

目录

- [一、md文件上传到GitHub之后，图片不显示了？](#)
 - [1.1 原因](#)
 - [1.2 技术介绍](#)
- [二、解决方案](#)
 - [2.1 方案一：存储到本地（不适合小编写文章）](#)
 - [2.2 方案二：Tyora PicGo GitHub（实现免费图片服务器）](#)
- [三、常遇到的问题](#)
 - [3.1 如果上传不成功？](#)
 - [3.2 上传成功了，也显示了，但重新打开后，又不显示了？](#)
 - [3.3 如何想要转移图集？重新转移动新的GitHub仓库？](#)

Created by [gh-md-toc](#)

一、md文件上传到GitHub之后，图片不显示了？

1.1 原因

1. Markdown文件上传到Github，或者把含有图片的md文件传给其他人。
2. Typora写的md文件中的图片都是另外存储的，不像word软件是存放到word文件里的。

比如：我在文件里面放了下面一张图片（雷神），但把md文件发给朋友的时候，图片就不能显示了。。。。。



解决办法：

- 方法1：转成Word格式或者PDF格式之类的
- 方法2：把图片和md文件一起打包，再进行分享
- 方法3：全部图片上传到Github服务器上，直接发md文件，但朋友需要联网打开，才能看到图片。

说明：Github服务器，是免费的，但对于新手来说，刚开始使用会觉得很痛苦！

1.2 技术介绍

- Tyora：
一个所见即所得的Markdown格式文本编辑器，支持Windows、macOS和GNU/Linux操作系统，包括对GitHub Flavored Markdown扩展格式的支持、拼写检查、自定义CSS样式、数学公式渲染（通过MathJax）等特性。
- PicGo（图床工具的其中之一）：
自动把本地图片转换成链接的一款工具。支持微博，七牛云，腾讯云COS，又拍云，GitHub，阿里云OSS，SM.MS，imgur 等8种常用图床，功能强大，简单易用
- GitHub：
简而言之，它就是存放代码、图片等资源的平台。
【官话：GitHub是一个面向[开源](#)及私有[软件](#)项目的托管平台，因为只支持git 作为唯一的版本库格式进行托管，故名GitHub。】

二、解决方案

2.1 方案一：存储到本地（不适合小编写文章）

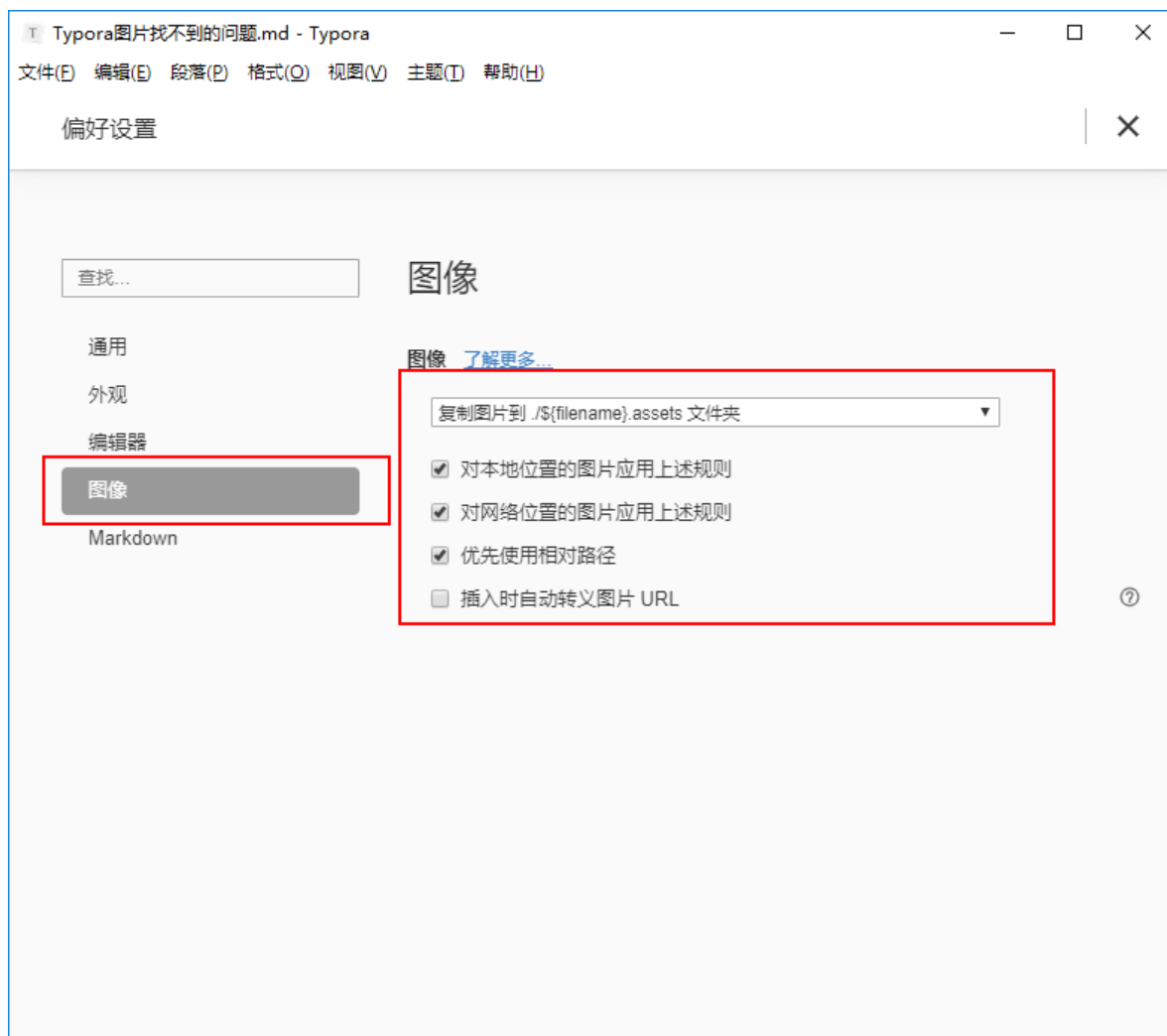
优点：

- 配置简单，免费

缺点：

- 文件分享麻烦，上传到博客导致图片找不到等问题。

设置如下，每次传md文件给其他人的时候，需要把图片文件夹也一并压缩转发才行，记住勾选“优先使用相对路径”



2.2 方案二：Tyora+PicGo+GitHub（实现免费图片服务器）

优点：

- 免费
- 搭建图床并自动上传（适合小编写文章）

缺点：

- 配置复杂
- 查看文件需要联网（图片全都是放到GitHub上的）
- 文件中的图片加载速度，取决于本机的网速🐢
- Typora和PicGo这项功能是2020年最近才有的，还存在不少的问题。

所谓图床工具，就是自动把本地图片转换成链接的一款工具，网络上有很多图床工具，就目前使用种类而言，PicGo 算得上一款比较优秀的图床工具。它是一款用 `Electron-vue` 开发的软件，可以支持微博，七牛云，腾讯云COS，又拍云，GitHub，阿里云OSS，SM.MS，imgur 等8种常用图床，功能强大，简单易用。

Typora要更新到最新版本，PicGo也要最新版【只有最近几个版本，才支持自动上传】

步骤一：创建GitHub账号和仓库

创建账号：https://blog.csdn.net/qq_43571415/article/details/100050781

创建仓库（很详细，记得另外保存好token值）：

<https://blog.csdn.net/acmman/article/details/77621253>

说明：

1. 创建的仓库可以是自己写的Github文章的仓库
2. 第一次创建仓库步骤很繁琐，以后就不会了。也劝大家不要自己去找教程，这类的资料很杂，很多都不全，这样对于小白更加痛苦！

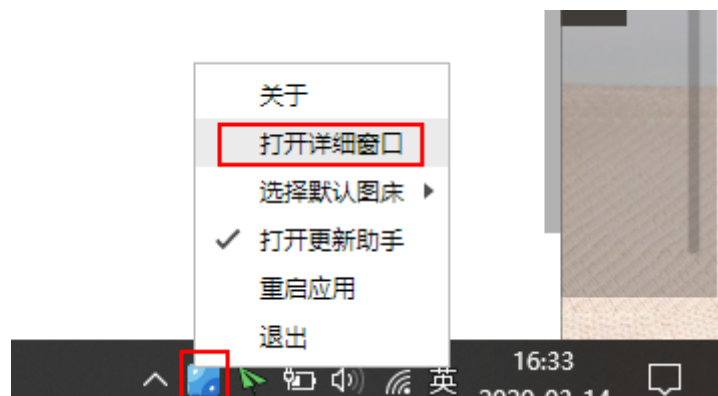
步骤二：下载安装Picgo（2.2.0以上的版本），Typora最新版

Picgo 2.2.2版本，下载链接：<https://www.lanzous.com/ia49ojg>

Typora 0.9.86 beta下载链接：<https://www.lanzous.com/ia49pgj>

Picgo设置步骤（很坑，所以步骤我自己写吧）：

①右击，打开详细窗口



②设置图床

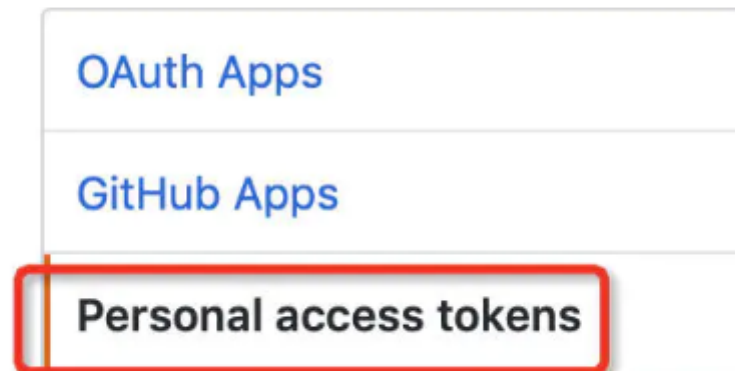


配置以下内容：

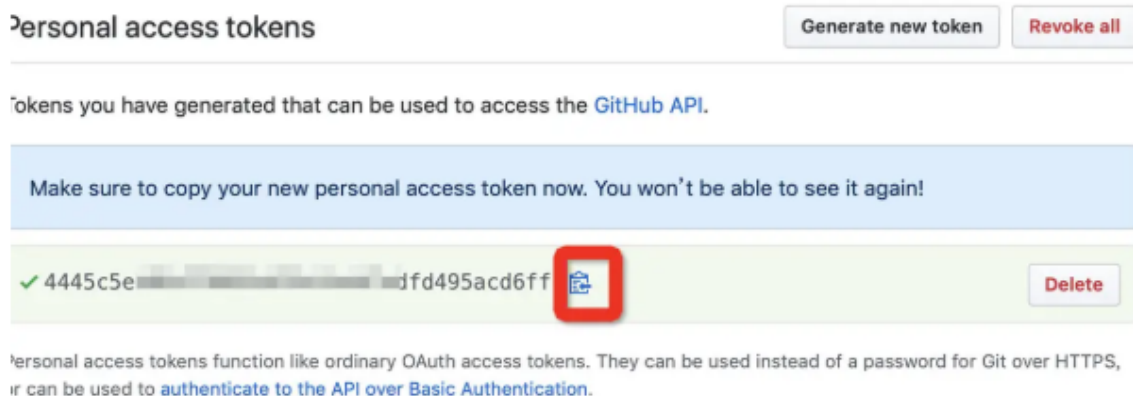
- 设定仓库名：用户名/仓库名
- 设定分支名：这里写入分支名称，一般直接用 `master` 即可。

- 设定Token（任意一个令牌都可以）：在 GitHub 上生成一个 *token* 以便 PicGo 来操作我们的仓库，来到个人中心，选择 *Developer settings* 就能看到 *Personal access tokens*，我们在这里创建需要的 *token*

Settings / Developer settings



点击 Generate new token 创建一个新 token，选择 repo，同时它会包含其中的都会勾选上，我们勾选这些就可以了。然后拉到最下方点击绿色按钮，Generate token 即可。之后就会生成一个 token，记得复制保存到其他地方，这个 token 只显示一次！！



- 指定存储路径：一般写 `/img`，会在设定的仓库中创建img文件夹，也可以取另外的名字。
- 设定自定义域名（很坑）：格式（<https://raw.githubusercontent.com/Github用户名/仓库名/分支名>）
raw.githubusercontent.com：固定的，不要修改
Github用户名：自己创建的Github登录用户名，不是昵称哦！
仓库名：上面创建的仓库名（尽量不要有中文符，或者特殊符号）
分支名：如果前面设定的分支名是master，这里也写master。

③手动上传测试（先确保能上传，再配置Typora）

有时候可能会上传不上，重启PicGo就可以了



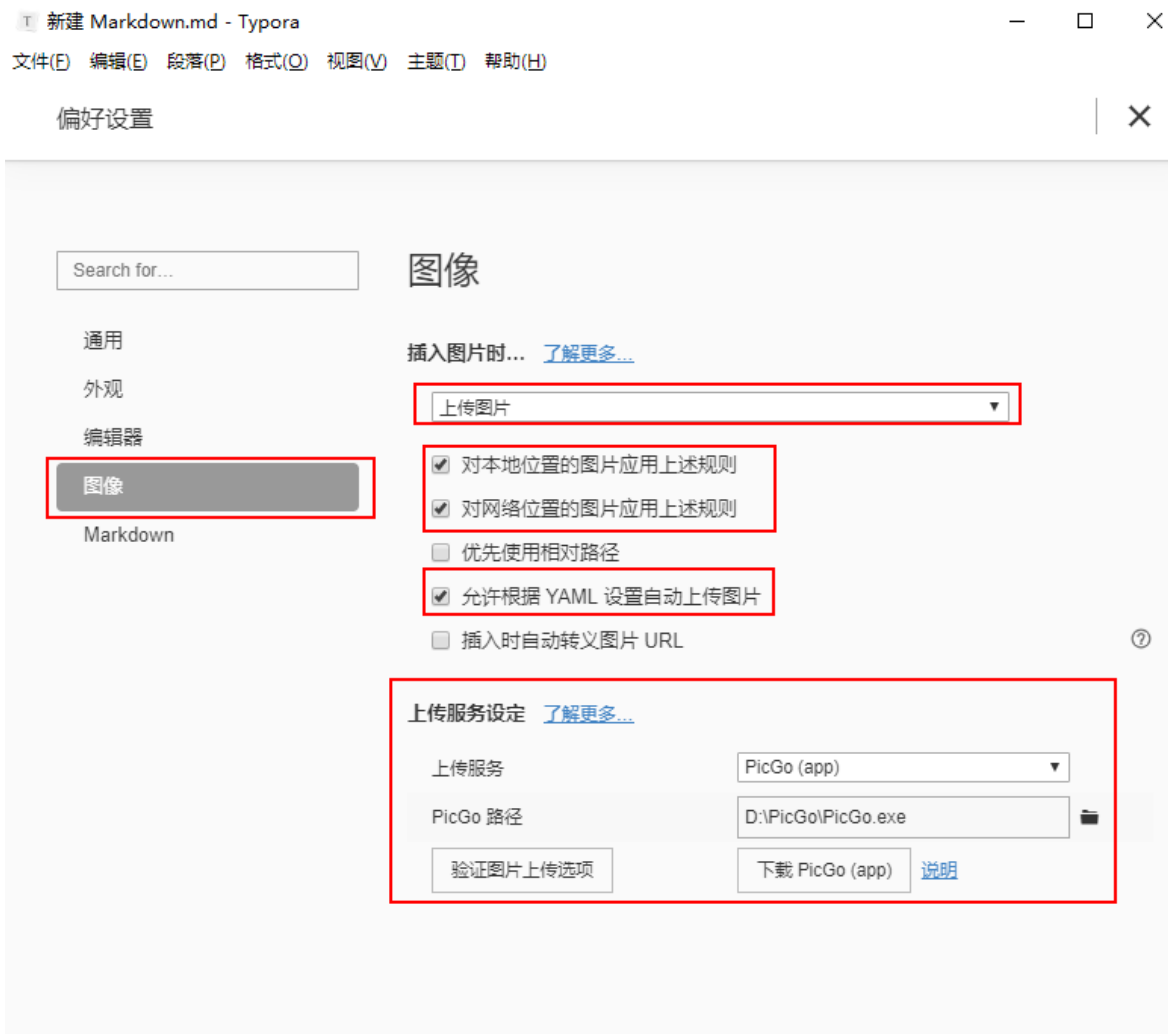
上传成功!



如果上传不成功?

- 仓库名有特殊符号? 上传文件有特殊符号? 或者有空格之类的?
建议全部改成英文。【GitHub中空格是用下划线来表示, _】
- token值不对?
解决: 重新复制粘贴一下, 或者重新创一个新的token。
- 自定义域名是否正确?
解决: 按照上面的步骤, 仔细观察书写格式。

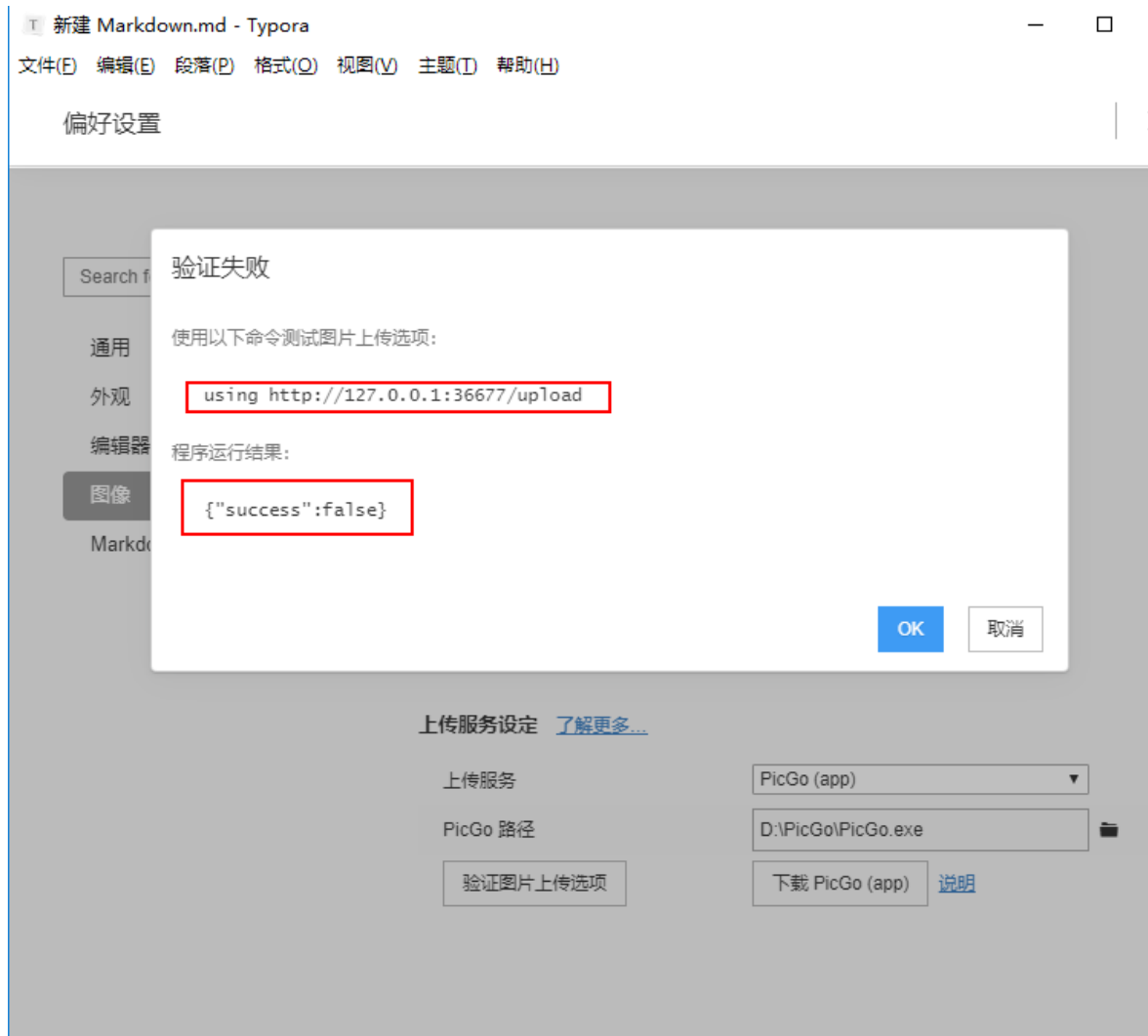
④Typora自动上传设置:



单击“验证图片上传选项”

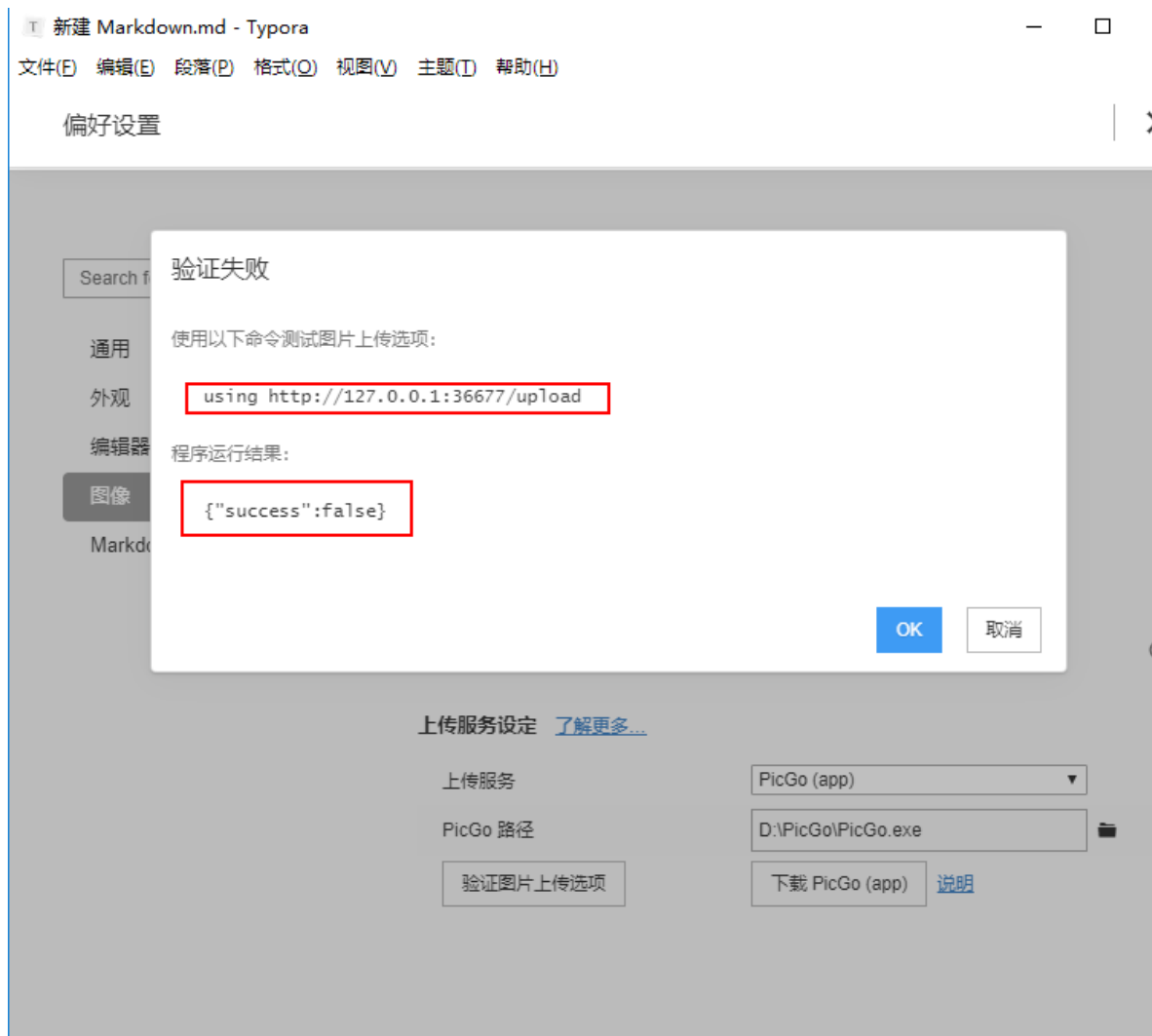
可以由下图获得PicGo需要设置的信息（记住，待会要用）：

- 需要监听的地址：127.0.0.1
- 需要监听的端口：36677



⑤PicGo设置Server





⑥**上传所有本地文件（单文件）**：它会把文件中用到的图片重新上传一次，路径也改变一次。



三、常遇到的问题

3.1 如果上传不成功？

- PicGo可能自己改了Server端口号？
解决：把它改过来，这个可能是个bug（毕竟是最新版，问题肯定会有）
- PicGo报错了？
解决：关掉PicGo，等一会，再启动。再等一会，最后上传图片。



3.2 上传成功了，也显示了，但重新打开后，又不显示了？

解决：应该是网速的原因，稍微等一下，或者做一下别的事情，过一会它就显示了。

3.3 如何想要转移图集？重新转移动新的GitHub仓库？

解决：

1. 把图片搜集到本地，再一个一个重新上传到新仓库（最老土的方法）
2. 下载GitHub之前仓库的图集，复制到新的仓库中。打开md文件，按ctrl+F，替换全部路径的仓库名。