**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

**---oOo---**

Logo

Description automatically generated

**BTVN-04**

**TÌM VÍ DỤ LIÊN QUAN ĐẾN THAY ĐỔI YÊU CẦU**

Họ và tên sinh viên: Huỳnh Tấn Thọ

Mã số sinh viên: 19120383

Lớp: 19\_3

Email: 19120383@student.hcmus.edu.vn

Môn học: Phân tích và quản lí yêu cầu phần mềm

Giảng viên: Lâm Quang Vũ

Thành phố Hồ Chí Minh, tháng 5 năm 2022

**MỤC LỤC**

[I. Mutable Requirements 3](#_Toc103631993)

[II. Emergent Requirements 4](#_Toc103631994)

[III. Consequential Requirements 5](#_Toc103631995)

[IV. Compatibility Requirements 6](#_Toc103631996)

1. Mutable Requirements
2. Định nghĩa

* Là loại yêu cầu thay đổi khi có sự thay đổi về môi trường mà ứng dụng hoặc hệ thống đang vận hành.

1. Ví dụ

* Trong đề tài của nhóm– Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, giả sử có sự thay đổi trong cách tính toán chi phí trung gian từ 20% tháng lương đầu tiên thành 5% lương hàng tháng và tối đa là 250 000 VNĐ.
  + Thay đổi này là thay đổi trong môi trường vận hành ứng dụng, do đó ứng dụng phải cập nhật theo.
* Trong đề tài của nhóm– Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, ban điều hành ứng dụng quyết định không muốn trực tiếp đứng ra thu giữa gia sư và học viên. Do đó hệ thống sẽ có thêm một loại người dùng mới là các Trung tâm gia sư, làm công việc trung gian giữa học viên và gia sư. Ứng dụng chỉ thu phí từ loại người dùng mới này.
  + Thay đổi này là thay đổi trong môi trường vận hành ứng dụng, do đó phải cập nhật về cơ sở dữ liệu, thêm một loại người dùng mới, và ứng dụng phải cập nhật các tính năng mới cho Trung tâm gia sư.
* Một ứng dụng quản lý bán vé chuyến bay ban đầu chỉ cho phép người dùng mua vé, thanh toán và nhận vé trực tiếp tại sân bay. Về sau, ban quản lý sân bay cho phép người dùng đặt vé, và thanh toán trực tuyến mà không cần phải đến trực tiếp sân bay.
  + Thay đổi này là thay đổi trong môi trường vận hành ứng dụng. Do đó, ứng dụng phải cập nhật thêm các tính năng mới cho phép người dùng mua vé trực tuyến.
* Một ứng dụng quản lý nhà thuốc ban đầu chỉ có một loại người dùng duy nhất quản lý tất cả thông tin về hóa đơn, thu chi, kho thuốc,… Sau này để mở rộng quy mô, chủ nhà thuốc quyết định chia ra làm 2 bộ phận riêng biệt và quản lý kho.
  + Thay đổi này là thay đổi trong môi trường vận hành ứng dụng. Do đó, ứng dụng phải tạo thêm một loại người dùng mới là “Quản lý kho” và cung cấp các tính năng cần thiết cho họ.
* Một hệ thống thi trực tuyến ở một trường đại học ban đầu chỉ có hình thức thi trắc nghiệm cho sinh viên, mỗi câu có 1 đáp án đúng, và số điểm được chia đều cho các câu. Sau này, lãnh đạo trường quyết định thêm 2 hình thức thi là Trả lời câu hỏi và Điền khuyết. Về bài thi trắc nghiệm, số điểm của các câu trong bài thi trắc nghiệm không nhất thiết phải đều nhau và mỗi câu có thể có nhiều đáp án đúng. Ngoài ra, mỗi môn học trong học kỳ đó chỉ được phép cho sinh viên làm bài trực tuyến trên hệ thống tối đa 5 lần.
  + Thay đổi này là thay đổi trong môi trường vận hành ứng dụng. Do đó, ứng dụng phải cập nhật thêm các tính năng cần thiết giúp giảng viên tạo được các bài kiểm tra đa dạng hơn, cũng như ràng buộc về số bài kiểm tra trực tuyến mỗi môn học.

1. Emergent Requirements
2. Định nghĩa

* Là loại yêu cầu chưa thể được định nghĩa rõ ràng khi đặc tả hệ thống, mà chỉ xuất hiện khi hệ thống được thiết kế và cài đặt.

1. Ví dụ

* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, khi đang demo sản phẩm cho khách hàng và họ cho rằng nên có thêm các tính năng giúp thay đổi giao diện (cỡ chữ, màu sắc,…).
  + Yêu cầu này xuất hiện sau khi hệ thống đã được thiết kế và xuất phát từ nhu cầu của người dùng khi họ có thêm hiểu biết về sản phẩm họ sắp sử dụng.
* Trong đề tài của nhóm– Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, khi đang demo sản phẩm cho khách hàng và họ cho rằng nên có thêm tính năng giúp gia sư và học viên có thể nhắn tin với nhau thông qua ứng dụng
  + Yêu cầu này xuất hiện sau khi hệ thống đã được thiết kế, cài đặt, và xuất phát từ nhu cầu của người dùng khi họ có thêm hiểu biết về sản phẩm họ sắp sử dụng.
* Trang web đặt hàng trực tuyến của Gà Rán KFC ban đầu chỉ cho phép thanh toán khi nhận hàng hoặc thanh toán thông qua thẻ tín dụng. Về sau, khách hàng sử dụng các ví điện tử ngày càng nhiều, do đó trang web phải thêm các hình thức thanh toán bằng ví điện tử cho khách hàng lựa chọn.
  + Yêu cầu này xuất hiện sau khi hệ thống đã được cài đặt và xuất phát từ nhu cầu của người dùng.
* YouTube ban đầu không có tính năng Video Ngắn. Tuy nhiên, với sự bùng nổ của TikTok, người dùng ngày càng có xu hướng thích xem dạng video ngắn dưới 1 phút. Do đó, đội ngũ phát triển đã triển tính năng này.
  + Yêu cầu này xuất hiện sau khi hệ thống đã được đưa vào vận hành và xuất phát từ nhu cầu, thị hiếu, xu hướng của người dùng.
* Grab ban đầu là một ứng dụng chuyên hỗ trợ đặt xe. Sau này, mọi người dần có xu hướng đặt thức ăn thông qua ứng dụng thay vì đi mua trực tiếp. Do đó, Grab đã phát triển thêm tính năng giao thức ăn để phục vụ nhu cầu người dùng.
  + Yêu cầu này xuất hiện sau khi hệ thống đã được đưa vào vận hành và xuất phát từ nhu cầu, thị hiếu, xu hướng của người dùng.

1. Consequential Requirements
2. Định nghĩa

* Là loại yêu cầu được đặc tả đựa trên các giả định khi hệ thống được đưa vào hoạt động. Khi sử dụng thực tế, một số giả định có thể sai.

1. Ví dụ

* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, đội ngũ phát triển cho rằng tại mọi thời điểm, có không quá 500 người truy cập đồng thời. Khi đưa vào sử dụng, có trên 2000 tài khoản truy cập cùng lúc, khiến máy chủ và cơ sở dữ liệu bị quá tải, hệ thống bị treo hoặc sập. Đội ngũ phát triển phải tiến hành nâng cấp hệ thống.
  + Yêu cầu của hệ thống thay đổi khi các giả định trong quá trình thiết kế và cài đặt hệ thống khác so với khi vận hành thực tế.
* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, ứng dụng sử dụng dịch vụ bản đồ từ Google Maps. Khi thiết kế và cài đặt ứng dụng, dịch vụ này không tốn phí. Tuy nhiên, sau khi ra mắt được một thời gian, dịch vụ có thể yêu cầu trả phí và đội ngũ phát triển phải tốn thêm chi phí vận hành hoặc tìm kiếm dịch vụ khác thay thế.
  + Yêu cầu của hệ thống thay đổi khi các giả định trong quá trình thiết kế và cài đặt hệ thống khác so với khi vận hành thực tế.
* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, ứng dụng cung cấp tính năng đánh giá gia sư nhằm giúp người dùng dễ dàng tìm được gia sư chất lượng, đáng tin cậy. Tuy nhiên, trong thực tế, học viên thường xuyên không để lại đánh giá dành cho gia sư, gây thiếu chính xác khi xếp hạng và lọc ra các gia sư chất lượng. Đội ngũ phát triển phải nghĩ ra giải pháp khuyến khích người dùng sử dụng tính năng này nhiều hơn, hoặc thay thế hoàn toàn tính năng này.
  + Yêu cầu của ứng dụng thay đổi khi các giả định trong quá trình thiết kế và cài đặt hệ thống khác so với khi vận hành thực tế.
* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, khi một tài khoản không sử dụng quá 5 năm, hệ thống sẽ xóa thông tin của tài khoản đó khỏi cơ sở dữ liệu nhằm tiết kiệm chi phí lưu trữ. Tuy nhiên, thực tế người dùng chỉ đăng ký tài khoản và sử dụng ứng dụng trong 6 tháng đến 1 năm sau đó không dùng nữa. Đội ngũ phát triển quyết định nếu tài khoản không sử dụng quá 2 năm thì sẽ xóa bỏ.
  + Yêu cầu của hệ thống thay đổi khi các giả định trong quá trình thiết kế và cài đặt hệ thống khác so với khi vận hành thực tế.
* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, hệ thống sử dụng Firebase của Google làm cơ sở dữ liệu. Tuy nhiên, sau khi vận hành được một thời gian, Firebase thay đổi chính sách khiến giá thuê máy chủ tăng cao. Đội ngũ phát triển phải tốn thêm chi phí vận hành hoặc tìm kiếm dịch vụ khác thay thế.
  + Yêu cầu của hệ thống thay đổi khi các giả định trong quá trình thiết kế và cài đặt hệ thống khác so với khi vận hành thực tế.

1. Compatibility Requirements
2. Định nghĩa

* Là loại yêu cầu phụ thuộc vào các thiết bị, tiến trình, hoặc hệ thống khác.

1. Ví dụ

* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, ứng dụng được thiết kế chạy trên các thiết bị Android 9 hoặc iOS 13 trở đi. Tuy nhiên, một bộ phận người dùng vẫn sử dụng các thiết bị cũ, do đó ứng dụng phải chạy được trên các thiết bị từ Android 8.1 hoặc iOS 12 trở xuống. Ngoài ra, nếu sau này ra mắt phiên bản hệ điều hành Android hoặc iOS mới hơn, ứng dụng phải đảm bảo cập nhật kịp thời.
  + Ứng dụng phải tương thích ngược với các phiên bản hệ điều hành cũ hơn hoặc tương thích với các phiên bản mới vừa ra mắt.
* Trong đề tài của nhóm – Ứng dụng hỗ trợ tìm kiếm gia sư Seeker, ứng dụng sử dụng một số dịch vụ thanh toán trực tuyến như Internet Banking, Zalo Pay,… Tuy nhiên, sau khi vận hành được một thời gian, các dịch vụ này có sự thay đổi, dẫn đến việc đội ngũ phát triển phải cập nhật mã nguồn cho sản phẩm của họ sao cho phù hợp.
  + Ứng dụng sử dụng dịch vụ của một bên khác và phải cập nhật để tương thích khi có sự thay đổi ở bên đó.
* Một trang web đọc sách và truyện được thiết kế để sử dụng trên máy tính. Tuy nhiên, người dùng ngày càng có xu hướng truy cập trang web qua điện thoại. Do đó, đội ngũ phát triển phải thiết kế giao diện sao cho hiển thị được trên các màn hình có kích thước khác nhau.
  + Ứng dụng phải tương thích với nhiều thiết bị có kích thước khác nhau.
* Một phần mềm hỗ trợ tính tiền trong siêu thị được thiết kế để chạy trên Windows 7 trở đi. Tuy nhiên, ở cửa hàng sử dụng phần mềm này phần lớn các thiết bị chạy Windows XP trở xuống. Đội ngũ phát triển phải cập nhật cho phần mềm sử dụng được trên Windows XP hoặc làm một phiên bản dành riêng.
  + Ứng dụng phải tương thích ngược với các thiết bị, các phiên bản hệ điều hành cũ hơn.
* Một phần mềm giúp đọc ảnh từ các máy ảnh kĩ thuật số. Tuy nhiên, mỗi loại máy ảnh có cách đọc dữ liệu khác nhau. Do đó, phần mềm phải đảm bảo tương thích được với hầu hết các loại máy ảnh phổ biến trên thị trường.
  + Ứng dụng phải tương thích ngược với các thiết bị khác nhau.