**HƯỚNG DẪN SỬ DỤNG GIT CHO PROJECT MANAGER**

1. **Cài đặt Git và cấu hình GitHub**

Link tham khảo: <http://help.github.com/win-set-up-git/>

* 1. **Dowload và cài đặt Git**

Thực hiện như trong link tham khảo.

* 1. **Cấu hình SSH key**
* Sử dụng tài khoản (ihbdev/namhieubienthanh68) để add key SSH.
  1. **Thiết lập thông tin cá nhân**

Thực hiện như trong link tham khảo.

1. **Khởi tạo dự án trên Github**

* Tạo dự án trên release repository sử dụng tài khoản (ihbihbvietnam/namhieubienthanh68)

1. **Triển khai môi trường làm việc tại local**
   1. **Tạo thư mục làm việc**

Theo chuẩn, bạn nên tạo thư mục MANAGER\_PROJECT trong htdocs để lưu trữ các dự án

* 1. **Clone và cấu hình project**
* Tạo dự án trên dev repository bằng cách fork dự án trên release repository.
* Clone dự án về local:

$git clone [git@github.com:ihbdev/{Project\_Name}.git](mailto:git@github.com:ihbdev/%7bProject_Name%7d.git)

Chú ý: Để thay đổi URL của remote repository

$git remote set-url origin git@github.com:ihbvietnam/{Project\_Name}.git

* Nếu dự án có sử dụng phần CORE tạo một remote repository để update phần CORE cho dự án

$git remote add core git@github.com:ihbvietnam/{Core\_Name}.git

$git fetch core

$git merge core/{Branch\_Name\_Version}

* Phân chia dự án và khởi tạo branch cho các hạng mục công việc

$git push origin origin:refs/heads/{Branch\_Name}

Chú ý:

* + Xóa remote branch trên local
    - $git branch –r –d origin/{Branch\_Name}
  + Xóa branch trên server
    - $git push origin :refs/heads/{Branch\_Name}
* Tạo file cấu hình .gitignore trong thư mục CODE, nội dung như sau:

assets/\*

protected/runtime/\*

upload/\*

.settings/\*

.buildpath

.project

error\_log

.gitignore

* Tạo file cấu hình .gitignore trong thư mục gốc, nội dung như sau:

.settings/\*

.buildpath

.project

error\_log

.gitignore

* Xóa chỉ số các file trong thư mục upload

$git rm –r –-cached upload/\*

* Bổ sung các thư mục sau vào thư mục DEV\_PROJECT/{Branch\_Name}:

assets/

protected/runtime

1. **Pull and push to remote repo**

* Pull dữ liệu về:

$git pull hoặc $git fetch origin & $git merge origin/{Branch\_Name}

* Merge branch master với các branch thành phần.
* Xử lý conflict do việc merge giữa các branch
  + Sử dụng lệnh $git diff để xem lại những file và những đoạn trên file bị conflict
  + Chỉnh sửa file đó để giải quyết conflict rồi dùng lệnh $git add và $git commit để commit lại.
  + Các lệnh hữu dụng khác:
    - $git reset --hard origin/{Branch\_Name} xóa bỏ mọi thay đổi do thao tác merge tạo ra và quay trở lại trạng thái commit gần nhất.
    - $git reset –soft HEAD^ origin/{Branch\_Name} cũng xóa bỏ mọi thay đổi do thao tác merge tạo ra nhưng vẫn giữ staging area để tiến hành chỉnh sửa.
    - $git checkout –ours {path/to/file} xử lý confict bằng cách giữ file của branch hiện tại.
    - $git checkout –theirs {path/to/file} xử lý conflict bằng cách lấy file của branch mà mình merge.
* Pull dữ liệu của phần CORE về:

$git fetch core & $git merge core/{Branch\_Name}

* Xử lý conflict do việc pull phần core về
* Pull request
* Merge trên release repository