

山东大学 计算机科学与技术 学院

云计算技术 课程实验报告

学号：202120130266	姓名：鲍俊杰	班级：计科 21.1
实验题目：利用主流云平台搭建个人博客或网站		
实验学时：2	实验日期：2024/3/22	
<p>实验目的：熟悉使用主流云平台并搭建个人博客或者网站。</p> <p>具体包括：</p> <p>参考方案：基于主流云平台，设计、实现个人博客或者网站的搭建，撰写实验报告（附带网站链接并可以访问），并在网站上呈现此次实验报告。</p> <p>网站地址为：<a href="https://hugbob.github.io">https://hugbob.github.io</a></p>		
<p>硬件环境：</p> <p>联网的计算机一台</p>		
<p>软件环境：</p> <p>Windows or Linux</p>		
<p>实验步骤与内容：</p> <p>一、 下载 node.js 和 git</p> <div><pre>C:\Users\DELL&gt;node -v v20.11.1  C:\Users\DELL&gt;npm -v 10.2.4  C:\Users\DELL&gt;git -v git version 2.37.3.windows.1</pre></div> <p>二、安装 hexo</p> <p>新建一个文件夹用来存放 Hexo 的程序文件，如 myblog。打开该文件夹，右键 -&gt; Git Bash Here。</p> <ul style="list-style-type: none"><li>● 使用 npm 安装 Hexo 博客：</li></ul> <div><pre>npm install -g hexo-cli</pre></div> <ul style="list-style-type: none"><li>● 初始化并安装所需组件：</li></ul> <div><pre>hexo init      # 初始化  npm install    # 安装组件</pre></div> <ul style="list-style-type: none"><li>● 完成后依次输入下面命令，启动本地服务器进行预览：</li></ul>		

```
hexo g # 生成页面
```

```
hexo s # 启动预览
```

访问 <http://localhost:4000>，出现 Hexo 默认页面，本地博客安装成功。

```
DELL@DESKTOP-UGGSD01 MINGW64 /d/myblog
$ hexo init
INFO Cloning hexo-starter https://github.com/hexojs/hexo-starter.git
INFO Install dependencies
INFO Start blogging with Hexo!

DELL@DESKTOP-UGGSD01 MINGW64 /d/myblog
$ npm install

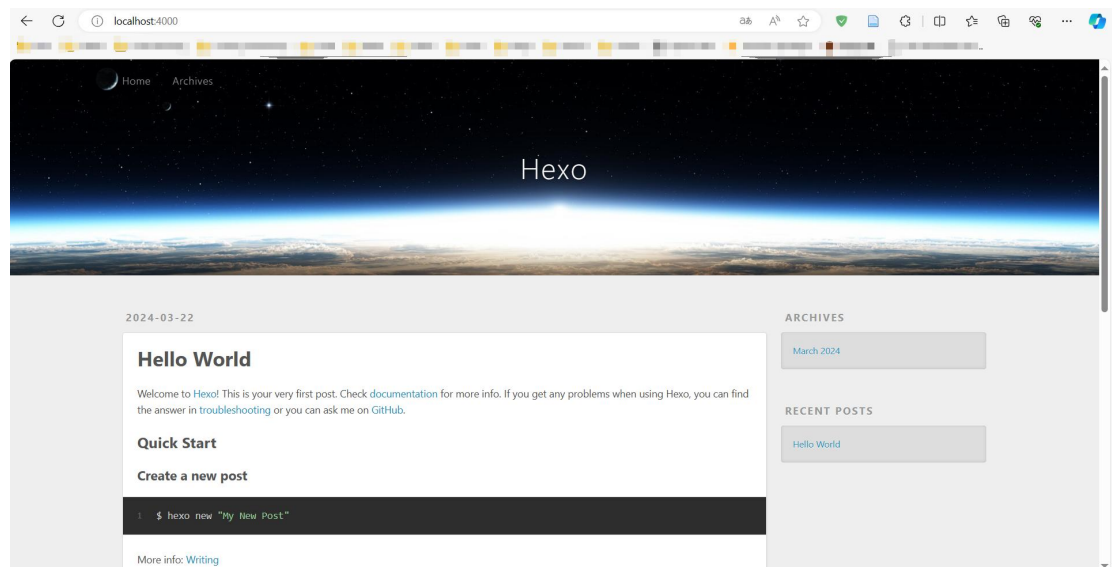
added 1 package in 48s

29 packages are looking for funding
  run `npm fund` for details

DELL@DESKTOP-UGGSD01 MINGW64 /d/myblog
$ hexo g
INFO Validating config
INFO Start processing
INFO Files loaded in 92 ms
INFO Generated: archives/2024/index.html
INFO Generated: archives/index.html
INFO Generated: archives/2024/03/index.html
INFO Generated: index.html
INFO Generated: fancybox/jquery.fancybox.min.css
INFO Generated: fancybox/jquery.fancybox.min.js
INFO Generated: css/style.css
INFO Generated: js/script.js
INFO Generated: js/jquery-3.6.4.min.js
INFO Generated: css/images/banner.jpg
INFO Generated: 2024/03/22/hello-world/index.html
INFO 11 files generated in 204 ms

DELL@DESKTOP-UGGSD01 MINGW64 /d/myblog
$ hexo s
INFO Validating config
INFO Start processing
INFO Hexo is running at http://localhost:4000/. Press Ctrl+C to stop.
```

在 localhost 运行结果如下所示：



三、本地博客测试成功后，上传到 GitHub 进行部署，使其能够在网络上访问。

首先安装 hexo-deployer-git：

```
npm install hexo-deployer-git --save
```

#### 四、创建并连接 github

新建一个名为 用户名.github.io 的仓库：在本实验中，以用户名为 hugbob 为例，新建名为 hugbob.github.io 的仓库，网站地址为：<https://hugbob.github.io>（还需要创建 SSH）

```
DELL@DESKTOP-UGGSD01 MINGW64 /d/myblog
$ ssh -T git@github.com
The authenticity of host 'github.com (20.205.243.166)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TujJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Hi hugbob! You've successfully authenticated, but GitHub does not provide shell
access.
```

#### 五、修改博客的配置文件

打开博客根目录下的\_config.yml 文件（博客的配置文件，修改与博客相关的各种信息）。在本实验中以用户名为 hugbob 为例，修改最后一行的配置为：

```
deploy:
  type: git
  repository: git@github.com:hugbob/hugbob.github.io.git
  branch: main
```

#### 六、将本地文件部署到 github

执行下面几条命令，将博客部署到 GitHub 上：

```
hexo clean
```

```
hexo generate
```

(若要本地预览就先执行 hexo server)

```
hexo deploy
```

快捷命令：

```
hexo g == hexo generate
```

```
hexo d == hexo deploy
```

```
hexo s == hexo server
```

```
hexo n == hexo new
```

还能组合使用，如：

```
hexo d -g
```

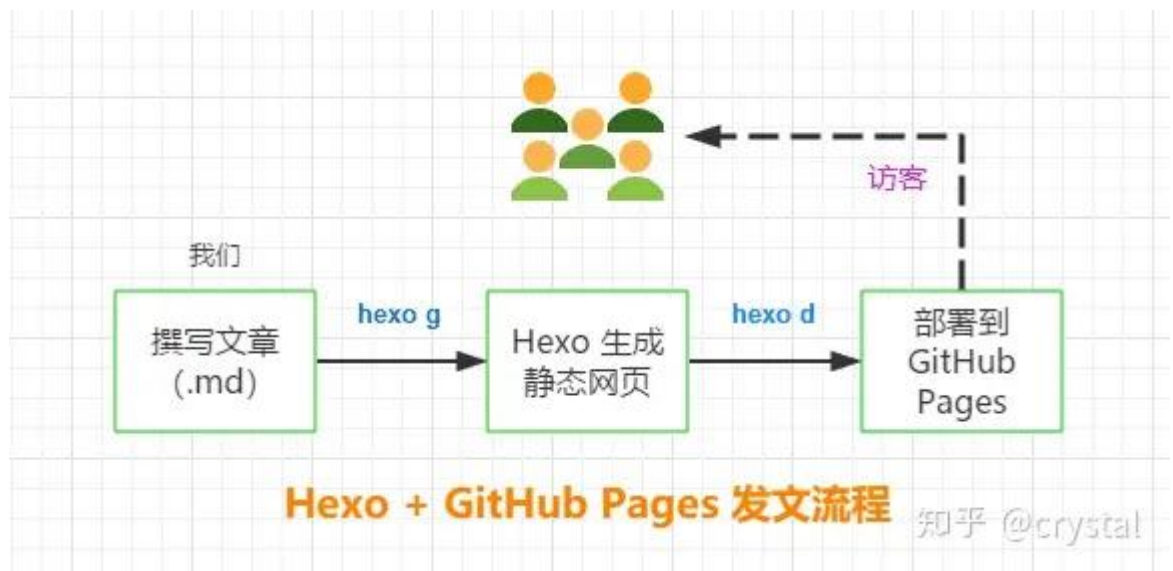
## 六、将实验报告上传至博客



## 结论分析与体会：

### 1. Hexo + GitHub 文章发布原理

在本地撰写 Markdown 格式文章后，通过 Hexo 解析文档，渲染生成具有主题样式的 HTML 静态网页，再推送到 GitHub 上完成博文的发布。



2. Hexo 很难对图片进行有效的管理，本地 markdown 文件中的图片会出现无法在网页中显示的情况，解决方法如下：

(1) 设置站点配置 config.yml  
post\_asset\_folder: true

(2) 安装插件

```
npm install hexo-asset-image -- save
```

### (3) 引用图片

运行 `hexo n "xxxx"` 来生成 md 博文时，`/source/_posts` 文件夹内除了 `xxxx.md` 文件还有一个同名的文件夹。接下来就是利用 markdown。

在 `xxxx.md` 中想引入图片时，先把图片复制到 `xxxx` 这个文件夹中，然后只需要在 `xxxx.md` 中按照 markdown 的格式引入图片：

![这里输入图片描述] (图片名.jpg)

必须只能是图片名，不能含有任何前缀，否则网页端无法显示（但这样会导致本地文件中不会显示），可以在编辑时加入相对地址能够在本地显示便于查看，部署时再删除相对路径只保留图片的文件名。

```
# hexo-asset-image
```

```
Give asset image in hexo a absolutely path automatically
```

```
# Usege
```

```
```shell
```

```
npm install hexo-asset-image --save
```

```
```
```

```
# Example
```

```
```shell
```

```
MacGesture2-Publish
```

```
├─ apppicker.jpg
```

```
├─ logo.jpg
```

```
└─ rules.jpg
```

```
MacGesture2-Publish.md
```

```
```
```

```
Make sure `post_asset_folder: true` in your `_config.yml`.
```

```
Just use `![[logo]](logo.jpg)` to insert `logo.jpg`.
```