**需求分析规格说明**

**文档名称：单页应用搜索引擎优化系统需求分析规格说明**

**项目名称：单页应用搜索引擎优化系统**

**编写人：赖信涛**

1 问题定义

本系统的核心模块是对页面中JavaScript的执行以及爬虫，这是基本的功能。何时进行对网站更新，保证与原网页的实时性又不对网站的访问太过频繁是系统优化的重点。

对于信息展示类的网站来说，大约有70%的流量来自于搜索引擎的结果引入。搜索引擎对于网站的收录，对网站来说是至关重要的实验数据显示约85 % 的用户只翻看搜索引擎 返回结果的前10 个结果,即返回结果页面的第一页。这个用户行为决定了尽管搜索引擎返回的结果 数目十分庞大,但真正可能被绝大部分用户所浏览的,只有排在最前面的很小一部分而已[8]。图中展示了用户的浏览习惯，基本上靠前的一小部分结果占据了搜索引擎的大部分流量，而互联网的主要流量来自于这里。要想提高网站在搜索引擎展示结果中的数量和排名（排名越靠前，被点击的概率越大），首先要保证网站的内容能正常并且全面地被搜索引擎收录。

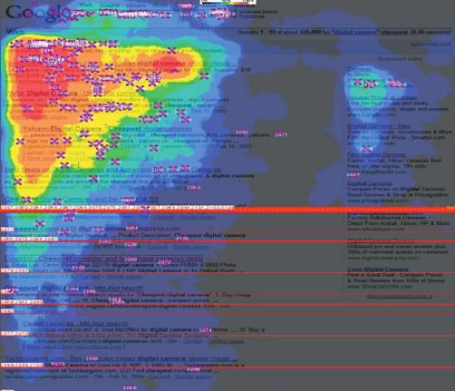


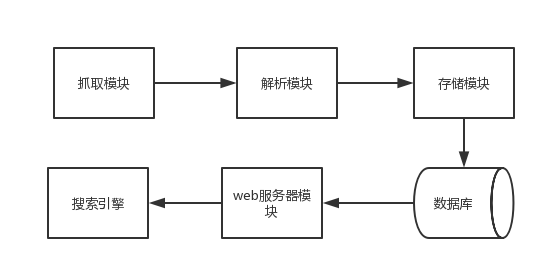
图 用户浏览习惯金三角

如果单页应用没有针对搜索引擎优化的话，是很难被正常收录的。因为单页面应用只有一个页面，搜索引擎只是下载页面的HTML文档，并不会执行文档中包含的脚本。这样，就造成了搜索引擎收录页面数量少、收录到的页面内容空洞、价值不高、网页的页面之间结构化差，从而会进一步导致网站在搜索引擎的结果中展示的频率低、排名靠后。会损失很多宝贵的搜索流量来源。

所以单页应用不同于一般的网页，除了常规的搜索引擎优化工作之外，还需要一些特殊的搜索引擎优化方案，来保证单页应用所有的页面能够被正常收录。本文的目的便是解决这个问题，如果能让单页应用被搜索引擎收录，就可以带来很多宝贵的搜索流量，提升网站的价值。目前流行的一些其他搜索引擎优化方案，也可以基于本文做跟深一步的优化。

2需求分析

针对本文要解决的问题，即让搜索引擎正常收录单页应用，需要将原网站的网页解析并保存下来，还要能在搜索引擎的网页爬虫发来请求的时候，返回解析后的网页作为响应。所以，需要爬虫模块来抓取网页，phantomjs模块解析网页，存储模块保存网页，web服务器模块响应网页，这几个模块相互配合。工作方式如图3-2所示。



各个模块的协调方式

将网页抓取下来，是系统的基础功能。能通过页面中的链接连续不断地自主发现新的页面，自己分配工作，是理想中的爬虫的工作方式。作为一个爬虫模块，任务调度必不可少，必须使用一种合理的任务队列，使用调度算法分配任务，才能使爬虫稳定、有序地执行。除此之外，还需要解析页面中的链接，将新的链接放回到队列中。这个过程应该注意发现链接的效率，和内存占用，以及判断是否是本站链接，如果不是，不应该作为任务执行。如果提供sitemap，爬虫可以根据sitemap进行更全面地抓取。

对HTML文档的解析是系统的重点功能，有一个专门的模块负责解析的任务。由于这部分解析JavaScript脚本需要使用JavaScript语言来编写程序，系统的其余模块需要使用Python编写，所以为了使这两个模块之间能互相配合，便将解析JavaScript的模块以服务的形式开发。运行的时候监听某一个端口，python程序通过往这个端口发送请求，此模块回应请求，以这种方式交互。

存储模块需要实现的是将保存下来的网页与新抓取的网页进行快速对比，判断网页是否已经更新。系统可以将网页提交给搜索引擎，被搜索引擎正常收录。其中对于文档的对比是一个技术难点，页面的对比必须满足程序执行速度块，占用内存少，占用存储空间少的几个要求。此外，存储使用的数据库可能要面对非结构化的数据，可以灵活扩展与管理。

Web服务器模块的功能比较简单，只是将已经保存的网页返回便可以。但是有一种特殊的情况，就是有请求的目标是尚未被索引的网页。这种情况下应该马上去处理这个页面，处理完之后将页面反回给浏览器。也就是说，Web服务器应该有往任务队列中放高优先级任务的机制。

此外，稳定性与安全性是本系统的目标。所以应该使用容器机制和合适的进程管理界面。应该提供直观地观察系统运行状态的方式，能够快速重启或维护系统各个组件。维护方面应该使用授权机制，保证基本的安全性。

性能需求

要保证系统的可用性，最基本的是保证系统能够对新产生的网页进行快速收录。对搜索引擎爬虫发送过来的请求，要保证及时响应，对于已经收录的网页，这点非常简单，只要读取数据库一次，读取文件系统一次，即可以完成对请求的响应。还要要尽量减少对源网站的请求次数，避免使原网站的负载压力过大[9]。初次之外，系统还应该保证以下几点。

1. 健壮性：对自身网站存在的错误，要能够处理，不能导致系统崩溃。
2. 易监控：提供监控页面，可以快速、直观地查看系统运行情况。
3. 易维护：系统出现问题后，可以快速定位到问题并排查。

4开发环境

对于本文提出的系统来说，稳定性是首先要考虑的因素，所以采用了Linux系统以及相关的工具进行开发，结合容器与进程管理程序，以求尽可能保证程序的正常运行和实时监控。

硬件要求： 苹果笔记本电脑。

软件： zsh shell（配合oh-my-zsh），tmux终端工具，vim编辑器，git版本控制工具。

开发语言：JavaScript，Python，shell。

5小结

本章对本系统和框架进行了详细介绍。详细地分析了系统的功能，并且定义了性能要求。证明了该方案是可能的，为后面的开发增加了信心，奠定了基础。此外，还介绍了开发环境。