

LAB 3

Câu 3. Phân tích rủi ro và giải pháp khắc phục

Rủi ro	Khả năng xảy ra	Độ ảnh hưởng	Độ khó	Giai đoạn	FMEA	Giải pháp
Crash app do bug	2	3	4	Lập trình phần mềm	24	Chạy công cụ debug để xác định bug nằm ở module nào. Phân cho người làm module đó fix bug.
App không tương thích với OS	3	5	5	Kiểm tra phân tích thiết kế	75	Leader xem lại kiến trúc hệ thống, toàn team ngồi đọc kỹ lại document của ngôn ngữ và hệ thống, các bản cập nhật,
Thiếu hụt nhân lực	3	5	4	Tổ chức thực hiện, Lập trình phần mềm	60	Xây dựng một bản kế hoạch tạm thời để dự án vẫn có thể tiếp tục được triển khai. Leader ưu tiên việc tuyển gấp nhân sự.
Thiếu / dư thừa yêu cầu cho dự án	5	4	3	Khảo sát và phân tích	60	Làm lại bước khảo sát và phân tích.
Lỗi phần cứng server	2	4	4	Lập trình phần mềm	32	Đưa ra bản kế bản kế hoạch tạm thời để dự án vẫn có thể tiếp tục được triển khai ở những phần không bị ảnh hưởng. Ưu tiên mua/ thuê phần cứng mới, khắc phục lỗi server.
Các yêu cầu hoạt động từ bên thứ 3 (thanh toán) bị đình trệ	2	5	5	Lập trình phần mềm	50	Leader cân đối lại ngân sách, làm việc lại với bên thứ 3.

Sửa thiết kế hệ thống	2	5	4	Tổ chức thực hiện, Lập trình phần mềm	45	Tạm dừng tất cả hoạt động, xây dựng lại bảng kế hoạch. Toàn team thảo luận để đưa ra thiết kế hệ thống tối ưu và nhanh nhất.
Yêu cầu hệ thống không rõ ràng	4	4	3	Lập trình phần mềm, Tổ chức thực hiện	48	Cử 1 nhóm đi xác nhận lại các yêu cầu tính năng, phi tính năng của hệ thống.
Thiếu phản hồi từ người dùng cuối	5	3	3	Tổ chức thực hiện, kiểm tra	45	Làm việc lại với khách hàng.
Thiết kế hệ thống không đủ rõ ràng	2	4	3	Phân tích thiết kế sơ đồ mô hình hóa	24	Tạm dừng việc code. Team tập trung phân tích và xây dựng lại thiết kế hệ thống
Bị hacker xâm nhập	3	5	5	Kiểm tra, Lập trình phần mềm, triển khai	75	Rà soát thiệt hại. Tìm hiểu nguyên nhân lỗ hổng. Khắc phục lỗ hổng. Báo cáo lại tiến độ và tình hình xảy ra với khách hàng.
Source code không rõ ràng	5	3	2	Lập trình phần mềm	30	Leader đưa ra đánh giá và yêu cầu nhân viên code phần đó phải khắc phục bằng cách thêm chú thích hoặc code lại theo đúng tiêu chuẩn.
Thiếu thiết bị, hệ điều hành để kiểm thử	5	4	2	Development & Testing	40	Leader đánh giá mức độ phù hợp của thiết bị và hệ điều hành hiện nay. Nếu cần thiết thì project manager đánh giá lại chi phí cho dự án

Một số rủi ro tổng quan về thời gian, con người, chi phí khác và giải pháp khắc phục

STT	Rủi Ro	Nguyên Nhân	Giải pháp
1	Ước lượng thời gian không chính xác	<ul style="list-style-type: none"> • Ước lượng không chính xác xảy ra khi độ dài của một dự án, các cột mốc quan trọng và nhiều yếu tố khác bị nhóm dự án đánh giá thấp. • Ước lượng phần mềm có thể gây ra vấn đề giữa nhà phát triển và khách hàng vì chúng dẫn đến nhiều hệ lụy như thời gian của dự án dẫn đến việc tăng chi phí của dự án. 	<ul style="list-style-type: none"> • Chỉ xây dựng công việc cần ưu tiên trước mắt. • Tính đến Tech Spikes trong ước lượng của bạn (tức là phân bổ thời gian để tập trung vào một số vấn đề kỹ thuật hoặc vấn đề thiết kế trong dự án cần được giải quyết trước và loại bỏ rủi ro đối với một phần đặc biệt phức tạp hoặc không quen thuộc của dự án). • Thêm hệ số phân bổ vào ước lượng (nghĩa là hệ số thời gian được tính toán mà nhóm phát triển dành trong tuần làm việc cho nhiệm vụ bên ngoài dự án). • Xem xét Mô hình nón của sự bất định (Cone of Uncertainty) khi ước lượng phần mềm.
2	Các thay đổi về phạm vi	<ul style="list-style-type: none"> • Xảy ra khi phạm vi của tác vụ thay đổi sau một khung thời gian đã được thống nhất. • Sự phản hồi thường xuyên của khách hàng, của các bên liên quan hoặc chủ sở hữu sản phẩm thường sẽ thường dẫn đến việc thay đổi phạm vi của một dự án. 	<ul style="list-style-type: none"> • Các lần lặp (iteration) ngắn, có thể quản lý được (hoặc sử dụng phương pháp Agile) cho phép có nhiều cơ hội thường xuyên hơn để phản ánh và thay đổi phạm vi dự án. • Xây dựng công việc ưu tiên duy nhất.

3	Tương tác của người dùng cuối	<ul style="list-style-type: none"> Rủi ro này là khi một sản phẩm được tung ra thị trường nhưng người dùng không chấp nhận các thay đổi hoặc xảy ra xung đột giữa những người dùng. 	<ul style="list-style-type: none"> Thử nghiệm và khảo sát người dùng. Làm việc với các nhóm tập trung (focus groups). Phát hành thường xuyên. Thử nghiệm beta.
4	Code chất lượng kém	<ul style="list-style-type: none"> Xảy ra khi các dự án bị đánh giá thấp và các nhà phát triển gấp rút hoàn thành các tác vụ. Code có thể khó đọc, làm cho các developers khác khó có thể xem xét hoặc thực hiện thay đổi. Không được kiểm thử, do đó còn nhiều các lỗi có thể ngăn chặn được. 	<ul style="list-style-type: none"> Xây dựng các Tiêu chí chấp nhận của người dùng (User Acceptance Criteria) để các bên liên quan xác nhận dự án đạt tiêu chuẩn. Đánh giá code. Các tiêu chuẩn và hướng dẫn lập trình rõ ràng. Kiểm tra tất cả các code. Chỉ định một Giám đốc Sản phẩm chuyên trách để giám sát chất lượng của dự án và nắm quyền sở hữu đối với tất cả các bên liên quan về sự thành công và thất bại. Chú ý Way of Working (đã đề cập bên trên).
5	Năng suất kém	<ul style="list-style-type: none"> Khi một nhóm dự án bị tụt lại so với kế hoạch về thời gian, ta có thể cần phải kiểm tra năng suất của nhóm phát triển. 	<ul style="list-style-type: none"> Văn hóa con người của công ty bạn. Lên kế hoạch thời gian có thể đạt được trong dài hạn để tránh tình trạng kiệt sức của nhân viên. Tìm một Giám đốc sản phẩm tham gia trực tiếp và cộng tác với nhóm.

6	Nguồn nhân lực không đủ	<ul style="list-style-type: none"> Xảy ra khi một bên liên quan hoặc thành viên nhóm phát triển phải rời khỏi dự án một cách đột xuất hoặc không xem xét kỹ lưỡng quy mô của dự án dẫn đến sự thiếu hụt nhân lực. 	<ul style="list-style-type: none"> Duy trì cập nhật tài liệu. Cung cấp các tài liệu hướng dẫn đầy đủ cho người mới tham gia. Theo dõi lịch trình làm việc của nhóm một cách thường xuyên.
7	Hiệu suất không đảm bảo	<ul style="list-style-type: none"> Không nắm rõ khả năng của từng người, chưa phân đúng người đúng việc. Các thành viên chưa nắm được rõ yêu cầu của dự án. Không có quy trình phát triển dự án rõ ràng. 	<ul style="list-style-type: none"> Xác định tiêu chuẩn hiệu suất một cách rõ ràng và đưa cho khách hàng review lại. Đảm bảo mọi nhân sự trong dự án nắm được yêu cầu cụ thể về hiệu suất cá nhân. Xây dựng hiệu suất mẫu cho các giao dịch quan trọng. Kiểm thử với dữ liệu mẫu nếu có thể.