

# Github 使用报告

LetsGo 安卓应用开发小组

June 5, 2014

## 修订历史

版 本	日 期	描 述	作 者
初始草案	2014 年 6 月 1 日	第一个草案。Github 与 Git 的报告	罗宇凡

## 1 概述

本次 project 我们全程使用 github 来同步我们的源代码。这一做法给我们带来很大的便利，尤其是在多人协同编程的情况下，借助于 github 或类似的分布式版本管理平台会明显提高工作效率，极大降低沟通带来的花费。

## 2 Github 是什么

要解答“github 是什么”这个问题，首先必须了解 git。简而言之，git 就是一个应用程序，它可以安装于 windows, linux 等操作系统上，用户可以像启动普通软件一样启动它。

既然 git 是一个应用程序，那么它肯定有其特定的功能。git 的基本功能是：用户可以通过 git 把自己的本地文件（一般是文本文件，尤其是源代码）上传到“某个地方”，同时用户也可以从“某个地方”把文件下载到本地。“某个地方”可以是任意一台服务器，而 github 就是这样的一台大型免费服务器。

综上所述，编程人士可以在本地安装 git，通过 git 把代码上传到 github；也可以通过 git 从 github 上下载源代码。这就是 github 的主要工作方式，同时这也从感性上解释了 github 是什么。

## 3 Git 的功能

因为 github 只是一个存放东西的空间，具体的存取功能都是 git 实现的，因此我们将重点放在 git 上。

### 3.1 创建项目

创建 git 项目相当于建立一个文件夹，这意味着以后该文件夹下的所有东西都会在本地和服务器两端进行同步。但是这并不是说我在本地随便创建一个新文件夹，就算成功创建一个 git 项目了。创建 git 项目通常有两种途径，且这两种途径都要借助于 github 网页：1，可以在 github 个人主页上新建一个 repository（网页中有专门执行该操作的按钮），这样就成功新建一个 git 项目了，用户可以在本地建立一个同名文件夹，并与 github 同步。2，可以在 github 上 fork 别人的项目（网页上有专门执行该操作的按钮），这样就可以将别人的项目克隆为自己的项目，用户可以在本地建立一个同名文件夹，并与 github 同步。

### 3.2 上传

要将本地的文件上传到 github，首先需要将需要用 add 命令添加待上传的文件（方便起见一般都用 `git add .`，意思是添加所有文件），接着用 `git commit` 命令确认修改，最后就可以通过 `git push [source] [branch]` 命令上传到 github 了。

### 3.3 下载

要将 github 上的文件下载到本地，有两种常用方法：1，`git pull [source] [branch]`，将 source 指定目录里 branch 指定分支下的所有文件下载到本地，并自动和本地当前分支进行合并。2，`git fetch [source] [branch]`，将 source 指定目录里 branch 指定分支下的所有文件下载到本次并自动添加到本地 source/branch 分支中。即该方法下载的文件不会自动与本地当前分支合并，需要之后手动合并。

## 4 Git 的使用心得

到目前为止，本文都在不厌其烦地强调 git 的上传下载功能。但如果仅此而已，git 就和普通网盘没多大区别了。而且，用网盘共享资源似乎并不会给多人协作编程带来多大的便利。因此，下面本文将介绍我们团队在使用 github 时印象最深的几点，供大家参考。

### 4.1 分支

git 可以为一个项目同时保存多个版本，每个版本就成为一个分支。这个特性很好地适应了多人同时开发同一个项目地情景。假设在某个时间点，每个人所看到的项目工程代码都是相

同的，接着每个人都在相同源代码的基础上各自开发自己负责的部分。这时可以通过 `git` 为每个人创建一个分支。假设存在分支 A 和分支 B，使用 `git checkout [branch]` 命令可以自由在 A 和 B 之间“移动”，并且对于同一个文件 C，在分支 A 看到的内容和在分支 B，看到的内容可能是不同的！这就保证了团队的每个人可以互不干扰，独立开发。

## 4.2 合并

一个项目发行时显然只能有一种版本，因此各个分支最终都避免不了合并的命运。如果要人工合并两个修改得面目全非的文件，绝对是吃力不讨好的工作，但 `git` 为我们提供了强大的合并功能！合并（merge）功能也是笔者认为的 `git` 最重要的功能！用户可以通过 `git merge [branch]` 命令来将 `branch` 分支和当前分支进行合并。对于相同的一个文件，如果当前分支该文件的内容与 `branch` 分支不同的话，`git` 会提示 `merge conflict`，并会在该文件中用“> > > > > >head”，“=====”，“< < < < < < <source”来清楚地标定出不同指出。因此我们只需人工处理这些有 `conflict` 的地方，并重新保存文件，就可以轻松完成合并了。