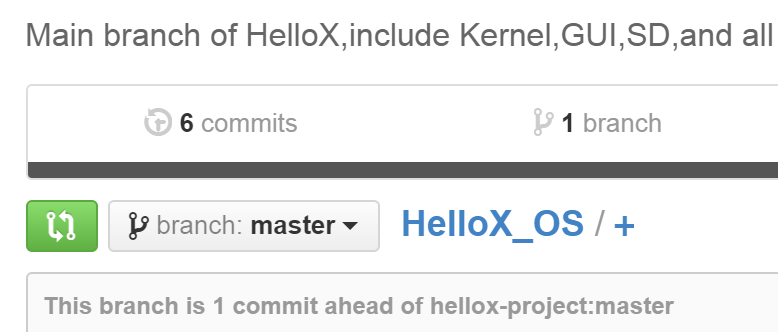
HelloX项目github协同开发指导

本文通过模拟试验，对如何使用github协同开发HelloX项目进行说明。需要注意的是，试用git的基本功能进行协同开发，确保版本代码的一致性，是非常简单易行的。虽然下面描述的步骤比较多，但是大多数的步骤，都是初始配置，只需要执行一次即可。

1. 注册一个github账号，需要访问github.com网站，点击“sign up”进行注册。比如，注册的账号为garryxin；
2. 使用新注册的账号登陆github，然后找到hellox-project所在的位置，即输入<http://github.com/hellox-project>，即可进入HelloX项目的代码库；
3. 点击希望参与开发的具体项目，比如“HelloX\_OS”，进入后，点击fork按钮，这样就拷贝了一个副本，到您的个人数据库中。比如，如果用garryxin进行上述操作，则在garryxin的代码库（<http://github.com/garryxin>）中，就会增加了一个HelloX\_OS的副本；
4. 试用您的账号，登陆github客户端。成功下载github客户端并安装后（比如，github for windows），缺省情况下，在桌面上会创建两个图标-gitshell和github。打开github，点击右上角的配置图标，选择“options…”菜单，在出现的对话框中输入您的账号和密码信息(以及其它相关信息)；
5. 试用gitshell命令行工具，clone一个副本到本地。比如，如果要clone HelloX\_OS项目，则在命令行中输入：git clone <http://github.com/garryxin/HelloX_OS.git>，输入回车，即可clone到本地。这时候，就可脱离gitshell，直接修改本地副本了；
6. 本地副本修改完毕之后，进入gitshell，并切换到副本所在目录下，运行git status，即可看到所做的修改。如果希望把所做的修改进行合并，则运行”git add –all”命令，把所做的修改合并到本地库中。注意all前面是两个横线；
7. 运行git commit –m “modify by XXX”，提交所做的修改。这里的提交，只是提交到本地库中，尚未提交到在线库中；
8. 运行”git push –u origin master”命令，把提交后的本地库，推送到在线库中；
9. 这时候，通过浏览器进入github(注意，是登陆到个人所在的目录，比如github.com/garryxin)，就会看到HelloX\_OS的更新，已经被同步到在线库了。这时候如果希望合并到主版本中，只需要创建一个pull request即可，只要点击下列绿色图标，按照提示输入相关内容，确认即可：



1. 提交成功之后，您所做的工作就完成了。这时候版本管理员会收到pull request的电子邮件，并审视所做的提交。如果没有问题，管理员会直接合并进去，您会收到确认邮件。如果有疑问，管理员会在这个pull request线索上进行回复讨论，必要时您需要进行修改，然后再次提交。

需要说明的是，后续github.com/hellox-project/下的任何代码库，都只作为合并和发布使用，不作为直接开发使用。也就是说，即使是当前的代码管理者garryxin，也必须创建另外一个个人账户（garryxin），通过上述协同开发步骤进行提交。

另外，如果希望跟踪主版本的修改，即当主版本做了修改，能够自动下载到本地库中，可以这样操作：

第一步：设置本地库的上游代码库。具体来说，切换到本地库所在的目录，运行git remote add upstream <http://github.com/hellox-project/HelloX_OS.git>

第二步：运行git fetch upstream，即可把跟踪的上游主版本下载下来，并存在一个叫做“upstream/master”的本地库中。注意，这时候与您的本地库尚未合并，而是两个完全独立的代码库；

第三步：再运行git merge upstream/master，即可把主版本与本地库进行合并。