# 本地git托管到github发布

最近不是买了个NAS么，就开始琢磨上搭些个乱七八糟的服务，下面开始正题。

## 原环境

编写工具：typora  
静态网站生成器：mkdocs，选这个的原因是生成的速度快些，同时由于我的文档写在乱七八糟的位置，原来有用过WordPress、typecho、hexo、jekyll、mdbook、gitbook等等，基本上常用的我好像都用过，因为平台不一样所以导致格式也是乱七八糟的，只有这个能够让我修改最少，他兼容的markdown格式也比较多，对于我的静态资源渲染也比较好  
网站平台：GitHub Pages  
自动化：自己写的bat脚本：

git add .  
git commit -m "更新本地文档"  
git push  
mkdocs build   
cd site  
git add .  
git commit -m "update"  
git push

这个脚本看起来并不复杂，实际上就是把先把当前的已更新的文档进行保存，然后使用mkdocs编译为静态站点代码，然后将编译后的site目录推送到github，至于GitHub Pages如何搭，请自行百度

## 修改原因

现在有了NAS，我的博客原始文档又不想被别人拿过来复制就发到什么微信公众平台啥的，再加上我本地已经搭建了自己的git服务端，所以更换了下

## 现环境

编写工具：typora+obsidian+Writerside（我日常更换编辑器~）  
静态网站生成器：mkdocs，这个依旧没换  
网站平台：GitHub Pages  
自托管的git服务端：gitea，资源占用小，支持各种扩展，常用的东西基本上都支持  
自动化：请看下文

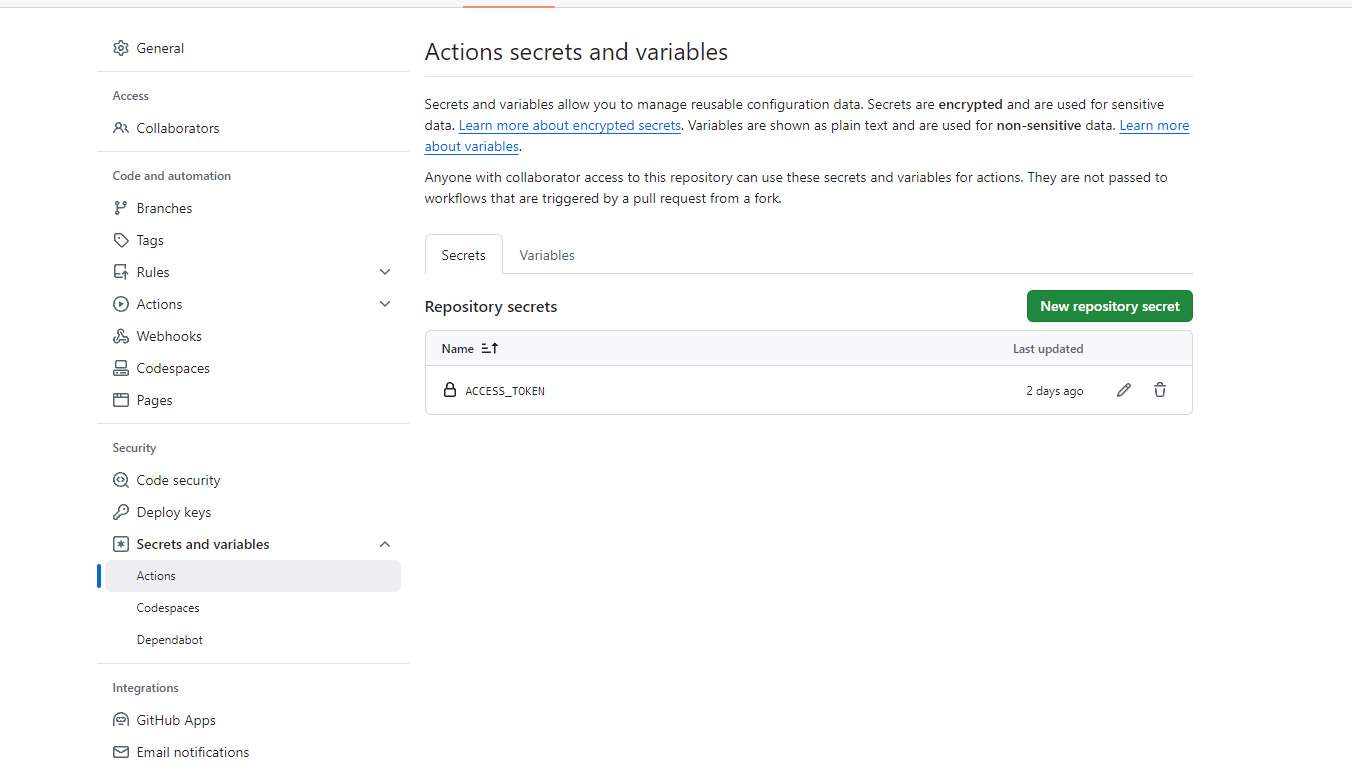
## 具体操作

先在自己自托管的git服务端搭建个仓库，然后在仓库上设置镜像仓库：  


这里要注意下，现在github不支持使用用户名密码来进行验证，可以申请个token，路径在：https://github.com/settings/tokens，repo的所有权限都要给到

然后本地正常的与自托管的git服务端关联

在github上新建一个私有仓库，我的叫blog，这个仓库用来存放你的文档源文件，这个仓库不一定要私有的哈，也可以公共的，看你自己，然后在仓库的设置中，新建一个Secrets and variables名字叫ACCESS\_TOKEN（如果修改请同步修改后面的yml文件内容）:



在文档的根目录下新建文件夹：.github\workflows，然后在workflows目录下新建一个build-and-deploy.yml，具体内容：

name: Build and Deploy  
  
on:  
 push:  
 branches:  
 - master # 或者你想要触发工作流的分支  
  
jobs:  
 build:  
 runs-on: ubuntu-latest  
  
 steps:  
 - name: Checkout Blog Repository  
 uses: actions/checkout@v3  
 with:  
 repository: Rabb1tQ/blog # 私有仓库  
 token: ${{ secrets.ACCESS\_TOKEN }} # 你的访问令牌  
 fetch-depth: 0  
  
 - name: Set up Python  
 uses: actions/setup-python@v4  
 with:  
 python-version: '3.8'  
  
 - name: Install MkDocs and dependencies  
 run: |  
 pip install mkdocs  
 pip install mkdocs-material  
  
 - name: Build with MkDocs  
 run: mkdocs build  
  
 - name: Deploy to GitHub Pages  
 env:  
 ACCESS\_TOKEN: ${{ secrets.ACCESS\_TOKEN }} # 使用环境变量传递访问令牌  
 run: |  
 # 克隆目标仓库  
 git clone https://x-access-token:${ACCESS\_TOKEN}@github.com/Rabb1tQ/Rabb1tQ.github.io.git temp-gh-pages  
 cd temp-gh-pages  
   
 # 清空目标文件夹  
 rm -rf \*  
   
 # 复制 MkDocs 生成的文件  
 cp -R ../site/\* .   
  
 # 提交并推送更改  
 git config --global user.email "Rabb1tQ"  
 git config --global user.name "Rabb1tQ@users.noreply.github.com"   
 git add .  
 git commit -m "Deploy to GitHub Pages" || echo "No changes to commit"  
 git push

需要更改的点：你的GitHub Pages地址、用户名、邮箱、仓库地址、分支等，这个yml并不复杂，自己看看剩下还应该修改哪些地方

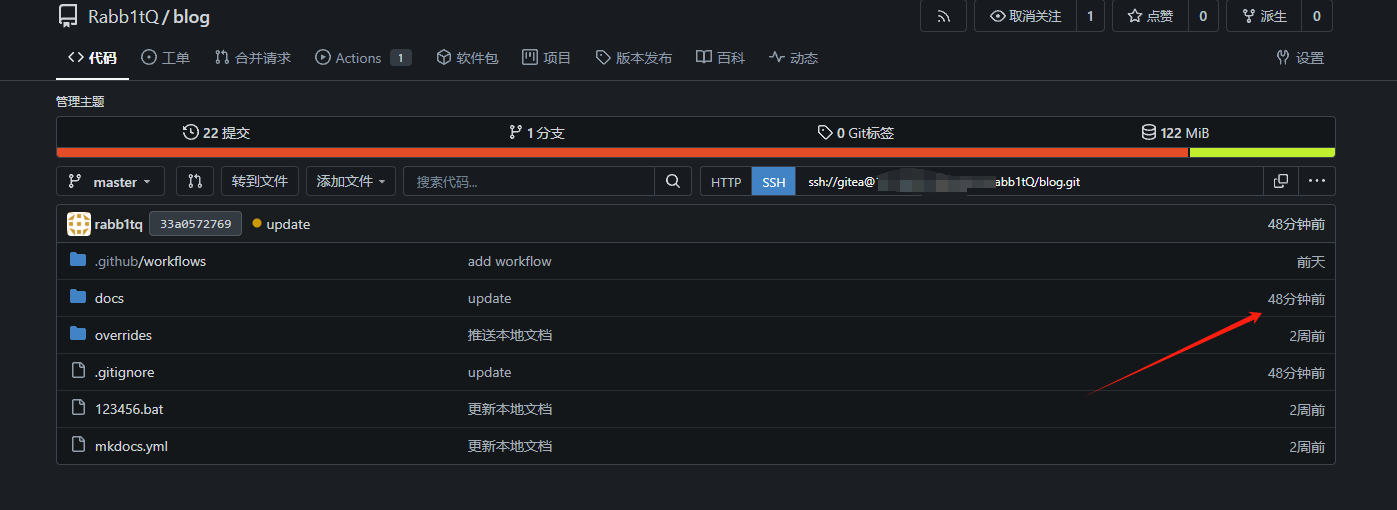
.gitignore文件中记得添加上

/site/

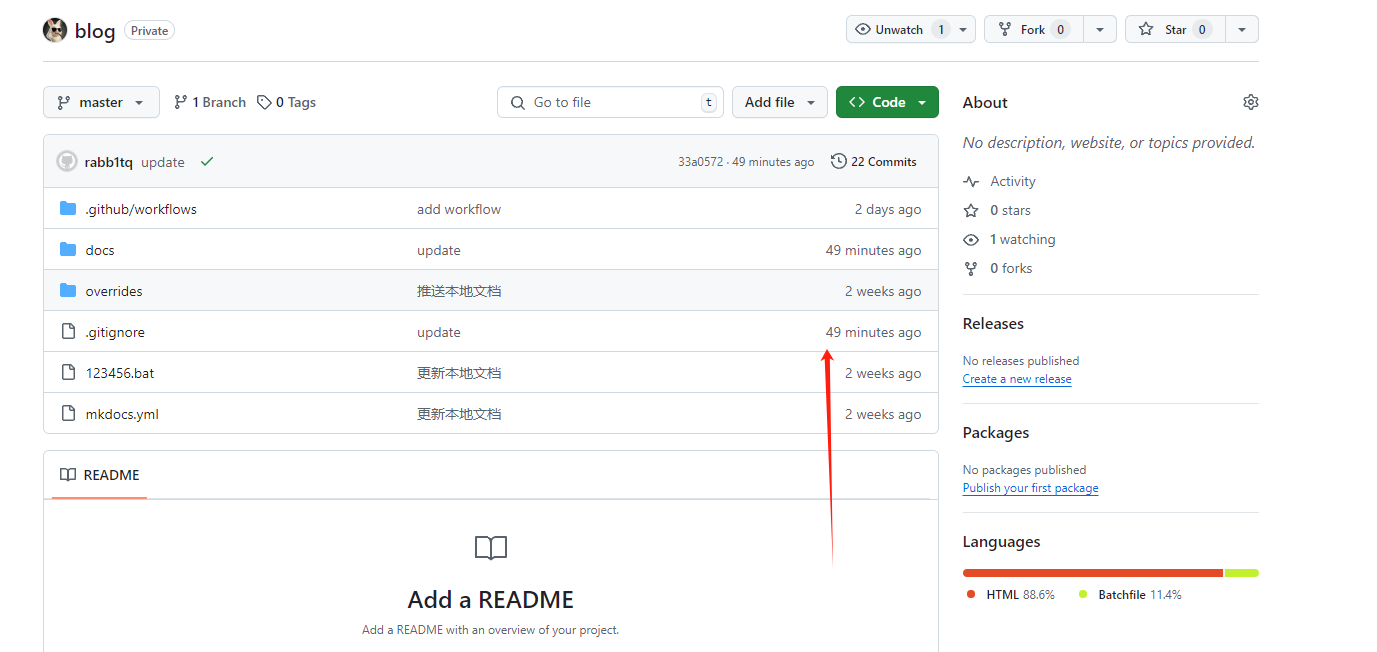
防止本地编译测试后推送给github，这玩意没啥用，不过不加其实也可以

## 最终效果：

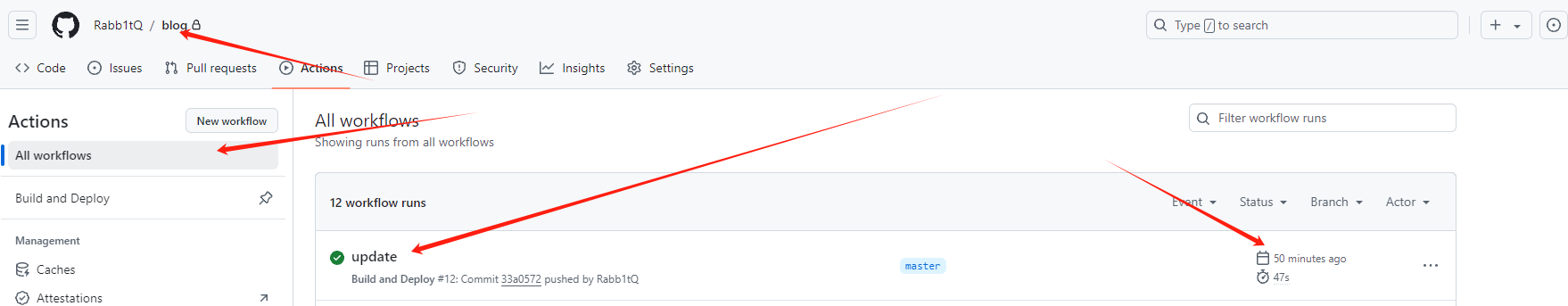
推送到本地git服务端：

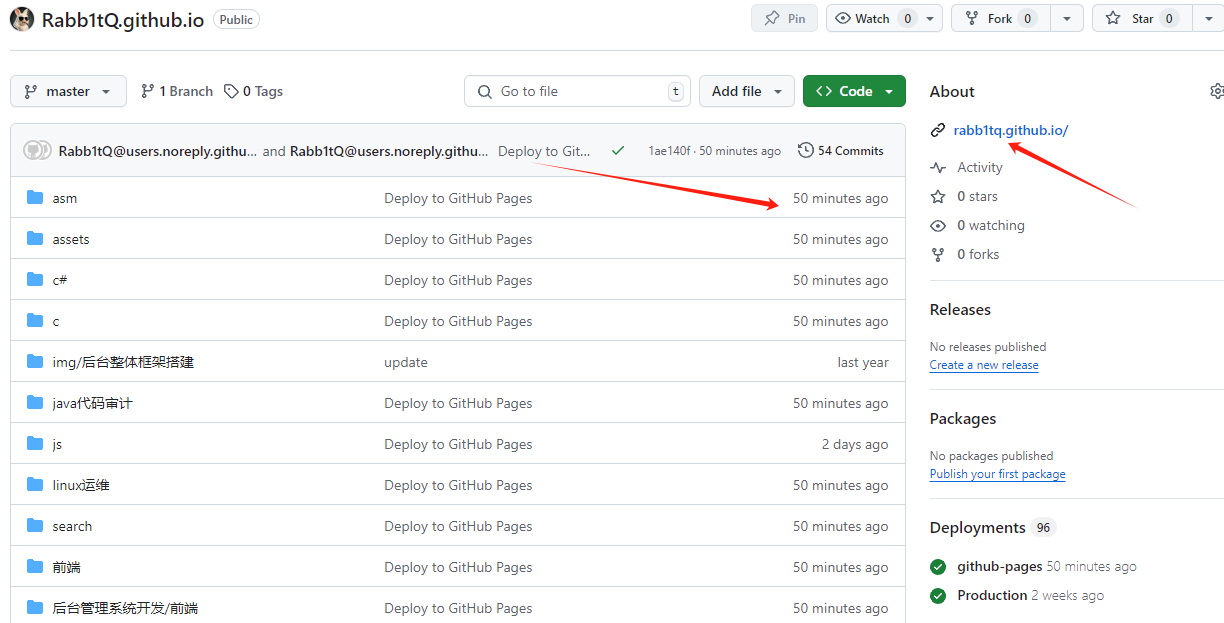


github仓库自动同步：

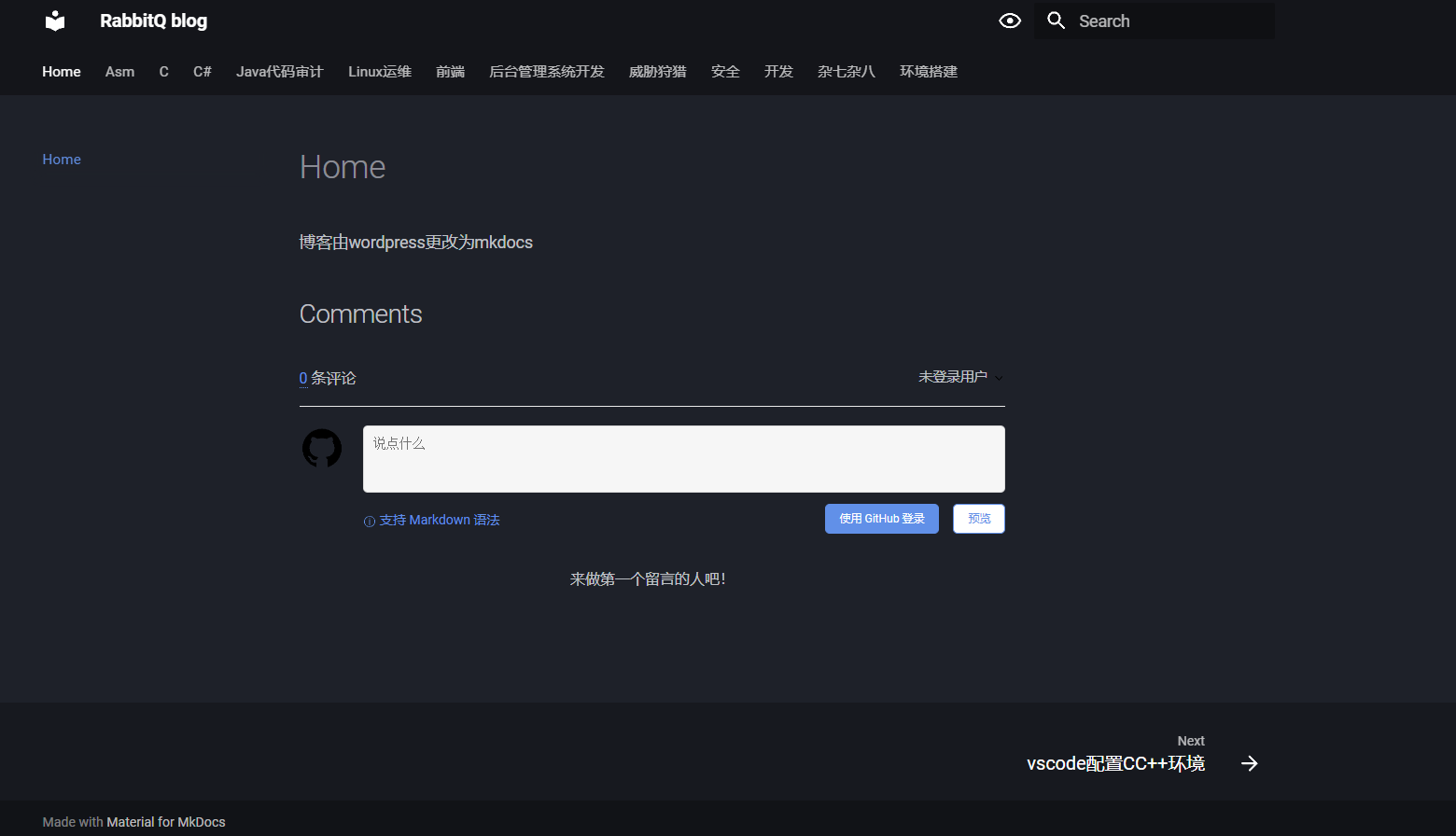


工作流执行：



Github Pages所在仓库更新：  


Blog效果：



## 写在最后

我的系统还在写~现在已经定了前后端框架了，但是写上才发现，有些表结构设计的不太合理，可能还需要重构一下