**Tài khoản quản lý**

* Kho code chính của OpenCPS (OpenCPS Repository) được quản lý thông qua tài khoản của tổ chức [VietOpenCPS Organization](https://github.com/VietOpenCPS).
* Tài khoản của tổ chức OpenCPS được quản lý bởi Ban quản trị OpenCPS và đội ngũ phát triển code chính.

**Nguyên tắc phát triển code**

* Tự do đóng góp: Tất cả mọi người dù là người quản lý kho code, thành viên đội code hay những người đóng góp thuộc các tổ chức, công ty, các cá nhân... đều có thể đóng góp phát triển code một cách tự do.
* Ghi công đóng góp: Việc phát triển code phải thực hiện trên kho code riêng, không được code trực tiếp trên kho code chính của OpenCPS (kể cả tài khoản của người quản lý kho code).
* Kiểm duyệt chéo: Những người quản lý kho code sẽ chịu trách nhiệm việc kiểm duyệt code được đóng góp vào kho code chính. (Nên hạn chế việc tự duyệt code của mình trên kho code chính).

**Quy trình tham gia phát triển code**

* Để tham gia phát triển code, chỉ cần 1 tài khoản trên Github. Sau đó truy cập kho code của OpenCPS, ấn nút Fork để copy project có sẵn thành project của mình, rồi sau đó ta có thể tự do chỉnh sửa project đó của mình.
* Chỉnh sửa xong thì ấn nút Pull Request để gửi yêu cầu đóng góp lên cho những người quản lý kho code.
* Những người quản lý kho code sẽ kiểm tra và phê duyệt các đóng góp này. Nếu đóng góp chưa đạt hoặc cần chỉnh sửa thì những người quản lý có thể viết góp ý ngay tại yêu cầu đóng góp và trả lại để người đóng góp chỉnh sửa theo góp ý và gửi lại sau khi đã sửa xong. Quá trình này diễn ra cho đến khi yêu cầu đóng góp được chấp nhận, đóng góp đó sẽ được trộn (merge) vào các nhánh phù hợp với dự án được đóng góp.
* Để tăng hiệu suất quản lý, toàn bộ công việc chỉnh sửa, xử lý xung đột... trước khi gửi đóng góp lên sẽ do người đóng góp xử lý. Người quản lý kho code sẽ chỉ việc kiểm tra và trả lời chứ không phải mất công để sửa lại code được đóng góp.

**Gợi ý cho các doanh nghiệp / nhóm tham gia phát triển code**

Nếu doanh nghiệp của bạn có nhiều nhân viên tham gia phát triển code thì thay vì tham gia dự án dưới tư cách của cá nhân, các bạn có thể tham gia phát triển dự án dưới tư cách của doanh nghiệp (trường hợp doanh nghiệp trả lương cho nhân viên để phát triển code của OpenCPS). Dưới đây là hướng dẫn:

* Đại diện doanh nghiệp đăng ký tài khoản code, tạo tổ chức mang tên doanh nghiệp và add các thành viên vào tổ chức của mình.
* Đại diện doanh nghiệp fork dự án về kho code của doanh nghiệp.
* Các thành viên trong doanh nghiệp fork dự án từ kho code của doanh nghiệp về kho code của cá nhân. Nhân viên doanh nghiệp Pull Request vào kho code của doanh nghiệp sau đó doanh nghiệp Pull Request vào kho code chung của OpenCPS.

Theo hình thức này, một cá nhân vừa có thể tham gia trực tiếp vào dự án của OpenCPS (phát triển ngoài giờ làm việc) và tham gia dưới tư cách là nhân viên của doanh nghiệp (phát triển trong giờ làm việc).

**Các chú ý**

* Trong quá trình làm phải commit thường xuyên lên nhánh develop, khi xong xuôi thì merge/rebase vào nhánh master. Không nên chờ đến khi có kết quả cuối cùng mới commit lên kho code chính, vì như thế sẽ gia tăng nguy cơ xung đột code giữa các thành viên trong nhóm và giữa các nhóm với nhau.
* Quá trình phát triển, kho cá nhân của bạn (fork) trên GitHub có thể cũ so với kho gốc chính của dự án (upstream). Tham khảo các thủ tục bên dưới để đồng bộ kho local với kho upstream rồi push lên kho cá nhân (fork) trên GitHub.

**Cấu hình upstream (để có thể đồng bộ các thay đổi với kho gốc chính của dự án)**

* Kiểm tra các kho remote hiện tại: $ git remote -v
* Thêm kho gốc chính thành upstream: $ git remote add upstream git@github.com:VietOpenCPS/OpenCPS.git
* Kiểm tra lại lần nữa (sẽ thấy kho upstream vừa được thêm vào)

**Đồng bộ code từ upstream (kho gốc chính của dự án)**

* Lấy các thay đổi từ kho gốc về: $ git fetch upstream => các thay đổi trong nhánh developsẽ được lấy về tại upstream/develop
* Chuyển sang nhánh develop trên kho local: $ git checkout develop
* Trộn các thay đổi trong upstream/develop vào master: $ git merge upstream/develop
* Đẩy các thay đổi vừa trộn lên kho cá nhân trên GitHub: $ git push

**Cấu trúc tổ chức kho code**

Cấu trúc tổ chức kho code của OpenCPS tuân theo [mô hình phân nhánh Git](http://nvie.com/posts/a-successful-git-branching-model/) như hình dưới:



* Hai nhánh (branch) đi suốt chiều dài phát triển code là Master và Develop. Trong đó:

\* Nhánh chính là nhánh Master, nhánh này luôn đảm bảo rằng code được lưu trữ trên đó là phiên bản chính thức mới nhất đang được phát hành. \* Nhánh Develop là nhánh được cập nhật liên tục các đóng góp của tất cả mọi người ở mọi thời điểm. Nhánh này sẽ tiếp nhận các đóng góp của mọi người gửi đến thông qua việc tiếp nhận Pull Request cũng như tiếp nhận việc nhập các nhánh khác vào (integration branch). Khi nhánh Develop đạt độ chín mùi, nó sẽ được nhập vào nhánh Master đồng thời được dán nhãn phiên bản và quá trình phát hành phiên bản được tiến hành. \* Ngoài 2 nhánh trên, kho code còn có các nhánh hỗ trợ phát triển, và nó chỉ tồn tại trong một giai đoạn nào đó của dự án nhằm phục vụ những mục đích nhất định, gồm có các nhánh sau: \* Feature branches - Các nhánh tính năng \* Release branches - Các nhánh hỗ trợ phát hành phiên bản \* Hotfix branches - Các nhánh hỗ trợ vá lỗi nhanh \* Để đánh dấu các mốc quan trọng (ví dụ các phiên bản được phát hành), ta sử dụng các tag. Các tag này sẽ giúp truy cập nhanh đến kho code tại thời điểm diễn ra dấu mốc sự kiện được đánh dấu.

**Phát hành phiên bản mới (dành cho Release Engineering team)**

Việc phát hành OpenCPS được thực hiện bằng cách tagging trực tiếp trên nhánh master [Kho git gốc](https://github.com/VietOpenCPS/opencps) theo qui trình sau:

* Clone kho gốc về: git clone git@github.com:VietOpenCPS/opencps.git
* Hoặc pull bản latest trên nhánh "Release Candidate branch" về (nếu đã clone kho local sẵn): git checkout <qa-rc-xxx>; git pull
* Chuyển về & cập nhật nhánh master: git checkout master; git pull
* Thực hiện merge nhánh RC về nhánh master: git merge <qa-rc-xxx>
* Tagging release: git tag <release-number>, rồi nhập tagging message vào, thường là URL đến Release Notes (cần phải viết & đưa lên trước tại: [Thông tin phát hành](https://github.com/VietOpenCPS/doc/wiki#th%C3%B4ng-tin-ph%C3%A1t-h%C3%A0nh)
* Nên thêm tham số -s để ký vào bản phát hành bằng khóa OpenPGP
* Push lại lên kho gốc trên GitHub: git push
* Optional. Merge nhánh RC về nhánh develop: trong nhiều trường hợp, đặc biệt là giai đoạn đầu của dự án, toàn bộ code trên nhánh RC branch sẽ được sử dụng tiếp trong các phiên bản tiếp theo => cần thực hiện merge tương tự như trên cho nhánh develop.
* Lưu ý:
* Release Notes cần được viết riêng trước lên một trang wiki mới và link vào trang wiki Home tại phần [Thông tin phát hành](https://github.com/VietOpenCPS/doc/wiki#th%C3%B4ng-tin-ph%C3%A1t-h%C3%A0nh)
* Sau khi push lên kho git gốc, cần viết thông báo phát hành gửi lên Mailing list. Tham khảo: [Thông báo phát hành OpenCPS 1.0](http://lists.opencps.vn/pipermail/opencps/2016-May/000210.html)