Echo 是如何支持 MarkDown 的

原创 飞天小牛肉 [飞天小牛肉](javascript:void(0);)

**飞天小牛肉**

微信号 CS-Wiki

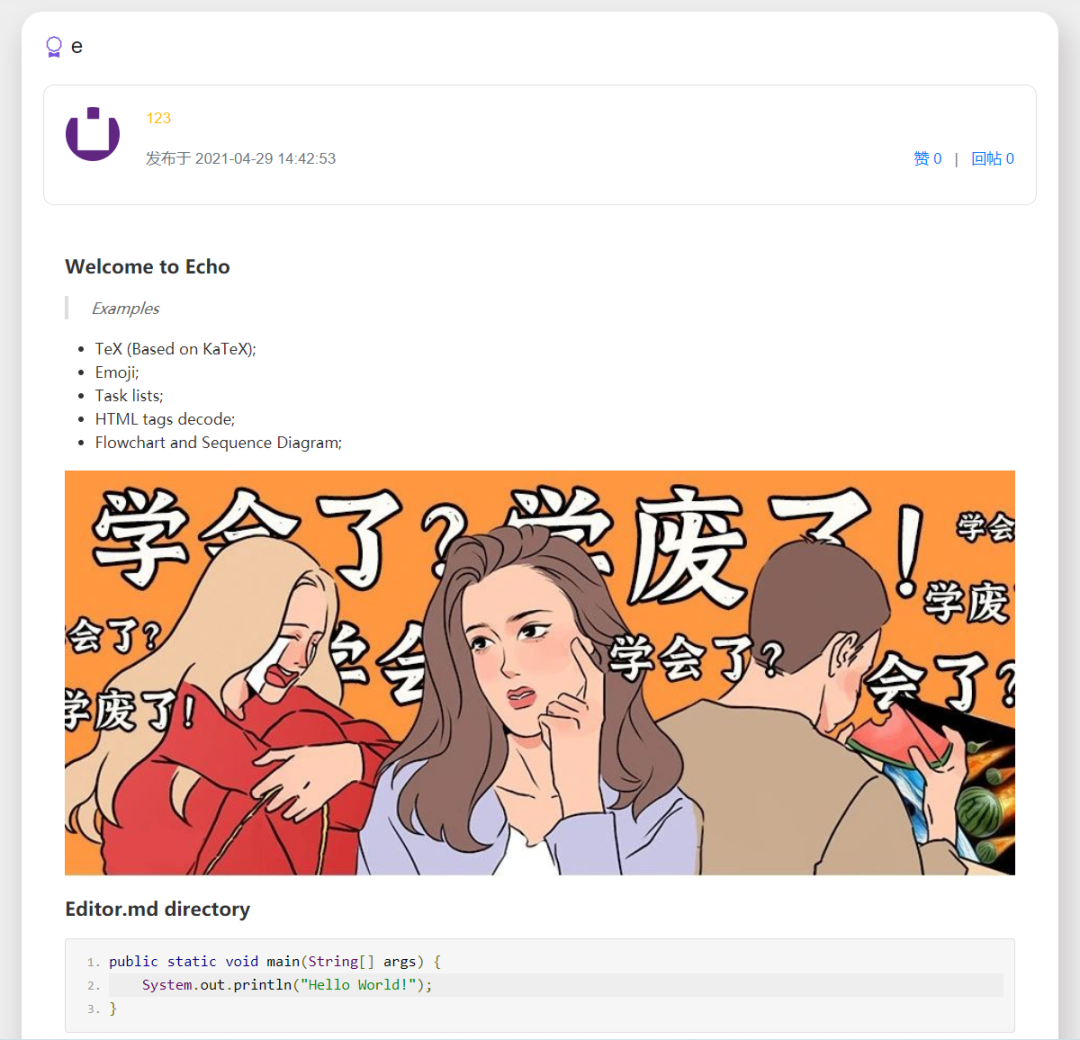
功能介绍 InfoQ & 阿里云签约作者，分享原创技术干货和成长经验

2021-08-03[原文](https://mp.weixin.qq.com/s?__biz=MzI0NDc3ODE5OQ==&mid=2247487734&idx=2&sn=f960d22fbb41ff615144f261df8b56ee&chksm=e959c54ede2e4c5899b4b2ce83bb47d3a20fc0ba1394fcffbd8a8a9217e123d6abc11d85c60e&scene=178&cur_album_id=1744497649518493697#rd) 发表于

收录于合集 #Echo 学习教程 18个

这个点其实没啥技术含量，就是集成一下开源的 MarkDown 编辑器，按照说明按部就班地走就行了。

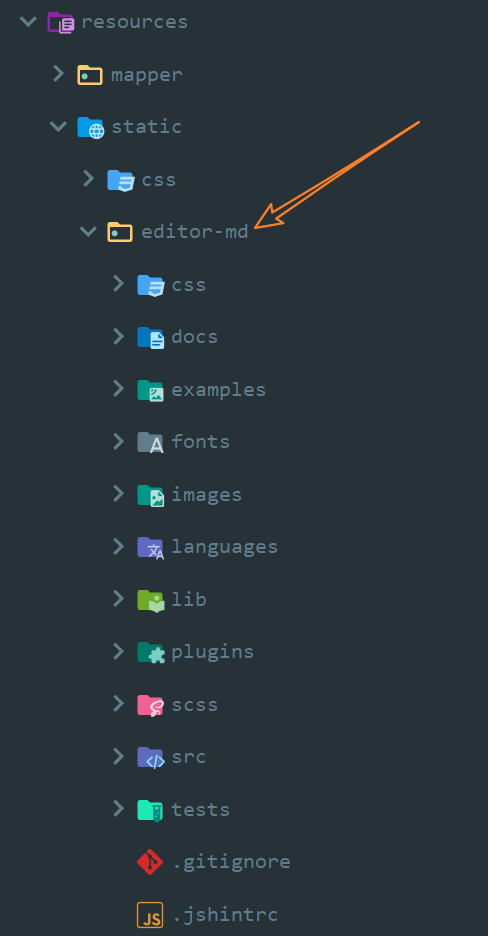
这里我选的 MarkDown 编辑器是 Editor.md。大家可能会碰壁的地方应该就是图片上传的部分。



**第一步：下载 Editor.md 源码**

Editor.md 源码下载地址：https://pandao.github.io/editor.md/index.html

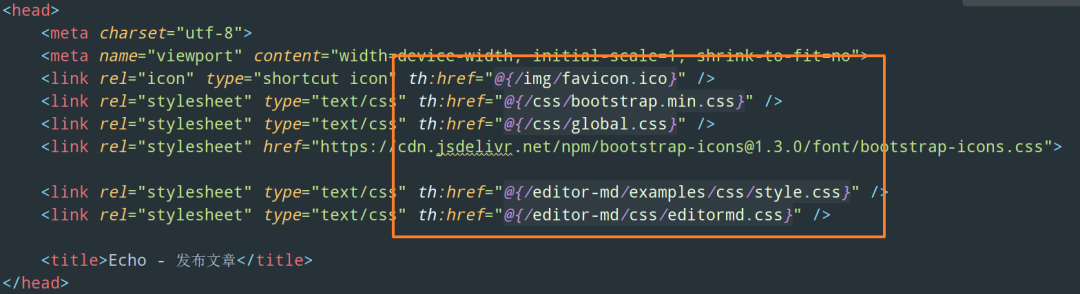
下载解压之后将文件夹整个放入 static 目录下：



做完这一步各位的第一想法应该就是要去配置下我们的拦截器（对除静态资源外所有路径进行拦截），不要拦截 editor-md 这个文件夹下的所有资源。

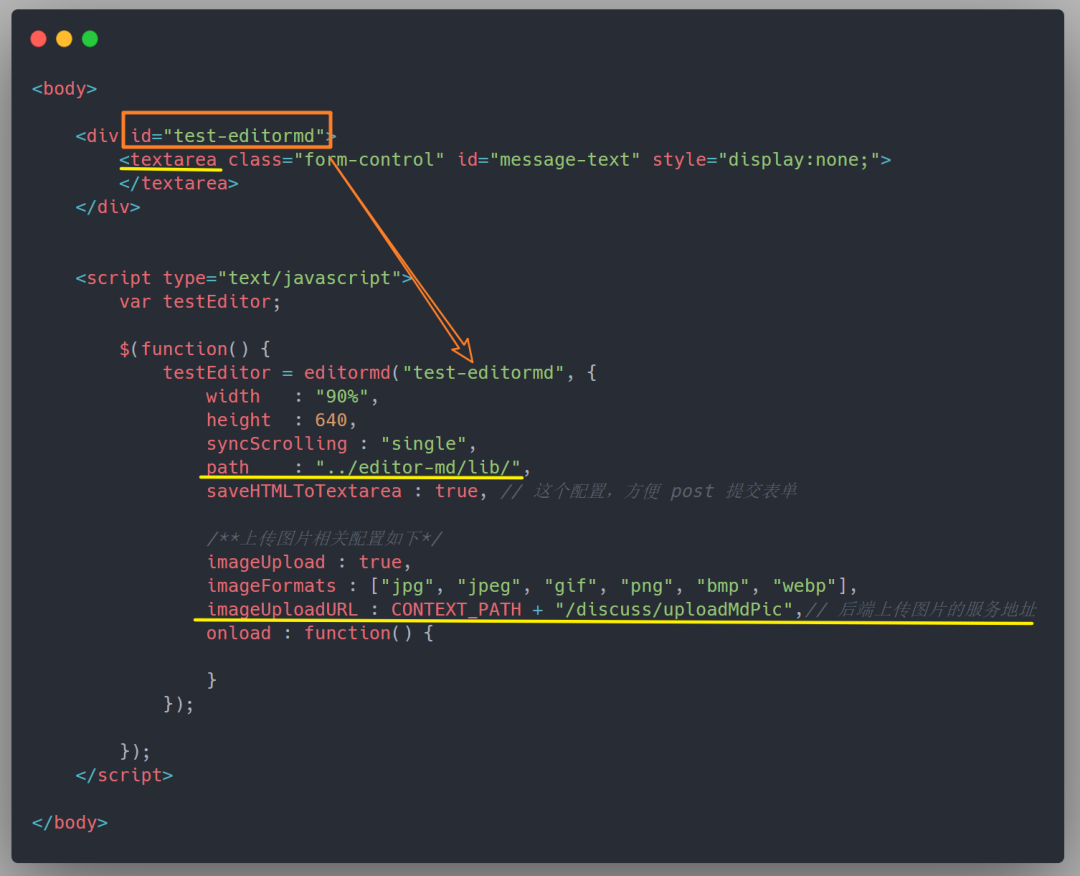


另外，editor.md 自带的 examples 文件夹下的 example.html 文件就是一个示例界面，我们直接将其拷贝到 templates/site 目录下，并重命名为 discuss-publish.html，作为我们的发帖界面。然后将里面所有的静态资源引用都改成这些引用的绝对路径，比如：



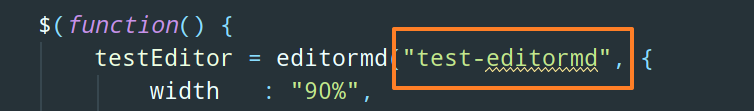
**第二步：修改 discuss-publish.html 代码**

下面贴一下核心的前端代码：



MarkDown 文本的渲染区域就是 textarea 标签包裹的那个区域

下图中框出来的地方填的就是包裹 textarea 的 div 的 id：



另外，需要注意的是 path 路径的配置（lib 文件夹的路径）以及 imageUploadURL（后端上传图片的方法地址），这个在上图中都用黄线标注出来了。

**第三步：编写后端上传图片代码**

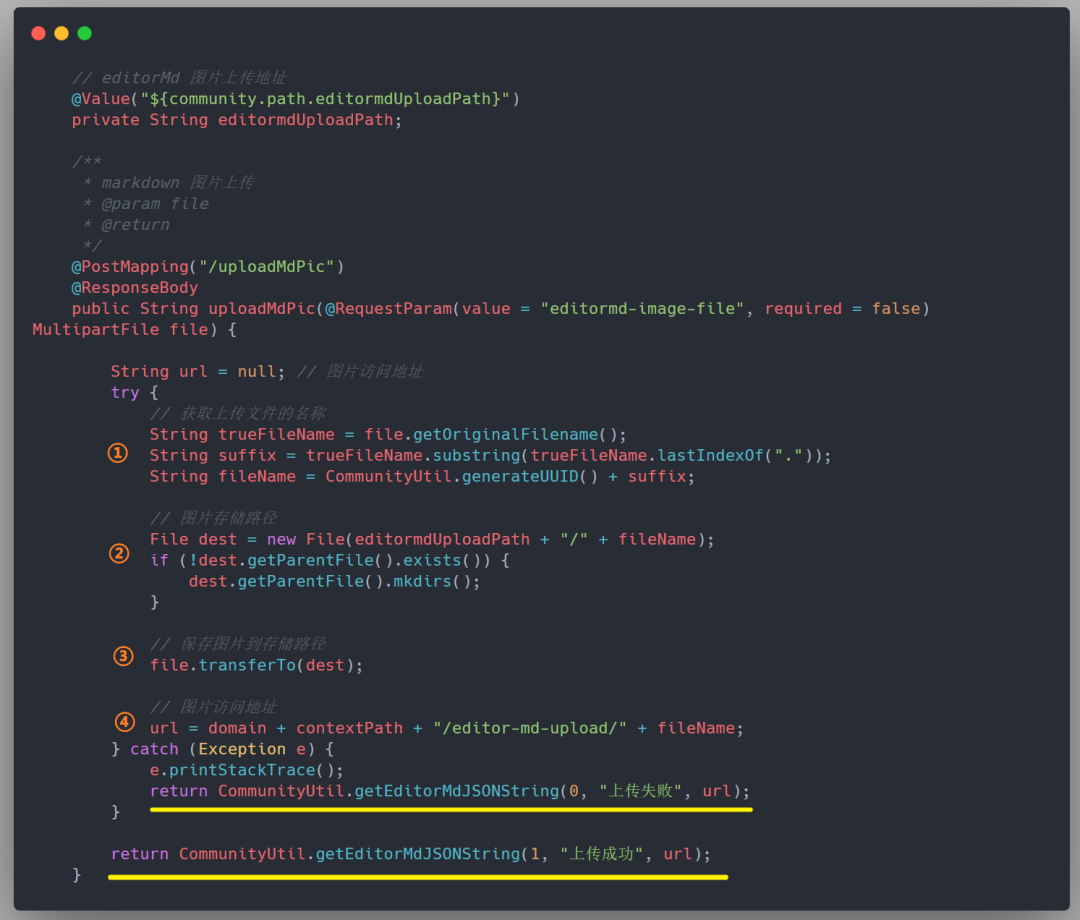
**关于上传图片这里，我就没有做图床服务器这种了，直接把图片上传到本地了**。

编写后端之前首先要做的就是与前端协调好接口格式，那 editor.md 官方说期望我们上传图片的服务返回如下 JSON 格式：

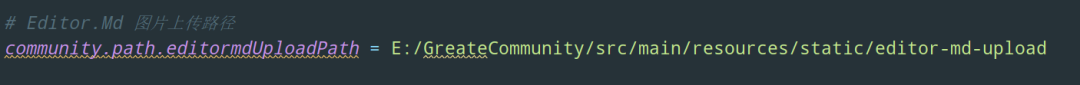
{  
    success : 0 | 1, *// 0 表示上传失败; 1 表示上传成功*    message : "提示的信息",  
    url     : "图片地址" *//上传成功时才返回*}

另外，**editor.md 中选择图片的控件 id 默认是 editormd-image-file**

下面这段上传图片的代码在 DiscussPostController.java 中：



图片上传路径 editormdUploadPath 定义在配置文件中（**这里如果各位小伙伴需要将项目部署到服务器上，需要将本地路径修改为线上路径**）：



图片上传四步走：

* ① 获取上传文件的名称
* ② 生成图片存储路径
* ③ 保存图片到存储路径
* ④ 生成图片访问地址

**第四步：配置虚拟路径映射访问**

不知道各位会不会遇到这样的问题，这主要是 IDEA 对服务器的保护措施导致的：SpringBoot 中上传图片到当前目录下，不能立即显示，需要重启服务器才会显示上传的图片。**服务器不能对外部暴露真实的资源路径，需要配置虚拟路径映射访问**。

在我们的 WebMvcConfig.java 中添加如下方法，对图片上传的文件夹 editor-md-upload 配置虚拟路径映射访问：



**第五步：解决跨域问题**

上述步骤做完后，可能会遇到报错跨域问题，需要在 Security 配置中增加一行配置：

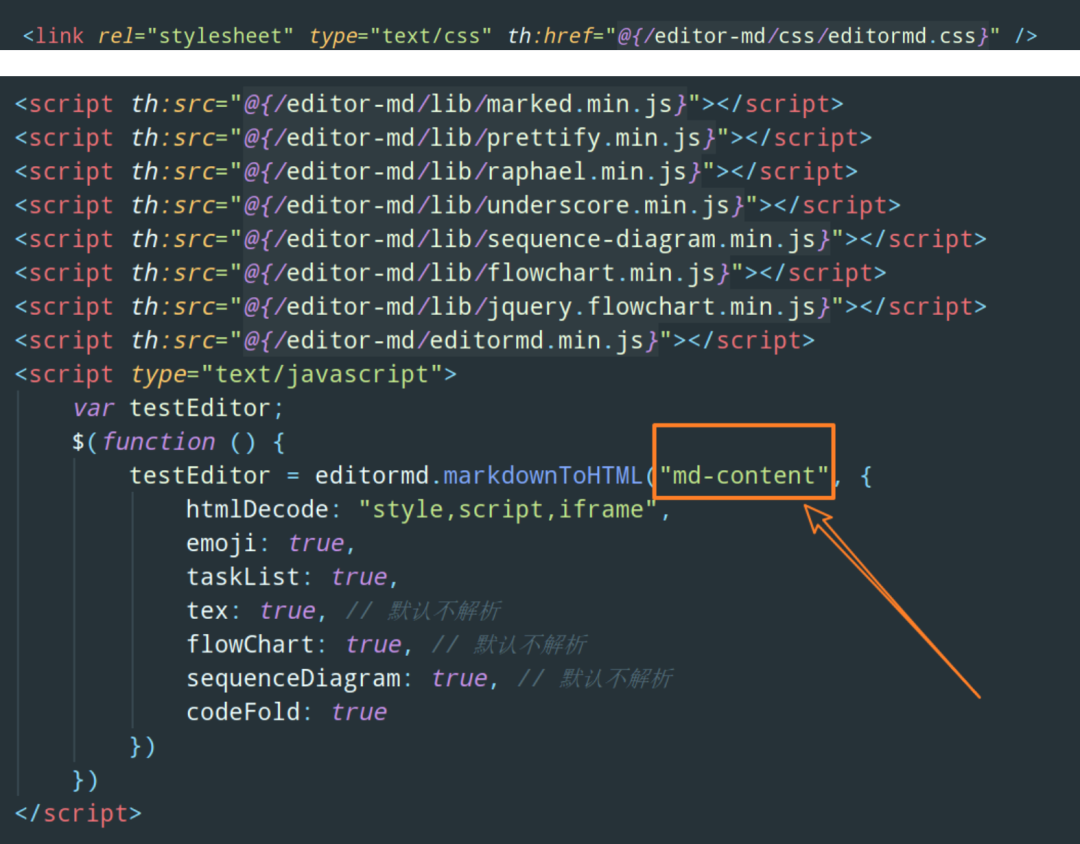
http.headers().frameOptions().sameOrigin();

**第六步：配置 MarkDown 渲染**

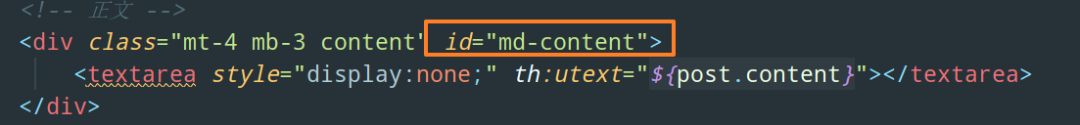
至此，我们做完了写 MarkDown 的工作，接下来做的就是将 MarkDown 渲染成 Html 显示在页面上。

我们需要在帖子详情页 dicuss-detail.html 中做一些修改。

引入 editor.md 的 css 和必要的 js 文件：



可以看见，上面配置了一些 MarkDown 渲染规则，图中标注出来的 md-content 就表示要渲染的区域，即包裹帖子内容的 div 的 id:



### 精选留言

暂无...