|  |
| --- |
| 北京交通大学软件学院 |
| JAVAEE-WEBGYM |
| Design Report |

|  |
| --- |
| 刘青羽 16301097 陈红 16301088  2019-5-20 |

[项目概述 2](#_Toc9244415)

[项目目录结构图 2](#_Toc9244416)

[开发环境 2](#_Toc9244417)

[基本功能 3](#_Toc9244418)

[Restful services 3](#_Toc9244419)

[实现思路（流程） 3](#_Toc9244420)

[主要技术 3](#_Toc9244421)

[程序运行截图 3](#_Toc9244422)

[api versioning. 4](#_Toc9244423)

[实现思路（流程） 4](#_Toc9244424)

[主要技术 4](#_Toc9244425)

[程序运行截图 4](#_Toc9244426)

[附加功能 5](#_Toc9244427)

[Rate Limiting 5](#_Toc9244428)

[实现思路（流程） 5](#_Toc9244429)

[主要技术 5](#_Toc9244430)

[程序运行截图 5](#_Toc9244431)

[Online API document（Swagger Document） 6](#_Toc9244432)

[实现思路（流程） 6](#_Toc9244433)

[主要技术 6](#_Toc9244434)

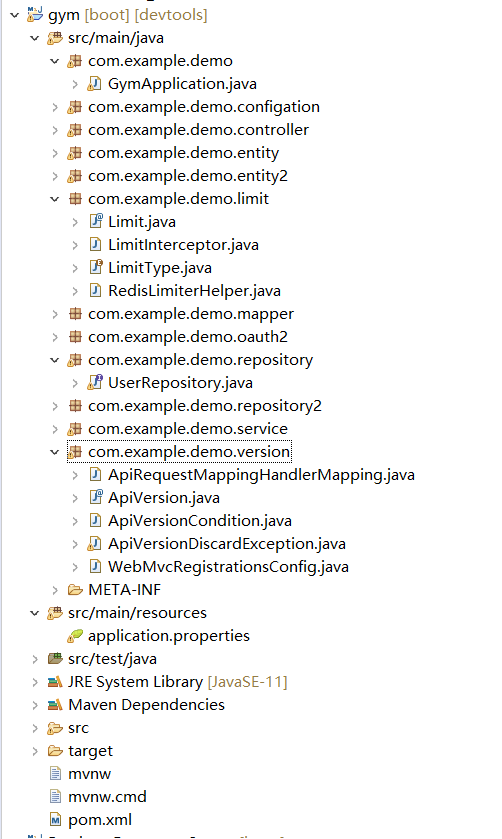
[程序运行截图 7](#_Toc9244435)

# 项目概述

此次项目基于第一次作业实现的基础之上，把传统的MVC框架变成了RESTful web service，以及增加了API接口的版本控制，实现了第二次作业的基本功能。

附加功能基于Redis实现了API接口访问速率的控制、基于Swagger document实现了在线的API文档。

# 项目目录结构图



# 开发环境

|  |  |
| --- | --- |
| **Tools** | **Version** |
| **Eclipse IDE** | **4.11.0** |
| **MySql** | **8.0.15** |
| **Redis** | **3.2.100** |
| **Github** |  |

# 基本功能

## Restful services

### 实现思路（流程）

RESTful web service和传统的MVC关键区别在于建HTTP响应体的方式，RESTful web服务控制器不依赖于视图技术来执行服务端对象数据到HTML端呈现，而是生成并返回Greeting对象，对象数据以JSON形式写入HTTP response响应

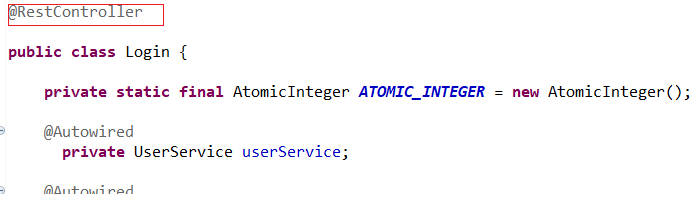
为它增加@RestController注解，表示这是一个处理RESTful请求的响处理类。返回的对象使用ModelAndView来假装实现返回页面的实现。

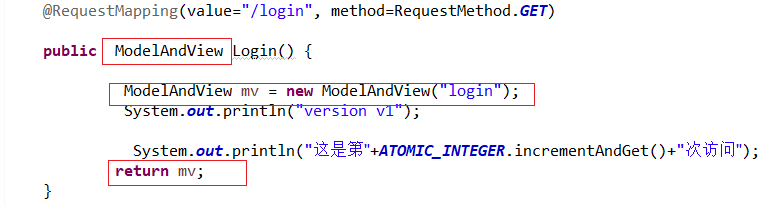
### 主要技术

使用 @RestController 代替@Controller

使用ModelAndView 来返回页面

### 程序运行截图





## api versioning.

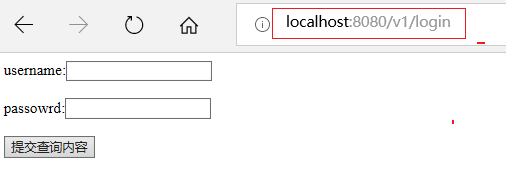
### 实现思路（流程）

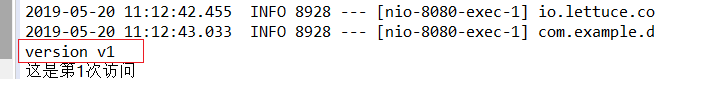
一个系统上线后会不断迭代更新，需求也会不断变化，有可能接口的参数也会发生变化，如果在原有的参数上直接修改，可能会影响线上系统的正常运行，这时我们就需要设置不同的版本，这样即使参数发生变化，由于老版本没有变化，因此不会影响上线系统的运行。

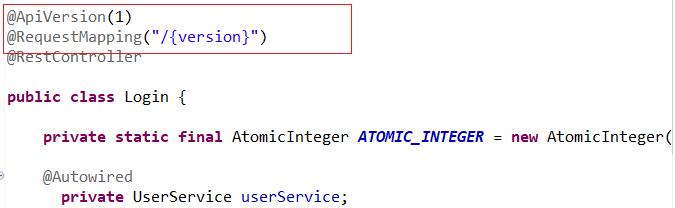
### 主要技术

Restful api 版本控制

### 程序运行截图







# 附加功能

## Rate Limiting

### 实现思路（流程）

假设一个用户（用IP判断）每分钟访问某一个服务接口的次数不能超过10次，那么我们可以在Redis中创建一个键，并此时我们就设置键的过期时间为60秒，每一个用户对此服务接口的访问就把键值加1，在60秒内当键值增加到10的时候，就禁止访问服务接口。

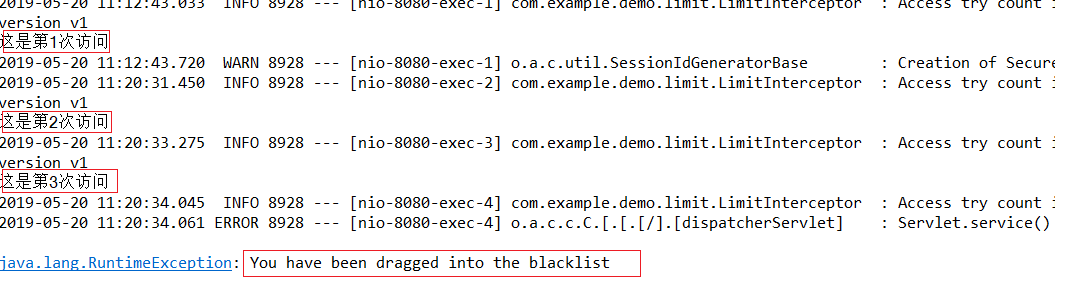
### 主要技术

Redis

RedisTemplate

### 程序运行截图





## Online API document（Swagger Document）

### 实现思路（流程）

随着互联网技术的发展，现在的网站架构基本都由原来的后端渲染，变成了：前端渲染、先后端分离的形态，而且前端技术和后端技术在各自的道路上越走越远。 前端和后端的唯一联系，变成了API接口；API文档变成了前后端开发人员联系的纽带，变得越来越重要，swagger就是一款让你更好的书写API文档的框架，而且swagger可以完全模拟http请求，入参出参和实际情况差别几乎为零。

### 主要技术

Swagger

SpringBoot

Restful

### 程序运行截图



