

给文字以岁月

“时光胶囊”一个人生活记录平台

说明文档&实验报告

2019300656 申世东 计算机科学与技术

目录

一、平台说明：	3
二、需求分析：	3
三、逻辑设计：	9
四、物理设计：	10
五、页面设计	12
六、环境配置方法：	16
七、定时触发方法	18
八、后端接口详细信息	19
九、前端路由	19

一、平台说明：

“时光胶囊”是一个日记型社交分享平台。用户可以上传**普通日记**和**胶囊日记**。普通日记上传后随时可见，而胶囊日记则需要待解封日期到来后才可见，并且会收到来自平台的到期提醒邮件。上述两种日记类型均可以选择是否将日记公开到“胶囊广场”。公开的日记和胶囊可以供所有用户查看和发表评论，最终搭建为一个个人情感记录和分享的平台。

本项目使用 Django、Vue、Element-UI 联合进行开发。其中 Django 版本为 3.2.8，Vue 版本为 2.5.17，Element-UI 版本为 2.15.6。

本项目的创新点是使用 Vue 进行前端渲染工作，减轻了后端服务器的压力，符合目前极为主流的前后端分离的技术开发方案，同时使用了 Element-UI 前端框架构造页面，使开发过程效率更高，页面更加美观。

此外，后端 Django 开发使用了 PyCharm 作为编辑器，前端则采用 HBuilder 和 VSCode 进行编辑和调试。后端接口的调试过程也使用了 Postman 进行接口测试。

本项目主要使用了用户表、文章（日记）表、评论表和注册验证码表四张数据表，核心功能有**注册、登录、登出、查看/新建/删除/修改文章（普通日记/胶囊日记）、发表评论、回复评论（父评论机制）、注册邮件验证以及胶囊到期的邮件提醒**。

二、需求分析：

主要需要完成的模块有用户模块、文章模块和评论模块。用户模块需要完成注册、用户邮件确认、登录、登出功能。文章模块需要完成文章发布、修改、到期邮件提醒等功能，评论模块需要完成评论和评论回复（父评论）功能。

此外，还要考虑全部文章的展示和修改功能。

同时列举了开发所需的数据流图和数据字典，可以根据下述分析得到的信息针对性的开发本应用。

以下是各个功能模块的数据流图：

登录和注册：

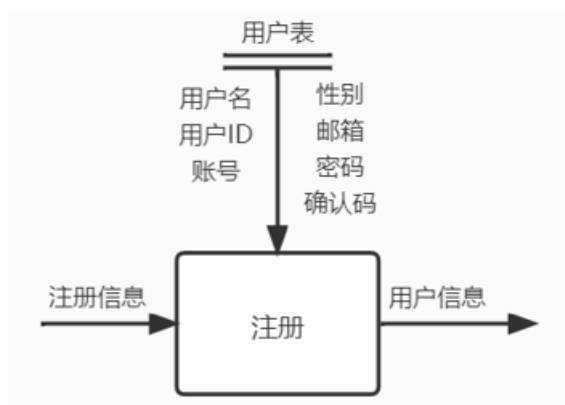


图 2-1 注册过程数据流图

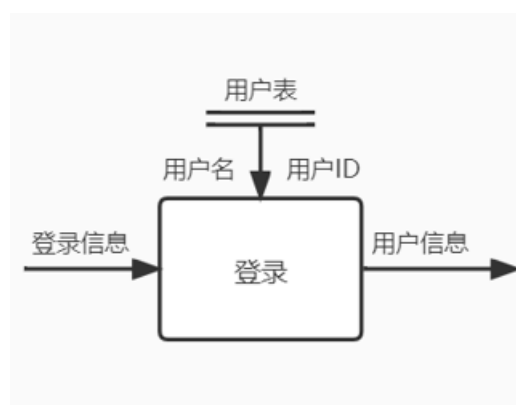


图 2-2 登录过程数据流图

日记和评论的相关操作：

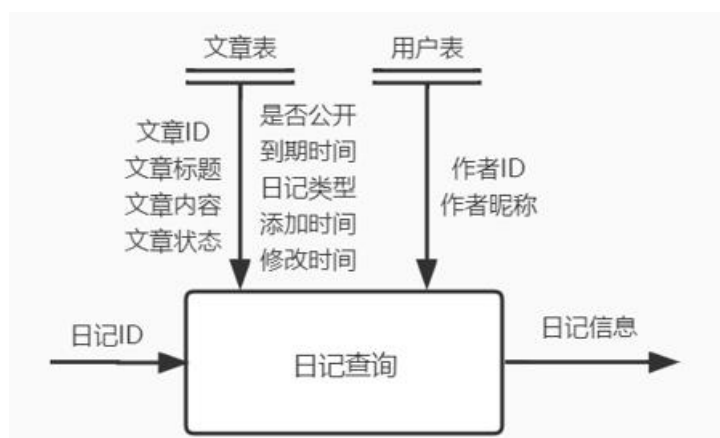


图 2-3 日记查询过程数据流图

此外，日记的增、删、改操作所的数据流图又上述查询过程大体相似，增加和修改时操作为流入日记，并保存相关信息，删除则是流入日记 ID 直接进行删除。在此不进行一一赘述。

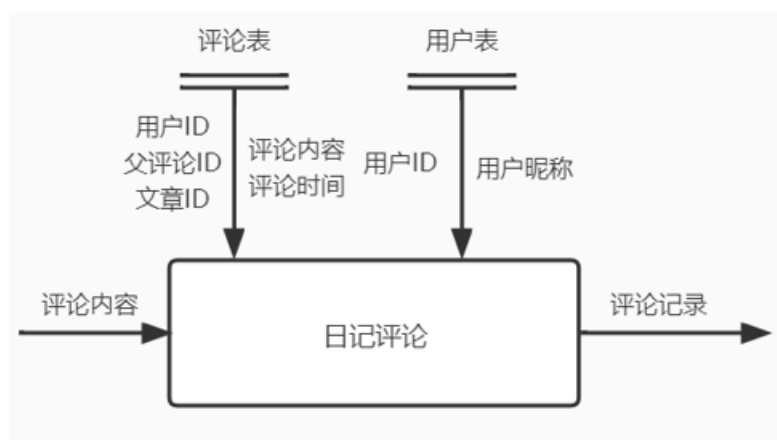


图 2-4 日记评论数据流图

数据字典如下：

用户表：

数据项名	用户 ID
说明	用户的唯一标识码
类型	数值型
长度	11 位 int 型
别名	/
取值范围	11 位二进制

数据项名	用户昵称
说明	用户的个性昵称
类型	字符型
长度	128
别名	/
取值范围	128 位唯一字符型

数据项名	用户密码
说明	用户的密码
类型	数值型
长度	256
别名	/
取值范围	256 位字符型

数据项名	用户性别
说明	用户的性别
类型	数值型
长度	2
别名	/
取值范围	男/女

数据项名	用户注册时间
说明	用户的注册时间
类型	日期型
长度	/
别名	/
取值范围	自动生成

数据项名	是否确认
说明	用户是否验证邮箱
类型	布尔型
长度	8
取值范围	是/否

邮件确认表：

数据项名	确认码
说明	用户邮箱验证的确认码
类型	字符型
长度	256
别名	/
取值范围	256 位

数据项名	用户
说明	邮箱和用户一对一的映射
类型	/
长度	/
别名	/
取值范围	自动生成

数据项名	用户注册时间
说明	用户的注册时间
类型	日期型
长度	/
别名	/
取值范围	自动生成

文章表：

数据项名	文章 ID
说明	文章的唯一标识码
类型	数值型
长度	11 位 int 型
别名	/
取值范围	11 位二进制

数据项名	文章标题
说明	文章标题
类型	字符型
长度	50
别名	/
取值范围	50 位字符型

数据项名	文章内容
说明	文章的内容
类型	字符型
长度	/
取值范围	/

数据项名	广场公开
说明	到期后是否在广场公开
类型	布尔型
长度	8
别名	/
取值范围	是/否

数据项名	是否到期
说明	胶囊日记到期与否
类型	布尔型
长度	8
别名	/
取值范围	是/否

数据项名	日记类型
说明	胶囊日记/普通日记
类型	字符型
长度	/
别名	/
取值范围	胶囊日记(pill)/普通日记(primary)

数据项名	作者 ID
说明	外键
类型	数值型
长度	/
别名	/
取值范围	/

数据项名	作者 ID
说明	外键
类型	数值型
长度	/
别名	/
取值范围	/

数据项名	创建时间
说明	文章的创建时间
类型	日期型
长度	/
别名	/
取值范围	自动生成

数据项名	修改时间
说明	文章的修改时间
类型	日期型
长度	/
别名	/
取值范围	自动生成

评论表:

数据项名	评论 ID
说明	评论的唯一标识码
类型	数值型
长度	11 位 int 型
别名	/
取值范围	11 位二进制

数据项名	评论内容
说明	评论的内容
类型	字符型
长度	/
别名	/
取值范围	/

数据项名	评论用户 ID
说明	外键
类型	数值型
长度	/
别名	/
取值范围	/

数据项名	评论时间
说明	评论的创建时间
类型	日期型
长度	/
别名	/
取值范围	自动生成

数据项名	父评论 ID
说明	指向自己的外键
类型	数值型
长度	/
别名	/
取值范围	/

数据结构的数据字典：

数据结构	用户
含义	用户的全部相关信息
组成成分	用户 ID、用户名、密码、电子邮箱、性别、注册时间、是否确认

数据结构	确认表
含义	用户的全部相关信息
组成成分	注册码、关联用户、注册提交时间

数据结构	文章
含义	文章的全部相关信息
组成成分	文章 ID、文章标题、文章内容、是否公开、到期时间、状态、文章类型、作者 ID、创建日记、最后修改日期

数据结构	评论
含义	评论的全部相关信息
组成成分	评论 ID、文章 ID、文章内容、评论时间、评论者 ID、父评论 ID

三、逻辑设计：

本项目共有四张表，分别是用户表、文章表、评论表和注册验证码表。其中用户表、文章表和评论表均设置有相应的唯一 ID 作为主键。在文章表中，加入用户 ID 为外键约束以调用作者的相关信息，完成身份信息的鉴权。在评论表中引入用户 ID 和文章 ID 作为外键。完成评论和评论人，评论和文章的连接。因此消除了消除非主属性对码的部分函数依赖以及消除非主属性对码的部分函数依赖。本设计至少达到了 3NF。

总 E-R 图和各个分 E-R 图的设计如下图所示：

总 E-R 图：



图 3-1 总 E-R 图

分 E-R 图：

用户：

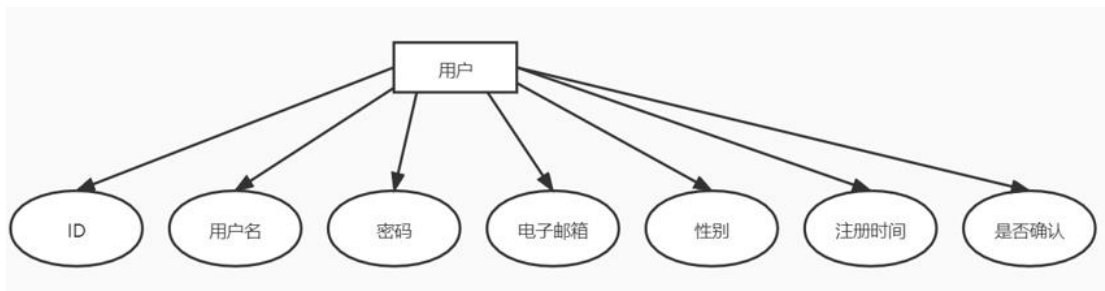


图 3-2 总 E-R 图

文章：

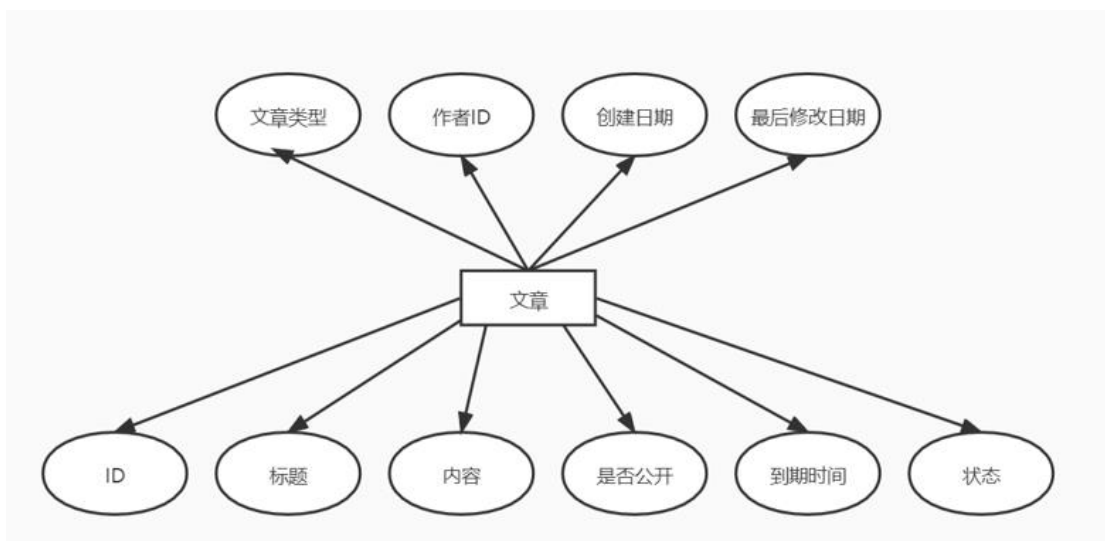


图 3-3 E-R 图

评论：

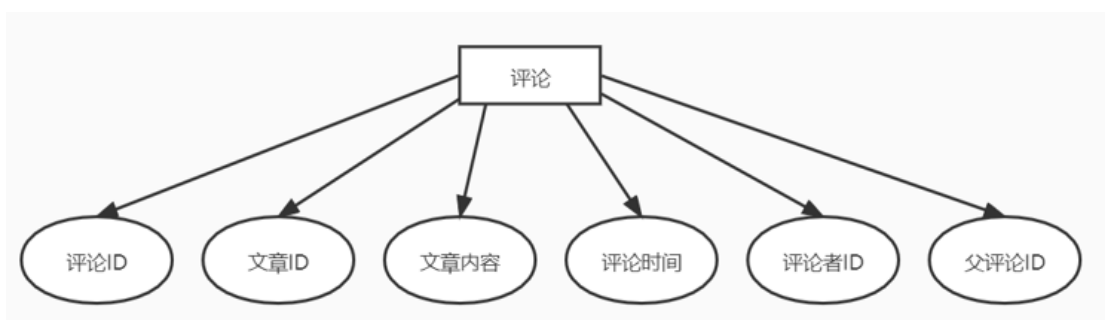


图 3-4 总 E-R 图

四、物理设计：

表的设计在上述设计中已经阐释，在物理设计方面，本项目使用了 Mysql 数据库搭配 Django 的数据处理功能实现了数据的存储。由于用户、文章（日

记)、评论基本按照 ID 进行查询,因此使用 B+树并在**对应 ID 上建立索引的方法进行查询,加快了数据读取的深度。**

存放位置和存放结构则有 Mysql 根据设置的引擎进行相应配置。MySQL 文件包括 MySQL 所建数据库文件和 MySQL 所用引擎创建的数据库文件。

由于 Mysql 的配置可能不同,因此在不同的 Mysql 引擎选择时引擎文件不同。

主要分为如下文件:

.frm 文件与操作系统和数据库引擎无关,都有这么个与表名同名文件。

MyISAM 引擎的文件:

.myd 即 my data, 表数据文件

.myi 即 my index, 索引文件

.log 日志文件

InnoDB 引擎的文件:

采用表空间 (tablespace) 来管理数据, 存储表数据和索引,

InnoDB 数据库文件 (即 InnoDB 文件集, ib-file set):

ibdata1、ibdata2 等: 系统表空间文件, 存储 InnoDB 系统信息和用户数据库表数据和索引, 所有表共用。

.ibd 文件: 单表表空间文件, 每个表使用一个表空间文件 (file per table), 存放用户数据库表数据和索引。

五、页面设计

本项目主要设计了注册页、登录页、写日记（新建日记）页、写胶囊（新建胶囊）页、胶囊广场页、日记内容页等页面。具体设计如下：

注册页：



The screenshot shows the 'User Registration' (用户注册) page of the 'Time Capsule' (时光胶囊) application. The page has a navigation bar at the top with links: 时光胶囊 (Time Capsule), 首页 (Home), 胶囊广场 (Capsule Plaza), 写日记 (Write Diary), 写胶囊 (Write Capsule), 注册 (Register), and 登录 (Login). The registration form includes fields for: 用户名 (Username) with the value 'ssd', 密码 (Password) with masked characters '...', 确认密码 (Confirm Password), 邮箱地址 (Email Address), and 性别 (Gender) with a dropdown menu set to '请选择性别' (Please select gender). There are '注册' (Register) and '重置' (Reset) buttons. At the bottom, it says '时光胶囊 © 2021 – 2021 All Copyright reserved'.

图 5-1 注册页页面图

在注册后邮箱内会收到一封注册确认邮件，验证成功后将完成注册。



图 5-2 邮箱验证图

登录页：

时光胶囊

首页

胶囊广场

写日记

写胶囊

注册

登录

用户登录

* 用户名

ssd

* 密码

...

提交

重置

时光胶囊 © 2021 – 2021 All Copyright reserved

图 5-3 登录页页面图

首页：

时光胶囊

首页

胶囊广场

写日记

写胶囊

Hello,ssd!

登出

写一颗胶囊

写一封日记

时光胶囊 © 2021 – 2021 All Copyright reserved

图 5-4 首页页面图

写胶囊（新建胶囊日记）页：

时光胶囊

首页

胶囊广场

写日记

写胶囊

注册

登录

* 日记标题

* 解封时间

选择日期

-

选择时间

* 是否公开

* 日记内容

立即创建

重置

时光胶囊 © 2021 – 2021 All Copyright reserved

图 5-5 胶囊编辑页页面图

写日记页：

时光胶囊

首页

胶囊广场

写日记

写胶囊

Hello,ssd!

登出

* 日记标题

* 是否公开

* 日记内容

立即创建

重置

时光胶囊 © 2021 – 2021 All Copyright reserved

图 5-6 日记编辑页页面图

胶囊广场页：

该页面用于展示公开的日记、我的日记和我的胶囊的相关信息。



图 5-7 胶囊广场页页面图

日记内容页：



图 5-8 日记详情页页面图

同时还可以对发布的日记进行评论以及回复评论，发表评论后页面将展示评论信息，同时本人发送的评论可以进行删除。



图 5-9 日记详情评论效果图

在设定的胶囊日历到达解封日期后，可以收到相应的提醒邮件，提醒用户查看相关解封胶囊的具体信息。



图 5-10 邮件内容图

六、环境配置方法：

配置 Mysql：

在启动 Django 之前，需要首先配置 Mysql 数据库，需要在源代码的 mysite/setting.py 文件中修改 DATABASE 项关于数据库的相关信息。如下述代码所示：

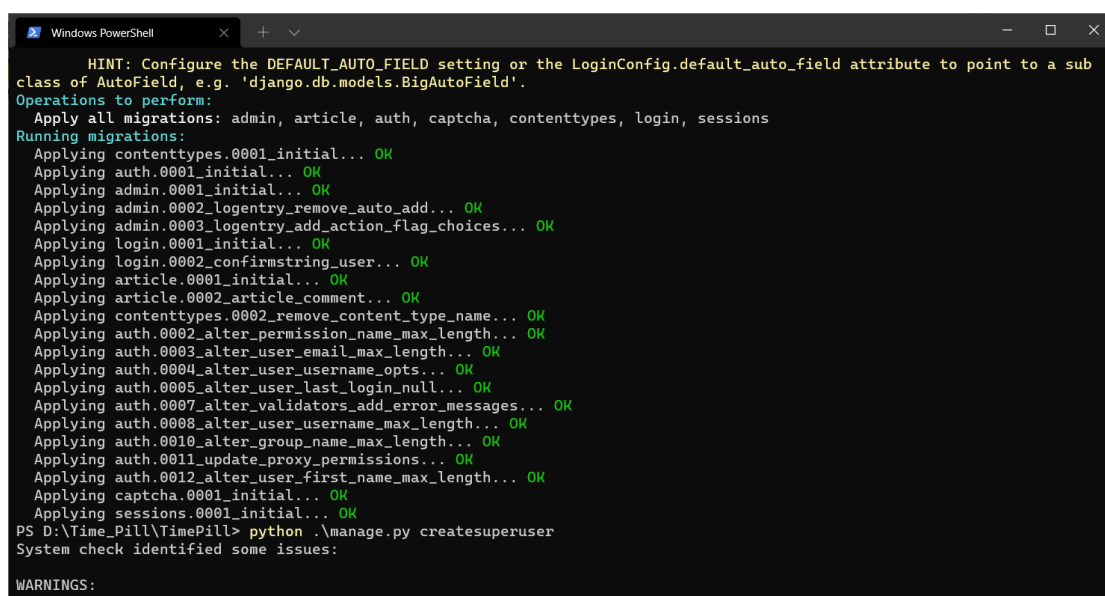
```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': '数据库名',
        'USER': '数据库用户',
        'PASSWORD': '数据库密码',
```



```
'HOST': '数据库地址',  
'PORT': '3306',  
  
}  
}
```

之后可以将作业提交文件中附录的“Timepill.sql”文件导入相应的数据库，即可直接使用本项目。后台超级用户的账号和密码都是 seddon，前端可以自行注册用户。同时也可以不导入上述数据库备份文件，直接在终端中执行“python manage.py makemigrations”、“python manage.py migrate”进行相应表的创建即可。

具体操作截图如下：



```
Windows PowerShell  
HINT: Configure the DEFAULT_AUTO_FIELD setting or the LoginConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.  
Operations to perform:  
  Apply all migrations: admin, article, auth, captcha, contenttypes, login, sessions  
Running migrations:  
  Applying contenttypes.0001_initial... OK  
  Applying auth.0001_initial... OK  
  Applying admin.0001_initial... OK  
  Applying admin.0002_logentry_remove_auto_add... OK  
  Applying admin.0003_logentry_add_action_flag_choices... OK  
  Applying login.0001_initial... OK  
  Applying login.0002_confirmstring_user... OK  
  Applying article.0001_initial... OK  
  Applying article.0002_article_comment... OK  
  Applying contenttypes.0002_remove_content_type_name... OK  
  Applying auth.0002_alter_permission_name_max_length... OK  
  Applying auth.0003_alter_user_email_max_length... OK  
  Applying auth.0004_alter_user_username_opts... OK  
  Applying auth.0005_alter_user_last_login_null... OK  
  Applying auth.0007_alter_validators_add_error_messages... OK  
  Applying auth.0008_alter_user_username_max_length... OK  
  Applying auth.0010_alter_group_name_max_length... OK  
  Applying auth.0011_update_proxy_permissions... OK  
  Applying auth.0012_alter_user_first_name_max_length... OK  
  Applying captcha.0001_initial... OK  
  Applying sessions.0001_initial... OK  
PS D:\Time_Pill\TimePill> python .\manage.py createsuperuser  
System check identified some issues:  
  
WARNINGS:
```

图 6-1 数据表创建操作图

启动 Django：

Python 需要安装 Django 库，具体方法为使用“pip install django”命令进行安装。安装完成后在代码目录使用终端执行“python manage.py runserver 127.0.0.1:8000”命令即可在本地的 8000 端口由 Django 提供的一个 Web 服务。

```
December 16, 2021 - 17:05:19  
Django version 3.2.8, using settings 'mysite.settings'  
Starting development server at http://127.0.0.1:8000/  
Quit the server with CTRL-BREAK.
```

图 6-2 Django 启动操作图

配置 Vue 和 Element-UI:

使用 Vue 和 Element-UI 需要配置 Node.js 环境, 相应方法请自行查阅相关资料。

代码包中已经包含了 Vue 和 Element-UI 的相关工程文件, 因此在所缺依赖安装完成后, 仅需在“element-starter-master”文件夹内执行“npm run dev”命令即可在本地 8010 端口启动前端。

```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS D:\Time Pill\TimePill\element-starter-master> npm run dev

> element-starter@ dev D:\Time Pill\TimePill\element-starter-master
> webpack-dev-server --inline --hot --env.dev

Project is running at http://127.0.0.1:8010/
webpack output is served from /assets/
404s will fallback to /assets/
webpack: wait until bundle finished: /assets/
webpack: wait until bundle finished: /assets/
Hash: cffad6cf041649509164
Version: webpack 2.4.1
Time: 16040ms

   Asset      Size  Chunks             Chunk Names
732389ded34cb9c52dd88271f1345af9.ttf    56 kB          0  [emitted]
535877f50039c0cb49a6196a5b7517cd.woff  28.2 kB          0  [emitted]
f091c8a059e3d53f917b4da1fed51d68.jpg   17.2 kB          0  [emitted]
  vendor.js    6.31 MB          0  [emitted] [big] vendor
   index.js    3.1 MB          1  [emitted] [big] index
  manifest.js   30.4 kB          2  [emitted]
  index.html    341 bytes          0  [emitted]
```

图 6-3 Vue 启动操作图

此外,还可以直接打开/element-starter-master/dist 目录下的 index.html 文件, 该文件夹是已经通过“npm run build”打包好的 Vue 工程文件, 无需 Node.js 环境即可直接打开运行, 如需修改前端源码还需通过安装环境对相关源代码进行修改。在项目发布时则可以采用“npm run build”进行编译生成。

七、定时触发方法

由于该平台目前在本人的一台 Windows 10 系统的笔记本电脑上搭建, 在 Windows 操作系统下并没有很好的定时任务触发工具, 因此本功能目前需要进行相应的定时触发设置, 具体方法为定时访问“/pron”路径, 访问该路径后 Django 将查询目前需要解封的时光胶囊并进行相应的解封操作, 同时发送邮件到相应的用户邮箱。而在 Linux 操作系统下则可以使用 Django 自带的定时任务工具定时执行相应的发送操作。

八、后端接口详细信息

由于采用前后端分离的方法进行开发，因此后端接口较多，另附“接口文档.pdf”文件介绍本项目开发的 Django 后端的各操作的接口，供参考。大致内容如下图所示：

/登录				
用于登录				
接口状态				
已完成开发				
接口URL				
http://127.0.0.1:8000/login/				
请求方式				
POST				
Content-Type				
urlencoded				
请求Body参数				
参数名	示例值	参数类型	是否必填	参数描述
username	ssd	Text	是	用户名
password	ssd	Text	是	密码

图 8-1 接口文档效果图

九、前端路由

由于采用 Vue+Django 进行前后端分离开发，因此前端页面需要通过路由进行页面跳转。该部分的代码在前端代码的 main.js 内定义。具体路由操作如下图所示：

```
28 let router = new VueRouter({
29   routes: [
30     //一个个对象
31     { path: '/', component: Index ,meta:{needLogin:false}},
32     { path: '/home', component: Home ,meta:{needLogin:true}},
33     { path: '/login', component: Login ,meta:{needLogin:false}},
34     { path: '/reg', component: Register ,meta:{needLogin:false}},
35     { path: '/edit_pre', component: Editor_pre ,meta:{needLogin:true}},
36     { path: '/edit_pre/:id', component: Editor_pre ,meta:{needLogin:true}},
37     { path: '/edit_pill', component: Editor_pill ,meta:{needLogin:true}},
38     { path: '/edit_pill/:id', component: Editor_pill ,meta:{needLogin:true}},
39     { path: '/article/:id', component: Article ,meta:{needLogin:true}},
40     { path: '/pill/:id', component: Pill ,meta:{needLogin:true}}
41   ]
42 });
43
```

图 9-1 前端路由代码图

主要有如下几个页面：

页面地址	功能
/	时光胶囊首页
/home	胶囊广场页（公开和个人日记展示）
/login	登录页
/reg	注册页
/edit_pre	写普通日记页
/edit_pre/:id	编辑普通日记页
/edit_pill	写胶囊日记页
/edit_pill/:id	编辑胶囊日记页
/article/:id	查看普通日记具体信息
/pill/:id	查看胶囊日记具体信息

表 9-1 Vue 路由配置信息