có thể bỏ qua một số vấn đề như thời gian phản hồi, bộ nhớ sử dụng, …

Giai đoạn 3: Lấy ý kiến về prototype (Review of the prototype)

* Prototype đã được phát triển sẽ được sử dụng bởi một số khách hàng. Sau đó, các phản hồi được tập hợp lại và được sử dụng cho việc nâng cấp, cải tiến phần mềm.

Giai đoạn 4: Xem xét và cải tiến prototype (Revise and Enhance the prototype)

* Các phản hồi của khách hàng được xem xét, thống nhất và một prototype mới được phát triển. (Lặp lại giai đoạn 2)

Giai đoạn 5: Xuất sản phẩm Product

3. Ví dụ: Phần mềm quản lý nhà sách

Giai đoạn 1: Xác định yêu cầu cơ bản

* Đối tượng sử dụng phần mềm: nhân viên quản lý nhà sách.
* Phần mềm cần quản lý các tiến trình nhập, xuất sách.
* Phần mềm cần quản lý các đầu sách được nhập, xuất như tên sách, tác giả, thể loại, số lượng, …
* Khi nhân viên nhập, xuất sách cần tiến hành làm phiếu nhập và xuất.
* Phiếu tính tiền khi khách hàng mua sách tại nhà sách.
* Thống kê số lượng sách sau mỗi tháng.
* Giao diện dễ sử dụng.
* …

Giai đoạn 2: Thiết kế prototype

**(Ảnh UI phần mềm ví dụ)**

Giai đoạn 3: Lấy ý kiến về prototype

* Xác định các điểm bất hợp lý trong prototype

**(Ảnh UI phần mềm ví dụ và chỉ ra các điểm bất hợp lý)**

Giai đoạn 4: Cải tiến prototype

**(Ảnh UI phần mềm ví dụ và ảnh sau khi cải tiến)**

Giai đoạn 5: Xuất sản phẩm

**(Ảnh phần mềm sau khi hoàn thành)**

II. Ngôn ngữ UML và mô hình phân tích động thái:

**UML** là ngôn ngữ trực quan được dùng cho việc đặc tả, hình dung, xây dựng cho các hệ thống phần mềm. UML là một tập các phần tử và một tập các quy tắc riêng. Các phần tử của UML là các đối tượng đồ họa như đường thẳng, hình chữ nhật và các quy tắc trong UML xác định các kết hợp giữa các phần tử này.

Trong UML, các đối tượng trao đổi với nhau bằng cách gửi các thông điệp cho nhau. Sự trao đổi này thể hiện sự tương tác giữa các đối tượng trong hệ thống. Mô hình động thái hệ thống được biểu diễn bằng các sơ đồ tương tác (Interaction Diagram)

- Sơ đồ trình tự (Sequence Diagram): mô tả sự trao đổi, tương tác của các đối tượng với nhau theo trình tự thời gian. Sơ đồ trình tự bao gồm các phần tử biểu diễn cho các đối tượng, các thông điệp được gửi và nhận trình tự theo thời gian để thực hiện các ca sử dụng của hệ thống.

- Sơ đồ trạng thái (State Diagram): mô tả các trạng thái, hành vi của các đối tượng. Sơ đồ trạng thái bao gồm những thông tin về những trạng thái khác nhau của các đối tượng, thể hiện các đối tượng chuyển từ trạng thái này sang trạng thái khác như thế nào, hành vi ứng xử của mỗi đối tượng khi có các sự kiện xảy ra để làm thay đổi trạng thái.

- Sơ đồ hoạt động (Activity Diagram): mô tả cách các đối tượng tương tác với nhau nhưng nhấn mạnh về công việc, xác định các hoạt động và thứ tự thực hiện những hoạt động đó.

- Sơ đồ cộng tác (Collaboration Diagram): mô tả sự tương tác của các đối tượng với nhau theo ngữ cảnh và không gian công việc.

Việc xây dựng các sơ đồ này nhằm mục đích mô hình hóa hoạt động của hệ thống.

Không phải tất cả các hệ thống đều cần cả bốn sơ đồ tương tác trên mà nó còn tùy thuộc vào mức độ phức tạp của của hệ thống. Đối với những hệ thống tương đối đơn giản thì chỉ cần sơ đồ trình tự và sơ đồ trạng thái là đủ.

1. Sơ đồ trình tự:

1.1. Các thành phần của sơ đồ trình tự:

Sơ đồ trình tự bao gồm các phần tử biểu diễn đối tượng, thông điệp và thời gian. Sơ đồ trình tự được thể hiện theo hai trục:

* Trục dọc trên xuống chỉ thời gian xảy ra các sự kiện, hay sự truyền thông điệp, được biểu diễn bằng các đường gạch - gạch thẳng đứng bắt đầu từ đỉnh đến đáy của sơ đồ.
* Trục ngang từ trái qua phải là dãy các đối tượng tham gia vào việc trao đổi các thông điệp với nhau theo chiều ngang, có thể có cả các tác nhân.

Đối tượng được biểu diễn bằng hình chữ nhật trong đó có tên đối tượng cụ thể và/hoặc tên lớp cùng được gạch dưới (hoặc tên lớp được gạch dưới biểu diễn cho một đối tượng bất kỳ của lớp đó).

Sơ đồ trình tự được đọc từ trên xuống dưới, từ trái sang phải. Thứ tự các đối tượng trong sơ đồ phải được sắp xếp sao cho đơn giản nhất có thể để dễ quan sát. Thời gian thực hiện một thông điệp của một đối tượng, hay còn gọi là hoạt động của đối tượng được biểu diễn bằng hình chữ nhật hẹp dọc theo trục thẳng đứng của đối tượng đó.

1.2. Các bước xây dựng sơ đồ trình tự:

- Xác định các tác nhân, các đối tượng tham gia vào ca sử dụng và vẽ chúng theo hàng ngang trên cùng theo đúng các ký hiệu,

- Xác định những thông điệp (lời gọi hàm) mà tác nhân cần trao đổi với một đối tượng nào đó, hoặc giữa các đối tượng tương tác với nhau theo trình tự thời gian và vẽ lần lượt các hoạt động đó từ trên xuống theo thứ tự thực hiện trong thực tế.

1.3. Ví dụ: Quản lý nhà sách

1.3.1. Sơ đồ trình tự mô tả ca sử dụng “Lập phiếu nhập sách”

Nhân viên nhà sách nhập vào các đầu sách sắp được nhập vào nhà sách. Khi nhập sách, nhân viên này cần thông tin và số lượng n của đầu sách được nhập. Những thông tin về đầu sách này được xác định thông qua mã sách id. Vậy đối tượng NhanVien gửi đến cho HeThong thông điệp nhập đầu sách với mã sách id và số lượng sách được nhập n nhapSach(id, n). Khi đã hoàn tất việc nhập tất cả các đầu sách thì cần phải báo cho hệ thống biết là đã nhập xong bằng cách nhấn nút HoanTatNhapSach. Điều này có nghĩa là thông báo cho hệ thống thông điệp capNhat().

Các hoạt động trên được mô tả bằng sơ đồ trình tự như sau:

capNhat()

nhapSach(id,n)

NhanVien

HeThong

1.3.2. Sơ đồ trình tự mô tả ca sử dụng “Lập hóa đơn bán sách”

Nhân viên nhà sách nhập vào các đầu sách sắp được bán. Khi thực hiện việc bán sách, nhân viên này cần thông tin và số lượng n của đầu sách được nhập. Những thông tin về đầu sách này được xác định thông qua mã sách id. Vậy đối tượng NhanVien gửi đến cho HeThong thông điệp bán đầu sách với mã sách id và số lượng sách được bán n banSach(id, n). Khi đã hoàn tất việc cung cấp thông tin tất cả các đầu sách được bán thì cần phải báo cho hệ thống biết là đã nhập xong thông tin bằng cách nhấn nút HoanTatLapHoaDon. Điều này có nghĩa là thông báo cho hệ thống thông điệp capNhat().

Các hoạt động trên được mô tả bằng sơ đồ trình tự như sau:

capNhat()

banSach(id,n)

NhanVien

HeThong

1.3.3. Sơ đồ trình tự mô tả ca sử dụng “Thanh toán”

Sau khi lập hóa đơn bán sách và nhân viên nhà sách bấm nút ThanhToan. Điều này có nghĩa là thông báo cho HeThong thông điện batDauThanhToan(). HeThong sau đó hiển thị số tiền khách hàng phải trả hienThiTienPhaiTra(). Tiếp theo, NhanVien nhập vào số tiền (soTien) mà khách hàng đưa và nhấn nút hoanTatThanhToan. HeThong nhận được thông điệp thanhToan(soTien) và tính toán để hiển thị số tiền thừa phải trả lại khách hienThiTienThua().

Sơ đồ trình tự:

hienThiTienPhaiTra()

hienThiTienThua()

thanhToan(soTien)

batDauThanhToan()

NhanVien

HeThong

2. Sơ đồ trạng thái:

2.1. Trạng thái và sự biến đổi trạng thái

Mọi đối tượng trong hệ thống đều có chu kỳ sống và mỗi thời điểm đều có một trạng thái nào đó.

Trạng thái là một trong các điều kiện có thể để đối tượng tồn tại, là kết quả của một hoạt động trước đó của đối tượng. Trạng thái của đối tượng thường được mô tả trong hình chữ nhật góc tròn và được xác định bởi:

* Tên gọi trạng thái, thường bắt đầu bằng động từ.
* Biến trạng thái mô tả các giá trị hiện thời của trạng thái.
* Hoạt động là hành vi mà đối tượng sẽ thực hiện khi nó ở vào trạng thái đó.

2.2. Xác định các trạng thái và các sự kiện:

Để xác định được các trạng thái và các sự kiện chúng ta cần trả lời cho các câu hỏi sau:

- Một đối tượng có thể ở những trạng thái nào? Liệt kê tất cả các trạng thái có thể có trong hệ thống của mỗi đối tượng.

- Những sự kiện nào có thể xuất hiện? Bởi vì sự kiện có thể làm biến đổi trạng thái, do vậy, từ các sự kiện có thể xác định được các trạng thái của đối tượng.

- Những trạng thái mới nào sẽ xuất hiện? Từ một trạng thái, đối tượng có thể chuyển sang trạng thái mới khi một số sự kiện xác định xuất hiện.

- Ở mỗi trạng thái, hoạt động của đối tượng là gì? - Sự tương tác giữa các đối tượng là gì? Sự tương tác giữa các đối tượng thường gắn chặt với các trạng thái của đối tượng.

- Những sự kiện, hay chuyển đổi trạng thái nào là không thể xảy ra? Một số sự kiện, hay trạng thái không thể chuyển đổi sang trạng thái khác được.

- Cái gì làm cho đối tượng được tạo ra? Đối tượng thường được tạo ra bởi một, hay một số sự kiện.

- Cái gì làm cho đối tượng bị hủy bỏ? Đối tượng thường được loại bỏ khi không còn cần thiết nữa?

2.3. Ví dụ: Quản lý nhà sách

2.3.1. Sơ đồ trạng thái mô tả “Lập phiếu nhập sách”

Một phiếu nhập sách được tạo lập

Nhập các đầu sách

Chưa nhập

Đã được nhập xong

Khi một phiếu nhập sách được tạo lập thì nó ở trạng thái chưa được nhập, sau khi được nhân viên nhập vào các đầu sách thì nó chuyển sang trạng thái đã được nhập xong.

2.3.2. Sơ đồ trạng thái mô tả “Lập hóa đơn bán sách”

Tương tự với “Lập phiếu nhập sách”, ta cũng có thể vẽ được sơ đồ:

Một hóa đơn bán sách được tạo lập

Nhập các đầu sách

Chưa nhập

Đã được nhập xong

Khi một hóa đơn bán sách được tạo lập thì nó ở trạng thái chưa được nhập, sau khi được nhân viên nhập vào các đầu sách được bán thì nó chuyển sang trạng thái đã được nhập xong.

2.3.3. Sơ đồ trạng thái mô tả “Thanh toán”

Trả tiền

Hiển thị tiền thừa

Kết thúc thanh toán

Khách hàng chuẩn bị thanh toán

Hiển thị tiền phải trả

Chờ để thanh toán

Chờ thanh toán

Chờ kiểm duyệt tiền

Lấy tiền thừa

Khi khách hàng chuẩn bị thanh toán tiền thì khách hàng ở trạng thai chờ để thanh toan. Sau khi được hệ thống hiển thị tiền phải trả thì sơ đồ chuyển sang trạng thái chờ thanh toán. Tiếp đó, khách hàng trả tiền và chuyển sang trạng thái chờ kiểm duyệt tiền. Sau khi được nhân viên trả tiền thừa và chuyển sang trạng thái lấy tiền thừa thì việc thanh toán kết thúc.