

短链接生成器

用户手册

作者：张翔宇



目录

写在前面	1
概述	2
文件结构.....	2
RESTful API 与前端	3
数据库模型	4
部署方法.....	4
预览	5

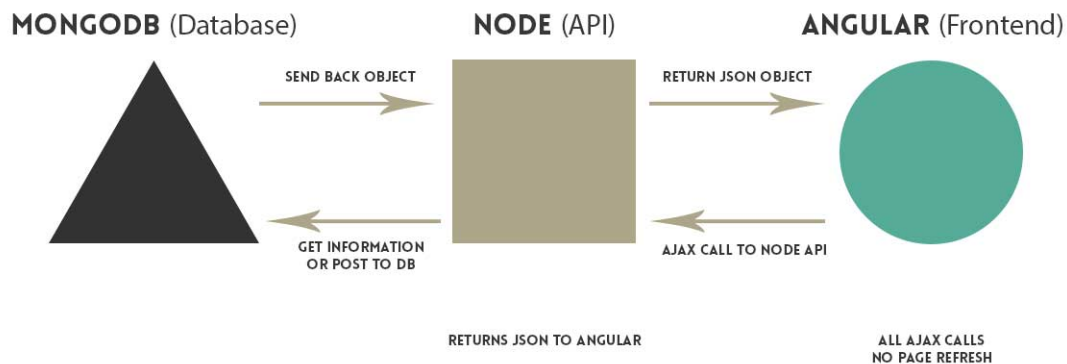
写在前面

谢谢燎原能个我这次尝试，项目用时 20 小时。实际上我之前完全没有关于全栈工程师的经验，这次算是极限创作。全部工作都是独立完成，虽然完成了任务，但是需要承认关于 NodeJS + AnjularJS + 数据库的知识体系还是很不完善，项目过程中我借鉴了很多有些的作品，下面是我用到的资源：

1. <http://broccoli.ecs.umass.edu:3000/b>（前端后端数据库协同工作的思路借鉴自这个资源）
2. <http://broccoli.ecs.umass.edu:3000/c>（数据库模型，API 思路借鉴自这个资源）
3. <http://broccoli.ecs.umass.edu:3000/d>（编码算法，例如 Base58, Base64 借鉴自 Wiki）
4. 其他资源均来自于互联网

概述

整个项目是通过 Express 框架设计的，根据作业要求，应用可以分成三个部分。首先搭建 HTML+CSS 的前端，并且嵌入 Angular JS 来发起事件，第二，当前端发起事件，用 NodeJS 提供 RESTful API 响应事件，最后搭建数据库为 API 提供数据服务。整体的结构可以用下图描述：



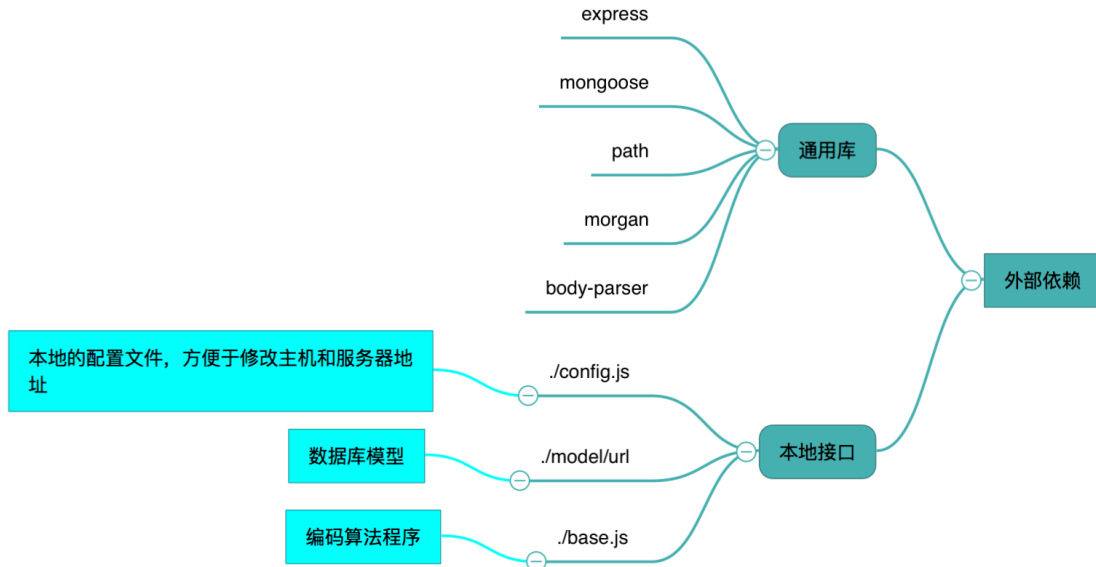
文件结构

因为用的是 Express 框架，所以原文件结构大同小异

- *models*(数据库模型)
- *node_modules*
- *public* (网页前端资源)
 - CSS*
 - javascript*(AngularJS 资源)
 - res* (媒体资源)
- *views* (HTML 外观)
- *doc* (用户手册)
- *app.js* (antry point)
- *base58.js* (编码算法)
- *config.js* (配置文件)

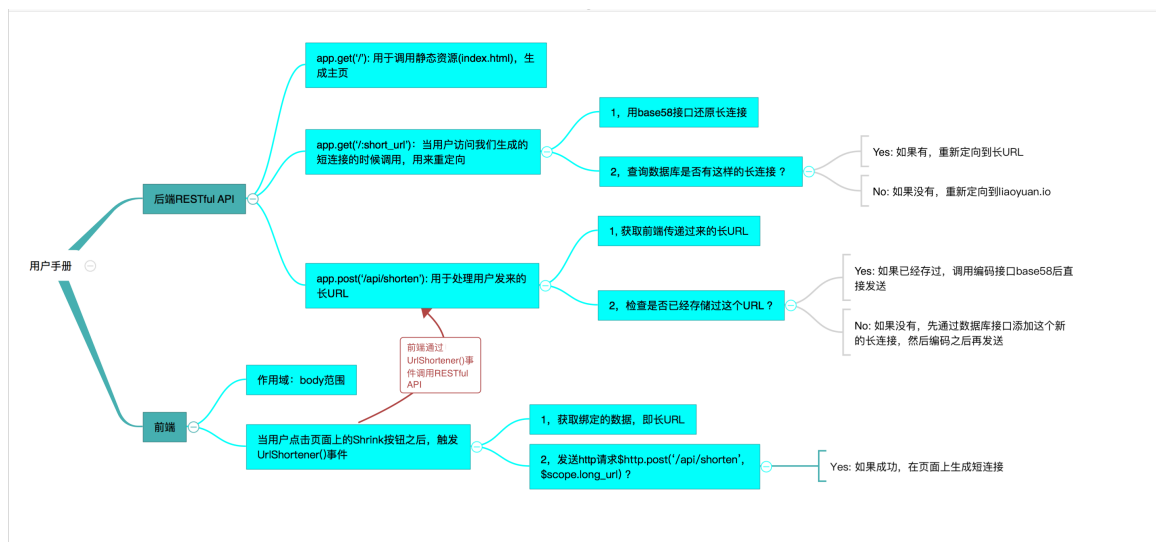
- `package.json` (npm 项目简介)

文件的依赖关系如下所示：



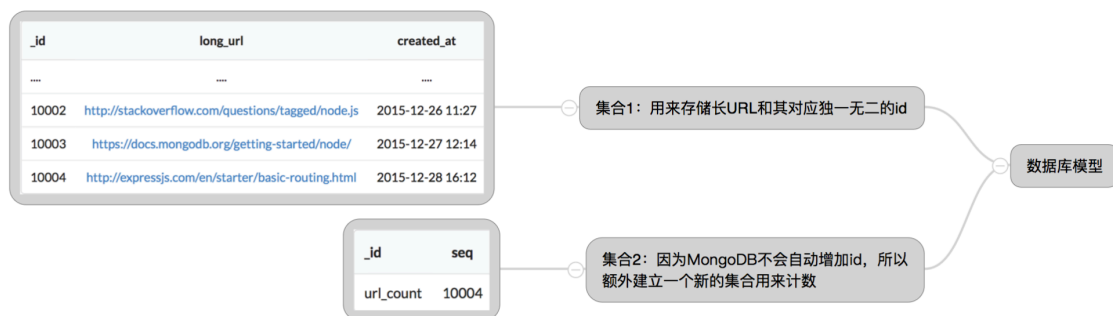
RESTFUL API 与前端

前端通过 `UriShortener()` 来发起事件，HTML 中的数据来自于与 Angular JS 绑定的数据。具体的前后端关系图如下：



数据库模型

需要两个集合来存储数据，第一个是用来存储长 URL 和其对应 ID。还有一个是计数器，因为 MONGODB 的_id 并不会自动增加，所以额外创建一个。具体的数据库模型如下：



部署方法

默认的环境是已经安装 NPM, NodeJS （或者 Node，视平台而定），还有 MongoDB。

1. 拷贝仓库

https://github.com/xiangyuzhang/liaoyuan_URL.git

2. 安装依赖

```
$ npm install express --save
$ npm install body-parser --save
$ npm install mongoose --save
```

3. 打开并创建数据库

在终端输入

```
$ mongo
$ use url_shortener (在程序里，我默认数据库是 url_shortener, 可以在配置文件 config.js 里面修改，但不建议)
```

`$ db.counters.insert({_id:'url_count', seq:1})`（这一步是为了创建一个 `counter`，我尝试过在后端自动创建，但是似乎不行。在数据库里面直接创建可以解决这个问题）

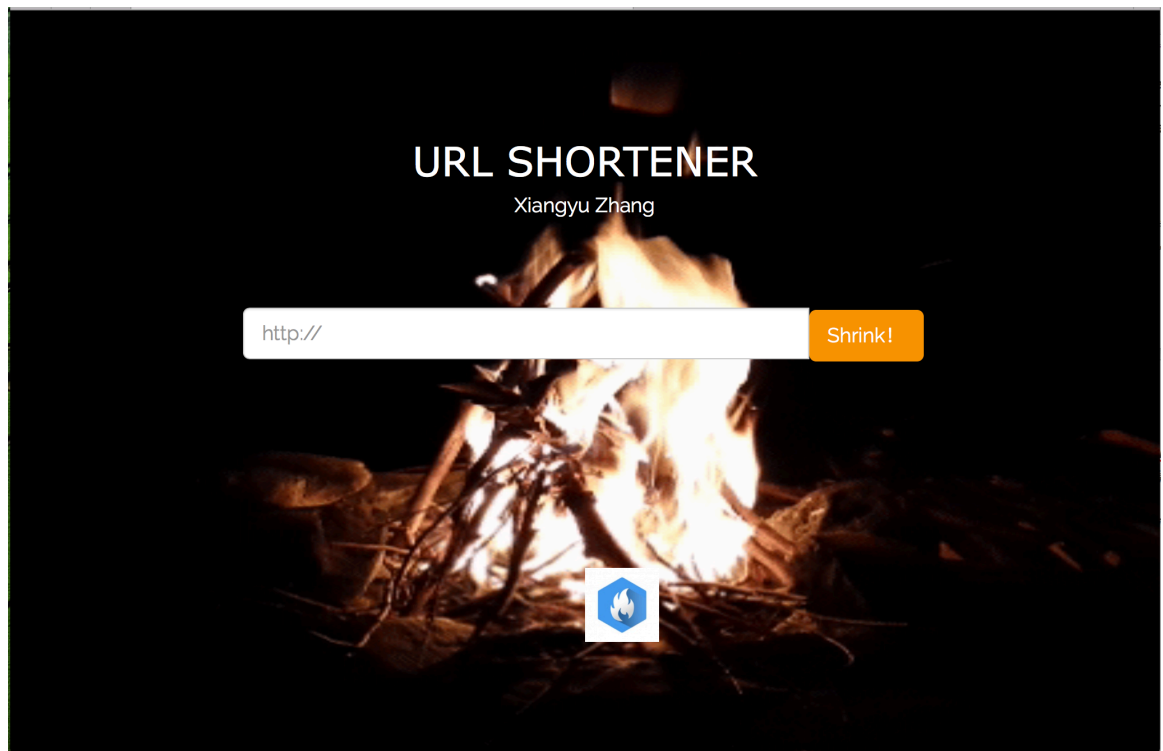
4. 根据需求更改配置文件 `config.js`

5. 运行：

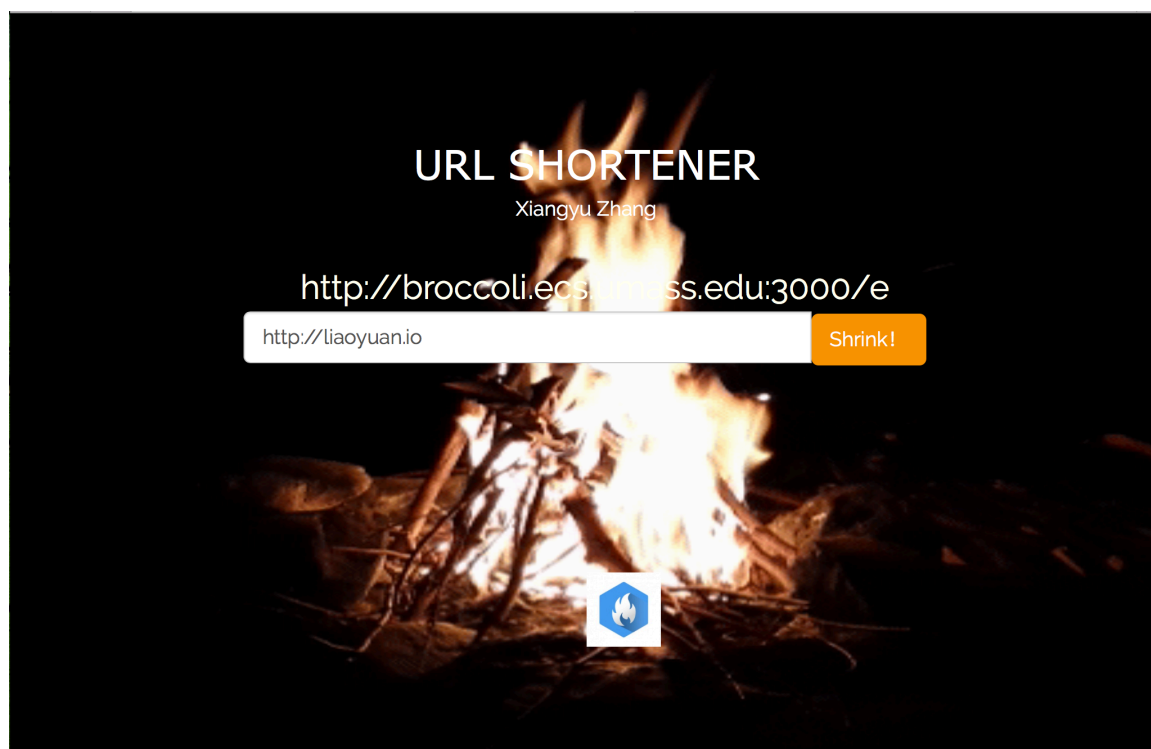
`$ node(或者 nodejs) app.js`

预览 APP.JS

1. 主界面



2. 输入并生成短连接



3. 重定向

