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

Qui trình phát triển phần mềm gồm 6 bước:

1. **Needs identification**: Xác định nhu cầu.
2. **Requirements Analytics**: Phân tích yêu cầu.
3. **Design**: Thiết kế.
4. **Development**: Lập trình.
5. **Testing**: Triển khai phần mềm.
6. **Deployment** & **Maintenance**: Triển khai & bảo trì
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:**

* Đánh giá khối lượng công việc hoàn thành so với yêu cầu đề ra.
* Đánh giá khả năng sáng tạo đề xuất cải tiến trong công việc.
* Kỹ năng làm việc nhóm
* Tồn tại cần khắc phục trong thời gian tới

**Tiêu chí đánh giá dựa trên SMART:**

* Specific: Cụ thể
* Measurable: Đo lường được
* Achievable: Khả thi
* Realistic: Thực tế
* Time-bound: Thời gian

**Kế hoạch phân tích thiết kế phần mềm (Dựa trên WBS - product backlog)**

* Mô tả phạm vi hệ thống, các phương án sơ khởi: bản mô tả hệ thống, phạm vi hệ thống, tóm lược các phương án và tiềm năng
* Phân chia các công việc cần thực hiện
* Ước lượng và xây dựng kế hoạch tài nguyên hệ thống: tài chính, tài sản, đặc biệt là nhân sự
* Phác thảo lịch thời gian thực hiện
* Xác định và đánh giá độ rủi ro
* Lập kế hoạch ngân sách: bảng chi phí - lợi nhuận
* Xây dựng tài liệu mô tả hệ thống

1. **Mô tả cách thức kiểm soát và kiểm tra một dự án.**
2. Mục tiêu dự án phù hợp không?
3. Nguồn lực phù hợp
4. Kế hoạch có cụ thể không?
5. Các tài nguyên đủ không?
6. Các cách thức quản lý rủi ro
7. Có kết quả chứng thực để kiểm tra/kiểm thử
8. **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.**

**Các rủi ro chính trong quá trình phát triển phần mềm:**

Bao gồm các yếu tố liên quan trong quá trình phát triển phần mềm. Các rủi ro có thể xảy ra như sau:

- Phần mềm bỏ qua một số chức năng quan trọng mà người dùng chỉ định, người sử dụng yêu cầu hoặc các bên liên quan đã thống nhất.

- Phần mềm không tin cậy và thường xuyên bị lỗi khi sử dụng.

- Phần mềm hỏng theo những cách gây ra thiệt hại về tài chính hoặc thiệt hại khác cho người dùng hoặc công ty mà người sử dụng phần mềm đang làm việc.

- Phần mềm có vấn đề liên quan đến một đặc tính chất lượng nào đó như vấn đề bảo mật của hệ thống, tính thân thiện.

**Cách thức kiểm soát:** Khắc phục và sữa chữa những khuyết điểm cũng như là những rủi ro đã phát sinh và đồng thời dự đoán,  phân tích những rủi ro có thể xảy ra trong tương lai nhằm phòng ngừa và có kế hoạch khắc phục phù hợp

**Các rủi ro (Slide thầy):**

* Khách hàng đổi Requirement liên tục
* Các module không thể tích hợp
* Khó bảo trì
* Trễ deadline
* Chất lượng sản phẩm kém
* Quy trình/ quản lý kém
* Nhân sự chưa có kinh nghiệm/nhân sự rời team bất ngờ
* Khi triển khai phần mềm gặp lỗi/thiếu thiết kế/thiết kế đọc khó hiểu
* Sản phẩm k phù hợp, không đúng yêu cầu KH

**Các giải pháp (Slide thầy):**

* Thực hiện lại quy trình phát triển lại sản phẩm
* Quản lý được Requirement mà KH đưa ra
* Tái sử dụng các cấu trúc phần mềm đã có
* Trực quan mô hình thiết kế phần mềm
* Quản lý sao cho dự án đảm bảo thực thi được
* Quản lý thay đổi (nhân sự thay đổi, nguồn lực, requirement)

1. **So sánh ưu điểm và khuyết điểm giữa việc Tích hợp giải pháp hệ thống CNTT và phát triển hệ thống CNTT.**

Tích hợp giải pháp vào một hệ thống sẵn có giúp tiết kiệm thời gian phát triển do đã có phần mềm từ đầu, chỉ cần tích hợp mở rộng tính năng cần thiết. Tuy vậy, nếu cấu trúc hay thiết kế của bản thân phần mềm khó mở rộng, việc tích hợp sẽ khó thực hiện hoặc gây rủi ro

Phát triển hệ thống thông tin giúp tạo ra được sản phẩm hệ thống theo yêu cầu một cách độc lập, thực hiện lại từ đầu sẽ dễ tạo ra sản phẩm phù hợp, nếu thiết kế tốt sẽ dễ thực hiện tích hợp sau này. Tuy nhiên do làm lại từ đầu nên sẽ mất thời gian để tạo được sản phẩm có thể tiếp cận với người dùng, ngoài ra còn sẽ gặp các rủi ro trong quá trình phát triển.

1. **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.**

* Mô hình quan niệm mô tả sản phẩm phần mềm về mặt chức năng, giúp khách hàng có thể hiểu được sản phẩm của mình có những chức năng thành phần gì. Tuy nhiên, mô hình quan niệm không phù hợp cho team Dev phát triển sản phẩm
* Các lược đồ: Form giao diện (Boundary), Usecase (Control), ERD(Database)

1. **Trong phân tích thiết kế, cho biết vai trò của mức logic và các lược đồ tương ứng.**

* Mô hình logic mô tả cách chức năng hoạt động thông qua input, các xử lý, output, luồng dữ liệu… nhằm chỉ ra cách hoạt động của chương trình, chức năng mà mô hình quan niệm miêu tả. Tuy nhiên, với mô hình này vẫn chưa đủ để làm ra một phần mềm cách hoàn chỉnh
* Các lược đồ: Focused Usecase, Activity Diagram, Sequence Diaggram, Class Diagram ver 1(Control), Mô hình dữ liệu quan hệ (Database)

1. **Trong phân tích thiết kế, cho biết vài trò của mức vật lí và các lược đồ tương ứng.**

* Mô hình mức vật lý mô tả chi tiết cách các dữ liệu được lưu trữ, thể hiện, triển khai trên thiết bị, máy tính dựa trên mô hình logic. Với mô hình này, team dev có thể thực hiện để tạo dựng và phát triển sản phẩm phần mềm.
* Các lược đồ: Class Diagram đầy đủ (Control), Table (Database).

1. **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.**

* Kiểm thử chấp nhận (Acception Testing): Nhằm Validate sản phẩm để xem sản phẩm có đáp ứng được yêu cầu nghiệp vụ của KH không
* Kiểm thử hệ thống (System Testing): Nhằm Verificate sản phẩm để kiểm tra sản phẩm có hoạt động đúng theo thiết kế, theo yêu cầu, đúng kỹ thuật hay không? Đảm bảo được các yêu cầu chức năng và phi chức năng không?
* Kiểm thử tích hợp (Intergration Testing): Nhằm kiểm tra các module hoạt động tích hợp với nhau có đúng theo mong muốn hay không?
* Kiểm thử đơn vị (Unit Testing): Nhằm kiểm tra một module hoạt động có đúng theo mong muốn không?

1. **Cho biết các tài liệu cần thiết cho mỗi mức độ kiểm thử.**

* **Unit test**: test case, test script
* **Integration test**: test scenarios, test case, test script
* **System test**: test plan, test case, test data
* **Acceptance test**: test case, test script, test data

1. **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.