# Review

Câu 1:

Câu 1.1

a/Các công việc của Project Manager:

* Lập kế hoạch cho dự án: PM là người có trách nhiệm lên kế hoạch cho việc phát triển dự án, ước lượng nhân lực cũng như phân công công việc
* Quản lý rủi ro: PM phải đánh giả rủi ro có thể gây bất lợi trong quá trình phát triển dự án, đánh giá nguy cơ rủi ro và lập kế hoạch để giảm thiểu rủi ro đó
* Quản lý nhân sự: PM có trách nhiệm quản lý đội ngũ phát triển. PM là người chọn ra các thành viên tham gia phát triển cũng như tổ chức cách làm việc để nhóm đạt hiệu quả tối ưu
* Viết báo cáo: PM là người báo cáo tiến trình của dự án đến khách hàng cũng như người quản lý của công ty phát triển phần mềm.
* Viết proposal – kế hoạch đề xuất: PM là người viết kế hoạch đề xuất để mang về hợp đồng cho cty. Kế hoạch đề xuất không đi quá sâu nhưng cho biết dự án sẽ được hoàn thành như thế nào.

b/ Các thành phẩm được tạo ra bởi PM trong quá trình phát triển phần mềm

* Các tài liệu báo cáo: Các tài liệu báo cáo có thể cho khách hàng biết về tiến trình dự án, báo cáo cho cấp trên, các biểu đồ UML cho biết về kiến trúc dự án, các báo cáo về quản lý rủi ro….
* Các kế hoạch về dự án: Các cột mốc quan trọng trong quá trình phát triển, đảm bảo dự án hoàn thành trong đúng thời gian cho phép

Câu 1.2

a/Phân biệt yêu cầu chức năng và yêu cầu phi chức năng

* Yêu cầu chức năng: Cho biết hệ thống có thể thực hiện được những việc gì, cách hệ thống xử lý các đầu vào cụ thể
* Yêu cầu phi chức năng: Là những ràng buộc đối với dịch vụ hay chức năng nào đó của hệ thống. Thường là những ràng buộc về mặt thời gian phản hồi, về độ tin cậy … Yêu cầu phi chức năng thường được áp dụng cho cả hệ thống chứ không chỉ một chức năng nào đó

b/Yêu cầu phi chức năng ảnh hưởng thế nào đến việc thiết kế kiến trúc phần mềm? Nêu 3 ví dụ

* Kiến trúc phần mềm được chọn phải đảm bảo thỏa mãn cả yêu cầu chức năng và yêu cầu phi chức năng của hệ thống. Tuy nhiên, nếu như kiến trúc phần mềm được chọn không thể đáp ứng được yêu cầu phi chức năng thì cả hệ thống sẽ không thể được chấp nhận. Do đó, chúng ta phải dựa vào đặc điểm (thường liên quan đến yêu cầu phi chức năng như: hiệu năng, tính bảo mật, tính an toàn, tính thường trực, tính dễ bảo trì…) của hệ thống để lựa chọn kiến trúc phần mềm phù hợp
* Ví dụ 1: Hệ thống bán vé xe cho toàn thành phố

Đây là hệ thống có cơ sở dữ liệu lớn (quản lý từng loại vé xe có điểm xuất phát, điểm đến khác nhau) và cần có sự đồng nhất dữ liệu (data synchronization) – còn 1 vé xe thì không thể bán cho 2 người

⇨ Đề xuất: Kiến trúc repository. Vì kiến trúc này thích hợp cho các hệ thống cần lưu trữ dữ liệu, tất cả các điểm bán vé đều truy cập vào csdl trung tâm (tránh được trình trạng không đồng nhất dữ liệu). Giả sử điểm bán vé xe tại điểm A có trục trặc thì các điểm bán vé xe khác cũng không bị ảnh hưởng (vì chúng không liên quan đến nhau)

* Ví dụ 2: Hệ thống xử lý hóa đơn tiền điện quốc gia

Đây là hệ thống chuyên về việc xử lý dữ liệu, ít tương tác với người dùng (data-driven model)

⇨ Đề xuất: Kiến trúc Pipe and Filter. Vì kiến trúc này thích hợp với việc xử lý input, đưa input qua nhiều bước xử lý (transformation) và đưa ra output thích hợp (tiền cần phải trả). Và hệ thống xử lý hóa đơn cũng là một hệ thống ít tương tác với người dùng (chỉ nhận input) do đó kiến trúc Pipe and Filter là một lựa chọn phù hợp

* Ví dụ 3: Hệ thống quản lý doanh thu của một cửa hàng

Đây là hệ thống thống kê, cho biết doanh thu của cửa hàng mỗi tháng và sự tương quan giữa doanh thu của các tháng với nhau

⇨ Đề xuất: Kiến trúc MVC.

Model: Liên kết với database (csdl hàng hóa của cửa hàng)

View: Giao diện thể hiện doanh thu của cửa hàng (có nhiều view khác nhau)

Controller: Xử lý tưởng tác của người dùng (nhập dữ liệu vào database, chọn các giao diện khác nhau để xem doanh thu)

View component sẽ có nhiều view khác nhau giúp người quản lý có thể xem doanh thu cửa hàng dưới nhiều góc độ (Pie chart – thể hiện doanh thu tính đến thời điểm hiện tại so với mục tiêu cả năm, Bar chart – giúp việc so sánh doanh thu của tháng này so với các tháng trước dễ dàng hơn … ) Trong tương lai, việc cửa hàng muốn thêm các giao diện mới, thêm những cách tương tác mới cũng có thể được xử lý vì cả 3 component của kiến trúc MVC hoạt động độc lập với nhau.

Câu 2:

Câu 2.1

a/ Viết 15 mong muốn đối với hệ thống phần mềm:

Vai trò: Cư dân thành phố - 9

* Là cư dân thành phố, tôi mong muốn ứng dụng có các hướng dẫn y tế cơ bản về việc bảo vệ bản thân trước tình hình dịch
* Là cư dân thành phố, tôi mong muốn ứng dụng có thể cho phép tôi gửi thông báo khẩn cấp đến chính quyền để thông báo về ca nghi nhiễm
* Là cư dân thành phố, tôi mong muốn có thể nhận được thông báo về các ca nhiễm, ca nghi nhiễm ở địa phương
* Là cư dân thành phố, tôi mong muốn ứng dụng cho phép tôi gửi thông báo khẩn để được hỗ trợ khi thiếu các nhu yếu phẩm vì không thể ra ngoài
* Là cư dân thành phố, tôi mong ứng dụng cập nhật đầy đủ về tình trạng tiêm ngừa vacxin của tôi
* Là cư dân thành phố, tôi mong ứng dụng sẽ có tính bảo mật, an toàn cao, không để lộ thông tin cá nhân của tôi ra bên ngoài
* Là cư dân thành phố, tôi mong ứng dụng sẽ gửi cho tôi các chỉ thị, thông báo mới nhất của chính phủ về quy định chống dịch
* Là cư dân thành phố, tôi mong muốn ứng dụng sẽ cập nhật các địa điểm đang bị phong tỏa
* Là cư dân thành phố, tôi mong muốn ứng dụng có chức năng truy vết, thông báo cho tôi nếu như tôi đã tiếp xúc với ca nghi nhiễm.

Vai trò: Cán bộ - 6

* Là cán bộ, tôi mong muốn ứng dụng thông báo cho tôi địa điểm của các cư dân thành phố cần được hỗ trợ về nhu yếu phẩm
* Là cán bộ, tôi mong muốn ứng dụng cho phép tôi định vị các ca nghi nhiễm
* Là cán bộ, tôi mong muốn ứng dụng thông báo cho tôi các địa điểm có nguy cơ bùng dịch để kịp thời lên kế hoạch xử lý
* Là cán bộ, tôi mong muốn ứng dụng cho tôi biết về tình hình tiêm ngừa vacxin ở địa phương (để có thể kịp thời nhắc nhở)
* Là cán bộ, tôi mong muốn ứng dụng thông báo cho tôi các chỉ thị mới nhất từ chính phủ về việc chống dịch
* Là cán bộ, tôi mong muốn ứng dụng có cập nhật các địa điểm cung cấp oxi ở gần nhất để tôi kịp thời hỗ trợ người dân nếu họ cần

Vai trò: Ca nghi nhiễm - 4

* Là ca nghi nhiễm, tôi mong muốn ứng dụng cho phép tôi gửi thông báo khẩn cấp đến cơ sở y tế nếu như tôi có triệu chứng hoặc cần hỗ trợ về y tế
* Là ca nghi nhiễm, tôi mong muốn ứng dụng gợi ý cho tôi các cơ sở y tế gần nhất
* Là ca nghi nhiễm, tôi mong muốn ứng dụng cho phép tôi gửi thông báo khẩn đến chính quyền nếu như tôi cần hỗ trợ về các nhu yếu phẩm
* Là ca nghi nhiễm, tôi mong muốn ứng dụng cho phép tôi gửi yêu cầu hỗ trợ nếu như tôi không có điều kiện để cách ly tại nhà

b/ 3 yêu cầu chức năng và 2 yêu cầu phi chức năng

3 yêu cầu chức năng:

R1: Ứng dụng phải cho phép cư dân thành phố được gửi thông báo khẩn đến chính quyền địa phương để được hỗ trợ khi thiếu nhu yếu phẩm

R2: Ứng dụng phải cập nhật các chỉ thị chống dịch mới nhất từ chính phủ

R3: Ứng dụng phải cập nhật tình trạng tiêm vacxin cho cư dân thành phố

Vẽ Usecase cho R1:

|  |  |
| --- | --- |
| ***Use case ID*** | ***UC01*** |
| Tên Use Case | Gửi thông báo khẩn đến chính quyền |
| Tóm tắt | Cho cư dân thành phố gửi thông báo khẩn đến chính quyền để được hỗ trợ về nhu yếu phẩm |
| Tác nhân | Cư dân thành phố |
| Điều kiện tiên quyết | Phải đăng nhập vào tài khoản trước. |
| Kết quả | Thông báo khẩn sẽ được gửi đến chính quyền, chờ được xử lý |
| Kịch bản chính | 1. Người dùng nhấn vào nút “Hỗ trợ” 2. Ứng dụng sẽ hiển thị một form yêu cầu người dùng nhập đầy đủ các thông tin liên quan 3. Người dùng nhấn vào nút “Gửi” sau khi điền xong 4. Ứng dụng sẽ hiện ra thông báo xác nhận với người dùng 5. Người dùng chọn “Có” để xác nhận 6. Ứng dụng sẽ gửi thông báo đến chính quyền 7. Ứng dụng hiển thị thông báo xác nhận đã gửi thành công |
| Kịch bản phụ | 1. Ở bước 5, nếu người dùng chọn “Không”, ứng dụng sẽ lưu lại form mà không gửi đi.  2. Ở bước 7, nếu đường truyền internet không ổn định, ứng dụng sẽ không thể gửi form và hiện lên thông báo thất bại, đồng thời lưu lại form đã điền |
| Ràng buộc phi chức năng | Giao diện dễ hiểu, sau 30’ làm quen, người dùng có thể sử dụng ứng dụng mà không mắc lỗi. |

2 yêu cầu phi chức năng:

* Ứng dụng có thể sử dụng được trên hệ điều hành Android 9.0 trở lên, hệ điều hành IOS 11.0. hệ điều hành Windows 10, Windows 11, hệ điều hành MacOS
* Ứng dụng phải đáp ứng được yêu cầu của 10.000 người cùng sử dụng trong một lúc
* Sau 2-4h training, cán bộ sẽ có thể sử dụng ứng dụng thành thạo

Câu 2.2

a/ Đề xuất: Kiến trúc Client-Server (giải thích thêm)

Diagram

Description automatically generated

b/ Đề xuất kế hoạch dự án

* Kế hoạch nhân sự:

20 dev: Bao gồm 15 người thực hiện việc cài đặt hệ thống, 5 người phụ trách về quản lý database

3 BA: Các BA sẽ là người lấy yêu cầu thực tế, phân tích và bàn bạc với đội ngũ thiết kế phần mềm

5 Tester: Bao gồm 3 Tester thực hiện kiểm thử trong giai đoạn phát triển, 2 Tester sẽ phụ trách kiểm thử giai đoạn Release [chưa ổn lắm]

* Thời gian phát triển: 6 sprint x 2 tuần = 3 tháng
* Các gian đoạn thực hiện, các mốc thời gian và chuyển giao

Sprint 1: Các BA lấy yêu cầu thực tế và phân tích, bàn bạc với đội ngũ để đưa ra đặc tả ban đầu cho hệ thống

Sprint 2: Bắt đầu phân chia công việc và cài đặt các component riêng lẻ cho từng đối tượng, tiến hành kiểm thử đơn vị (Unit Testing)

Sprint 3: Tích hợp các component và cài đặt để chúng có thể tương tác với nhau, cài đặt các yêu cầu chức năng đã được đưa ra

Sprint 4: Tiếp tục cài đặt yêu cầu chức năng, đảm bảo hệ thống đáp ứng yêu cầu phi chức năng, tiến hành kiểm thử component và kiểm thử hệ thống (Component Testing và System Testing)

Sprint 5: Đưa ứng dụng ra bên ngoài đội ngũ phát triển, thực hiện kiểm thử Release, kiểm tra hệ thống có đáp ứng được yêu cầu và có thể đưa ra sử dụng hay không

Sprint 6: Thực hiện User Testing và bàn giao sản phẩm.