

# 测试文档

## 一、 测试目的

1. 通过对测试结果的分析，得到对软件质量的评价
2. 分析测试的过程，产品，资源，信息，为以后制定测试计划提供参考
3. 分析系统存在的缺陷，为修复和预防 bug 提供建议

## 二、 测试环境和工具

Window10

Python3.6.0

Django1.10.3

使用 Pagespeed 测试工具

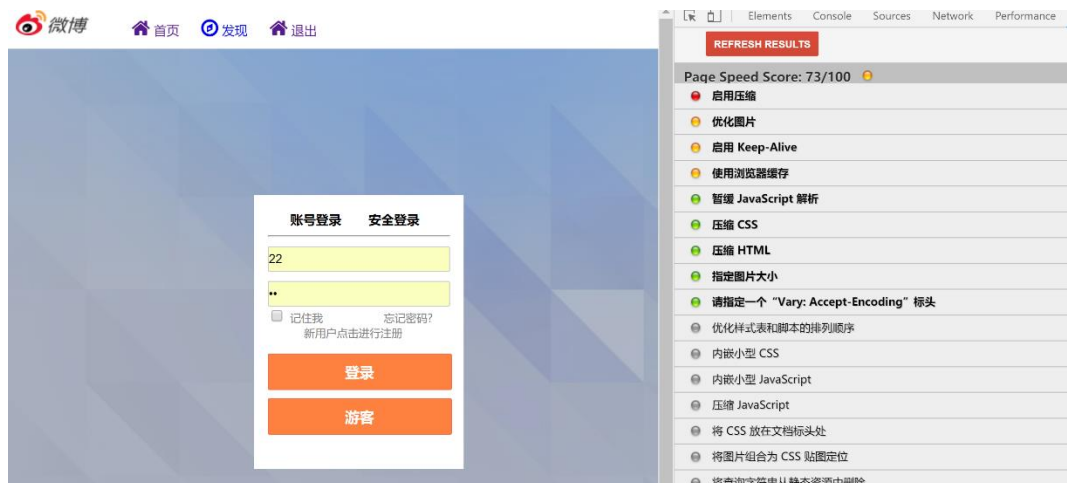
## 三、 测试的主要内容

使用 pagespeed 插件进行性能测试，测试的主要内容有：

- ① 优化缓存——让你应用的数据和逻辑完全避免使用网络
- ② 减少回应时间——减少一连串请求-响应周期的数量
- ③ 减小请求大小——减少上传大小
- ④ 减小有效负荷大小——减小响应、下载和缓存页面的大小
- ⑤ 优化浏览器渲染——改善浏览器的页面布局

## 四、 测试结果

- ① 登录界面：



主要问题在于启用压缩：

## 启用压缩

为以下资源启用压缩可将其传送大小减少150.6 KiB (67%)。

- 压缩 <http://127.0.0.1:8000/static/Commons/JavaScript/jquery.min.js> 可减少58.6 KiB (64%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/JavaScript/jquery-2.1.4.min.js> 可减少53.5 KiB (64%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Plugins/font-awesome/css/font-awesome.css> 可减少79%)。
- 压缩 <http://127.0.0.1:8000/login/> 可减少6.6 KiB (76%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Css/login.css> 可减少2.4 KiB (74%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Css/layout.css> 可减少2.1 KiB (71%)。

[More information](#)

启用压缩可以在整个服务器范围内或特定目录下启用 HTTP 压缩，HTTP 压缩不仅可以改善带宽使用情况，还可以提高网站性能。此外还可以响应静态响应和动态响应。

② 注册界面:

The screenshot shows the Weibo registration page with a performance audit overlay on the right. The page has a blue header with the Weibo logo and the text '注册微博'. The registration form includes fields for '用户名:' (Username), '简介:' (Bio), '密码:' (Password), '确认密码:' (Confirm Password), '邮箱:' (Email), '性别:' (Gender) with a dropdown menu showing 'male', and '年龄:' (Age). The performance audit overlay shows a 'Page Speed Score: 70/100' and a list of recommendations. The first recommendation is '提供压缩后的图片' (Provide compressed images), which is highlighted. Other recommendations include '优化图片' (Optimize images), '压缩 JavaScript', '使用浏览器缓存', '启用 Keep-Alive', '暂缓 JavaScript 解析', '优化样式表和脚本的排列顺序', '启用压缩', '压缩 HTML', '请指定一个 "Vary: Accept-Encoding" 标头', '内嵌小型 CSS', '内嵌小型 JavaScript', '压缩 CSS', '将 CSS 放在文档标头处', '将图片组合为 CSS 贴图定位', '将查询字符串从静态资源中删除', '尽量减少请求的数据量', '尽量减少重定向', '指定图片大小', '指定字符集', and '由同一网址提供资源'.

主要问题是优化图片

## 提供压缩后的图片

以下图片的尺寸在 HTML 或 CSS 中经过了调整，提交调整后的图片可减少165.5 KiB (99%)。

- <http://127.0.0.1:8000/static/Commons/Images/weibo-logo.png> 的尺寸在 HTML 或 CSS 中从1759x558调整到70x50。提供调整后的图片可减少165.5 KiB (99%)。

[More information](#)

## 优化图片

优化以下图片可将其大小减少123.0 KiB (45%)。

- Lossy compressing [http://127.0.0.1:8000/static/Commons/Images/body\\_bg\\_login.jpg](http://127.0.0.1:8000/static/Commons/Images/body_bg_login.jpg) could save 82.3 KiB (75% reduction). [Download](#) optimized version.
- Lossy compressing <http://127.0.0.1:8000/static/Commons/Images/weibo-logo.png> could save 40.7 KiB (25% reduction). [Download](#) optimized version.

优化图片也是改善性能的一个方法，通过查阅资料发现，图片的 Alt 标签是目前公

认对图片优化最重要的部分，是搜索引擎判断图片内容的重要因素，其次，Alt 标签的文本内容会在图片无法读取时得以显现，所以给图片加上 Alt 标签是比较好的习惯。

- ③ 忘记密码界面，存在的可优化部分和登陆注册相同
- ④ 微博主页：



主要问题是启动 Keep-Alive 和启动压缩

Page Speed Score: 64/100

**启用 Keep-Alive**

主机 应启用 Keep-Alive，它提供了以下资源。

- [http://127.0.0.1:8000/get\\_new\\_wb/?user\\_id=2](http://127.0.0.1:8000/get_new_wb/?user_id=2)
- [http://127.0.0.1:8000/get\\_old\\_wb/?user\\_id=2](http://127.0.0.1:8000/get_old_wb/?user_id=2)
- <http://127.0.0.1:8000/index/>
- <http://127.0.0.1:8000/static/Commons/Css/face.css>
- <http://127.0.0.1:8000/static/Commons/Css/index.css>
- <http://127.0.0.1:8000/static/Commons/Css/layout.css>
- <http://127.0.0.1:8000/static/Commons/Css/reset.css>
- [http://127.0.0.1:8000/static/Commons/Images/WB\\_logo.png](http://127.0.0.1:8000/static/Commons/Images/WB_logo.png)
- [http://127.0.0.1:8000/static/Commons/Images/body\\_bg\\_login.jpg](http://127.0.0.1:8000/static/Commons/Images/body_bg_login.jpg)
- <http://127.0.0.1:8000/static/Commons/Images/bz.jpg>
- <http://127.0.0.1:8000/static/Commons/Images/icon.png>
- <http://127.0.0.1:8000/static/Commons/Images/touxiang.jpg>
- <http://127.0.0.1:8000/static/Commons/Images/touxiang.png>
- <http://127.0.0.1:8000/static/Commons/JavaScript/jquery-2.1.4.min.js>
- <http://127.0.0.1:8000/static/Commons/JavaScript/jquery.qqFace.js>
- <http://127.0.0.1:8000/static/Commons/Plugins/arlist/53.gif>
- <http://127.0.0.1:8000/static/Commons/Plugins/arlist/62.gif>
- <http://127.0.0.1:8000/static/Commons/Plugins/font-awesome/css/font-awesome.css>
- <http://127.0.0.1:8000/static/Uploads/1/temp/111120190108173101.png>

[More information](#)

**启用压缩**

为以下资源 **启用压缩** 可将其传送大小减少 142.6 KiB (74%)。

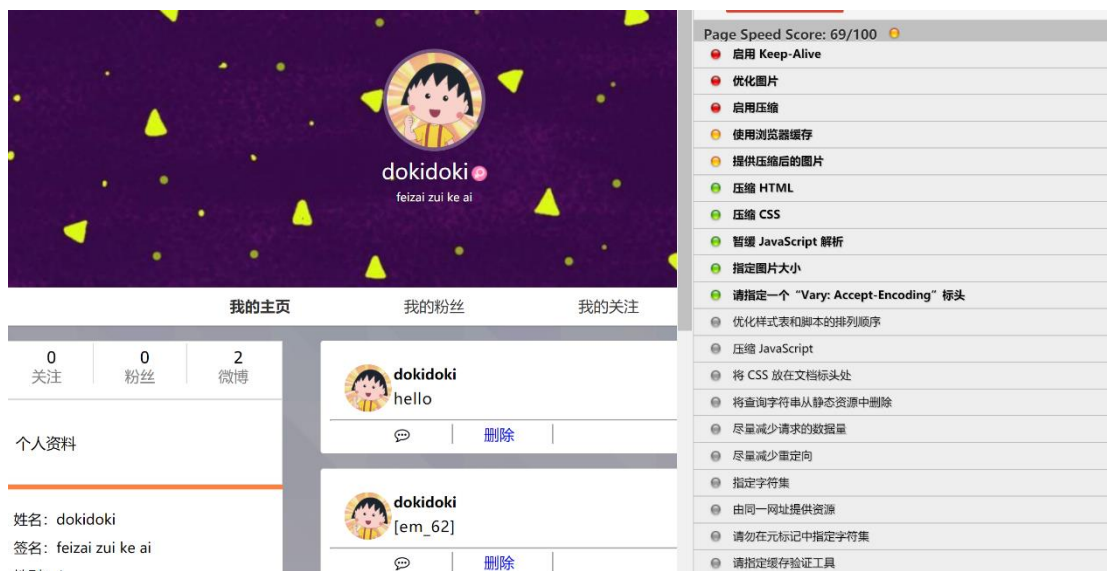
- 压缩 <http://127.0.0.1:8000/static/Commons/JavaScript/jquery-2.1.4.min.js> 可减少 53.5 KiB (64%)。
- 压缩 <http://127.0.0.1:8000/index/> 可减少 49.3 KiB (86%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Plugins/font-awesome/css/font-awesome.css> 可减少 27.4 KiB (79%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Css/index.css> 可减少 5.6 KiB (75%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Css/layout.css> 可减少 2.1 KiB (71%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/JavaScript/jquery.qqFace.js> 可减少 1.9 KiB (60%)。
- 压缩 <http://127.0.0.1:8000/static/Commons/Css/face.css> 可减少 1.3 KiB (61%)。
- 压缩 [http://127.0.0.1:8000/get\\_old\\_wb/?user\\_id=2](http://127.0.0.1:8000/get_old_wb/?user_id=2) 可减少 723 B (70%)。
- 压缩 [http://127.0.0.1:8000/get\\_new\\_wb/?user\\_id=2](http://127.0.0.1:8000/get_new_wb/?user_id=2) 可减少 718 B (70%)。

Keep-Alive

我们知道 HTTP 协议采用“请求-应答”模式，当使用普通模式，即非 KeepAlive 模

式时，每个请求/应答客户和服务器都要新建一个链接，完成之后立即断开连接（HTTP 协议为无连接的协议）；当使用 Keep-Alive 模式（又称持久连接，连接重用）时，Keep-Alive 功能使客户端到服务器端的连接持续有效，当出现对服务器的后继请求，Keep-Alive 功能避免了建立或者重新建立连接。

#### ⑤ 个人主页：



问题大致类似，可以通过启用 Keep-Alive，优化图片和启用压缩来优化性能。

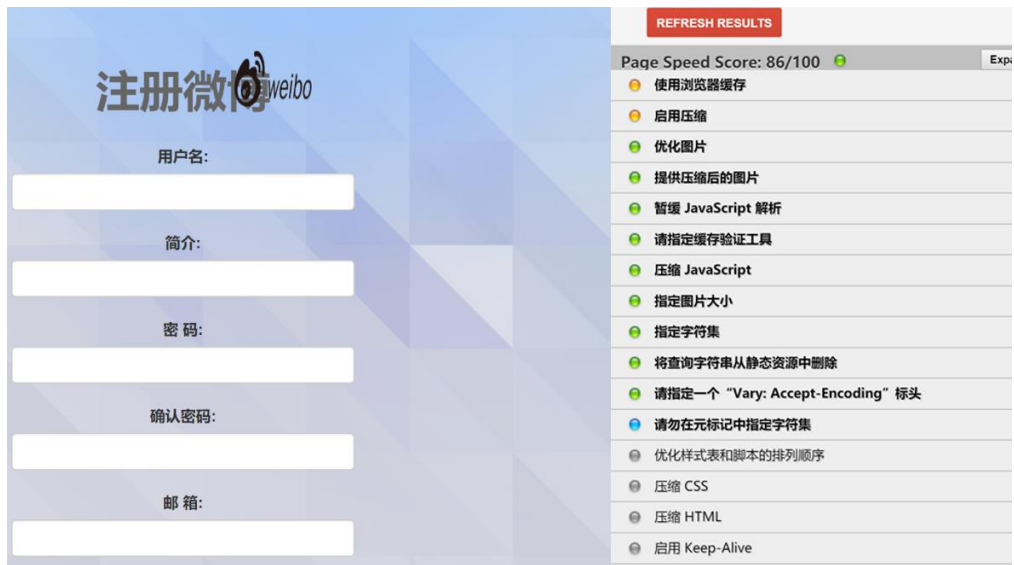
#### ⑥ 搜索结果



搜索功能的性能较好，得分较高，为 85 分

## 五、 测试结果分析：

该网站的测试结果总体较好，问题主要集中于启用压缩和优化图片，我们通过 Gzip 压缩来解决问题，为 Gzip 开启以后会将输出到用户浏览器的数据进行压缩的处理，这样就会减小通过网络传输的数据量，使得网站的性能有了较大的改善，例如注册界面。



优化性能的方法还有很多，通过查阅资料，我们得知还有很多方法：

网页内容	服务器	Cookie	图片
<a href="#">减少http请求次数</a>	<a href="#">使用CDN</a>	<a href="#">减少Cookie大小</a>	<a href="#">优化图像</a>
<a href="#">减少DNS查询次数</a>	<a href="#">添加Expires或Cache-Control报头</a>	<a href="#">页面内容使用无cookie域名</a>	<a href="#">优化CSS Sprite</a>
<a href="#">避免页面跳转</a>	<a href="#">Gzip压缩传输文件</a>	<b>CSS</b>	<a href="#">不要在HTML中缩放图片</a>
<a href="#">缓存Ajax</a>	<a href="#">配置ETags</a>	<a href="#">将样式表置顶</a>	<a href="#">使用小且可缓存的favicon.ico</a>
<a href="#">延迟加载</a>	<a href="#">尽早flush输出</a>	<a href="#">避免CSS表达式</a>	<b>移动客户端</b>
<a href="#">提前加载</a>	<a href="#">使用GET Ajax请求</a>	<a href="#">用&lt;link&gt;代替@import</a>	<a href="#">保持单个内容小于25KB</a>
<a href="#">减少DOM元素数量</a>	<a href="#">避免空的图片src</a>	<a href="#">避免使用Filters</a>	<a href="#">打包组建成符合文档</a>
<a href="#">根据域名划分内容</a>		<b>Javascript</b>	
<a href="#">减少iframe数量</a>		<a href="#">将脚本置底</a>	
<a href="#">避免404</a>		<a href="#">使用外部Javascript和CSS文件</a>	
		<a href="#">精简Javascript和CSS</a>	
		<a href="#">去除重复脚本</a>	
		<a href="#">减少DOM访问</a>	
		<a href="#">使用智能事件处理</a>	

其中减少 http 请求次数，避免页面跳转，优化图像，优化 CSS 等对该网站的性能改善都有较大的可行性。