利用git上传文件至github是特别常用的，总结以下内容供参考使用。   
**第一步：**下载git工具，[这里是链接](https://git-scm.com/downloads)，选择适合自己的版本进行安装。

**第二步：**安装完成后，找到Git bash，双击打开。

**第三步：**输入自己的用户名和邮箱（为注册GITHUB账号时的用户名和邮箱）

$ git config --global user.name "coliyin@163.com"

$ git config --global user.email "coliyin@163.com"

* 1
* 2

**第四步：**设置SSH key

  众所周知ssh是加密传输。加密传输的算法有好多，git可使用rsa，rsa要解决的一个核心问题是，如何使用一对特定的数字，使其中一个数字可以用来加密，而另外一个数字可以用来解密。这两个数字就是你在使用git和github的时候所遇到的public key也就是公钥以及private key私钥。

  其中，公钥就是那个用来加密的数字，这也就是为什么你在本机生成了公钥之后，要上传到github的原因。从github发回来的，用那公钥加密过的数据，可以用你本地的私钥来还原。

  如果你的key丢失了，不管是公钥还是私钥，丢失一个都不能用了，解决方法也很简单，重新再生成一次，然后在github.com里再设置一次就行。

言归正传，我们首先检验本机是否生成密钥，执行命令：

$ cd ~/.ssh

$ ls

* 1
* 2

若结果如下，则说明密钥已存在。

如果没有密钥，则执行以下命令来生成密钥：

$ ssh-keygen -t rsa -C "coliyin@163.com"

* 1

生成过程中按3次回车键就好（默认路径，默认没有密码登录），生成成功后，去对应默认路径里用记事本打开id\_rsa.pub，得到ssh key公钥。

**第五步：**为github账号配置SSH key

  接下来，切换到个人github账号里，点击右上角用户头像下的小三角，找到setting，在右侧菜单栏中找到SSH and GPG keys，选择new SSH key，输入title，下面key的内容就是本机ssh key 公钥，直接将id\_rsa.pub中的内容粘贴过来就可以，然后点击下面的add SSH key即可完成。

**第六步：**上传本地文件

**1、创建一个本地文件**

**2、建立本地仓库**

（1）首先进入layout文件夹：

（2）执行指令进行初始化，会在原始文件夹中生成一个隐藏的文件夹.git：

$ git init

* 1

（3）执行指令将文件添加到本地仓库：

$ git add . //添加当前文件夹下的所有文件

$ git add \*\*.cpp //添加当前文件夹下的\*\*.cpp这个文件

* 1
* 2

（4）输入本次的提交说明，准备提交暂存区中的更改的已跟踪文件，单引号内为说明内容：

$ git commit -m "layout" //引号中的内容为对该文件的描述

* 1

**3、关联github仓库**

在github中新建一个repository，复制仓库地址：

执行命令：

//新建一个repository时会出现下面的代码，直接复制即可

$ git remote add origin https://github.com/CongliYin/CSS.git

* 1
* 2

**注意：**

如果出现错误：fatal: remote origin already exists，则执行以下语句：

$ git remote rm origin

* 1

再重新执行：

$ git remote add origin https://github.com/CongliYin/CSS.git

* 1

即可成功。

最后执行命令：

$ git push origin master

* 1

如果出现错误failed to push som refs to…….，则执行以下语句，先把远程服务器github上面的文件拉先来，再push 上去。：

$ git pull origin master

* 1

增加或删除文件之后需要的操作是

（3）执行指令将文件添加到本地仓库：

$ git add . //添加当前文件夹下的所有文件

$ git add \*\*.cpp //添加当前文件夹下的\*\*.cpp这个文件

（4）输入本次的提交说明，准备提交暂存区中的更改的已跟踪文件，单引号内为说明内容：

$ git commit -m "layout" //引号中的内容为对该文件的描述

最后执行命令：

$ git push origin master