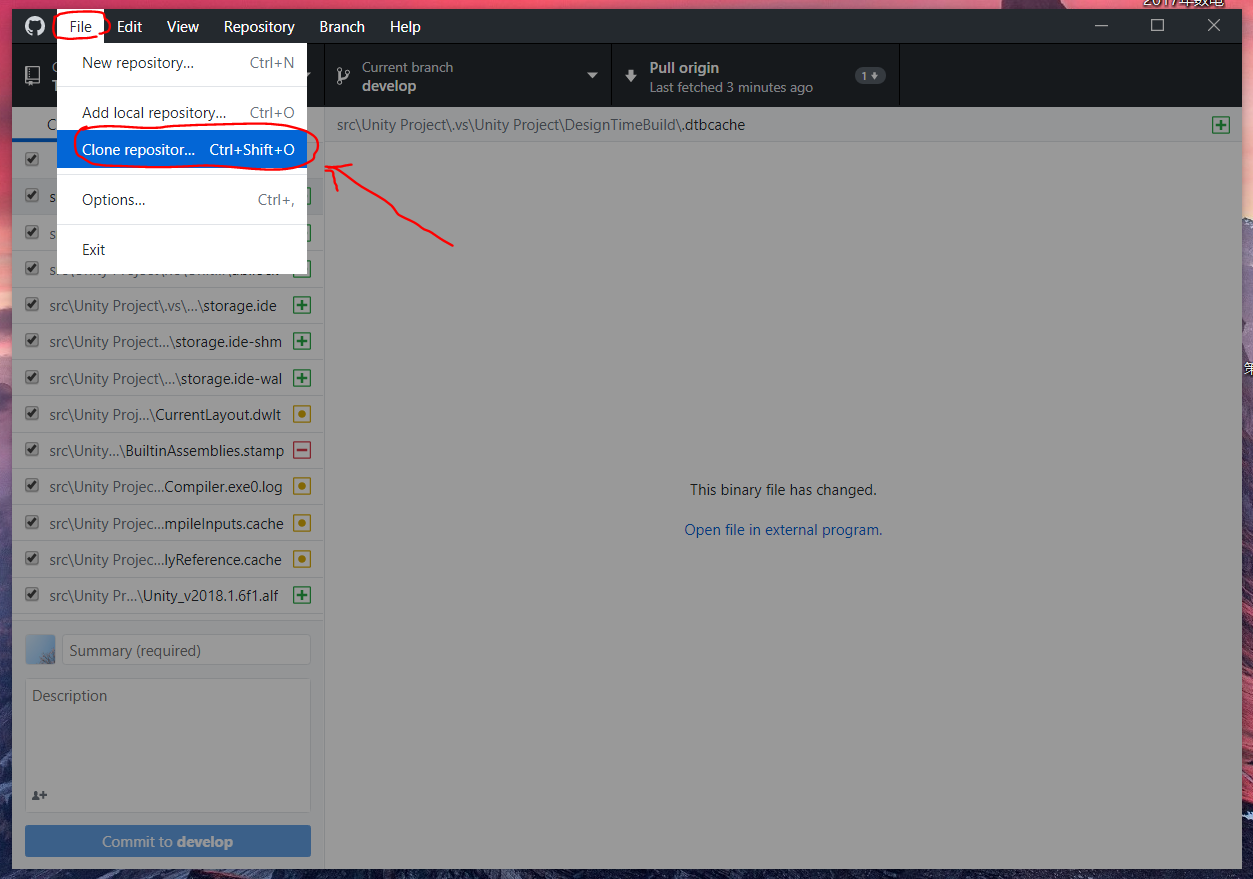
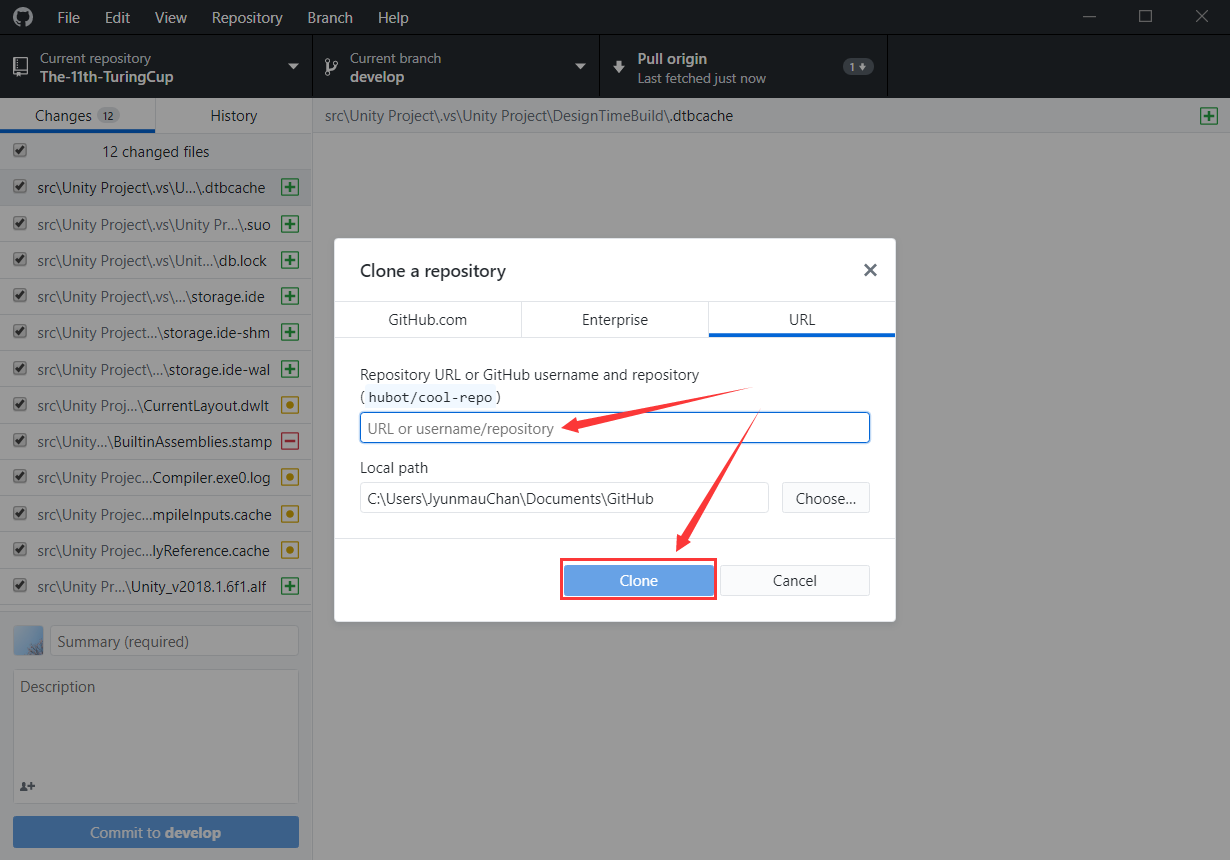
**项目部署步骤及说明**

1. 加入团队并申请权限
   1. 向项目组长申请加入团队权限（提供GitHub账号），本项目所有文档及代码均托管在GitHub上。
   2. 加入团队后项目组长给新组员分配权限，技术部、美工部成员可拥有读/写权限，其他参与者仅赋予读取权限。
2. clone并创建本地仓库
   1. 下载GitHub Desktop并安装（官网地址<https://desktop.github.com/>）
   2. 从远程仓库clone项目



* 1. 在地址栏中填写项目地址<https://github.com/TuringCup-11th/The-11th-TuringCup.git> ，将项目clone到本地，记住仓库路径

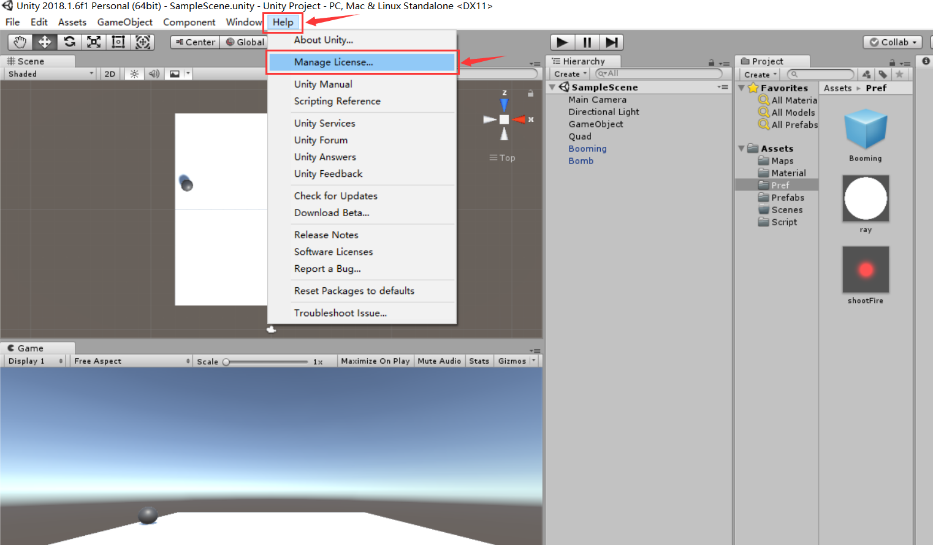


* 1. 如果你熟悉git的操作，也可以在命令行上完成，具体过程请另行百度
  2. 目前最新的项目在“Observe-Mode”分支上，下载主分支后自行同步

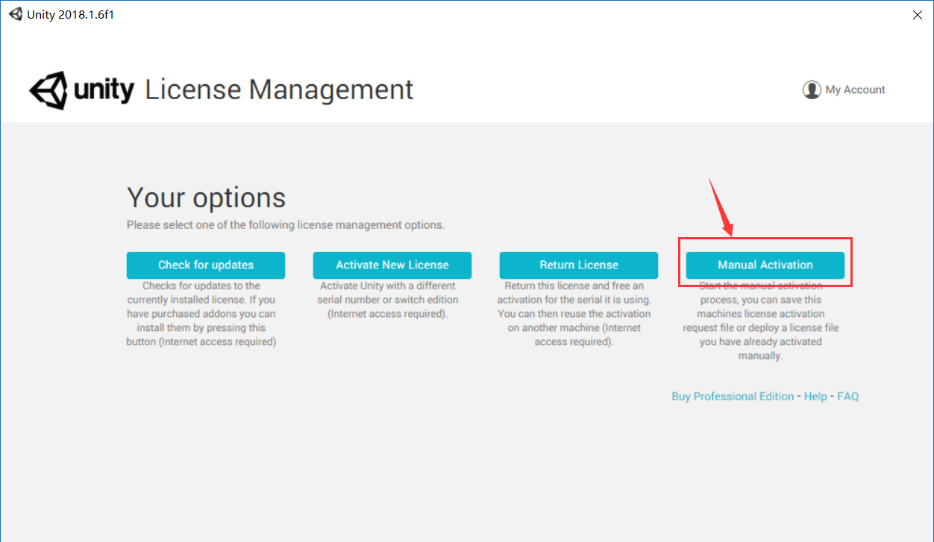
1. 用unity打开项目
   1. 注册账号，
   2. 从官网下载unity2018.1.6f1
   3. 在线模式下部署

打开uinity后登陆账号即可，登陆后操作与离线状态同，见下方e)

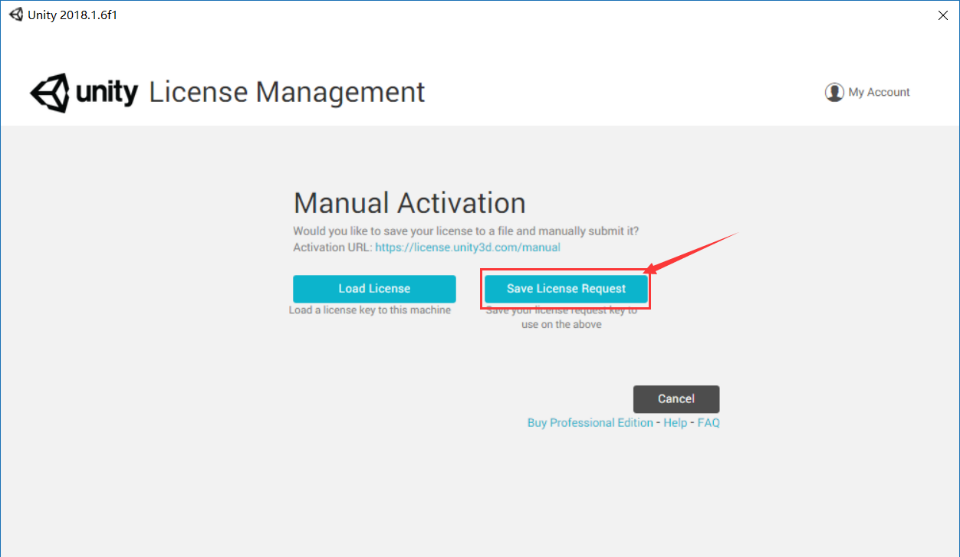
* 1. 离线模式下部署前的准备
     1. 在联网状态下，打开或新建一个项目，点击Help -> Manage License。



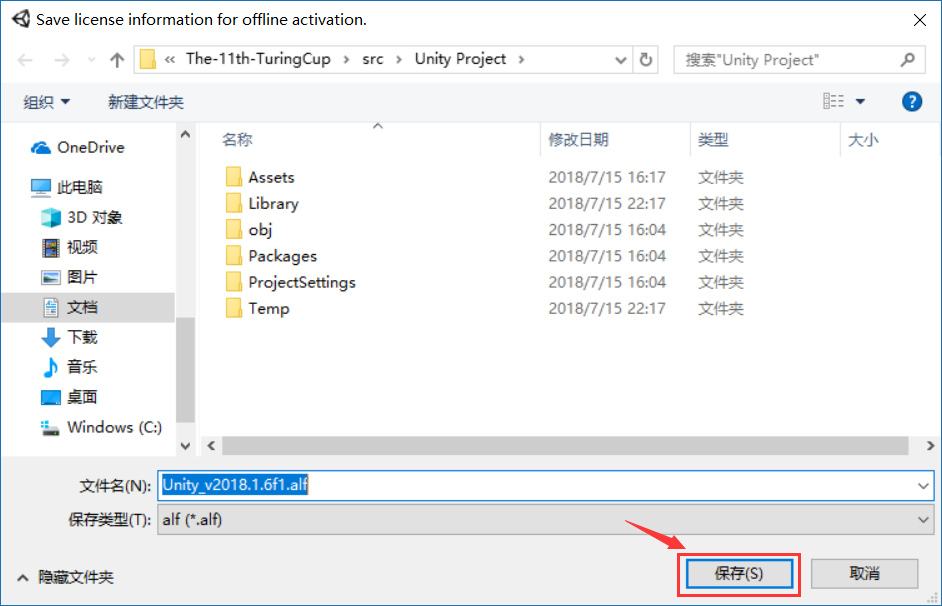
* + 1. 在弹出的窗口中，点击“Manual Activation”



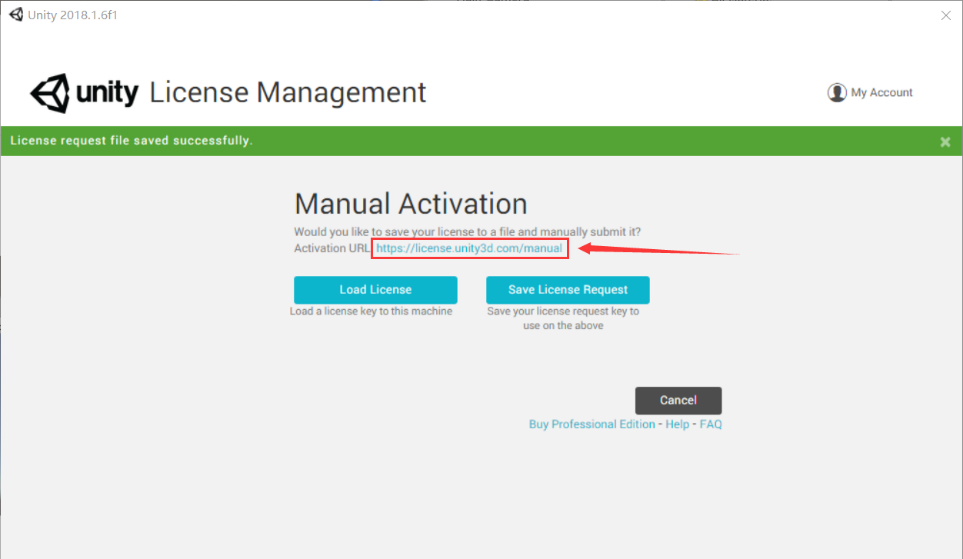
* + 1. 点击“Save License Request”



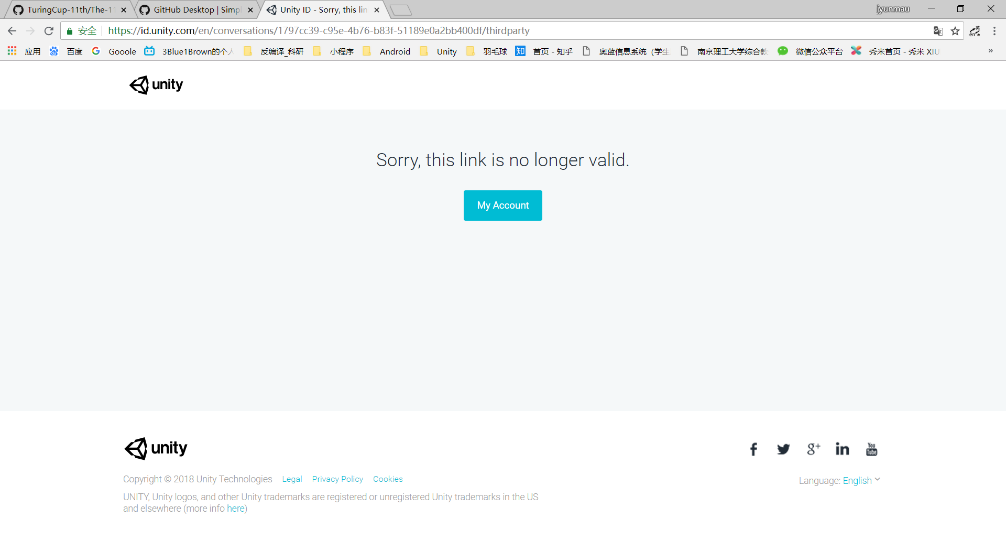
* + 1. 选择一个路径保存该 \*.alf 文件，并记住该路径

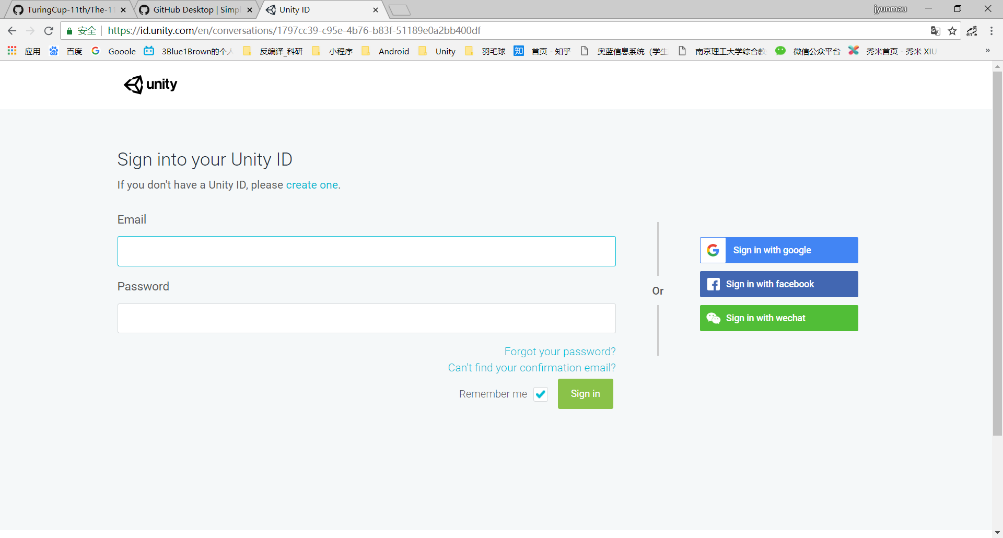


* + 1. 点开如图所示URL

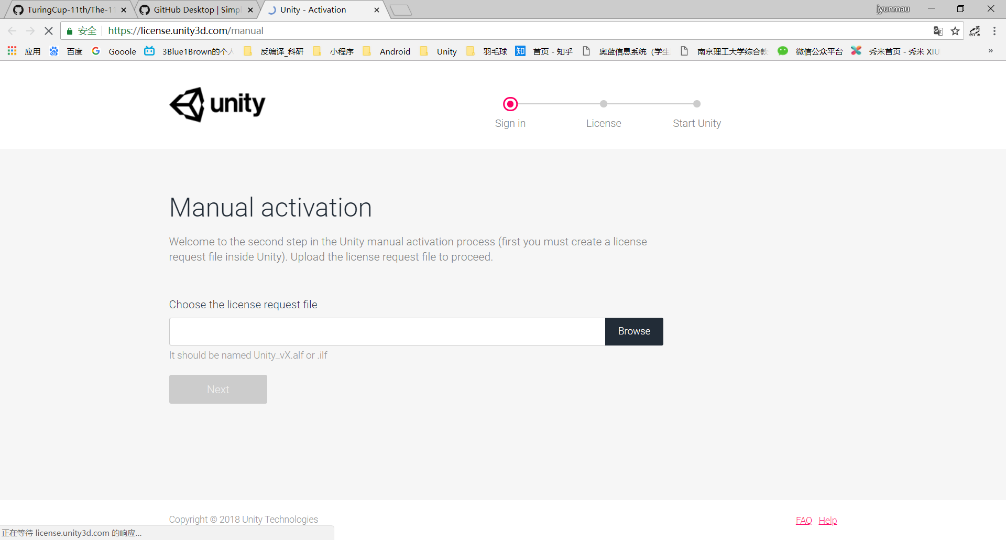


* + 1. 网页会自动跳转登陆，出现如下页面时，请耐心刷新

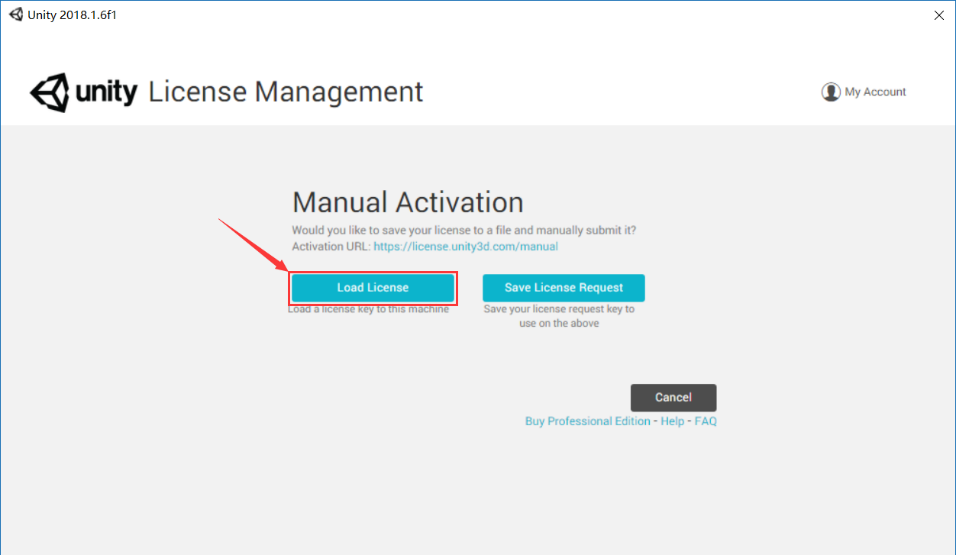




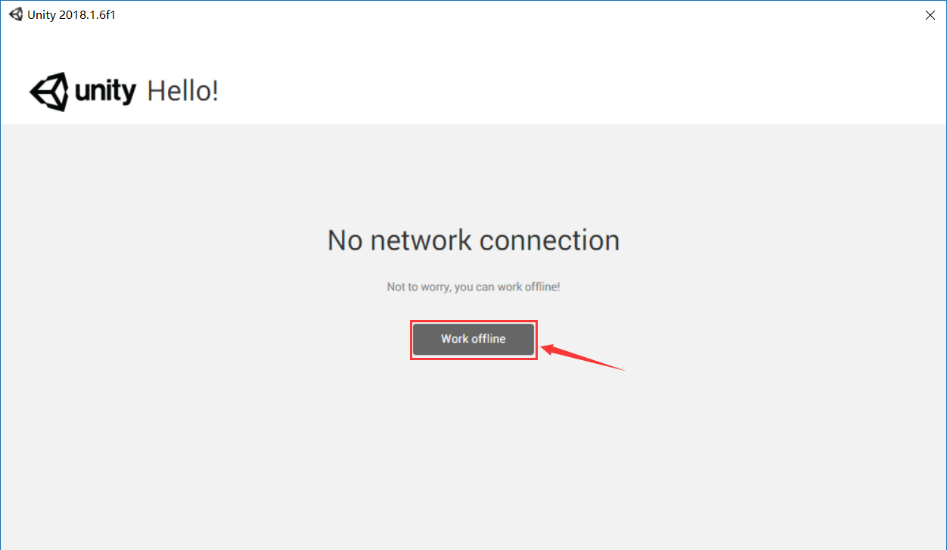
* + 1. 登陆成功后，在如图所示页面点击“Brow”，选择先前保存的 \*.alf 文件，确定后点击“Next”，等待 \*.ulf 文件下载完毕



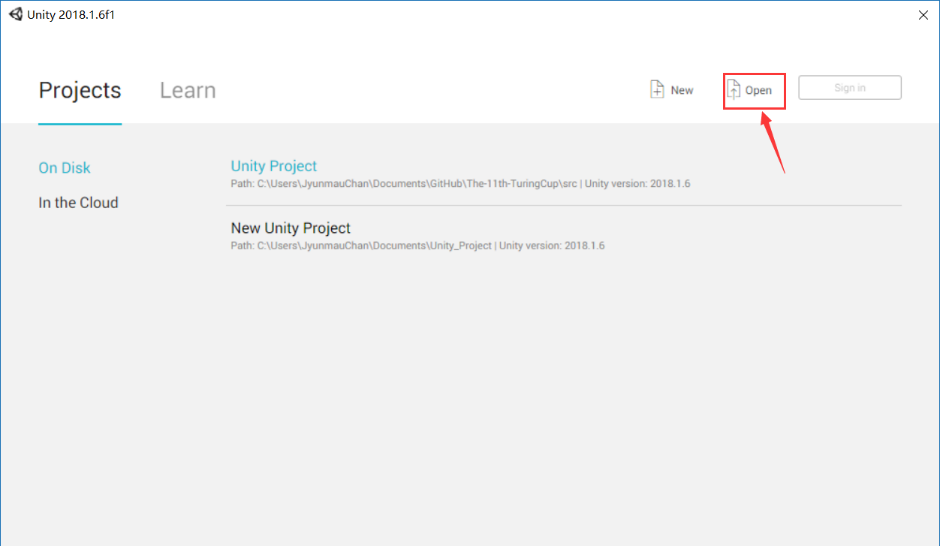
* + 1. 返回刚才的窗口，点击“Load License”，选择刚刚保存的 \*.ulf 文件，加载后即可完成离线使用的准备



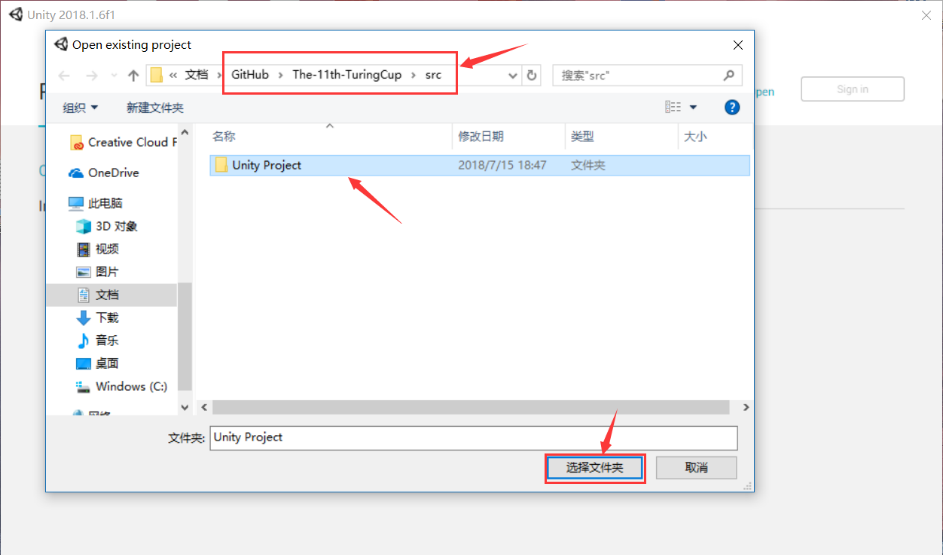
* 1. 离线模式下部署
     1. 打开unity后，出现如下图所示界面，点击“Work offline”。



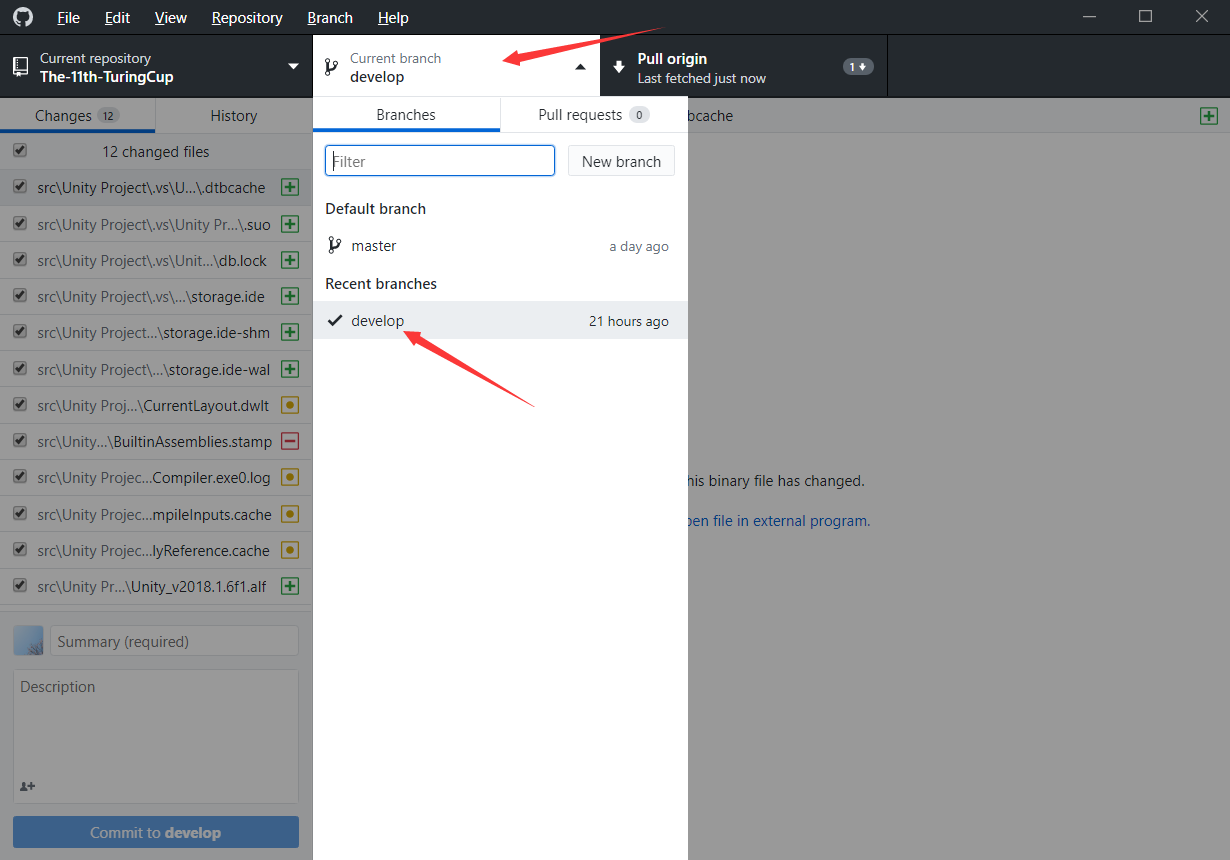
* + 1. 点击右上角的“Open”。



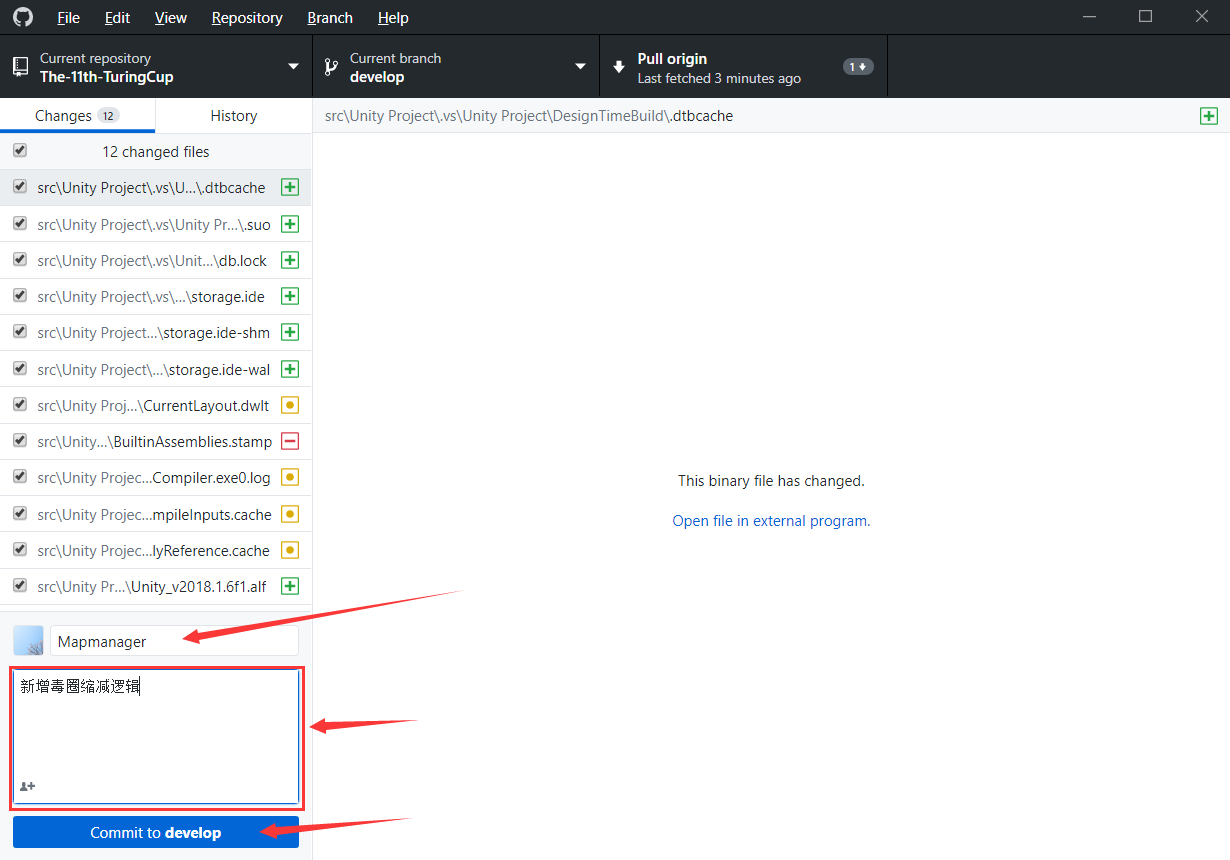
* + 1. 选择项目src文件夹下的Unity Project文件夹（注意unity打开的必须要是包含所有工程文件的文件夹），然后点击“选择文件夹”，即可打开项目。



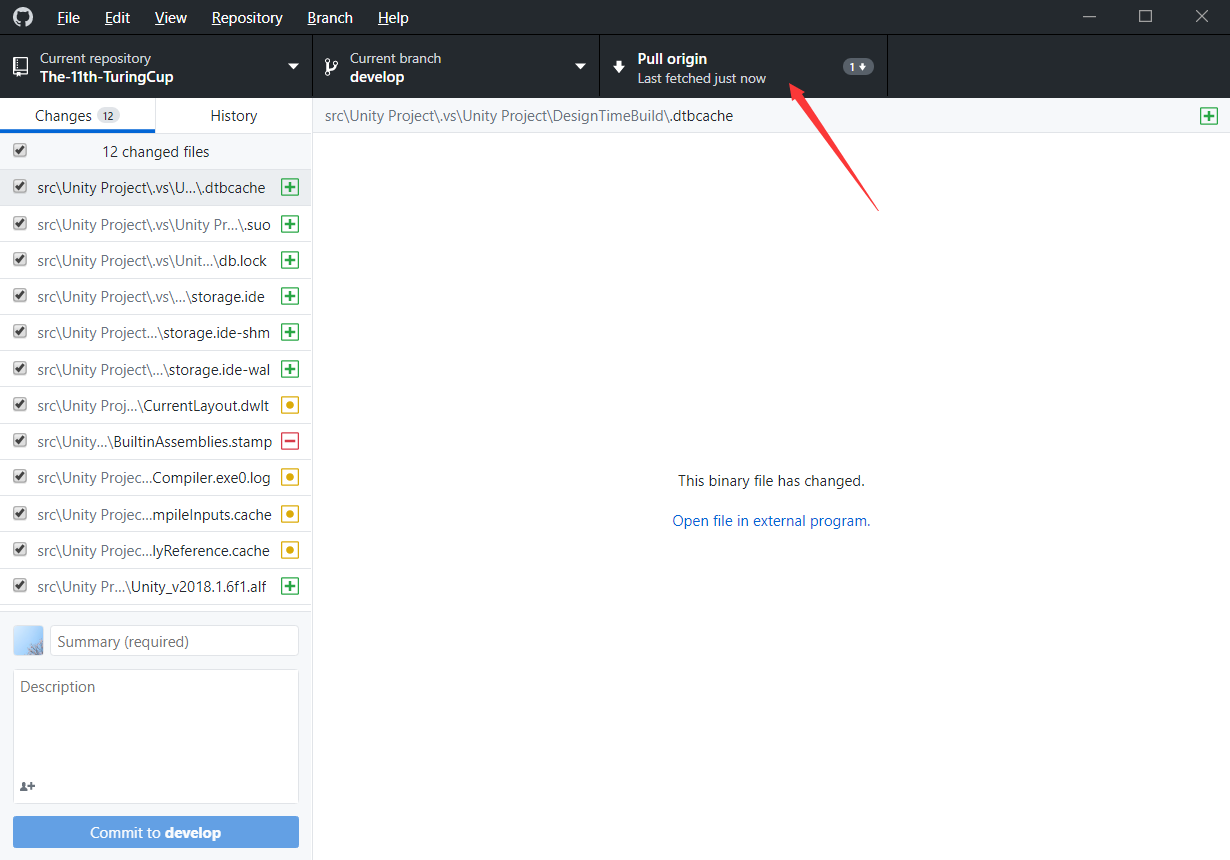
1. 项目修改及提交规范
   1. 每次修改前，先将GitHub上的项目同步到本地（master分支）
   2. 仔细阅读需求文档，如有疑问，向项目组长询问需求的具体要求，自行测试过所写模块没有bug后再进行提交
   3. 完成更改后打开GitHub Desktop，看到页面提示已经产生更改的文件，此时打开“Current branch”选项卡，选择“Obser-Mode”分支



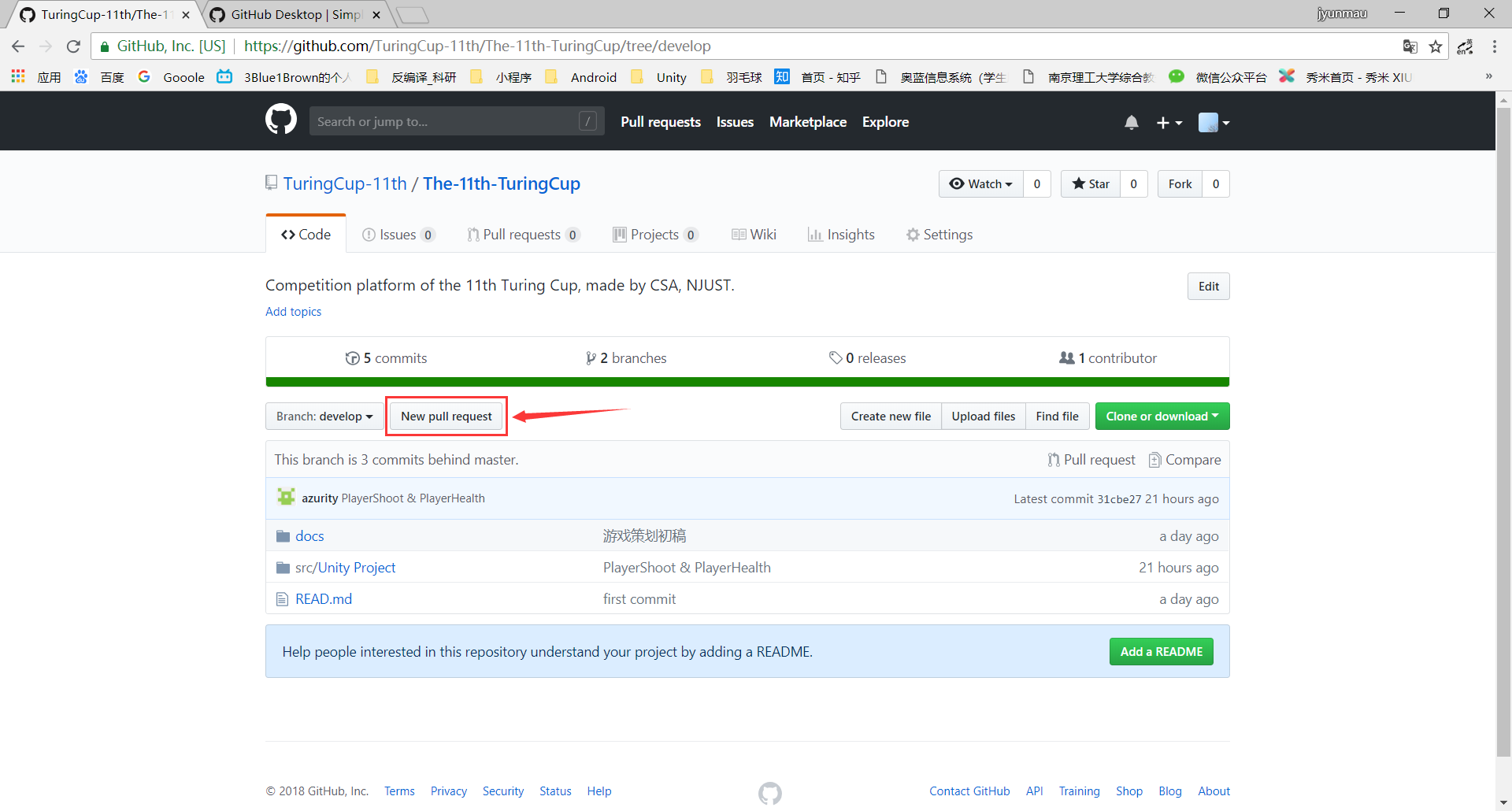
* 1. 在“Summary(required)”框内填写修改的脚本文件或特定需求（debug、测试等），在下方的“Description”框内填写对于本次修改的描述，之后点击“Commit to develop”。此时本地仓库已更新。



* 1. 点击“Pull origin”，即可将本地仓库内容更新到GitHub上。



* 1. 在网页上登陆github，找到本项目发起“Pull request“，填写清楚修改的事由



* 1. **（仅项目组长和负责人有权限）**检查Pull request，确认无误后合并到master分支
  2. 项目组长和负责人应当及时更新master分支，以防多人提交造成更改丢失。有多人同时更改项目时，每人更新一个独立分支，合并到develop，无误后再合并到master分支。