

HỌC VIỆN KỸ THUẬT QUÂN SỰ

VIỆN CÔNG NGHỆ THÔNG TIN TRUYỀN THÔNG

**BÁO CÁO THỰC TẬP**

**Quản lý trang thiết bị công nghệ thông tin**

**cấp trung, lữ đoàn**

|  |  |  |
| --- | --- | --- |
| **Giáo viên hướng dẫn** | **: 1//** | **- Trần Hữu Phi** |
|  |  |  |
| **Nhóm thực hiện** | **: 3/** | **- Trần Hồng Quân** |
|  | **: 4/** | **- Nguyễn Thế Anh** |
|  | **: 4/** | **- Đỗ Đức Trung** |
|  | **: 3/** | **- Phạm Quang Duẩn** |

**Vĩnh Phúc, tháng 01 năm 2024**

1. Lập nhóm 3 – 4 người.
2. Tìm hiểu quy trình, công nghệ trong phát triển dự án phần mềm (gồm có: DevOpt, Docker, Git,...).
3. Tự chọn một dự án trong danh sách ở dưới, hoặc có thể tự đề xuất.
4. Lập Kế hoạch thực hiện dự án (theo mẫu), chỉ rõ ai phụ trách công việc gì, tài liệu gì, mục tài liệu nào.
5. Xác định danh mục các tài liệu cần phải có của dự án (đặc tả yêu cầu, đặc tả thiết kế, mã nguồn,…). Trong kế hoạch thực hiện dự án phải chỉ rõ thành viên nào phụ trách tài liệu nào (có thể nhiều người cùng phụ trách một tài liệu, mỗi người một mục).
6. Lập tài liệu đặc tả yêu cầu phần mềm (theo mẫu). Trong phần phân tích quy trình nghiệp vụ, chỉ cần phân tích những nghiệp vụ quan trọng nhất, đặc trưng nhất đối với phần mềm. Lập tài liệu thiết kế và tài liệu testcase.
7. Triển khai dự án phần mềm.
8. Lập báo cáo thưc tập về tổng thể về nội dung thực hiện (30-40 trang trình bày tổng quan quy trình, công nghệ, áp dụng và các kết quả đạt được)
9. Hoàn thiện bổ sung: Nếu có dự án tương đối hoàn chỉnh (Phần mềm/ ứng dụng hoàn chỉnh).
10. Hoàn thiện và nộp tài liệu cần thiết: Kế hoạch dự án, Tài liệu đặc tả yêu cầu, Tài liệu thiết kế, Tài liệu testcase, Báo cáo thực tập.

# Mở đầu

Giới thiệu về đề tài thực tập: Trình bày mục tiêu, phạm vi và giới hạn của đề tài thực tập.

Đề tài thực tập: Quản lý trang thiết bị công nghệ thông tin cấp trung, lữ đoàn

Mục tiêu của dự án phần mềm là quản lý trang thiết bị CNTT tại đơn vị cấp trung lữ đoàn. Chức năng chính của phần mềm là cho phép người quản trị hệ thống thông qua việc tạo tài khoản và phân quyền người dùng; cho phép người quản lý hồ sơ và người quản lý chỉ huy các cấp xem, thêm mới, sửa được hồ sơ thông tin, báo cáo. Giúp cho quá trình quản lý quản lý trang thiết bị cũng như việc tiếp cận thông tin hệ thống của người quản lý hồ sơ và người quản lý chỉ huy các cấp được thuận tiện, nhanh chóng và chính xác hơn.

Phạm vi gồm các công việc chính phân tích yêu cầu, thiết kế, lập trình, kiểm thử, triển khai phần mềm quản lý trang thiết bị CNTT tại đơn vị cấp trung lữ đoàn. Dự án là kết quả của việc khảo sát yêu cầu, tìm hiểu các tài liệu liên quan đến các chức năng cần thiết của phần mềm quản lí trang thiết bị CNTT tại đơn vị cấp trung lữ đoàn. Dự án là cơ sở giao tiếp giữa các thành viên trong nhóm, là căn cứ để xây dựng các tài liệu khác: tài liệu thiết kế phần mềm, tài liệu thiết kế CSDL, tài liệu testcase…

Đồng thời, Nghiên cứu, tìm hiểu quy trình phát triển và vận hành phần mềm; Sử dụng thành thạo công cụ quản lý dự án phần mềm; Thực hiện lên kế hoạch, phân công công việc cho các thành viên trong nhóm; Xây dựng kịch bản kiểm thử; Lựa chọn một dự án phần mềm/ ứng dụng để áp dụng lý thuyết và quy trình quản lý, phát triển và vận hành; Triển khai trên máy chủ online.

Giới thiệu về DevOps, CI/CD: Trình bày khái niệm, lợi ích, các nguyên tắc và các công cụ của DevOps, CI/CD.

DevOps là một phương pháp tiếp cận phát triển phần mềm kết hợp phát triển (Dev), vận hành (Ops) và các hoạt động liên quan khác. DevOps tập trung vào việc giảm thiểu khoảng cách giữa các đội phát triển và vận hành, đồng thời tự động hóa các quy trình phát triển và triển khai. CI/CD là một tập hợp các thực tiễn và công cụ giúp tự động hóa quy trình phát triển phần mềm từ khi viết mã đến khi triển khai.

DevOps và CI/CD là hai khái niệm có mối quan hệ chặt chẽ với nhau. DevOps là một phương pháp tiếp cận phát triển phần mềm kết hợp phát triển (Dev) và vận hành (Ops), trong khi CI/CD là một tập hợp các thực tiễn và công cụ giúp tự động hóa quy trình phát triển phần mềm từ khi viết mã đến khi triển khai.

# Chương 1: Tổng quan về quy trình phát triển phần mềm theo hướng DevOps

### 1. Phương pháp DevOps

DevOps là một phương pháp tiếp cận phát triển phần mềm kết hợp phát triển (Dev) và vận hành (Ops). DevOps tập trung vào việc giảm thiểu khoảng cách giữa các đội phát triển và vận hành, đồng thời tự động hóa các quy trình phát triển và triển khai.

Các giai đoạn chính của quy trình phát triển phần mềm theo hướng DevOps

Quy trình phát triển phần mềm theo hướng DevOps có thể được chia thành các giai đoạn chính sau:

* Phân tích và thiết kế
* Xây dựng và kiểm thử
* Tích hợp và triển khai

Giai đoạn phân tích và thiết kế

Giai đoạn phân tích và thiết kế bao gồm các hoạt động sau:

* Tìm hiểu nhu cầu của người dùng
* Phân tích các yêu cầu của người dùng
* Thiết kế hệ thống

Trong giai đoạn này, các đội phát triển và vận hành cần làm việc cùng nhau để hiểu rõ nhu cầu của người dùng và thiết kế hệ thống đáp ứng được nhu cầu đó.

Giai đoạn xây dựng và kiểm thử

Giai đoạn xây dựng và kiểm thử bao gồm các hoạt động sau:

* Viết mã
* Tích hợp mã
* Kiểm thử đơn vị
* Kiểm thử tích hợp
* Kiểm thử hệ thống

Trong giai đoạn này, các đội phát triển sẽ viết mã và tích hợp mã của họ với nhau. Các đội kiểm thử sẽ thực hiện các bài kiểm thử để đảm bảo rằng ứng dụng hoạt động đúng như mong đợi.

### 2. Giai đoạn tích hợp và triển khai

Giai đoạn tích hợp và triển khai bao gồm các hoạt động sau:

* Tích hợp liên tục (CI)
* Xây dựng và triển khai liên tục (CD)

Trong giai đoạn này, các đội phát triển và vận hành sẽ làm việc cùng nhau để tự động hóa quy trình tích hợp và triển khai ứng dụng.

CI/CD là một tập hợp các thực tiễn và công cụ giúp tự động hóa quy trình phát triển phần mềm từ khi viết mã đến khi triển khai. CI/CD bao gồm các bước sau:

* Tích hợp mã nguồn (CI): Tự động tích hợp mã nguồn của các nhà phát triển thường xuyên.
* Xây dựng và kiểm thử (CD): Tự động xây dựng và kiểm thử ứng dụng.
* Triển khai (CD): Tự động triển khai ứng dụng lên môi trường sản xuất.

CI/CD là một thành phần quan trọng trong quy trình phát triển phần mềm theo hướng DevOps. CI/CD giúp các tổ chức đạt được nhiều lợi ích, bao gồm:

* Tăng tốc độ phát triển: CI/CD giúp giảm thiểu thời gian cần thiết để tạo ra các bản phát hành mới.
* Tăng chất lượng: CI/CD giúp phát hiện và sửa lỗi sớm hơn, từ đó giảm thiểu rủi ro phát hành phần mềm lỗi.
* Tăng khả năng mở rộng: CI/CD giúp các tổ chức có thể dễ dàng mở rộng quy mô hoạt động phát triển phần mềm.

Các kỹ thuật và công cụ CI/CD

Jenkins là một công cụ tự động hóa mã nguồn phổ biến. Jenkins cung cấp một tập hợp các công cụ và plugin giúp tự động hóa các quy trình CI/CD, bao gồm tích hợp mã nguồn, xây dựng ứng dụng, kiểm thử ứng dụng và triển khai ứng dụng. Jenkins có thể được sử dụng để tự động hóa các quy trình CI/CD cho nhiều loại dự án phần mềm, bao gồm ứng dụng web, ứng dụng di động và ứng dụng máy tính để bàn.

GitHub là một hệ thống quản lý mã nguồn tích hợp CI/CD. GitLab cung cấp một nền tảng tích hợp cho quản lý mã nguồn, phát triển và triển khai ứng dụng. GitLab bao gồm các tính năng CI/CD tích hợp sẵn, giúp các tổ chức có thể dễ dàng bắt đầu với CI/CD.

Docker là một công cụ tạo và quản lý các ứng dụng đóng gói. Docker giúp các tổ chức tạo ra các ứng dụng có thể triển khai được, có thể được vận hành ở bất kỳ môi trường nào. Docker là một công cụ quan trọng trong CI/CD, vì nó giúp các tổ chức tự động hóa quy trình xây dựng và triển khai ứng dụng.

Kubernetes là một hệ thống quản lý container. Kubernetes giúp các tổ chức tự động hóa việc triển khai và quản lý các ứng dụng container. Kubernetes là một công cụ quan trọng trong CI/CD, vì nó giúp các tổ chức triển khai ứng dụng lên môi trường sản xuất một cách nhanh chóng và hiệu quả.

Dưới đây là một số ví dụ cụ thể về cách sử dụng các kỹ thuật và công cụ CI/CD này:

Jenkins có thể được sử dụng để tự động hóa quy trình tích hợp mã nguồn bằng cách sử dụng một plugin như GitLab CI. Plugin này sẽ tự động tích hợp mã nguồn của các nhà phát triển vào một kho lưu trữ GitLab.

GitLab có thể được sử dụng để tự động hóa quy trình xây dựng ứng dụng bằng cách sử dụng một plugin như Gitlab Runner. Plugin này sẽ tự động xây dựng ứng dụng từ mã nguồn đã được tích hợp.

Docker có thể được sử dụng để tự động hóa quy trình triển khai ứng dụng bằng cách sử dụng một plugin như Gitlab Runner. Plugin này sẽ tự động triển khai ứng dụng được đóng gói bằng Docker vào một môi trường sản xuất.

Việc lựa chọn các kỹ thuật và công cụ CI/CD phù hợp sẽ phụ thuộc vào nhu cầu cụ thể của từng tổ chức. Tuy nhiên, các kỹ thuật và công cụ được đề cập ở trên là những lựa chọn phổ biến và hiệu quả.

Chương 2: Phân tích yêu cầu

Phân tích yêu cầu nghiệp vụ: Xác định các nghiệp vụ cần thiết của hệ thống, phân tích các đầu vào, đầu ra, ràng buộc của các nghiệp vụ.

Phân tích yêu cầu hệ thống: Xác định các chức năng, yêu cầu chất lượng, yêu cầu bảo mật, yêu cầu khả năng mở rộng của hệ thống.

Chương 2 được cụ thể hóa trong Tài liệu Đặc tả yêu cầu người dùng kèm theo.

Chương 3: Thiết kế hệ thống

Thiết kế tổng quan hệ thống: Xác định kiến trúc hệ thống, các thành phần của hệ thống, mối quan hệ giữa các thành phần.

Thiết kế chi tiết hệ thống: Thiết kế chi tiết các thành phần của hệ thống, bao gồm:

Thiết kế giao diện người dùng

Thiết kế cơ sở dữ liệu

Thiết kế kiến trúc phần mềm

Chương 3 được cụ thể hóa trong Tài liệu Đặc tả phần mềm, tài liệu thiết kế phần mềm, tài liệu thiết kế cơ sở dữ liệu kèm theo.

Chương 4: Triển khai hệ thống

Xây dựng mã nguồn: Xây dựng mã nguồn cho các thành phần của hệ thống, sử dụng các công cụ và kỹ thuật DevOps.

Tích hợp mã nguồn: Tích hợp mã nguồn của các nhà phát triển thường xuyên, sử dụng các công cụ và kỹ thuật CI/CD.

Xây dựng và triển khai ứng dụng: Xây dựng và triển khai ứng dụng lên môi trường sản xuất, sử dụng các công cụ và kỹ thuật DevOps.

Chương 5: Kết quả và đánh giá

Kết quả thực hiện dự án: Trình bày các kết quả chính của dự án, bao gồm:

Các yêu cầu của dự án đã được đáp ứng như thế nào?

Các công nghệ DevOps, CI/CD đã được sử dụng hiệu quả như thế nào?

Các khó khăn, thách thức trong quá trình thực hiện dự án và cách giải quyết.

Đánh giá kết quả thực hiện dự án: Đánh giá kết quả thực hiện dự án theo các tiêu chí sau:

Tính phù hợp với yêu cầu: Dự án đã đáp ứng được các yêu cầu của người dùng như thế nào?

Tính khả thi: Dự án có thể triển khai và đưa vào sử dụng như thế nào?

Tính hiệu quả: Dự án mang lại hiệu quả như thế nào?

Kết luận

Tóm tắt lại các nội dung chính của báo cáo.

Đưa ra các ý kiến đánh giá, đề xuất.

Một số lưu ý khi viết báo cáo thực tập