Package,release,update daily

1.Tổng quản

Package là đóng gói phần mềm

Release là phát hành

Update daily là cập nhật,phát triển phần mềm

Đề tài này nói về cách các dự án mã nguồn mở đóng gói,phát hành,cập nhật và những vấn đề xung quanh để có thể hoàn thành mục tiêu.

Phần mềm mã nguồn mở là mô hình mà nhà phát triển công bố mã nguỗn ứng dụng,cho phép mọi người xem xét,sửa đổi và sử dụng mã nguồn đó

Với dự án phát triển phần mềm mã nguồn mở thì khi phát triển phần mềm sẽ có những nhánh khác nhau cùng được duy trì đồng thời,phần mềm luôn được cập nhật,sửa chửa.

2.Đánh số bản phát hành

Trước khi đến với cách tạo bản phát hành,mình sẽ giới thiệu về cách đặt tên bản phát hành

Mỗi bản phát hành mới nghĩa là:

+Có các lỗi cũ đã được sửa

+Lỗi mới đã được thêm vào

+Các tính năng mới được thêm vào

+Các tùy chọn cấu hình mới hoặc thay đổi ý nghĩa của việc cấu hình cũ

+Các thay đổi không tương thích có thể đã được ra

Không phải tất cả điều trên đều tốt,vì vậy có những người dùng sẽ cảm thấy lo lắng khi tiếp cận những bản phát hành mới và đặc biệt là họ cảm thấy quen,thỏa mản với phiên bản phần mềm đang sử dụng.

Mục đích của việc đánh số phiên bản có hai mặt là truyền đạt thứ tự của bản phát hành và cũng chỉ ra một cách ngắn gọn nhất về sự thay đổi trong bản phát hành.Khó có một tiêu chuẩn,chiến lược trong việc đánh số bản phát hành.Tuy nhiên có những chiến lược đánh số được mọi nhiều người đồng ý và đem lại tiện lợi cho người dùng

3.Chiến lược đơn giản

*Sự thay đổi sẽ theo mức tăng tiến của phiên bản,*

*2.1.2 ->2.1.4 là sự gia tăng chỉ số vi mô thì cho thấy các bản sửa lõi[không có tính năng mới],còn ->2.2.1 thì gia tăng chỉ số phụ thì sẽ có cải tiến nh còn 3.1.1 gia tăng chỉ sô lớn thì sẽ có thêm chức năng mới,thay đổi lớn*

*4.Chiến lược chẵn/lẽ*

*Một số dự án sử dụng tính chẳn lẽ của thành phần số phụ để biểu thị tính ổn định,chẵn là ổn định,lẻ là không ổn định*

*Sự gia tăng về số vi mô, chính vẫn như chiến lược đơn giản*

V.Thành phần của số phát hành

Số phát hành là nhóm chữ số được phân tách dấu chấm.Ví dụ là nhân linux với chuỗi “0.96:.. “0.99” đãn đến LINUX

Gồm 3 nhãn phổ biến là alpha,beta và **RC (Release Candidate].Alpha là bản dùng thử đầu tiên,tiếp đó là beta** là phiên bản dùng thử để test lỗi còn RC là phiên bản dùng thử trước khi sản phẩm chính thức ra đời còn RTM là phiên bản hoàn thiện

Sự gia tăng của các con số giữa các bản thể hiện sự khác biệt giữa các bản

5.Nhánh phát hành

Một dự án phần mềm mã nguồn mở luôn ở trạng thái phát hành liên tục.Khi có sự xung đột,tranh luận sẽ ảnh hưởng tiến độ phát triển.Việc sử dụng sao lưu sẽ cản trở vì vậy biện pháp cho vấn đề này là nhánh phát hành.Cơ chế tạo nhánh tùy vào hệ thống kiểm soát phiên bản sử dụng.

Ổng định là quá trình đưa nhánh phát hành sang trạng thái có thể phát hành được.Điều này cần phải được cân nhắc kĩ càng.

Việc sở hữu bản phát hành có thể chia thành một người độc tài sở hữu,nghĩa là có tranh luận,thảo luận nhưng người so sẽ có quyền hạn nhất định hoặc là dùng bỏ phiếu thay đổi,công bằng như nhau.Nếu dự án chọn bỏ phiếu sẽ có tệt gọi là status,votes

Có dạng như sau:

\* r2401 (issue #49)  
Prevent client/server handshake from happening twice.  
Justification:  
Avoids extra network turnaround; small change and easy to review.  
Notes:  
This was discussed in http://.../mailing-lists/message-7777.html  
and other messages in that thread.  
Votes:  
+1: jsmith, kimf  
-1: tmartin (breaks compatibility with some pre-1.0 servers;  
admittedly, those servers are buggy, but why be  
incompatible if we don't have to?)

Trong trường hợp này, sự thay đổi đã nhận được hai phiếu thuận, nhưng bị phủ quyết bởi tmartin, người đã đưa ra lý do phủ quyết trong một ghi chú trong ngoặc đơn.

6.Quản lý phát hanh

Bất kỳ nhà phát triển nào cũng có thể thực hiện quá trình hợp nhất thực tế (xem Merge (còn gọi là Port)) các thay đổi được phê duyệt trong nhánh phát hành. Không cần phải có một người có nhiệm vụ Merge các thay đổi; nếu có nhiều thay đổi, tốt hơn hết là bạn nên chia sẻ gánh nặng ra xung quanh.

7.Đóng gói

Hình thức chuẩn mực để phân phối phần mềm miễn phí là mã nguồn. Điều này đúng bất kể phần mềm thường chạy ở dạng nguồn (tức là có thể được thông dịch, như Perl, Python, PHP, v.v.) hay cần được biên dịch trước (như C, C++, Java, v.v.).Có tiêu chuẩn khá nghiêm ngặt và việc đóng gói,có những sai lệch nhưng là ngoại lệ chứ không là quy luật

8.Định dang

Mã nguồn phải được gửi ở các định dạng tiêu chuẩn để vận chuyển,với Unix và các hệ tương tự là sử dụng định dạng Tar.Với windows là định dạng zip

.Tên và bố cục

Tên của gói nên bao gồm tên của phần mềm cộng với số phát hành, cộng với hậu tố định dạng phù hợp với loại lưu trữ.Và tuân thủ các quy tắc nhưu viết hoa,kèm kí hiệu phiên bản như alpha,be ta. Ví dụ, Scanley 2.5.0, được nén vào Unix bằng GNU-Zip (gzip), trông như sau:

scanley-2.5.0.tar.gz

scanley-2.3.0-alpha1.tar.gz

scanley-2.3.0-alpha2.tar.gz

hoặc sử dụng zip để nén trên Windows:

scanley-2.5.0.zip

Sau khi giải nén,sẽ tạo thư mục với được sắp xếp theo một bố cục nhất định.Gồm các file như readme,install hoặc COPYING hoặc LICENSE cung cấp các điều khoản phân phối phần mềm,hoặc new là những gì mới trong phiên bản này.Tên của gói dự án

9.Biên soạn và cài đặt

Đối với phần mềm cần được biên dịch hoặc cài đặt từ mã nguồn ta phải tuân thủ các tiêu chuẩn để có thể thưc hiện.Dễ dàng thấy ta cần phải cài đặt môi trường,tuân thủ ngôn ngữ, và lẫn framwork

10.Gói nhị phân

Mặc dù các gói mã nguồn được phát hành chính thức, hầu hết người dùng sẽ cài đặt chúng từ các gói nhị phân được cung cấp bởi cơ chế phân phối phần mềm của hệ điều hành hoặc được mua thủ công từ trang web của dự án hoặc một số bên thứ ba. "nhị phân" ở đây không nhất thiết có nghĩa là "đã biên dịch"; Nó chỉ đơn giản đề cập đến bất kỳ dạng gói cấu hình sẵn nào cho phép người dùng cài đặt nó trên máy tính của họ mà không phải trải qua quá trình xây dựng và cài đặt dựa trên mã nguồn thông thường. Trên RedHat GNU/Linux, đó là hệ thống RPM; Trên Debian GNU/Linux, đó là hệ thống APT (.deb); Trên MS Windows, nó thường là. Tệp .MSI hoặc tệp .exe tự cài đặt.

11.Kiểm tra và phát hành

Sau khi tạo ra bản gốc từ nhánh phát hành ổn định,quá trình phát hành công khai bắt đầu.Trước khi bản ấy được công bố,nó được kiểm thử và chấp nhận bởi ít nhất 3 nhà phát triển

Nhà phát triển sẽ tải bản và kiểm tra các tiêu chí,,nếu vượt qua sẽ ký điên tử vào bằng cách sử dụng GnuPG,hoặc PGP.Càng nhiều nhà phát triển kí thì càng tin cậy

Khi được chấp nhận,sẽ được đặt để tải về kèm chữ ký số và mã kiểm tra MD5/SHA1. Mục tiêu của quá trình ký và kiểm tra này là cung cấp cho người dùng một cách để xác minh rằng bản sao họ nhận được không bị thay đổi độc hại

12. Phiên bản phát hành dùng thử

Đối với các bản phát hành quan trọng có nhiều thay đổi thì sẽ được ra bản dùng thử phát hành trước.Để kiểm tra và khắc phụ các sự cố,việc phát hiện và phát hành bản mới cho đến khi chấp nhận được,sự khác biết giữa bản dùng thử cuối cùng và bản chính thức là tên.Các phiên bản alpha,beta,rc cảnh báo cho người dùng và sẽ thu thập phản hồi để phát triển

13.Thông báo phát hành

Thông báo phát hành cũng như thông báo sự kiến.Nhưng cần phải lưu ý khi cung cấp địa chỉ để tải bản thì cũng phải cung cấp MD5/SHA1 và chữ kí số để đảm bảo bảo mật.Dùng hai cặp khóa và thuật toán băm để đảm bảo tính toán vẹn của tệp.Kí bằng khóa riêng và xác minh bằng khóa công khai.Sẽ giới thiệu những thay đổi,nâng cấp cho bản và phát hành và phải kèm lời cảm ơn nhóm phát triển,người thử nghiệm..

14.Phát hành bản cập nhật bảo mật

Bản cập nhật bảo mật là bản phát hành chỉ nhằm mục đích khác phục lỗ hổng bảo mât.Bảng cập nhật bảo mật không được kiểm tra công khai trước khi được phát hành.Vì thiếu việc kiểm tra,nên bản cập nhật bảo mật chỉ khắc phục lỗ hủng để tránh gây thêm lỗi

15.Phát hành và phát triển hàng ngày

Vì các dự án luốn được phát hành và phát triển hàng ngày nên phải tuân thủ kỉ luật ơhats triển phần mềm.Khi thay đổi bất cứ điều gì liên quan đến dự án ta phải commit,cam kết sự thay đổi

Nếu sự thay đổi quá lớn ta cần phải tách nhỏ và mới commit.Việc này không chỉ giúp ổn định bản phát hành,tính logic,mà còn giúp dễ kiểm soát,xem xét,khôi phục nếu gặp lại sự cố

16.Dự kiến phát hành

Một trong những sự khác biệt giữa các dự án mã nguồn mở và các dự án mã nguồn đóng trong quá khứ là việc lập kế hoạch phát hành.Mã nguồn đóng có thời gian chặt chẽ hơn khi phải đáp ứng nhu cầu của khách hàng trong khi các dự án mã nguồn mở thì tự do hơn,thúc đẩy bởi đam mê.

Ngày nay,nhiều dự án mã nguồn mở được tài trợ bởi các công ty và ngày càng chịu ảnh hưởng nhiều hơn bởi văn hóa công ty nên ý thức về thời hạn.Có điểm tốt nhưng sẽ gây ra xung đột giữa những nhà phát triển có lương và những người phát triển theo đam mê.Không có giải phát chung cho vấn đề này mà chỉ có thể thảo luận và thỏa hiệp

17.Thực tế

Quá trình đóng gói, phát hành, và cập nhật các phiên bản mã nguồn mở của các dự án nổi tiếng thường được thực hiện theo mô hình quản lý phiên bản và quy trình phát triển linh hoạt. Dưới đây là một trình bày tổng quan về cách một số dự án mã nguồn mở hàng đầu thực hiện các hoạt động này

Linux Kernel:

Đóng gói: Nhân Linux không có một hình thức đóng gói "chuẩn" như các ứng dụng thông thường. Thay vào đó, hạt nhân thường đi kèm với các bản phân phối Linux (distros) như Ubuntu, Fedora, hoặc Debian. Mỗi bản phân phối có quy trình đóng gói riêng để tích hợp nhân vào hệ thống của họ.

Phát hành: Linux Kernel có các phiên bản chính thức (major release) được phát hành hàng đầu năm và hàng tháng. Mỗi phiên bản đều được kèm theo nhiều bản vá và cải tiến tính năng.

Cập nhật: Cập nhật kernel thường được thực hiện thông qua hệ thống quản lý gói của bản phân phối, như apt cho Ubuntu hoặc dnf cho Fedora. Người dùng có thể nhận cập nhật tự động hoặc tự cập nhật theo yêu cầu.

WordPress:

Đóng gói: WordPress được đóng gói dưới dạng một mã nguồn mở và có thể tải về từ trang web chính thức. Ngoài ra, nhiều bản phân phối hosting cung cấp cài đặt tự động WordPress.

Phát hành: WordPress có các phiên bản chính thức định kỳ. Các bản vá và cập nhật nhỏ được phát hành khi cần thiết để sửa lỗi bảo mật hoặc cải thiện tính ổn định.

Cập nhật: Cập nhật WordPress có thể được thực hiện thông qua bảng quản trị trực tuyến, nơi người dùng có thể nhận thông báo về các bản cập nhật mới và thực hiện cập nhật với một cú click.

6. Node.js:

Đóng gói: Node.js sử dụng npm (Node Package Manager) để quản lý các gói. Gói Node.js có thể được cài đặt thông qua lệnh npm install. Các dự án Node.js thường có một tập tin package.json mô tả các phụ thuộc và cấu hình dự án.

Phát hành: Node.js có lịch phát hành LTS (Long-Term Support) và Current định kỳ. Các phiên bản LTS được hỗ trợ lâu dài và thường được chọn cho môi trường sản xuất.

Cập nhật: Cập nhật Node.js thường được thực hiện thông qua npm, với lệnh npm install -g node để cập nhật toàn bộ hệ thống.

Các kiến thức thêm về github

GIT HUB

+Source control là hệ thống giúp lưu trữ mọi thay đổi của souce code,hỗ trợ nhiều người làm việc,xem thay đổi code,đưa code về phiên bản cũ

+Git là 1 source control,toàn bộ code và histrory được lưu trữ trên máy người dùng,còn 3 khái niệm quan trong là repo,commit,branch

-Repo là 1 dự án chứa tất cả code,gồm local trên máy và remote trên máy chủ,commit là khi thay đổi,branch là nhánh

+Github là 1 dịch vụ cung cấp lưu trữ code,hỗ trợ nhiều hơn git

Github Desktio là một ứng dụng quản lý các dự ản trên github một cách dễ dàng thuận tiện

Các lệnh git cơ bản hay dụng

+git init:Tạo rp trên máy

+git clone:Lấy rp trên mạng về

+pull:đồng bộ từ trên mạng về

+add:Thêm,sửa

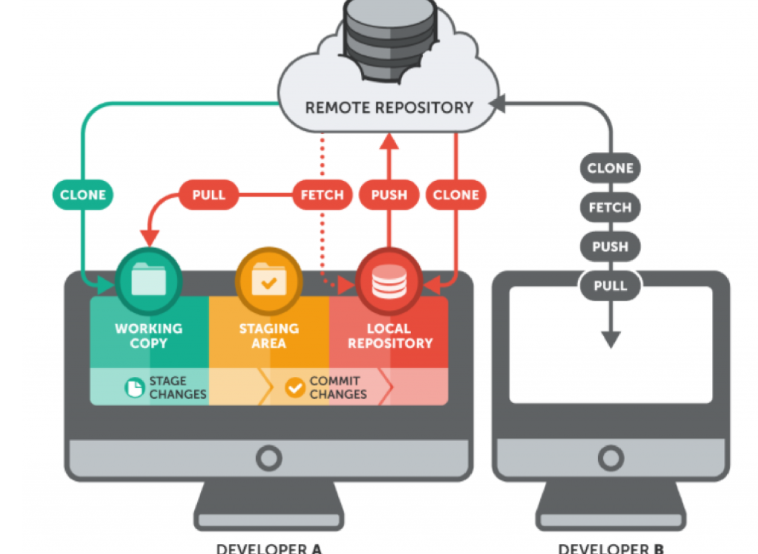
+commit:thêm commit

+push:đồng bộ từ local lên mạng

+Git branch:Tạo nhánh mới,ok rồi thì merge vào master

+git checkout:Đổi nhánh

+log:Xem trạng thái



Bugzilla là hệ thống phần mềm theo dõi lỗi mã nguồn mở, cho phép cá nhân hoặc nhóm các nhà phát triển theo dõi các lỗi xác suất xảy ra trong dự án của họ một cách hiệu quả

Đội ngũ kiểm tra chất lượng phần mềm QC ( Quality Control) có trách nhiệm quản lí hệ thống này

Chức năng – Bugzilla giúp quản lí quy trình sửa lỗi phần mềm miễn phí. – Cho phép quản lí quy trình hoạt động cũng như tiến độ test lỗi của từng dự án – Cho phép nhiều user làm việc cùng lúc, dễ tìm kiếm và phân bổ công việc cho từng thành viên – Cập nhập thông tin cho từng thành viên tham gia dự án thông qua chức năng gửi thư điện tử

Demo

+Tao project SCCC,Type scipt

+TypeScipt là phiên bản nâng cao của javascipt,nhiều chức năng,cải tiến hơn, nền tảng cũng là js

Hoạt động rộng rãi cho angular và node jss

+css preprocessor là ngôn ngữ kịch bản mở rộng của CSS được biên dịch thành cú pháp,là chương trình tiền xử lý Css peprocessor.Sass,scss là chương trình tiền xử lý preprocessor,Giúp bạn viết Css theo cách ngôn ngữ lập tình đơn giản,dễ bảo trì,phát triển

+Caì node js để có thể chạy được javascript phía máy chủ,còn cung cấp gói npm để quản lý các thư viện mà ta dùng,còn yarn tương tự như vậy nhưng hiệu quả hơn

Serach nodejs trên google

+Cài giao diện thư viện

Npm install - g @angular/cli

+Fix không tạo được work space =>ng new shophandmade

<https://stackoverflow.com/questions/37991556/ng-is-not-recognized-as-an-internal-or-external-command>

+routing để chuyển qua lại giữa các màn hình

+Bật cmd ,cd vào thư mục angular mới tạo rồi bật yarn star thì mới chạy được

+tạo component home

ng generate component home

+Đóng gói ứng dụng bằng ng build

Sẽ tạo một thư mục ddisst/ chứa ứng dụng đã được biên dịch và tối ưu hóa

DEMO:

+Dùng frame-work angular dùng đồ án chuyên ngành để demo dự án shop handmade ,hiện thực front end.

+ng build tạo ra một file thư mục,thêm các file để hiển thực như COPYRIGHT,LICENSE.

+Đóng gói lại thành một file nén.Đánh số phiên bản

+Up lên github

+Chỉnh sửa lại

+Làm tương tự như vậy

+Up lên git hub lại