

**Plan to build a
project**

1. Quy trình phát triển phần mềm

6 giai đoạn chính

4 vai trò đặc thù

2 điều kiện tiên quyết

1 cái nhìn tổng thể

Giai đoạn 1: Khảo sát dự án

- Bước 1:

+ Khảo sát sơ bộ: tìm hiểu các yếu tố cơ bản (tổ chức, văn hóa, đặc trưng, con người,...) tạo tiền đề để phát triển PM phù hợp với dự án và doanh nghiệp.

+ Khảo sát chi tiết: thu thập thông tin chi tiết của hệ thống (chức năng xử lý, thông tin được phép nhập và xuất khỏi hệ thống, ràng buộc, giao diện cơ bản, nghiệp vụ) phục vụ cho việc phân tích và thiết kế.

- Bước 2: Đặt ra các vấn đề trọng tâm cần phải giải quyết, như:

Thông tin đưa vào hệ thống phải như thế nào?

Dữ liệu hiển thị và xuất ra khác nhau ở những điểm nào?

Ràng buộc giữa các đối tượng trong hệ thống cần xây được dựng ra sao?

Chức năng và quy trình xử lý của hệ thống phải đảm bảo những yêu cầu nào? Cần sử dụng những giải pháp nào? Tính khả thi của từng giải pháp ra sao?

Thực hành: Khảo sát một dự án thực tế

- Website cho ai, nhằm mục đích gì?
- Phong cách?
- Màu sắc?
- Thời gian? Chi phí?
- Các tính năng?
- Quản lý hệ thống?
- Ngôn ngữ nào phù hợp, framework nào hợp lý,...

Giai đoạn 2: Phân tích hệ thống

- Xác định yêu cầu gồm: các chức năng chính – phụ; nghiệp vụ cần phải xử lý đảm bảo tính chính xác, tuân thủ đúng các văn bản luật và quy định hiện hành; đảm bảo tốc độ xử lý và khả năng nâng cấp trong tương lai.
- Phân tích và đặc tả mô hình phân cấp chức năng tổng thể thông qua sơ đồ BFD (Business Flow Diagram), từ mô hình BFD sẽ tiếp tục được xây dựng thành mô hình luồng dữ liệu DFD (Data Flow Diagram) thông qua quá trình phân rã chức năng theo các mức 0, 1, 2 ở từng ô xử lý.
- Phân tích bảng dữ liệu. Cần đưa vào hệ thống những bảng dữ liệu (data table) gồm các trường dữ liệu (data field) nào? Xác định khóa chính (primary key), khóa ngoại (foreign key) cũng như mối quan hệ giữa các bảng dữ liệu (relationship) và ràng buộc (constraint) dữ liệu cần thiết.

Thực hành: Phân tích hệ thống thực tế

- Các chức năng chính phụ ra sao, đặc tả sản phẩm như thế nào?
- Biểu đồ phân rã chức năng? Các diagrams?
- Các bảng dữ liệu cần có + mối liên hệ?

Giai đoạn 3: Thiết kế

Hai bước cơ bản của thiết kế:

- **Thiết kế tổng thể**

Trên cơ sở các bảng dữ liệu đã phân tích và đặc tả sẽ được thiết kế dưới dạng mô hình ý tưởng bằng các phần mềm chuyên dụng như Sybase PowerDesigner, CA ERwin Data Modeler,...

- **Thiết kế chi tiết** (nhiều công đoạn)

- ✓ **Thiết kế cơ sở dữ liệu (Database):** Với mô hình mức vật lý hoàn chỉnh ở giai đoạn thiết kế đại thể sẽ được kết sinh mã thành file sql.
- ✓ **Thiết kế truy vấn, thủ tục, hàm:** thu thập, xử lý thông tin nhập và đưa ra thông tin chuẩn xác theo đúng nghiệp vụ.
- ✓ **Thiết kế giao diện** chương trình đảm bảo phù hợp với môi trường, văn hóa và yêu cầu của doanh nghiệp thực hiện dự án.
- ✓ **Thiết kế chức năng** chương trình đảm bảo tính logic trong quá trình nhập liệu và xử lý cho người dùng.
- ✓ **Thiết kế báo cáo.** Dựa trên các yêu cầu của mỗi doanh nghiệp và quy định hiện hành sẽ thiết kế các mẫu báo cáo phù hợp hoặc cho phép doanh nghiệp tự tạo mẫu báo cáo ngay trên hệ thống.
- ✓ **Thiết kế các kiểm soát** bằng hình thức đưa ra các thông báo, cảnh báo hoặc lỗi cụ thể tạo tiện lợi và kiểm soát chặt chẽ quá trình nhập liệu với mục tiêu tăng độ chính xác cho dữ liệu.

Thực hành: Thiết kế một mô hình cụ thể

- Database?
- Model diagram?
- Pattern design
- Class diagram?
- Giao diện với Figma, AI, Photoshop,...

Giai đoạn 4: Thực hiện

- Lựa chọn Database, Framework, Server,
- Dựng Frontend
- Xây dựng Backend
- Số người tham gia, vai trò mỗi người,...

Giai đoạn 5: Kiểm thử

- ❖ Kiểm chứng các modules chức năng của hệ thống thông tin, chuyển các thiết kế thành các chương trình (phần mềm).
- ❖ Thử nghiệm hệ thống thông tin.
- ❖ Khắc phục các lỗi (nếu có).
- ❖ Viết test case theo yêu cầu.

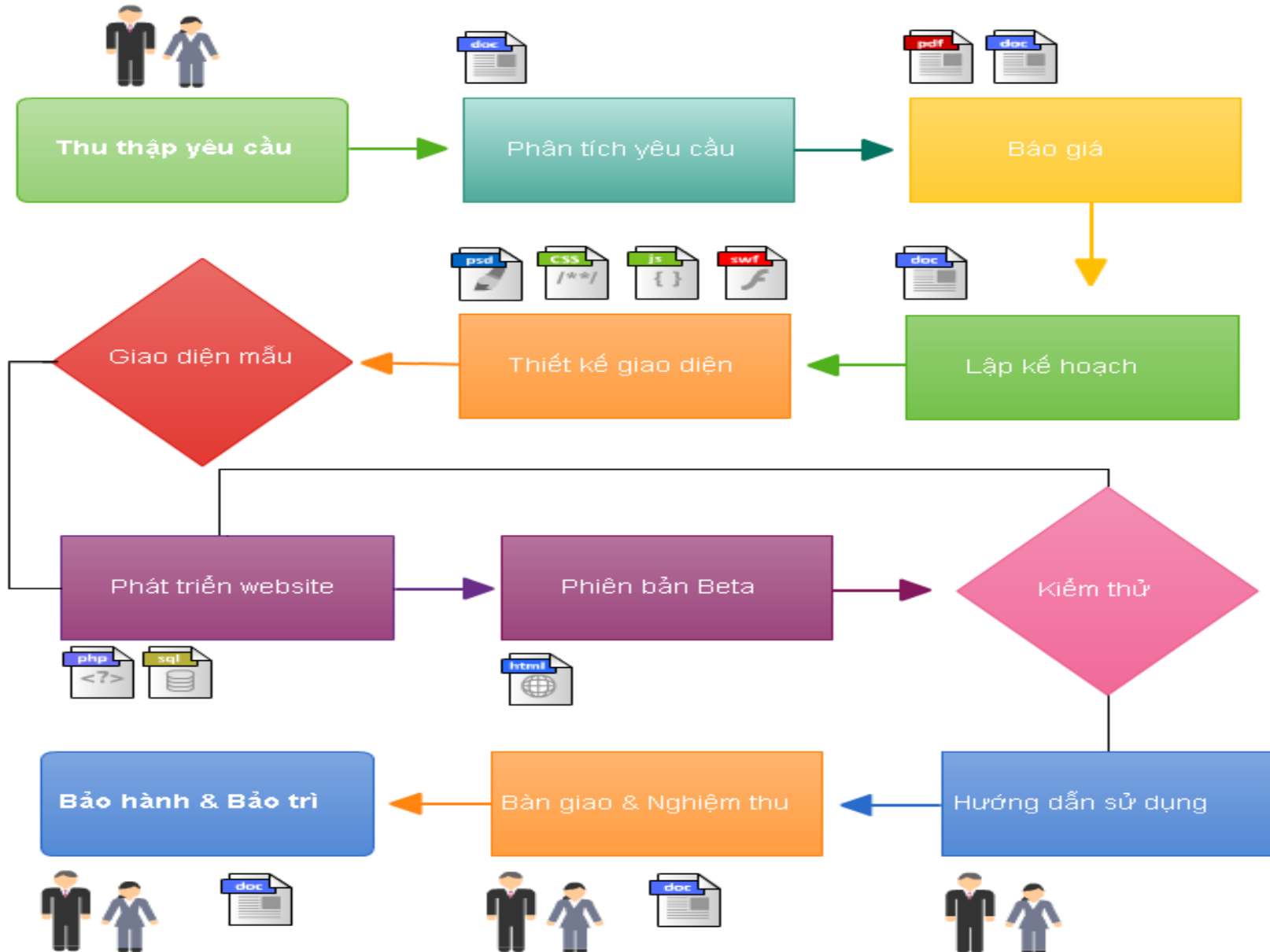
Giai đoạn 6: Triển khai và bảo trì

- Lắp đặt phần cứng để làm cơ sở cho hệ thống.
- Cài đặt phần mềm.
- Chuyển đổi hoạt động của hệ thống cũ sang hệ thống mới, gồm có: chuyển đổi dữ liệu; bố trí, sắp xếp người làm việc trong hệ thống; tổ chức hệ thống quản lý và bảo trì,...
- Phát hiện các sai sót, khuyết điểm của hệ thống thông tin.
- Đào tạo và hướng dẫn sử dụng.
- Cải tiến và chỉnh sửa hệ thống thông tin.
- Bảo hành.
- Nâng cấp chương trình khi có phiên bản mới.

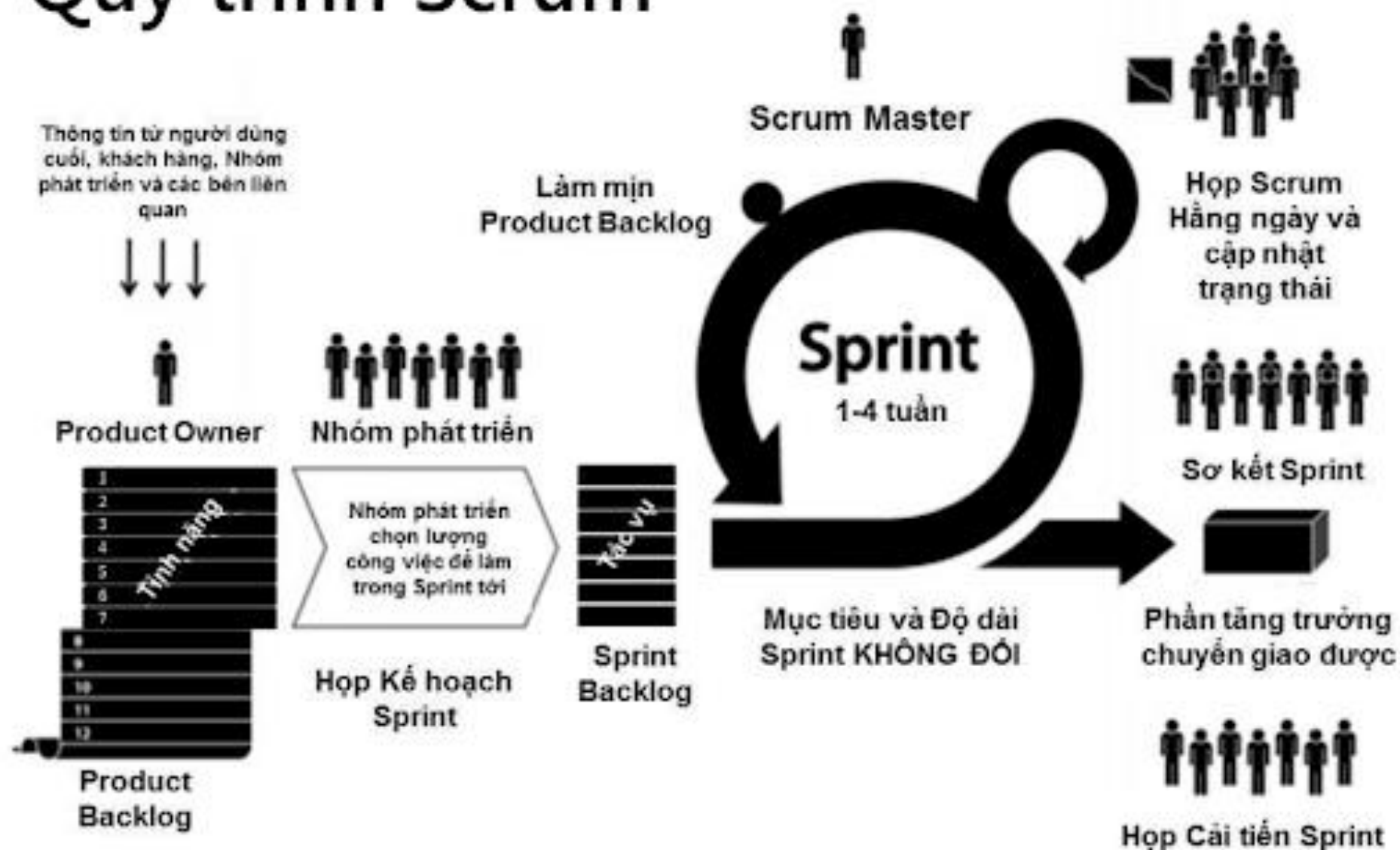
2. Xây dựng kế hoạch sản xuất website

- ❖ Xác định khách hàng.
- ❖ Xác định mục đích thiết kế
- ❖ Xác định đối tượng truy cập vào website.
- ❖ Tìm địa chỉ cung cấp tên miền, hosting.
- ❖ Xây dựng sơ đồ, bố cục website.
- ❖ Chuẩn bị đồ họa, hình ảnh cần thiết.
- ❖ Lên kế hoạch thời gian thực hiện.
- ❖ Chuẩn bị nội dung cho website.

3. Quy trình phát triển phần mềm



Quy trình Scrum



Thực hành: Tạo nhóm theo quy trình phát triển phần mềm

- Nhóm có bao nhiêu người?
- Tên của nhóm là gì?
- Vai trò, nhiệm vụ của mỗi người?
- Quy trình làm việc như thế nào?
- Tại sao lại là những người bạn này?

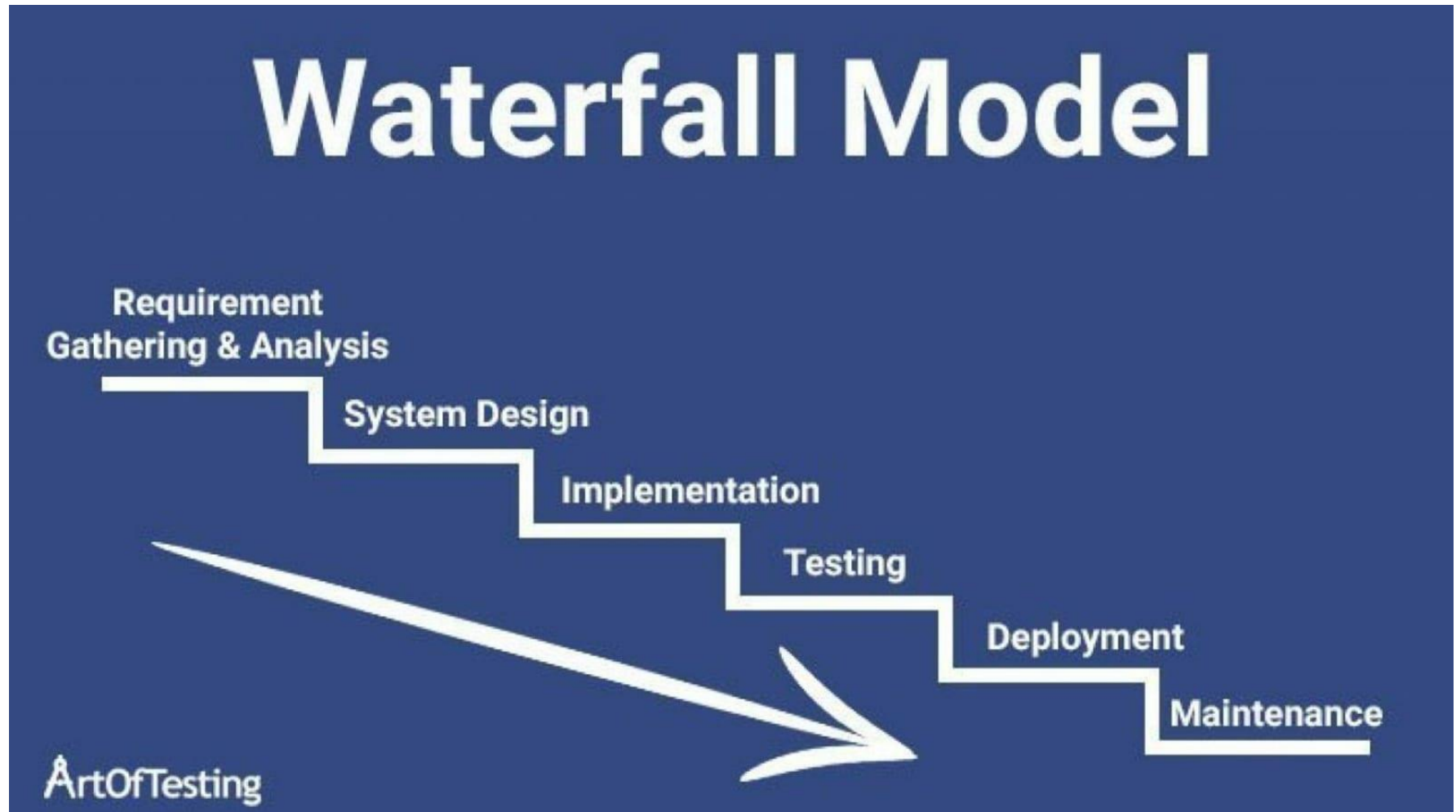
Lưu ý: Đây sẽ là nhóm làm dự án cuối khóa, vậy nên hãy lập một team thực chiến thật tốt!

Chúc các bạn may mắn!

Một vài mô hình phát triển phần mềm

- Waterfall
- V-Model
- Scrum

1. Waterfall



Mô hình này gồm các giai đoạn xử lý nối tiếp nhau như sau:

- . **Thu thập yêu cầu (Requirement gathering):** Đây là giai đoạn xác định các yêu cầu chức năng và phi chức năng mà hệ thống phần mềm cần có. Kết quả của giai đoạn này là bản tài liệu đặc tả yêu cầu. Tài liệu này sẽ là nền tảng cho những giai đoạn tiếp theo cho đến cuối dự án.
- . **Phân tích hệ thống (System Analysis):** Là giai đoạn định ra làm thế nào để hệ thống phần mềm đáp ứng đúng yêu cầu của khách hàng. Giai đoạn này thực hiện phân tích, thiết kế hệ thống phần mềm.
- . **Coding:** Là giai đoạn thực hiện sản phẩm dựa trên đặc tả yêu cầu và tài liệu thiết kế module.
- . **Testing:** Tester sẽ nhận sản phẩm từ developer và thực hiện kiểm thử cho nhóm các thành phần và kiểm thử hệ thống. Khâu kiểm thử cuối cùng sẽ là Kiểm thử chấp nhận, giai đoạn này còn có sự tham gia của khách hàng.
- . **Implementation:** Triển khai hệ thống ra môi trường của khách hàng.
- . **Operations & Maintenance:** Đây là giai đoạn cài đặt, cấu hình và đào tạo cho khách hàng. Giai đoạn này sửa chữa những lỗi của sản phẩm (nếu có) và phát triển những thay đổi mới được khách hàng yêu cầu.

Đặc điểm:

Thường áp dụng cho các phần mềm có quy mô vừa và nhỏ.

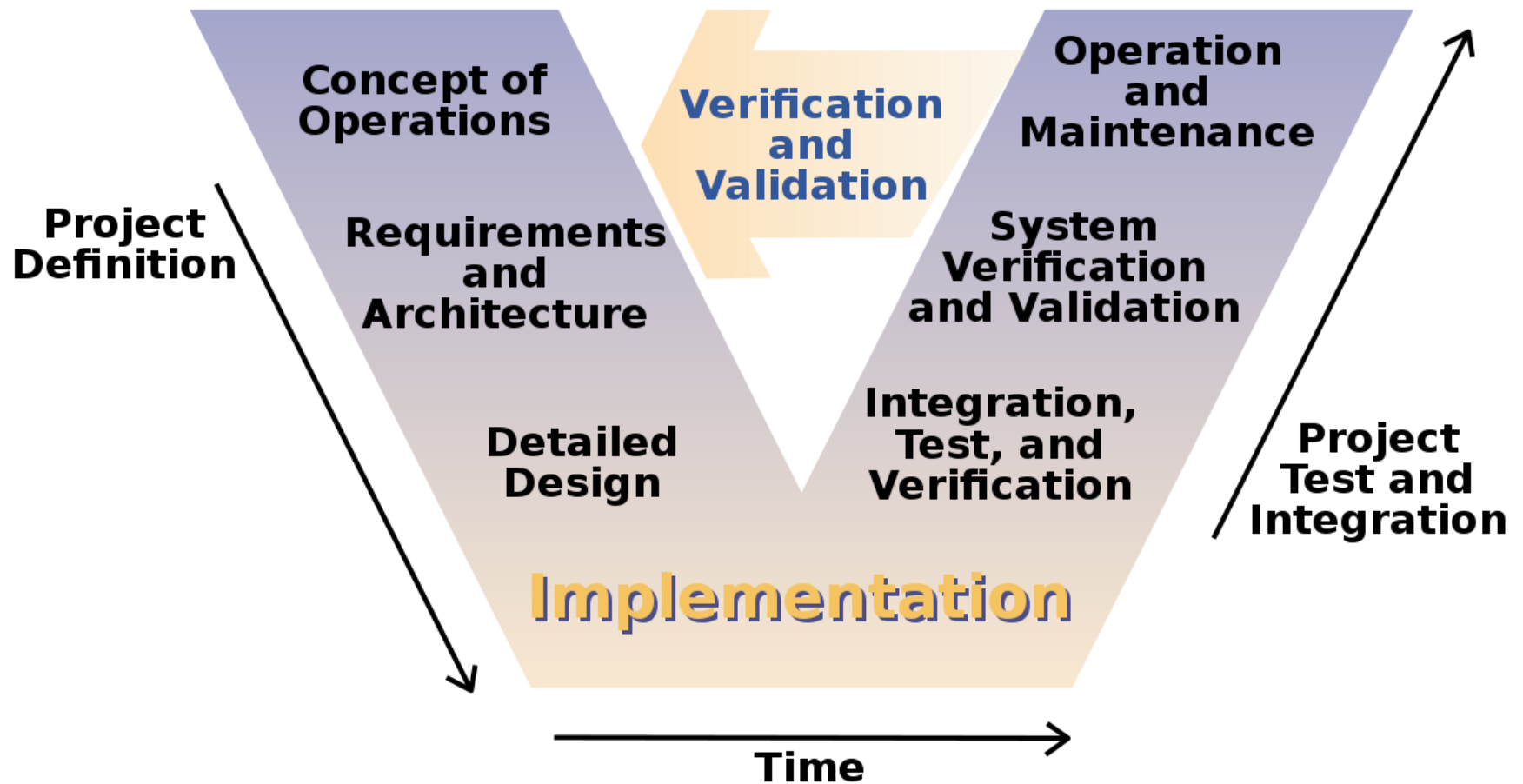
Các dự án có yêu cầu rõ ràng, ít thay đổi.

Nguồn lực được đào tạo và sẵn sàng.

Ưu điểm: Vì có yêu cầu rõ ràng nên dễ hiểu, dễ áp dụng. Dễ phân công công việc, bố trí, giám sát

Nhược điểm: Thực tế cho thấy rằng đến những giai đoạn cuối cùng của dự án mới có khả năng nhận ra sai sót trong những giai đoạn trước để có thể quay lại sửa chữa.

2. V-Model



- Hoạt động tốt với các dự án có quy mô vừa và nhỏ.
- Dễ dàng quản lý vì mỗi giai đoạn có các mục tiêu và mục tiêu được xác định rõ ràng.
- Toàn bộ quy trình được chia thành 2 nhóm giai đoạn tương ứng nhau là phát triển và kiểm thử. Mỗi giai đoạn phát triển sẽ tiến hành song song với một giai đoạn kiểm thử tương ứng. Do đó, các lỗi được phát hiện sớm ngay từ đầu.

Ưu điểm

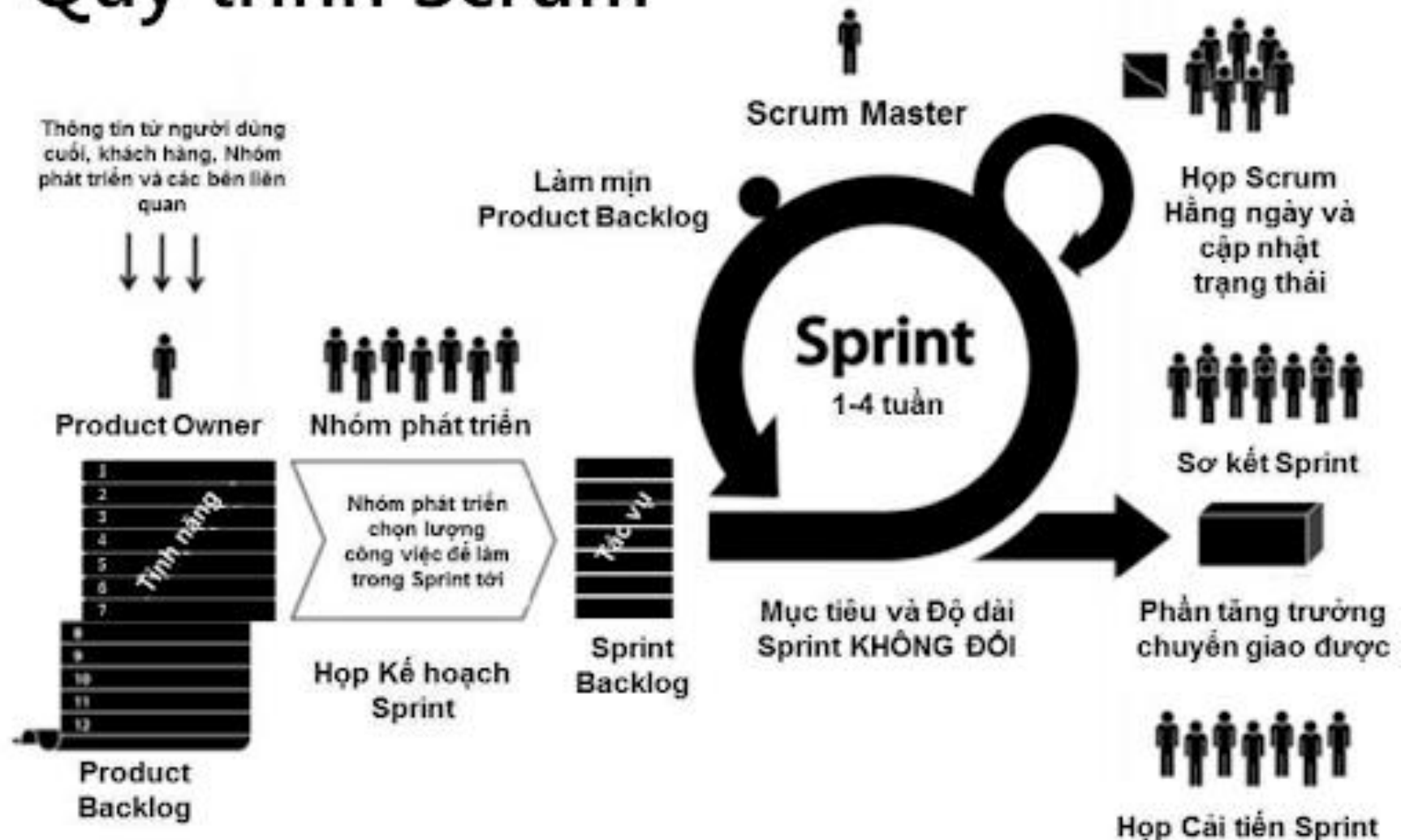
Ngay từ lúc nhận được tài liệu đặc tả yêu cầu, tester sẽ tham gia vào review tài liệu đặc tả yêu cầu sau đó lên kế hoạch và thực hiện viết test case. Lỗi được phát hiện từ giai đoạn này sẽ ít tốn thời gian và chi phí hơn các giai đoạn sau.

Nhược điểm

Trong mô hình chữ V, các yêu cầu vẫn được đưa vào thực hiện cùng 1 lúc mà rủi ro về thay đổi yêu cầu từ phía khách hàng là rất lớn. Do đó, mô hình này vẫn có thể gặp rắc rối khi khách hàng thường xuyên thay đổi yêu cầu.

3. Scrum

Quy trình Scrum



Product backlog: Đây là danh sách ưu tiên các tính năng (feature) hoặc đầu ra khác của dự án, có thể hiểu như là danh sách yêu cầu (requirement) của dự án. Product Owner chịu trách nhiệm sắp xếp độ ưu tiên cho từng hạng mục (Product Backlog Item) trong Product Backlog dựa trên các giá trị do Product Owner định nghĩa

Sprint backlog: Đây là bản kế hoạch cho một Sprint, là kết quả của buổi họp lập kế hoạch (Sprint Planning). Với sự kết hợp của Product Owner, nhóm sẽ phân tích các yêu cầu theo độ ưu tiên từ cao xuống thấp để hiện thực hóa các hạng mục trong Product Backlog dưới dạng danh sách công việc (TODO list).

- Product Owner tạo ra Product Backlog chứa các yêu cầu của dự án với các hạng mục được sắp theo thứ tự ưu tiên. Đội sản xuất sẽ thực hiện việc hiện thực hóa dần các yêu cầu của Product Owner với sự lặp đi lặp lại các giai đoạn từ 1 đến 4 tuần làm việc (gọi là Sprint) với đầu vào là các hạng mục trong Product Backlog, đầu ra là các gói phần mềm hoàn chỉnh có thể chuyển giao được (Potentially Shippable Product Increment)

- Đội sản xuất cùng họp với Product Owner để lập kế hoạch cho từng Sprint. Kết quả của buổi lập kế hoạch (theo cách làm của Scrum) là Sprint Backlog chứa các công việc cần làm trong suốt một Sprint.
- Các Sprint sẽ được lặp đi lặp lại cho tới khi nào các hạng mục trong Product Backlog đều được hoàn tất
- Trong suốt quá trình phát triển, nhóm sẽ phải cập nhật Sprint Backlog và thực hiện công việc họp hằng ngày (Daily Scrum) để chia sẻ tiến độ công việc cũng như các vướng mắc trong quá trình làm việc cùng nhau. Nhóm được trao quyền để tự quản lý và tổ chức lấy công việc của mình để hoàn thành công việc trong Sprint.
- Khi kết thúc Sprint, nhóm tạo ra các gói phần mềm có chức năng hoàn chỉnh, sẵn sàng chuyển giao (shippable) cho khách hàng. Buổi họp Sơ kết Sprint (Sprint Review) ở cuối Sprint sẽ giúp khách hàng thấy được nhóm đã có thể chuyển giao những gì, còn những gì phải làm hoặc còn gì phải thay đổi hay cải tiến.
- Sau khi kết thúc việc đánh giá Sprint, Scrum Master và nhóm cùng tổ chức họp Cải tiến Sprint (Sprint Retrospective) để tìm kiếm các cải tiến trước khi Sprint tiếp theo bắt đầu, điều này sẽ giúp nhóm liên tục học hỏi và trưởng thành qua từng Sprint.

Bao gồm 4 cuộc họp như sau:

1. Sprint Planning (Họp Kế hoạch Sprint): Nhóm phát triển họp với Product Owner để lên kế hoạch làm việc cho một Sprint. Công việc lập kế hoạch bao gồm việc chọn lựa các yêu cầu cần phải phát triển, phân tích và nhận biết các công việc phải làm kèm theo các ước lượng thời gian cần thiết để hoàn tất các tác vụ. Scrum sử dụng cách thức lập kế hoạch từng phần và tăng dần theo thời gian, theo đó, việc lập kế hoạch không diễn ra duy nhất một lần trong vòng đời của dự án mà được lặp đi lặp lại, có sự thích nghi với các tình hình thực tiễn trong tiến trình đi đến sản phẩm.

2. Daily Scrum (Họp Scrum hằng ngày): Scrum Master tổ chức cho Đội sản xuất họp hằng ngày trong khoảng 15 phút để Nhóm Phát triển chia sẻ tiến độ công việc Trong cuộc họp này, từng người trong nhóm phát triển lần lượt trình bày để trả lời 3 câu hỏi sau:

- Hôm qua đã làm gì?
- Hôm nay sẽ làm gì?
- Có khó khăn trở ngại gì không?

3. Sprint Review (Họp Sơ kết Sprint): Cuối Sprint, nhóm phát triển cùng với Product Owner sẽ rà soát lại các công việc đã hoàn tất (DONE) trong Sprint vừa qua và đề xuất các chỉnh sửa hoặc thay đổi cần thiết cho sản phẩm.

4. Sprint Retrospective (Họp Cải tiến Sprint): Dưới sự trợ giúp của Scrum Master, nhóm phát triển sẽ rà soát lại toàn diện Sprint vừa kết thúc và tìm cách cải tiến quy trình làm việc cũng như bản thân sản phẩm.

Bao gồm 3 vai trò:

1. Product Owner: Là người chịu trách nhiệm về sự thành công dự án, người định nghĩa các yêu cầu cho sản phẩm và đánh giá đầu ra cuối cùng của các nhà phát triển phần mềm.

2. Scrum Master: Là người đảm bảo các sprint được hoàn thành theo đúng quy trình Scrum, giúp đỡ loại bỏ các trở ngại cho đội dự án.

3. Development Team: Là tập hợp của từ 5 đến 9 thành viên chịu trách nhiệm trực tiếp tham gia sản xuất. Tùy theo quy mô của dự án để bố trí số thành viên cho phù hợp.

Ưu điểm

- . Phù hợp với các yêu cầu / nghiệp vụ hay thay đổi, hoặc hệ thống nghiên cứu do làm theo từng giai đoạn ngắn ngày, có thể nhìn thấy những rủi ro hay những điểm chưa phù hợp để thay đổi.

Nhược điểm

- . Thiếu sự nhân mạnh về thiết kế và tài liệu cần thiết
- . Quy mô nhân lực thường giới hạn , sẽ có trở ngại lớn nếu nguồn nhân lực yêu cầu vượt quá con số này ví dụ trong các cuộc họp trao đổi.
- . Yêu cầu nguồn nhân lực phải có kiến thức và am hiểu về Agile

Tasks Management with Trello

Version Control with Github

Rủi do trong dự án phát triển phần mềm

Rủi ro là yếu tố luôn tồn tại trong mọi hoạt động sản xuất và kinh doanh, và dự án phần mềm cũng không ngoại lệ. Tuy nhiên, với đặc thù riêng của mình, nhận diện và kiểm soát rủi ro trong dự án phần mềm là điều không đơn giản. Trong thực tế, nhiều dự án phần mềm đã bỏ qua hoặc kiểm soát rủi ro sơ sài, chiếu lệ dẫn đến kết quả thất bại, khách hàng phàn nàn về chất lượng hoặc lỗ vốn do chi phí tăng cao.

Có 7 loại rủi ro cơ bản:

1. Thiếu nhân lực được huấn luyện về kỹ thuật

Đây là rủi ro dễ thấy ngay từ khi dự án mới bắt đầu. Bởi vậy, nhà quản lý dự án có thể chủ động đưa ra giải pháp thay vì bị động như các rủi ro khách quan khác.

Cách khắc phục:

- . Dành một phần chi phí để đào tạo nhân sự ngay từ ban đầu
- . Dành bộ đệm cho nhân sự bổ sung
- . Lập một chương trình đào tạo riêng cho dự án
- . Tổ chức các buổi đào tạo chéo giữa các thành viên tay nghề cứng với người mới
- . Yêu cầu thay đổi quá nhiều

2. Rủi ro đến từ khách hàng

Rủi ro này đến từ phía khách hàng, khi họ đưa ra các thay đổi không theo trật tự nào. Nhiều nhà quản lý dự án do cuốn theo các yêu cầu thất thường này khiến dự án bị rối tung.

Cách khắc phục:

- . Có hợp đồng, chữ ký bằng văn bản của khách hàng cho các yêu cầu cụ thể khi nhận dự án.
- . Cảnh báo khách hàng về nguy cơ chậm tiến độ nếu làm theo các thay đổi đột xuất.
- . Xây dựng quy trình cụ thể, thủ tục rõ ràng (chữ ký của các đầu mối quan trọng liên quan) nếu muốn thay đổi.
- . Đàm phán rõ ràng để nhận thù lao tương xứng với những thay đổi ngoài hợp đồng.

3. Yêu cầu không rõ ràng

Cách khắc phục:

- . Lập một số trường hợp giả định để đưa cho khách hàng tham khảo, từ đó làm theo mẫu có sẵn.
- . Phát triển mẫu phần mềm thử nghiệm (prototype), đưa ra yêu cầu mô tả chi tiết và gửi tới khách hàng

4. Nhân sự rời dự án

Cách khắc phục:

- . Đảm bảo mọi nhân sự đều được đảm nhiệm phần việc quan trọng của dự án
- . Họp mặt thường niên (ngoài công việc) để xây dựng tình đoàn kết, teamwork
- . Thỉnh thoảng xoay vòng nhân sự vào các mảng việc khác nhau
- . Có từ 1,2 nhân sự back up kịp thời cho dự án bất cứ lúc nào
- . Có cơ chế lưu trữ, quản lý tài liệu công việc của tất cả nhân sự

5. Nhiều quyết định từ bên ngoài tác động lên dự án

Trong quá trình thực hiện dự án, một số quyết định khách quan (luật pháp, đối tác, quy chế công ty...) có thể tác động tiêu cực lên dự án.

Cách khắc phục:

- . Liệt kê danh sách các quyết định bất lợi, cùng với đó là thông tin liên quan.

- . Bàn bạc với nhân viên chịu ảnh hưởng trực tiếp bởi các quyết định này.
- . Nếu cần, xác định đây như một rủi ro bắt buộc và lên kế hoạch giảm thiểu trong dài hạn.

6. Hiệu suất không đảm bảo

Cách khắc phục:

- . 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.

7. Loại rủi ro: Schedule

Phân tích các nguyên nhân gây ra :

- . Lịch trình không thực tế, chỉ có “trường hợp tốt nhất”
- . Nhiệm vụ quan trọng bị xót từ lịch trình
- . Một sự chậm trễ trong một nhiệm vụ gây ra sự chậm trễ các nhiệm vụ phụ thuộc trong dự án
- . Các công việc không quen thuộc của các sản phẩm mất nhiều thời gian hơn dự kiến để thiết kế và thực hiện

Như vậy, chúng ta cần phải lưu ý:

- . Dự án càng lớn thì rủi ro càng nhiều.
- . Việc dự báo rủi ro phụ thuộc vào kinh nghiệm của người Quản lý dự án
- . Kiểm soát rủi ro không nhằm loại bỏ rủi ro, chỉ nhằm hạn chế tối thiểu thiệt hại của rủi ro.
- . Không thể loại trừ được triệt để
- . Không phải cứ tập trung hết sức để ngăn chặn và đề phòng rủi ro đã là tốt, vì có thể phải trả giá đắt, nếu rủi ro không xảy ra. Do đó, cần dự báo rủi ro chính xác.

PHÂN CÔNG CÔNG VIỆC

Thường gặp một số trường hợp sau:

- . **Phân công không đồng đều:** Khi phân công nhiệm vụ qua giao tiếp, nhà quản lý sẽ không có được cái nhìn tổng quát về khối lượng công việc đã chia cho các phòng ban cũng như các nhân viên. Do đó có cá nhân sẽ phải làm nhiều hơn, có cá nhân lại vô cùng nhẹ nhàng. Như vậy không chỉ gây ảnh hưởng tới tiến độ công việc mà còn làm giảm sút tinh thần làm việc.
- . **Phân công thiếu nhiệm vụ:** Số lượng công việc quá nhiều đôi khi cũng làm nhà quản lý lâm vào tình thế thiếu sót. Đôi khi họ tưởng rằng mọi thứ đã được dàn xếp ổn thỏa và đang trong quá trình thực hiện, nhưng thực tế vẫn còn vài đầu công việc còn sót lại và chưa ai đảm nhận. Điều này là vô cùng tai hại, có thể làm hỏng tất cả sự cố gắng, nỗ lực của nhân viên, làm giảm uy tín và doanh thu doanh nghiệp.

- . **Quên nhiệm vụ đã phân công:** Khi nhà quản lý phân công công việc một các thiếu hệ thống, rất có thể họ sẽ sớm quên nhân viên nào đảm nhận nhiệm vụ nào. Từ đó họ sẽ không có sự giao tiếp, kiểm soát tiến độ và chất lượng về đầu công việc đó, làm giảm sút hiệu quả.
- . **Kiểm soát tiến độ làm việc rời rạc:** Phân công công việc trực tiếp có thể truyền được tinh thần làm việc từ ban đầu, tuy nhiên trong quá trình thực hiện, nhà quản lý khó có thể kiểm soát được tiến độ của nó. Bởi thật mất thời gian khi phải hỏi từng cá nhân, từng phòng ban về tiến độ công việc và hệ thống nó với nhau để báo cáo lại với đối tác, khách hàng. Hơn nữa, cá nhân người quản trị cũng có rất nhiều việc riêng cần phải giải quyết, ngoài nhiệm vụ kiểm soát tiến độ làm việc của cấp dưới.

Để giải quyết vấn đề, chúng ta có thể:

Xây dựng mục tiêu và kế hoạch công việc

Làm việc triền miên mà không biết đích đến thì bạn sẽ chẳng bao giờ hoàn thành tốt công việc được. Trước khi bắt tay vào thực hiện, đặt ra một mục tiêu cụ thể cũng như lập kế hoạch thực hiện sẽ giúp nhân viên nắm bắt được tinh thần làm việc sao cho kịp tiến độ và hiệu quả nhất. Bước này đặc biệt có ích khi bạn cần phải thực hiện nhiều công việc và phải phối hợp giữa nhiều phòng ban trong cùng một thời điểm. Nhờ việc hoạch định mục tiêu và kế hoạch, các bộ phận sẽ không bị nhỡ nhịp mà hoạt động trơn tru, suôn sẻ hơn.

Phân chia ra từng giai đoạn cụ thể cho công việc

Với mỗi một công việc, đặc biệt là những dự án dài hơi, việc phân chia ra từng giai đoạn là điều vô cùng cần thiết: Giai đoạn này cần phải hoàn thành những công việc nào? Cần dành bao nhiêu thời gian cho từng giai đoạn?

Công đoạn này sẽ tạo sự rõ ràng, rành mạch để dễ dàng theo dõi công việc. Đồng thời giảm thiểu sai sót, nhầm lẫn trong quá trình hoạt động.

Đánh giá kết quả làm việc theo từng giai đoạn

Thực hiện đánh giá công việc theo từng giai đoạn để luôn kiểm soát được chất lượng công việc cũng như tiến độ có theo kịp mục tiêu đã đề ra hay chưa. Khi ấy, sai sót trong quá trình làm việc sẽ được phát hiện nhanh chóng, giúp bạn có được đánh giá khách quan nhất và đưa ra phương pháp điều chỉnh kịp thời, đảm bảo không ảnh hưởng lâu dài và làm giảm chất lượng của kết quả cuối cùng.

Lưu trữ hồ sơ làm việc cụ thể

Đây là bằng chứng ghi lại quá trình thực hiện công việc của các nhân viên và phòng ban. Trong hồ sơ làm việc có thể có những văn bản, những quy chế, quyết định quan trọng liên quan đến công việc. Đây sẽ là nguồn tham khảo cho các công việc trong tương lai hoặc là bản đối chứng với những thắc mắc của nhà quản lý.

LÊN KẾ HOẠCH CHI PHÍ PHÁT TRIỂN?

1. Chi phí thuê Hosting, Domain, Server
2. Chi phí thiết kế và phát triển (Từng page + chức năng)
3. Các chi phí khác
 - Tối ưu hóa công cụ tìm kiếm (SEO)
 - Tạo landing page
 - Chi phí bảo trì
 - Mua phần mềm bảo mật an ninh nâng cao,...
 - Tích hợp thanh toán, hệ thống mail,...

DANH MỤC CÁC THIẾT BỊ CẦN DÙNG