1. **Mô tả qui trình phát triển phần mềm.**

Qui trình phát triển phần mềm gồm 6 bước:

1. Requirement: Lấy yêu cầu từ khách hàng.
2. Analysis & Design: Phân tích thiết kế phần mềm.
3. Implement: Thực thi/Xây dựng phần mềm.
4. Testing: Kiểm thử phần mềm.
5. Deloyment: Triển khai phần mềm.
6. Maintain: Duy trì/Bảo trì sản phẩm cho khách hàng
7. **Phân biệt giữa chiến lược tiếp cận và kế hoạch hành động.**

* Chiến lược tiếp cận là các chiến lược như Scrum, Waterfall. Đấy là chiến lược lâu dài và có kế hoạch hành động cụ thể các bước để thể hiện chiến lược đó. VD: thời gian, yêu cầu đầu vào/đầu ra
* Kế hoạch hành động là kế hoạch được phân tích rõ ràng, cụ thể, đường đi và kế hoạch đạt được các mục tiêu.

1. **Tiêu chí đánh giá một kế hoạch? Lập kế hoạch phân tích thiết kế phần mềm.**

* Tiêu chí đánh giá một kế hoạch hành động là phải đảm bảo yêu cầu của nguyên tắc S.M.A.R.T
* Specific – cụ thể, dễ hiểu: Chỉ tiêu phải cụ thể vì nó định hướng cho các hoạt động trong tương lai.
* Measurable – đo lường được: Phải tính toán đo lường từ thời gian thực hiện tới nguồn tiền cần chi tiêu để hoàn thành dự án.
* Achievable – vừa sức: Chỉ tiêu phải có tính thách thức để cố gắng, nhưng cũng đừng đặt chỉ tiêu loại không thể đạt nổi.
* Realistics – thực tế: Sự cân bằng giữa khả năng thực hiện so nguồn lực của doanh nghiệp bạn (thời gian, nhân sự, tiền bạc)
* Timebound – có thời hạn: Mọi công việc phải có thời hạn hoàn thành, nếu không nó sẽ bị trì hoãn. Thời gian hợp lý giúp bạn vừa đạt được mục tiêu lại vừa dưỡng sức cho các mục tiêu khác
* Kế hoạch phân tích thiết kế phần mềm WBS (Work Breakdown Structure)
* Top-down: là tạo cấu trúc phân chia công việc từ tổng quan đến chi tiết.
* Bottom-up: là tạo cấu trúc phân chia công việc từ dưới lên hay từ chi tiết đến tổng quan.
* Analogy: là tạo cấu trúc phân chia công việc dựa trên kinh nghiệm từ các dự án tương tự đã từng được triển khai.
* Brainstorming: là dùng tư duy, tổng hợp ý kiến của nhiều người để tạo lập cấu trúc phân chia công việc. Có thể kết hợp Top-down và Brainstorming trên cùng một WBS.

1. **Mô tả cách thức kiểm soát và kiểm tra một dự án.**

- Ước lượng dự án: ước lượng các nguồn lực, chi phí và lịch trình….

+ Các nguồn lực: con người, môi trường, các phần mềm hổ trợ

+ Phạm vi dự án: các tính năng được chuyển giao đến người dùng, hoạt động hiệu quả với hệ thống, các dữ liệu đầu vào và kết quả cho ra

- Lập kế hoạch dự án: sắp xếp công việc, phân chia vụ

+ Xác định các nhiệm vụ: xác định loại dự án, đánh giá yêu cầu, chọn các phần mềm thích hợp cho dự án

+ Lập biểu đồ thời gian

+ Theo dõi tiến trình và báo cáo

- Quản lý rủi ro: các dự án đều có thể xảy ra các rủi ro, phải lên kế hoạch cách khắc phục nếu có xảy ra rủi ro

+ Xác định rủi ro: Sự thay đổi yêu cầu của khách hàng, thay đổi đột ngột về số lượng nhân viên, môi trường phát triển bị thay đổi hay tiến độ không đúng.

1. **Trình bày cách thức kiểm soát và kiểm tra một kế hoạch Kiểm thử phần mềm.**

- Lập kế hoạch:

+ Phân tích rủi ro

+ Uớc tính thời gian Test

+ Lập kế hoạch

+ Tổ chức xây dựng quy trình Test

-Thực thi:

+ Điều khiển và giám sát việc Test

+ Quản lý vấn đề

+ Đánh giá và báo cáo kết quả Test

1. **Ưu điểm của quy trình phát triển phần mềm theo phương pháp SCRUM.**

* Đảm bảo chất lượng phần mềm tốt hơn
* Phần mềm được hoàn thành đúng tiến độ với khoảng thời gian phát triển ngắn.
* Tiết kiệm chi phí phát triển phần mềm bằng việc tiếp nhận phản hồi từ khách hàng trong giai doạn phát triển, tránh việc lặp đi lặp lại gây hao tổn chi phí.
* Chi phí phát triển giảm thì lợi nhuận tăng => Đem lại lợi ích lớn.

1. **Ưu điểm của quy trình phát triển phần mềm theo phương pháp Waterfall.**

* Là mô hình đơn giản dễ sử dụng, có kết cấu chặt chẽ. Ngoài ra, nó còn có quy trình rõ ràng theo từng bước và từng mốc thời gian cụ thể.
* Dễ quản lí, bảo trì bởi cách tiếp cận tuyến tính và cố định theo từng bước, ngoài ra còn cho phép chúng ta thay đổi bản thiết kế phần mềm từ sớm.
* Các tiêu chí đầu vào và đầu ra được xác định rõ ràng nên giúp cho việc kiểm tra chất lượng trở nên dễ dàng.
* Hoạt động hiệu quả đối với những dữ án nhỏ với các yêu cầu rõ ràng.

1. **Nêu các rủi ro trong qui trình phát triển phần mềm và cách thức kiểm soát rủi ro.**

Nguyên nhân nào tạo ra các rủi ro?

- Hiểu biết không chính xác về yêu cầu của khách hàng

- Không có khả năng đối mặt với yêu cầu thay đổi

- Các module không khớp với nhau

- Phần mềm khó duy trì và mở rộng

- Chất lượng phầm mền kém

- Quy trình cứng nhắc không linh hoạt và không chịu được các thay đổi

- Quản lý yếu kém

Hướng giải quyết?

- Phải có kế hoạch chuẩn bị trước cho các vấn đề có thể gặp phải trong việc phát triển

1. **So sánh ưu điểm và khuyết điểm giữa việc Tích hợp giải pháp hệ thống CNTT và phát triển hệ thống CNTT.**
2. **Hãy mô tả các loại nguồn lực trong lĩnh vực phát triển dự án phần mềm.**

* Manpower – Nguồn nhân lực: Người giữ trách nhiệm thực hiện công việc có đầy đủ các tiêu chí về: trình độ, kỹ năng, kinh nghiệm, phẩm chất,…
* Money – Ngân sách: Nguồn tiền để thực hiện dự án.
* Material – Nguyên vật liệu/Hệ thống cung ứng: Nguyên vật liệu/Hệ thống cung ứng cần để thực hiện dự án.
* Machines – Máy móc/Công cụ/Công nghệ: Các thiết bị công nghệ hoặc các phần mềm công nghệ cần để thực hiện dự án.
* Methods – Phương pháp: Cách mà nguồn nhân lực vận hành

1. **Trong phân tích thiết kế, cho biết vài trò của mức quan niệm và các lược đồ tương ứng.**
2. **Trong phân tích thiết kế, cho biết vai trò của mức logic và các lược đồ tương ứng.**
3. **Trong phân tích thiết kế, cho biết vài trò của mức vật lí và các lược đồ tương ứng.**
4. **Phân biệt 2 khái niệm Verification và Validation trong giai đoạn kiểm thử.**

|  |  |
| --- | --- |
| **Verification (Xác minh / Kiểm định)** | **Validation (Xác nhận / Thẩm định)** |
| Đánh giá sản phẩm trung gian để kiểm tra nó có đáp ứng các yêu cầu cụ thể của từng giai đoạn hay không | Đánh giá sản phẩm cuối cùng để xem nó đáp ứng được yêu cầu nghiệp vụ hay không |
| Kiểm tra xem sản phẩm có được xây dựng theo đúng yêu cầu và đặc điểm kĩ thuật thiết kế không | Xác định xem phần mềm có phù hợp với nhu cầu sử dụng và đáp ứng nghiệp vụ hay không |
| Kiểm tra xem “Chúng tôi xây dựng sản phẩm đúng không” | Kiểm tra xem “Chúng tôi xây dựng đúng sản phẩm không” |
| Thực hiện mà không chạy phần mềm | Thực hiện cùng với việc chạy phần mềm |

1. **Tại sao phải kiểm thử hồi qui (Regression Test).**

* Kiểm thử hồi quy để xác nhận rằng một tính năng mới được thêm không ảnh hưởng xấu đến các tính năng hiện có.
* Kiểm thử hồi quy là kiểm thử lại các trường hợp đã được thực hiện để đảm bảo các chức năng hiện có hoạt động tốt.
* Kiểm thử này được thực hiện để đảm bảo rằng những thay đổi source code mới sẽ không ảnh hưởng tới các chức năng hiện có, đảm bảo rằng code cũ vẫn hoạt động sau khi thực hiện thay đổi source code.

1. **Mô tả các mức độ kiểm thử từ tổng quát đến chi tiết.**
2. **Cho biết các tài liệu cần thiết cho mỗi mức độ kiểm thử.**
3. **Phân biệt vai trò giữa lược đồ Activity và Sequence.**

* **Vai trò của lược đồ Activity** là mô tả hoạt động của một chức năng, nó mô tả cho ta thấy chức năng đó hoạt động như thế nào, có luồng thực thi và trình tự tương tác ra sao.
* **Vai trò của lược đồ Sequence** là biểu diễn các trình tự sự kiện của một nhóm đối tượng tương tác với nhau trong một chức năng, nó miêu tả chi tiết các thông điệp gửi và nhận giữa các đối tượng và chú trọng về mặt trình tự thời gian của việc gửi và nhận thông điệp đó.

1. **Phương pháp xác định các phương thức trong một lớp, cho ví dụ.**

Ta xác định các phương thức trong một lớp một lớp bằng cách:

* Các thông điệp của actor trong Đặc tả usecase, tìm kiếm các động từ hoặc nhóm động từ liên quan tới đối tượng đang bị xét.
* Chú ý xem đối tượng được tạo và hủy bỏ đi như nào? Trong thời gian đó nó gửi/nhận thông điệp ra sao?
* Xác định phương thức thông qua các thông điệp được gửi và nhận trong Sequence Diagram mà ta phân tích.
* Sau khi xác định thông điệp message trong Sequence, ta chuyển nó thành thông điệp dạng phương thức => Xác định được phương thức mà lớp đó sử dụng.

1. **Phương thức xác định các table cần thiết trong một CSDL hệ thống.**

Ta xác định các table cần thiết trong một CSDL hệ thống theo từng bước:

1. Xác định các đối tượng dữ liệu cần lưu trữ => Ta xác định được Thực thể
2. Xác định các thuộc tính của thực thể => Ta xác định khóa chính và các dữ liệu khác của thực thể.
3. Xác định các mối kết hợp giữa các thực thể => Ta xác định các khóa phụ giữa các thực thể hoặc các dữ liệu dẫn xuất từ các thực thể đó.
4. Hoàn thiện Lược đồ ER => Ta hoàn thiện lược đồ ER mức cơ bản.
5. Hoàn thiện Lược đồ ER mở rộng => Ta xác định các thực thể yếu, các mối kết hợp đệ qui, các mối kết hợp mở rộng, các cấu trúc TQH, CBH
6. Chuyển Lược đồ ER sang Lược đồ CSDL quan hệ => Lúc này ta đã xác định đầy đủ được những table cùng với chi tiết các thuộc tính của table cần thiết trong một hệ thống.