

# Report

## 1. 关于设计思路的简单描述

本次设计后端用node.js+express框架，前端采用Angular js.

1) 设计一个页面（前端Angular js index.html），输入网站，点击generate，输出短地址

2) 点击生成时，存储 key value形式，key为短地址 value 为 网站地址。

（PS:文档中要求同一个地址被转化，短地址必须相同，所以存完上述的key value后，然后再存一个 md5(url),key, 对请求的url进行md5加密算法，每次请求前通过md5(url)进行判定是否存在）

3)key 规则为5位 数字+ 字母。用js写个随机生成规则

（转化规则具体代码见server.js getRandom. Math.random().toString(32) 这个方法生成一个小于0的随机数，从第二位开始，截取位）

4)Value由于是url带很多参数，传递的时候会出问题,需要进行base64编码，再存储

5)生成完key, value之后，把url/key 形式返回给用户，即输出短地址

6)用户访问url/key地址（短地址）时，后端获取key，然后取到value，进行base64解码，然后301跳转到用户网站地址

## 2. 简单的数据库文档，API文档

本次作业采用文件存储方式（PS:Mongodb 因为总显示占用端口号，一直无法安装成功，迫于时间关系，采用文件存储方式）

数据库存储有两项： 一个是 key( md5(url) ), value ( 短地址（5位随机数））

一个是 key ( 短地址（5位随机数） ), value (base64 (网址) )

API文档： RESTful api :/save/:msg和/:shortsite

## 3 如何运行，测试

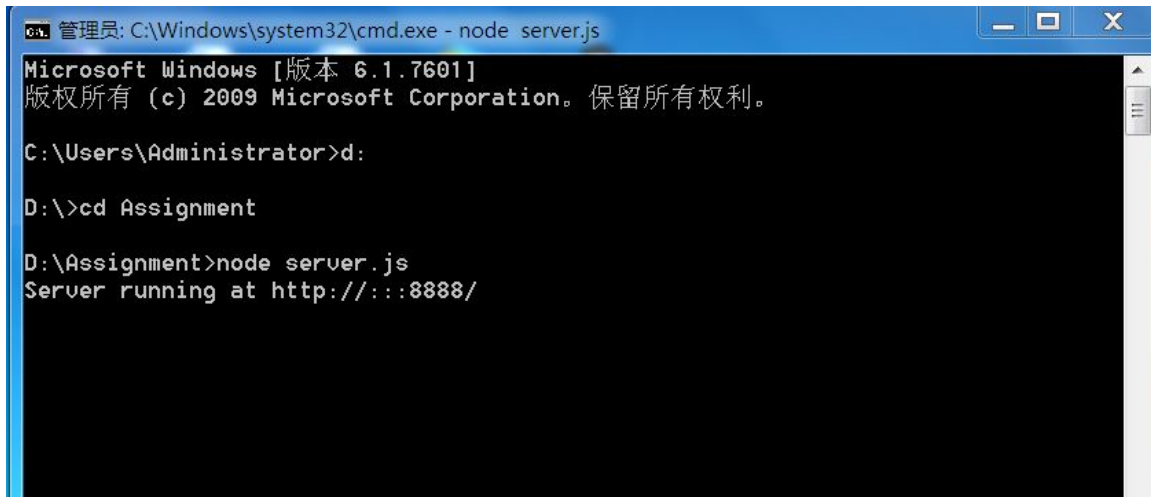
打开浏览器,输入<http://127.0.0.1:8888/index.html>,输入your site, 点击generate按钮,生成short site。访问short site,直接跳转到your site.

最重要的一点,同一个长网页连续点击generate两次,出现的短地址应相同。

如果满足上述功能,即为成功。

#### 4. 实际部署一份实例

cmd, 命令行启动node server.js



```
管理员: C:\Windows\system32\cmd.exe - node server.js
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>d:

D:\>cd Assignment

D:\Assignment>node server.js
Server running at http://:::8888/
```

2.打开浏览器 输入 <http://127.0.0.1:8888/index.html>,



please input your site:

---

output short site:

输入你想转化的网址，点击 generate



The screenshot shows a web browser window with a tab titled '127.0.0.1:8888/index.html'. Below the address bar, there is a form with two input fields and a button. The first input field is labeled 'please input your site:' and contains the text 'www.google.com'. To its right is a button labeled 'generate'. Below this, the second input field is labeled 'output short site:' and contains the text 'http://127.0.0.1:8888/kur44'.

把生成的短地址，复制粘贴进浏览器



重定向到 google

Zhuowen Gmail In



**注解：**文件夹中 `server.js` 是最终版。带 `bak` 的是我研究的时候测试用的。总共有两个 `html` 文件，`index.html` 是用 `angularjs` 写的前端，`jquery.html` 是我一开始用 `jquery` 尝试写的，可以忽略哈！