**BÀI THỰC HÀNH 1.2**

1. Hãy nêu quy trình của các mô hình lặp, nêu ưu điểm và nhược điểm của các mô hình.

* Có các mô hình sau:

1. Mô hình thác nước ( Waterfall model)
2. Mô hình xoắn ốc ( Spiral model)
3. Mô hình agile
4. Mô hình tiếp cận lặp ( Iterative model)
5. Mô hình tăng trưởng ( Incremental model)
6. Mô hình chữ V ( V model)
7. Mô hình Scrum
8. RAD model ( Rapid Application Development)
9. Mô hình thác nước ( Waterfall model):

* Đây được coi như là mô hình phát triển phần mềm đầu tiên được sử dụng.
* Mô hình này áp dụng tuần tự các giai đoạn của phát triển phần mềm.
* Đầu ra của giai đoạn trước là đầu vào của giai đoạn sau. Giai đoạn sau chỉ được thực hiện khi giai đoạn trước đã kết thúc. Đặc biệt không được quay lại giai đoạn trước để xử lý các yêu cầu khi muốn thay đổi.
* Thường dùng cho các dự án nhỏ , ngắn hạn.
* Các dự án có ít thay đổi về yêu cầu và không có những yêu cầu không rõ ràng.

Phân tích mô hình:

* **Requirement gathering**: Thu thập và phân tích yêu cầu được ghi lại vào tài liệu đặc tả yêu cầu trong giai đoạn này.
* **System Analysis**: Phân tích thiết kế hệ thống phần mềm, xác định kiến trúc hệ thống tổng thể của phần mềm.
* **Coding**: Hệ thống được phát triển theo từng unit và được tích hợp trong giai đoạn tiếp theo. Mỗi Unit được phát triển và kiểm thử bởi dev được gọi là Unit Test.
* **Testing**: Cài đặt và kiểm thử phần mềm. Công việc chính của giai đoạn này là kiểm tra và sửa tất cả những lỗi tìm được sao cho phần mềm hoạt động chính xác và đúng theo tài liệu đặc tả yêu cầu.
* **Implementation**: Triển khai hệ thống trong môi trường khách hàng và đưa ra thị trường.
* **Operations and Maintenance**: Bảo trì hệ thống khi có bất kỳ thay đổi nào từ phía khách hàng, người sử dụng.
  1. **Ưu điểm**
* Dễ sử dụng, dễ tiếp cận, dễ quản lý.
* Sản phẩm phát triển theo các giai đoạn được xác định rõ ràng.
* Xác nhận ở từng giai đoạn, đảm bảo phát hiện sớm các lỗi.
  1. **Nhược điểm**
* Ít linh hoạt, phạm vi điều chỉnh hạn chế.
* Rất khó để đo lường sự phát triển trong từng giai đoạn.
* Mô hình không thích hợp với những dự án dài, đang diễn ra, hay những dự án phức tạp, có nhiều thay đổi về yêu cầu trong vòng đời phát triển.
* Khó quay lại khi giai đoạn nào đó đã kết thúc.

1. Mô hình xoắn ốc:

* Là mô hình kết hợp giữa các tính năng của mô hình prototyping và mô hình thác nước.
* Mô hình xoắn ốc được ưa chuộng cho các dự án lớn, đắt tiền và phức tạp.
* Mô hình này sử dụng những giai đoạn tương tự như mô hình thác nước, về thứ tự, plan, đánh giá rủi ro, …
* Mô hình này thường được sử dụng cho các ứng dụng lớn và các hệ thống được xây dựng theo các giai đoạn nhỏ hoặc theo các phân đoạn.

### Phân tích mô hình:

* **Objective identification- Thiết lập mục tiêu**: xác định mục tiêu, đối tượng cho từng pha của dự án.
* **Alternate evaluation- Đánh giá và giảm thiểu rủi ro**: đánh giá rủi ro và thực hiện các hành động để giảm thiểu rủi ro.
* **Product development- Phát triển sản phẩm**: Lựa chọn mô hình phù hợp để phát triển hệ thống.
* **Next phase planning- Lập kế hoạch**: đánh giá dự án và lập kế hoạch cho pha tiếp theo.

### Ưu điểm

* Tốt cho các hệ phần mềm quy mô lớn.
* Dễ kiểm soát các mạo hiểm ở từng mức tiến hóa.
* Đánh giá thực tế hơn như là một quy trình làm việc, bởi vì những vấn đề quan trọng đã được phát hiện sớm hơn.

### Nhược điểm

* Manager cần có kỹ năng tốt để quản lý dự án, đánh giá rủi ro kịp thời.
* Chi phí cao và mất nhiều thời gian để hoàn thành dự án.
* Phức tạp và không thích hợp với các dự án nhỏ và ít rủi ro.
* Yêu cầu thay đổi thường xuyên dẫn đến lặp vô hạn.
* Chưa được dùng rộng rãi.

1. Mô hình Agile:

* Agile là một phương pháp phát triển phần mềm linh hoạt để làm sao đưa sản phẩm đến tay người dùng càng nhanh càng tốt và được xem như là sự cải tiến so với những mô hình cũ như mô hình “Thác nước (waterfall)” hay “CMMI”. Phương thức phát triển phần mềm Agile là một tập hợp các phương thức phát triển lặp và tăng dần trong đó các yêu cầu và giải pháp được phát triển thông qua sự liên kết cộng tác giữa các nhóm tự quản và liên chức năng.

Mô tả :

* Dựa trên mô hình iterative and incremental.
* Các yêu cầu và giải pháp phát triển dựa trên sự kết hợp của các function.
* Trong Agile, các tác vụ được chia thành các khung thời gian nhỏ để cung cấp các tính năng cụ thể cho bản phát hành cuối.
* Có thể được sử dụng với bất kỳ loại hình dự án nào, nhưng cần sự tham gia và tính tương tác của khách hàng.
* Sử dụng khi khách hàng yêu cầu chức năng sẵn sàng trong khoảng thời gian ngắn.

### Ưu điểm

* Tăng cường tình thần làm việc nhóm và trao đổi công việc hiệu quả.
* Các chức năng được xây dựng nhanh chóng và rõ ràng, dế quản lý.
* Dễ dàng bổ sung, thay đổi yêu cầu.
* Quy tắc tối thiểu, tài liệu dễ hiểu, dễ sử dụng.

### Nhược điểm

Mô hình Agile được sử dụng rộng rãi trên thế giới nhưng cũng không đồng nghĩa với phù hợp với tất cả các dự án phần mềm.

* Không thích hợp để xử lý các phụ thuộc phức tạp.
* Có nhiều rủi ro về tính bền vững, khả năng bảo trì và khả năng mở rộng.
* Cần một team có kinh nghiệm.
* Phụ thuộc rất nhiều vào sự tương tác rõ ràng của khách hàng.
* Chuyển giao công nghệ cho các thành viên mới trong nhóm có thể khá khó khăn do thiếu tài liệu.

1. Mô hình cận lặp:

* Một mô hình được lặp đi lặp lại từ khi start cho đến khi làm đầy đủ spec.Quá trình này sau đó được lặp lại, tạo ra một phiên bản mới của phần mềm vào cuối mỗi lần lặp của mô hình.
* Thay vì phát triển phần mềm từ spec đặc tả rồi mới bắt đầu thực thi thì mô hình này có thể review dần dần để đi đến yêu cầu cuối cùng.

### Ứng dụng:

* Yêu cầu chính phải được xác định; tuy nhiên, một số chức năng hoặc yêu cầu cải tiến có thể phát triển theo thời gian.
* Một công nghệ mới đang được sử dụng và đang được học tập bởi nhóm phát triển trong khi làm việc trong dự án.
* Phù hợp cho các dự án lớn và nhiệm vụ quan trọng.

### Ưu điểm

* Xây dựng và hoàn thiện các bước sản phẩm theo từng bước.
* Thời gian làm tài liệu sẽ ít hơn so với thời gian thiết kế.
* Một số chức năng làm việc có thể được phát triển nhanh chóng và sớm trong vòng đời.
* Ít tốn kém hơn khi thay đổ phạm vi, yêu cầu.
* Dễ quản lý rủi ro.
* Trong suốt vòng đời, phần mềm được sản xuất sớm để tạo điều kiện cho khách hàng đánh giá và phản hồi.

### Nhược điểm

* Yếu cầu tài nguyên nhiều.
* Các vấn đề về thiết kế hoặc kiến trúc hệ thống có thể phát sinh bất cứ lúc nào.
* Yêu cầu quản lý phức tạp hơn.
* Tiến độ của dự án phụ thuộc nhiều vào giai đoạn phân tích rủi ro.

1. Mô hình tăng trưởng:

* Spec được chia thành nhiều phần.
* Chu kỳ được chia thành các module nhỏ, dễ quản lý.
* Mỗi module sẽ đi qua các yêu cầu về thiết kế, thực hiện, … như 1 vòng đời phát triển thông thường.
* Áp dụng cho những dự án có yêu cầu đã được mô tả, định nghĩa và hiểu một cách rõ ràng.
* Khách hàng có nhu cầu về sản phẩm sớm.

### Ưu điểm

* Phát triển nhanh chóng.
* Mô hình này linh hoạt hơn, ít tốn kém hơn khi thay đổi phạm vi và yêu cầu.
* Dễ dàng hơn trong việc kiểm tra và sửa lỗi.

### Nhược điểm

* Cần lập plan và thiết kế tốt.
* Tổng chi phí là cao hơn so với mô hình thác nước.

1. Mô hình chữ V:

* Mô hình chữ V là một phần mở rộng của mô hình thác nước và được dựa trên sự kết hợp của một giai đoạn thử nghiệm cho từng giai đoạn phát triển tương ứng. Đây là một mô hình có tính kỷ luật cao và giai đoạn tiếp theo chỉ bắt đầu sau khi hoàn thành giai đoạn trước.
* Với V model thì công việc test được tham gia ngay từ đầu.
* Yêu cầu được xác định rõ ràng.
* Xác định sản phẩm ổn định.
* Công nghệ không thay đổi và được hiểu rõ bởi nhóm dự án.
* Không có yêu cầu không rõ ràng hoặc không xác định.
* Dự án ngắn.

### Ưu điểm

* Đây là một mô hình có tính kỷ luật cao và các giai đoạn được hoàn thành cùng một lúc.
* Hoạt động tốt cho các dự án nhỏ, khi các yêu cầu được hiểu rất rõ.
* Đơn giản và dễ hiểu và dễ sử dụng, dễ quản lý.

### Nhược điểm

* Khó quản lý kiểm soát rủi ro, rủi ro cao.
* Không phải là một mô hình tốt cho các dự án phức tạp và hướng đối tượng.
* Mô hình kém cho các dự án dài và đang diễn ra.
* Không thích hợp cho các dự án có nguy cơ thay đổi yêu cầu trung bình đến cao.

1. Mô hình Scrum:

* Chia các yêu cầu ra làm theo từng giai đoạn. Mỗi 1 giai đoạn(sprint) chỉ làm 1 số lượng yêu cầu nhất định.
* Mỗi một sprint thường kéo dài từ 1 tuần đến 4 tuần ( ko dài hơn 1 tháng).
* Đầu sprint sẽ lên kế hoạch làm những yêu cầu nào. Sau đó, sẽ thực hiện code và test. Cuối sprint là 1 sản phẩm hoàn thiện cả code lẫn test có thể demo và chạy được.
* Hoàn thành sprint 1, tiếp tục làm sprint 2, sprint... cho đến khi hoàn thành hết các yêu cầu.
* Trong mỗi 1 sprint thì sẽ có họp hàng ngày – daily meeting từ 15 – 20 phút. Mỗi thành viên sẽ báo cáo: Hôm qua tôi đã làm gì? Hôm nay tôi sẽ làm gì? Có gặp khó khăn gì không?
* Scrum là mô hình hướng khách hàng (Customer oriented).

**Tổ chức (Organization)**

* 1. Tổ chức nhóm dự án và Roles: Vài trò.
  2. Product Owner: Người sở hữu sản phẩm.
  3. ScrumMaster: Người điều phối.
  4. Development Team: Nhóm phát triển.

**Tài liệu (Atifacts):** đó chính là các kết quả đầu ra.

* 1. Product Backlog: Danh sách các chức năng cần phát triển của sản phẩm.
  2. Sprint Backlog: Danh sách các chức năng cần phát triển cho mỗi giai đoạn.
  3. Estimation:Kết quả ước lượng của team.

**Qui trình(Process)**: Qui định cách thức vận hành của SCRUM.

* 1. Sprint Planning meeting: Hoạch định cho mỗi giai đoạn.
  2. Review: Tổng kết cho mỗi giai đoạn.
  3. Daily Scrum Meeting: Review hàng ngày.

### Ưu điểm

* Một người có thể thực hiện nhiều việc ví dụ như dev có thể test.
* Phát hiện lỗi sớm.
* Có khả năng áp dụng được cho những dự án mà yêu cầu khách hàng không rõ ràng ngay từ đầu.

### Nhược điểm

* Trình độ của nhóm cần có một kỹ năng nhất định.
* Phải có sự hiểu biết về mô hình aglie.
* Khó khăn trong việc xác định ngân sách và thời gian.
* Luôn nghe ý kiến phản hồi từ khách hàng và thay đổi theo nên thời gian sẽ kéo dài.
* Vai trò của PO rất quan trọng, PO là người định hướng sản phẩm. Nếu PO làm không tốt sẽ ảnh hưởng đến kết quả chung.

1. Mô hình RAD:

* Mô hình RAD là một phương pháp phát triển phần mềm sử dụng quy hoạch tối thiểu có lợi cho việc tạo mẫu nhanh.
* Các mô-đun chức năng được phát triển song song như nguyên mẫu và được tích hợp để tạo ra sản phẩm hoàn chỉnh để phân phối sản phẩm nhanh hơn.
* Đảm bảo rằng các nguyên mẫu được phát triển có thể tái sử dụng được.
* Module hóa rõ ràng. Nếu dự án không thể được chia thành các mô-đun, RAD có thể không thành công.
* RAD nên được sử dụng khi có nhu cầu để tạo ra một hệ thống có yêu cầu khách hàng thay đổi trong khoảng thời gian nhỏ 2-3 tháng.
* Nên được sử dụng khi đã có sẵn designer cho model và chi phí cao.

### Ưu điểm

* Giảm thời gian phát triển.
* Tăng khả năng tái sử dụng của các thành phần.
* Đưa ra đánh giá ban đầu nhanh chóng.
* Khuyến khích khách hàng đưa ra phản hồi.

### Nhược điểm

* Trình độ của nhóm cần có một kỹ năng nhất định.
* Chỉ những hệ thống có module mới sử dụng được mô hình này.

II. So sánh mô hình tuyến tính với mô hình lặp:

1. Mô hình tuyến tính:

* Đây được coi như là mô hình phát triển phần mềm đầu tiên được sử dụng.
* Mô hình này áp dụng tuần tự các giai đoạn của phát triển phần mềm.
* Đầu ra của giai đoạn trước là đầu vào của giai đoạn sau. Giai đoạn sau chỉ được thực hiện khi giai đoạn trước đã kết thúc. Đặc biệt không được quay lại giai đoạn trước để xử lý các yêu cầu khi muốn thay đổi.
* Thường dùng cho các dự án nhỏ , ngắn hạn.
* Các dự án có ít thay đổi về yêu cầu và không có những yêu cầu không rõ ràng.

Phân tích mô hình:

* **Requirement gathering**: Thu thập và phân tích yêu cầu được ghi lại vào tài liệu đặc tả yêu cầu trong giai đoạn này.
* **System Analysis**: Phân tích thiết kế hệ thống phần mềm, xác định kiến trúc hệ thống tổng thể của phần mềm.
* **Coding**: Hệ thống được phát triển theo từng unit và được tích hợp trong giai đoạn tiếp theo. Mỗi Unit được phát triển và kiểm thử bởi dev được gọi là Unit Test.
* **Testing**: Cài đặt và kiểm thử phần mềm. Công việc chính của giai đoạn này là kiểm tra và sửa tất cả những lỗi tìm được sao cho phần mềm hoạt động chính xác và đúng theo tài liệu đặc tả yêu cầu.
* **Implementation**: Triển khai hệ thống trong môi trường khách hàng và đưa ra thị trường.
* **Operations and Maintenance**: Bảo trì hệ thống khi có bất kỳ thay đổi nào từ phía khách hàng, người sử dụng.
  1. **Ưu điểm**
* Dễ sử dụng, dễ tiếp cận, dễ quản lý.
* Sản phẩm phát triển theo các giai đoạn được xác định rõ ràng.
* Xác nhận ở từng giai đoạn, đảm bảo phát hiện sớm các lỗi.
  1. **Nhược điểm**
* Ít linh hoạt, phạm vi điều chỉnh hạn chế.
* Rất khó để đo lường sự phát triển trong từng giai đoạn.
* Mô hình không thích hợp với những dự án dài, đang diễn ra, hay những dự án phức tạp, có nhiều thay đổi về yêu cầu trong vòng đời phát triển.
* Khó quay lại khi giai đoạn nào đó đã kết thúc.

1. Mô hình lặp:

* Một mô hình được lặp đi lặp lại từ khi start cho đến khi làm đầy đủ spec.Quá trình này sau đó được lặp lại, tạo ra một phiên bản mới của phần mềm vào cuối mỗi lần lặp của mô hình.
* Thay vì phát triển phần mềm từ spec đặc tả rồi mới bắt đầu thực thi thì mô hình này có thể review dần dần để đi đến yêu cầu cuối cùng.

### Ứng dụng:

* Yêu cầu chính phải được xác định; tuy nhiên, một số chức năng hoặc yêu cầu cải tiến có thể phát triển theo thời gian.
* Một công nghệ mới đang được sử dụng và đang được học tập bởi nhóm phát triển trong khi làm việc trong dự án.
* Phù hợp cho các dự án lớn và nhiệm vụ quan trọng.

### Ưu điểm

* Xây dựng và hoàn thiện các bước sản phẩm theo từng bước.
* Thời gian làm tài liệu sẽ ít hơn so với thời gian thiết kế.
* Một số chức năng làm việc có thể được phát triển nhanh chóng và sớm trong vòng đời.
* Ít tốn kém hơn khi thay đổ phạm vi, yêu cầu.
* Dễ quản lý rủi ro.
* Trong suốt vòng đời, phần mềm được sản xuất sớm để tạo điều kiện cho khách hàng đánh giá và phản hồi.

### Nhược điểm

* Yếu cầu tài nguyên nhiều.
* Các vấn đề về thiết kế hoặc kiến trúc hệ thống có thể phát sinh bất cứ lúc nào.
* Yêu cầu quản lý phức tạp hơn.
* Tiến độ của dự án phụ thuộc nhiều vào giai đoạn phân tích rủi ro.

III. Hãy nêu khái niệm về phần mềm? Công nghệ phần mềm và chu kỳ phát triển phần mềm:

Phần mềm: là một tập hợp những câu lệnh được viết bằng một hoặc nhiều ngôn ngữ lập trình theo một trật tự xác định nhằm tự động thực hiện một số chức năng hoặc giải quyết một bài toán nào đó.

Công nghệ phần mềm: hay kỹ thuật phần mềm (tiếng Anh: software engineering) là sự áp dụng một cách tiếp cận có hệ thống, có kỷ luật, và định lượng được cho việc phát triển, sử dụng và bảo trì phần mềm. Ngành học kỹ sư phần mềm bao trùm kiến thức, các công cụ, và các phương pháp cho việc định nghĩa yêu cầu phần mềm, và thực hiện các tác vụ thiết kế, xây dựng, kiểm thử (software testing), và bảo trì phần mềm. Kỹ sư phần mềm còn sử dụng kiến thức của các lĩnh vực như kỹ thuật máy tính, khoa học máy tính, quản lý, toán học, quản lý dự án, quản lý chất lượng, công thái học phần mềm (software ergonomics), và kỹ sư hệ thống (systems engineering).

Quá trình phát triển về mặt công nghệ có sự thay đổi phần cứng, phần mềm như hệ điều hành thay đổi từ Windows XP qua Windows 7 hay Linux thì các phần mềm nghiệp vụ cũng phải thay đổi theo để tương thích với hệ thống mới.

IV. Nêu quy trình mô hình tăng trưởng, nêu ưu và nhược.

Mô hình tăng trưởng:

* Spec được chia thành nhiều phần.
* Chu kỳ được chia thành các module nhỏ, dễ quản lý.
* Mỗi module sẽ đi qua các yêu cầu về thiết kế, thực hiện, … như 1 vòng đời phát triển thông thường.
* Áp dụng cho những dự án có yêu cầu đã được mô tả, định nghĩa và hiểu một cách rõ ràng.
* Khách hàng có nhu cầu về sản phẩm sớm.

### Ưu điểm

* Phát triển nhanh chóng.
* Mô hình này linh hoạt hơn, ít tốn kém hơn khi thay đổi phạm vi và yêu cầu.
* Dễ dàng hơn trong việc kiểm tra và sửa lỗi.

### Nhược điểm

* Cần lập plan và thiết kế tốt.
* Tổng chi phí là cao hơn so với mô hình thác nước.

V. Quy trình phát triển phần mềm là gì? bao gồm các giai đoạn nào, nêu cụ thể từng giai đoạn.

* **Xác định yêu cầu**: Khảo sát hệ thống, vạch ra các vấn đề tồn tại và các cơ hội của hệ thống,xác định phạm vi của hệ thống, lập kế hoạch các hoạt động của nhóm, xác định thời gian, nguồn lực cần thiết, chi phí đầu tư và lợi ích mang lại từ hệ thống. Kết quả của giai đoạn là chấp nhận để phát triển, bị từ chối,định hướng lại.
* **Phân tích:** Thu thập yêu cầu hệ thống, xác định tất cả những gì mà người dùng mong muốn từ hệ thống đề xuất. Nghiên cứu các yêu cầu và cấu trúc hóa (mô hình hóa) để dễ dàng nhận biết và loại bỏ những yếu tố dư thừa.
* **Thiết kế:** Kết quả của giai đoạn phân tích sẽ được chi tiết hóa để trở thành một giải pháp kỹ thuật để thực hiện. Các đối tượng và các lớp mới được xác định để bổ sung vào việc cài đặt yêu cầu và tạo ra một cơ sở kỹ thuật về kiến trúc.
* **Pha thiết kế** là quá trình thiết kế phần mềm dựa trên đặc tả. Pha thiết kế gồm:
  + Thiết kế giao diện
  + Thiết kế thành phần
  + Thiết kế cấu trúc dữ liệu và thiết kế thuật toán
* **Lập trình (code):** Bao gồm việc triển khai các tài liệu thiết kế bằng ngôn ngữ lập trình để đưa ra các mô đun chức năng. Cuối giai đoạn này sẽ cho ra được mã nguồn của chương trình để làm đầu vào cho quá trình kiểm thử tiếp theo.
* **Kiểm thử:** Là vận hành phần mềm để phát hiện lỗi trong phần mềm, lên kế hoạch kiểm tra kết hợp với các bộ tài liệu thiết kế và dữ liệu kiểm thử. Cuối giai đoạn này sẽ đưa ra được báo cáo về các lỗi của phần mềm trong khi kiểm nghiệm.

VI. Hãy nêu quy trình của các mô hình tuyế tí , êu ưu đ ểm và ược đ ểm của các mô hình.

* Đây được coi như là mô hình phát triển phần mềm đầu tiên được sử dụng.
* Mô hình này áp dụng tuần tự các giai đoạn của phát triển phần mềm.
* Đầu ra của giai đoạn trước là đầu vào của giai đoạn sau. Giai đoạn sau chỉ được thực hiện khi giai đoạn trước đã kết thúc. Đặc biệt không được quay lại giai đoạn trước để xử lý các yêu cầu khi muốn thay đổi.
* Thường dùng cho các dự án nhỏ , ngắn hạn.
* Các dự án có ít thay đổi về yêu cầu và không có những yêu cầu không rõ ràng.

Phân tích mô hình:

* **Requirement gathering**: Thu thập và phân tích yêu cầu được ghi lại vào tài liệu đặc tả yêu cầu trong giai đoạn này.
* **System Analysis**: Phân tích thiết kế hệ thống phần mềm, xác định kiến trúc hệ thống tổng thể của phần mềm.
* **Coding**: Hệ thống được phát triển theo từng unit và được tích hợp trong giai đoạn tiếp theo. Mỗi Unit được phát triển và kiểm thử bởi dev được gọi là Unit Test.
* **Testing**: Cài đặt và kiểm thử phần mềm. Công việc chính của giai đoạn này là kiểm tra và sửa tất cả những lỗi tìm được sao cho phần mềm hoạt động chính xác và đúng theo tài liệu đặc tả yêu cầu.
* **Implementation**: Triển khai hệ thống trong môi trường khách hàng và đưa ra thị trường.
* **Operations and Maintenance**: Bảo trì hệ thống khi có bất kỳ thay đổi nào từ phía khách hàng, người sử dụng.
  1. **Ưu điểm**
* Dễ sử dụng, dễ tiếp cận, dễ quản lý.
* Sản phẩm phát triển theo các giai đoạn được xác định rõ ràng.
* Xác nhận ở từng giai đoạn, đảm bảo phát hiện sớm các lỗi.
  1. **Nhược điểm**
* Ít linh hoạt, phạm vi điều chỉnh hạn chế.
* Rất khó để đo lường sự phát triển trong từng giai đoạn.
* Mô hình không thích hợp với những dự án dài, đang diễn ra, hay những dự án phức tạp, có nhiều thay đổi về yêu cầu trong vòng đời phát triển.
* Khó quay lại khi giai đoạn nào đó đã kết thúc.

VII. Hãy nêu các công cụquản lý phần mềm hin nay và nêu chức ă của chúng.

## [**Microsoft Project**](https://products.office.com/vi-vn/project/project-and-portfolio-management-software?rtc=1)

Microsoft Project được tạo ra để giúp các nhà quản lý dự án lập kế hoạch, phân phối nguồn lực, theo dõi tiến độ và phân tích khối lượng công việc. Mặc cho tính chất phức tạp, MS Project lại sở hữu giao diện trực quan. Đây là một giải pháp phần mềm được đánh giá cao; có thể đóng góp rất lớn vào thành công của dự án. Nó vận hành thành ba khối: Task, Resources, Calendar; và có sự liên kết giữa các khối này. Phần mềm này về cơ bản là một cơ sở dữ liệu với khả năng tự động hóa khá đơn giản.

Ưu điểm chính của việc sử dụng Microsoft Project là:

* Tổ chức công việc hiệu quả bằng công cụ MS Project scheduling
* Theo dõi đầy đủ và thuận tiện tất cả các thay đổi và phát triển dự án
* Tiến độ có khả năng tùy biến cao để phù hợp với các dự án cụ thể
* Khả năng đại diện cho các dự án, thêm ý tưởng, lọc dữ liệu
* Dự đoán và phòng ngừa rủi ro
* Tóm tắt báo cáo và quản lý hồ sơ hiệu quả

Microsoft Project là một lựa chọn tuyệt vời cho các tổ chức, đặc biệt là khi các dự án yêu cầu sự tương tác thường xuyên giữa các bộ phận khác nhau.

## [**Microsoft Visio**](https://products.office.com/vi-vn/visio/flowchart-software?rtc=1)

MS Visio là một ứng dụng đồ họa vector được tạo cho Hệ Điều hành Windows và được các nhà quản lý dự án sử dụng rộng rãi. Nó là một công cụ mạnh mẽ để trình bày thông tin phức tạp dưới dạng sơ đồ, hình khối và flowchart.

Ngay cả khi bạn không có năng khiếu về thiết kế hoặc kỹ năng kỹ thuật, Visio vẫn có khả năng giúp bạn phác thảo tư duy của mình thành hình ảnh trực quan nhất. Visio sử dụng một set diagram để đơn giản hóa và tăng cường tính trực quan của các quy trình trừu tượng. Cho dù bạn muốn tạo sơ đồ tổ chức, mạng lưới hoặc sơ đồ quy trình, Microsoft Visio vẫn cung cấp được nhiều mẫu và hình dạng được thiết kế sẵn. Hơn nữa, với Visio, bạn có thể tương tác với người dùng khác và minh họa ngắn gọn một số điểm nhất định. Ví dụ, từ một sơ đồ tổ chức, bạn có thể nêu lên một vấn đề và thảo luận về giải pháp, sau đó có thể áp dụng cho một dự án mới. Với Visio, bạn cũng có thể nhập dữ liệu từ MS Excel để tạo đồ họa rõ ràng cho các bản trình bày và báo cáo công khai.

## [**Evernote**](https://evernote.com/)

Evernote là một công cụ có thể giúp xây dựng một nền tảng kiến thức khổng lồ cho các nhà quản lý dự án. Nó có thể được sử dụng để tạo danh sách việc cần làm và lưu trữ tất cả các loại thông tin về các dự án của bạn trong các tệp và ghi chú. Ngoài ra, Evernote có thể hỗ trợ các công cụ cần thiết cho các PM. Evernote quản lý dự án dựa trên "đám mây", đồng nghĩa là bạn có thể truy cập từ bất kỳ thiết bị nào - smartphone, máy tính bảng hoặc PC. Có một số công cụ trong Evernote, khi kết hợp lại cho phép bạn quản lý các nhiệm vụ của mình một cách hiệu quả. Evernote cung cấp quyền truy cập được chỉ định cho những người dùng khác nhau. Đối với các nhiệm vụ cá nhân, bạn có thể sử dụng notebook cá nhân. Nếu bạn làm việc với một nhóm và muốn thử thách nhân viên của mình, bạn nên sử dụng quyền truy cập công cộng, cho phép bạn xác định mức độ truy cập của các đối tượng đồng nghiệp khác nhau.

## [**VSDX Annotator**](https://nektony.com/products/vsdx-annotator-mac)

VSDX Annotator là trình xem Visio cho những người làm việc trên Mac. Nó được sử dụng rộng rãi bởi các nhà quản lý dự án để cộng tác trên các tệp Visio trong môi trường đa nền tảng. Nó bao gồm 12 công cụ chú thích cho phép bạn thực hiện các thay đổi trong bản vẽ ngay trên máy Mac cũng như thêm ghi chú, câu hỏi và nhận xét của bạn. Sau đó, bạn có thể lưu dưới dạng tiện ích mở rộng PDF (.pdf) hoặc Visio (.vsdx) để chỉnh sửa thêm trong Visio. Ứng dụng này hiển thị tất cả các bản vẽ nội dung với hình ảnh sắc nét trực quan.

## [**GanttPro**](https://ganttpro.com/?coupon=Ho-ho-holidays&utm_campaign=go-Ho-ho-holidays)

GanttPro là giải pháp phần mềm dựa trên biểu đồ Gantt. Nó cho phép bạn lập kế hoạch và quản lý các dự án trực tuyến, trực quan hóa các quy trình, tạo và phân công nhiệm vụ cho các thành viên trong nhóm, đặt thời hạn và xem tỷ lệ phần trăm các nhiệm vụ riêng lẻ đã hoàn thành so với các mốc thời gian dự kiến. Bạn có thể chia sẻ biểu đồ Gantt mà bạn đã tạo với nhóm và khách hàng, gán cho người dùng chuyên dụng quyền xem hoặc chỉnh sửa. GanttPro có một bộ các tính năng chính, một số tính năng tương thích với các phương thức Agile, ví dụ:

* phân bổ thời gian các nhiệm vụ dựa trên mức độ ưu tiên
* phân phối nhiệm vụ giữa những người tham gia dự án
* phân tích critical path để tăng tốc tiến độ
* lịch trình tự động
* theo dõi tiến độ hiện tại để biết giai đoạn thực hiện dự án