1. **TỔNG QUAN VỀ QUY TRÌNH PHÁT TRIỂN PHẦN MỀM:**

Một quy trình tốt và hợp lí luôn tạo ra những sản phẩm đạt tiêu chuẩn. Nó giúp tương tác hóa các hoạt động và yếu tố với nhau một các nhịp nhàng, đem lại hiệu quả.

Có thể cho rằng quy trình phần mềm đem lại chất lượng, năng suất, giá thành phần phềm, từ đó tăng tính cạnh tranh và đem lại lợi nhuận cao cho doanh nghiệp.

**1. Khái niệm**

Quy trình phát triển phần mềm là một tập hợp các hoạt động tổ chức mà mục đích của chúng là xây dựng và phát triển phần mềm.

* Những câu hỏi được đặt ra ở đâu là:
  + Nhân sự: Ai sẽ làm? Ai làm gì?
  + Thời gian: Khi nào làm? Làm mất bao nhiêu thời gian?
  + Phương pháp: Làm như thế nào?
  + Công cụ: Dùng công cụ gì để làm công việc này?
  + Chi phí: Chi phí bỏ ra bao nhiêu? Thu về bao nhiêu? (ước tính)
  + Mục tiêu: Mục tiêu hướng đến là gì?
    - Mỗi loại hệ thống khác nhau thì cần những quy trình phát triển khác nhau.

**2. Các hoạt động cơ bản của quy trình phát triển phần mềm**

* Có 4 thao tác là nền tảng của hầu hết các quy trình phát triển phần mềm:
  + Đặc tả phần mềm: Định nghĩa được các chức năng, điều kiện hoạt động của phần mềm.
  + Phát triển phần mềm: Là quá trình xây dựng các đặc tả.
  + Đánh giá phần mềm: Phầm mềm phải được đánh giá để chắc chắn rằng ít nhất có thể thực hiện những gì mà tài liệu đặc tả yêu cầu.
  + Tiến hóa phần mềm: Đây là quá trình hoàn thiện các chức năng cũng như giao diện để ngày càng hoàn thiện phần mềm cũng như các yêu cầu đưa ra từ phía khách hàng.

**3) Một số mô hình phát triển phần mềm:**

- Mô hình thác nước.

- Mô hình phát triển tiến hoá.

+ Mô hình bản mẫu.

+ Mô hình xoắn ốc.

+ Mô hình tăng trưởng.

- Mô hình lập trình cực đoan XP/Agile.

1. **KHÁI NIỆM:**

**Mô hình thác nước** ([tiếng Anh](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh): *waterfall model*) là một mô hình của quy trình [phát triển phần mềm](https://vi.wikipedia.org/wiki/Ph%C3%A1t_tri%E1%BB%83n_ph%E1%BA%A7n_m%E1%BB%81m), trong đó quy trình phát triển trông giống như một dòng chảy, với các pha được thực hiện theo trật tự nghiêm ngặt và không có sự quay lui hay nhảy vượt pha. Mỗi giai đoạn phải được hoàn thành đầy đủ trước khi giai đoạn tiếp theo có thể bắt đầu. Có 5 pha: phân tích, thiết kế, mã hóa, kiểm thử và bảo trì.

**III) CÁC PHA THỰC HIỆN:**

1. **Phân tích:**

* Đối tác tham gia: khách hàng, người dùng, người phân tích
* Tài liệu yêu cầu: Danh sách yêu cầu, Mô tả chi tiết yêu cầu
* Xác định và phân tích các yêu cầu cho hệ thống

• Chức năng

• Ràng buộc

**2) Thiết kế:**

- Xây dựng giải pháp thiết kế cho yêu cầu phần mềm

- Đầu vào: tài liệu yêu cầu, môi trường triển khai

- Đầu ra: tài liệu thiết kế

- Tài liệu thiết kế:

+ Mô hình kiến trúc hệ thống

+ Mô hình thiết kế chi tiết

Dữ liệu, Thuật toán, Giao diện

**3) Mã hóa:**

* Viết chương trình.
* Kiểm tra, giám sát mã lệnh.
* Gỡ lỗi (*Debugging*).
* Đầu vào: tài liệu thiết kế
* Đầu ra: chương trình thực hiện được tài liệu chương trình

**4) Kiểm thử:**

* Phát hiện và sửa lỗi phần mềm.
* Đảm bảo phần mềm thỏa mãn yêu cầu khách hàng.
* Đầu vào: tài liệu yêu cầu, tài liệu thiết kế, chương trình, tài liệu chương trình
* Đầu ra: tài liệu kết quả kiểm thử

**5) Bảo trì:**

- Đưa hệ thống vào vận hành (triển khai)

- Sửa lỗi phần mềm

- Làm thích nghi phần mềm với môi trường mới

- Thay đổi phần mềm đáp ứng yêu cầu mới.

**IV) ƯU ĐIỂM VÀ NHƯỢC ĐIỂM CỦA WATERFALL:**

**1) Ưu điểm:**

* Xuất hiện sớm nhất
* Mô hình này đơn giản, dễ hiểu và dễ sử dụng.
* Dễ quản lí.
* Bảo trì thuận lợi.
* Hoạt động tốt cho các dự án nhỏ hơn, nơi yêu cầu được hiểu rất rõ
* Vì các giai đoạn của mô hình thác nước cứng nhắc và chính xác, một pha được thực hiện một lần, nó rất dễ dàng để duy trì.
* Các tiêu chí đầu vào và đầu ra được xác định rõ ràng, do đó nó dễ dàng và có hệ thống để tiến hành chất lượng.
* Thấy được trình tự công nghệ từ đầu đến sản phẩm cuối

1. **Nhược điểm:**

- Tách biệt giữa các pha, tiến hành tuần tự.

• Khó tuân thủ tuần tự: dự án lớn thường phải lặp lại

• Khó đáp ứng yêu cầu thay đổi của khách hàng

* Không có phần mềm làm việc được tạo ra cho đến cuối trong suốt chu trình sống. Đòi hỏi khách hàng phải kiên nhẫn.
* Rủi ro cao do sai sót phát hiện muộn có thể là thảm họa
* Không phải là một mô hình tốt cho các dự án phức tạp và hướng đối tượng và dự án dài hạn.

**Vậy khi nào có thể sử dụng mô hình Waterfall?**

• Mô hình này chỉ được sử dụng khi các yêu cầu rõ ràng và cố định.

• Định nghĩa sản phẩm ổn định.

• Hiểu rõ công nghệ.

• Không có yêu cầu mơ hồ

• Nguồn lực phong phú với chuyên môn được yêu cầu có sẵn một cách tự do

• Dự án ngắn hạn.

Trong mô hình thác nước, điều rất quan trọng là đi theo dấu hiệu của các sản phẩm của từng giai đoạn. Sự tương tác từ khách hàng rất ít đến sự phát triển của sản phẩm phần mềm. Một khi sản phẩm đã sẵn sàng sau đó thì đã là lúc demo cho end-user. Một khi sản phẩm được phát triển và nếu có bất kỳ sự cố nào xảy ra thì chi phí sửa các vấn đề như vậy là rất cao, bởi vì chúng ta cần phải cập nhật ở mọi nơi từ tài liệu đến logic và code.

**Quy trình Waterfall cải tiến:**

Mô hình thác nước có thể được cải tiến bằng cách cho phép quay lui khi phát hiện lỗi trong giai đoạn phía trước.

- Có ý kiến cho rằng không có một dự án thực tế nào lại thực hiện dạng mô hình này mặt khác thì lại có các ý kiến trái ngược, rằng hầu hết các dự án đều tiến hành theo mô hình thác đổ. Sự trái ngược đó là kết quả của sự khác nhau trong cách nhìn nhận, đánh giá mô hình này.

- Cách nhìn chặt chẽ coi các bước như các giai đoạn trong khi thực hiện phần mềm, tiếp theo nhau và không có vòng lặp.

- Cách nhìn tổng quát coi các bước chỉ có tính khái niệm, cho phép chúng chồng nhau, cho phép vòng lặp - quay lại các bước trước.

**V) SO SÁNH GIỮA MÔ HÌNH WATERFALL VÀ AGILE:**

|  |  |
| --- | --- |
| ***WATERFALL*** | ***AGILE*** |
| **Dễ dàng theo dõi và quản lí**: Vì Waterfall rất đơn giản, có cùng cấu trúc và cách tiếp cận ở mọi giai đoạn, nên dù dự án bạn đang thực hiện là gì, bạn vẫn thấy dễ hiểu, dễ theo dõi và dễ quản lý. Nó cũng rất cứng nhắc trong vấn đề về các thành phẩm và bản sửa đổi (mỗi giai đoạn đều có một danh sách cụ thể các công việc / hoạt động / cột mốc cần được thực hiện trước khi chuyển sang giai đoạn tiếp theo), vì vậy dự án rất dễ để kiểm soát. | **Khó lên kế hoạch**: Khá là khó để xác định rõ ràng thời gian bàn giao sản phẩm cuối cùng, vì dự án được chia nhỏ thành các phần khác nhau và mỗi phần lại có thời gian bàn giao riêng biệt. Trên hết, các nhà quản lí dự án thường hay thay đổi thứ tự ưu tiên, dẫn đến việc kế hoạch ban đầu bị rối tung lên và phải bổ sung thêm các sprint vào thời hạn lúc đầu. |
| **Thực hiện thay đổi khó:** các pha được thực hiện theo trật tự nghiêm ngặt | **Thực hiện thay đổi dễ dàng**: Bởi vì dự án được chia thành các phần nhỏ, riêng biệt, không phụ thuộc lẫn nhau, nên những thay đổi được thực hiện rất dễ dàng, ở bất kỳ giai đoạn nào của dự án. |
| **Tài liệu hóa mọi thứ:** Waterfall chú trọng vào tài liệu trong từng giai đoạn, giúp bạn dễ dàng truyền đạt cách tiếp cận của mình đến khách hàng và các bên liên quan. Hơn nữa, khách hàng có thể tham khảo tài liệu mỗi khi họ cần thêm thông tin chi tiết (như các thông tin về chi phí, qui mô, thời gian, v.v...) | **Ít tài liệu hướng dẫn**: Vì Agile thay đổi rất nhiều nên các tài liệu thích hợp cũng thường bị bỏ qua, vì không xác định rõ được kỳ vọng và thành phẩm ngay từ đầu. Mặc dù tài liệu không phải là yếu tố quan trọng nhất, nhưng chúng vẫn rất cần thiết. |
| **Thời hạn dài:** Bạn không thể tự lựa chọn sẽ làm gì vào thời điểm nào, bạn cần theo đúng quy trình của dự án và phải hoàn thành một giai đoạn trước khi bắt đầu giai đoạn tiếp theo, nên bạn sẽ mất rất nhiều thời gian để đạt được kết quả cuối cùng, hoặc thậm chí chỉ để giới thiệu một bản demo cho khách hàng. | **Bàn giao nhanh hơn**: Việc chia nhỏ dự án cho phép bạn tập trung hơn vào chất lượng và bàn giao kịp thời từng phần riêng lẻ. Đội ngũ cũng có thể tiến hành kiểm tra theo từng phần, xác định và sửa chữa vấn đề nhanh hơn, nhờ đó việc bàn giao công việc sẽ nhất quán và thành công hơn. |
| **Xác định dữ liệu ngay từ đầu:** Giai đoạn đầu của phương pháp Waterfall có lẽ là thời điểm quan trọng và khó khăn nhất. Nó đòi hỏi phải có sự giao tiếp với khách hàng và các bên liên quan, phải hiểu chính xác kết quả cuối cùng mà họ muốn đạt được là gì | **Không cần phải nắm mọi thông tin ngay từ đầu, chú ý đến phản hồi của khách hàng và người dùng** |