QUY TRÌNH PHÁT TRIỂN PHẦN MỀM

1. Một số khái niệm cơ bản

* Software : Phần mềm - là chương trình máy tính được viết bằng mã lệnh để thực hiện một số chức năng và mục đích nào đó.
* Domain : Nghiệp vụ chính của tổ chức, hệ thống , công ty
* Technology : là công nghệ được sử dụng ( C# , Java, ReactJs…) , database (MySQL..)
* Platforms : chạy trên môi trường (window..)

1. Quy trình tạo ra sản phẩm phần mềm

|  |  |  |  |
| --- | --- | --- | --- |
| Phases(Giai đoạn ) | What(làm gì?) | Who (người thực hiện) | Main Product( Sp chính) |
| Bắt đầu dự án(Initiation) | Xác định phạm vi , quy trình , ngân sách …(scope identifying, schedule , budget..) | PM, BA | Sơ đồ(URD)  Kế hoạch dự kiến(Project plan) |
| Phân tích yêu cầu (analyst) | Thu thập và phân tích yêu cầu của khách hàng (requirement gathering and analysis) | BA | SRS ( đặc tả yêu cầu của khách hàng) |
| Thiết kế(Design) | Design | Architect | Thiết kế chi tiết(Design docs) |
| Development | Coding | Developer | program files  source code |
| Testing | Tìm ra lỗi | Tester | testcase |
| Development | install | Developer/ system admin | Manual install |
| Test nghiệm thu ( UAT/ User Training | Dựa trên yêu cầu đã đề ra ngay từ đầu ( Acceptance test  Training) | Customer  BA | User Manual |

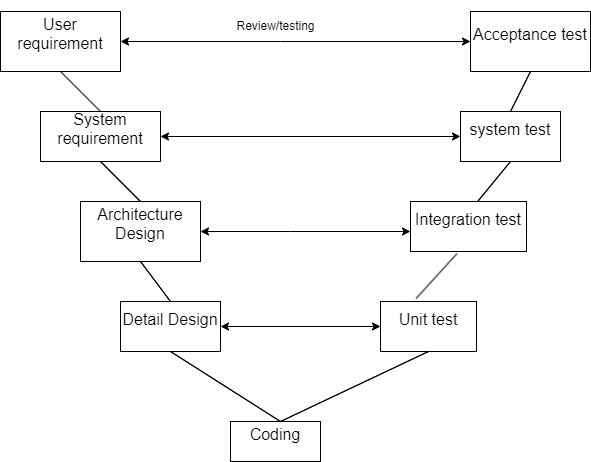
1. Mô hình phát triển phần mềm

* Waterfall model( mô hình thác nước ): Requirement analysis -> design -> Implementation(development) -> testing -> deployment -> maintenance(bảo trì hệ thống).

|  |  |
| --- | --- |
| Ưu điểm | Nhược điểm |
| * Các giai đoạn phát triển là độc lập với nhau, dễ kiểm soát * Dễ ước lượng chi phí và nhân lực làm dự án | * Không hiệu quả. * Hệ thống ban đầu đã có mô hình sẵn , thay đổi yêu cầu rất khó. * Phần mềm được xây dựng muộn , tìm lỗi sẽ khó , chi phí sẽ tăng. * không được sửa chữa sớm |

+ Sử dụng waterfall khi: Yêu cầu của hệ thống rõ ràng , ổn định; yêu cầu của hệ thống không thay đổi nhiều ; ứng dụng không phức tạp , không lớn ; dự án có vòng đời ngắn , không quá dài; công nghệ và công cụ được sử dụng phải ổn định ; nguồn nhân lực sẵn sàng từ kiến thức đến kỹ năng.

* V model :Các hoạt động test thực hiện song song với các giai đoạn phát triển dự án để tìm ra lỗi sớm nhất có thể tránh rủi ro, nâng cao sự thành công của dự án.



|  |  |
| --- | --- |
| Ưu điểm | Nhược điểm |
| * Phát hiện ra lỗi sớm, chi phí sửa lỗi càng giảm * Tìm ra các lỗi quan trọng * Ngăn ngừa các lỗi trong các giai đoạn tiếp theo * Giảm các yêu cầu không cần thiết , lên lịch thiết kế lại. * Phát hiện ra các yêu cầu bị sai , không chính xác hoặc thay đổi yêu cầu. | * Không linh hoạt * Nếu có thay đổi vẫn phải cập nhật và sửa code |

|  |  |
| --- | --- |
| verification | validation |
| * Là quy trình đánh giá sản phẩm nhưng không phải là sp cuối cùng mà là sản phẩm của các giai đoạn dự án để xem sản phẩm đó có thỏa mãn yêu cầu của dự án đó hay không * *phần mềm có thỏa mãn yêu cầu đặc tả hay không* | * Là quy trình đánh giá phần mềm trong quá trình hoặc khi kết thúc phần mềm để xem phần mềm đó có thỏa mãn với yêu cầu của khách hàng hay không . * là phần cuối của dự án. * *Phần mềm có thỏa mãn yêu cầu khách hàng hay không* |

* Incremental Model
* Là sản phẩm sẽ được chia nhỏ theo từng giai đoạn và chạy song song với nhau để phát triển phần mềm (vẫn bao gồm các giai đoạn : phân tích , thiết kế , code , kiểm thử); những yêu cầu có độ ưu tiên cao nhất sẽ được thực hiện sớm nhất. Sau khi hoàn thiện tất cả các giai đoạn thì sẽ kết nối lại với nhau để tạo dự án hoàn chỉnh

|  |  |
| --- | --- |
| Ưu điểm | Nhược điểm |
| * Phần mềm sẽ được đưa đến tay khách hàng sớm trong vòng đời phát triển. * mất ít chi phí nếu như có yêu cầu thay đổi. * Các thay đổi không quá lớn. Khách hàng có thể xem xét và thay đổi sớm trước khi bàn giao * test dễ dàng hơn và những lỗi sẽ nhỏ hơn | * kế hoạch dự án tốt, thiết kế hệ thống tốt. * Kiến trúc hệ thống phát triển lên thì rất khó. * Không được trùng lên nhau. |