

Law_Help-Server

指南：

基本信息

后端代码地址：https://github.com/lugf027org/Law_Help-Server

前端代码地址：https://github.com/lugf027org/Law_Help-MP

需要的软件：

- mysql 5.5 数据库（见redis&maven&db文件夹）
- PDMan 协助构建数据库，可以不要（见redis&maven&db文件夹）
- jdk 1.8 java环境
- apache-maven-3.6.0 java包管理（见redis&maven&db文件夹）
- Redis-x64-3.2.100 数据库缓存、登录校验码的存储（见redis&maven&db文件夹）
- idea专业版 后端IDE
- 微信开发者工具 前端IDE
- postMan api调试

部分软件安装说明：

执行sql脚本文件：略。

jdk安装：略。

maven安装：

将redis&maven&db文件夹中提供的maven直接解压复制到"D:\java\maven"文件夹下，配置环境变量

- 新建->变量名"MAVEN_HOME",变量值"D:\java\maven"（maven的路径）
- 编辑->变量名"Path",在原变量值后面加上"; %MAVEN_HOME%\bin"
- 测试dos界面，执行命令：mvn -version

redis安装：

先是同上。解压、添加到环境变量。

- 测试dos界面，执行命令：redis-server

idea安装：略。

idea: 菜单File-Settings。然后搜索maven，可以修改用内置的，还是刚刚自己安装的。

微信开发者工具安装：略。

postMan安装：略

项目配置

后端依赖包pom.xml，用处见注释(基本不用改):

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>

  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>

  <!-- database-->
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
  </dependency>

  <!--MyBatis-Plus 连接数据库-->
  <dependency>
    <groupId>com.baomidou</groupId>
    <artifactId>mybatisplus-spring-boot-starter</artifactId>
    <version>1.0.5</version>
  </dependency>

  <dependency>
    <groupId>com.baomidou</groupId>
    <artifactId>mybatis-plus</artifactId>
    <version>2.3</version>
  </dependency>

  <dependency>
    <groupId>org.mybatis.spring.boot</groupId>
    <artifactId>mybatis-spring-boot-starter</artifactId>
    <version>1.3.1</version>
  </dependency>

  <!-- redis -->
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-redis</artifactId>
    <exclusions>
      <exclusion>
        <groupId>redis.clients</groupId>
        <artifactId>jedis</artifactId>
      </exclusion>
      <exclusion>
        <groupId>io.lettuce</groupId>
        <artifactId>lettuce-core</artifactId>
      </exclusion>
    </exclusions>
  </dependency>
```

```

<!-- redis (client with jedis) -->
<dependency>
    <groupId>redis.clients</groupId>
    <artifactId>jedis</artifactId>
</dependency>

<!-- logs -->
<dependency>
    <groupId>com.googlecode.log4jdb</groupId>
    <artifactId>log4jdb</artifactId>
    <version>1.2</version>
</dependency>

<!-- 热部署 -->
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
    <scope>runtime</scope>
</dependency>

<!-- json api-->
<dependency>
    <groupId>com.alibaba</groupId>
    <artifactId>fastjson</artifactId>
    <version>1.1.36</version>
</dependency>

<!-- http client 用于微信api-->
<dependency>
    <groupId>org.apache.httpcomponents</groupId>
    <artifactId>httpClient</artifactId>
    <version>4.3.5</version>
</dependency>

<!-- security -->
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
</dependency>

</dependencies>

```

后端Yml/Properties配置（端口号、数据库地址与账户密码、媒体文件地址根据实际情况改）

```

server:
  port: 8089 # 以这个为准
servlet:
  context-path: /lawhelp
tomcat:
  uri-encoding: UTF-8

Spring:
  datasource:
    # y-cloud-tech 补上数据库名称

```

```
url: jdbc:log4jdbc:mysql://127.0.0.1:3306/law_help?
useUnicode=true&characterEncoding=utf8&serverTimezone=UTC
username: root # 数据库 用户
password: root@000 # 密码
driver-class-name: net.sf.log4jdbc.DriverSpy
hikari:
    max-lifetime: 60000
    maximum-pool-size: 10
    idle-timeout: 60000
    connection-timeout: 60000
    validation-timeout: 3000
    LoginTimeout: 60000

xcloud:
    uploadPath: F:/attach_file/law_help/ # 媒体文件地址, 见 (redis&maven&db文件夹)
    staticPath: resources

# ...
# 看到这里就好的, 下面的不用管
```

此外后端logback.xml文件第十五行, 设置日志文件路径, 要根据本地适配。

此外controller/userController第三十行左右, 配置好appId和appSecret。

前端util/ajax.js 第一行设置后端的路径(含端口号), 需要与后端统一。

前端的小程序appId确认好。

关键代码说明

先读redis&maven&db文件夹, 数据库db-help.doc, 了解需求与数据表各字段含义。

后端:

controller层、service层、mapper层, 依次看。熟悉一两个接口是咋回事儿, 然后自己试着改改。

要注意的有:

util/MyConstants.java 定义了数据库相关参数, 有需要的话直接用, 减少手写预定义的字段含义。

auth会检测header过滤很多没有登陆码(sessionKey, 可以理解为cookie)的url请求(不安全)。如果需要开放一些请求, 到tech.linjuliwhu.config.WebSecurityConfig.java 的50行左右, 照着写, 设置成permitAll就能直接访问(用postMan测试)了。

注意, 出于安全, 目前把url访问分为了visitor、user、admin三大类。visitor游客下的header可以不携带登录校验码, 允许所有人访问。user是登陆后的用户的。admin要求是管理员身份。其中"/user/login", "/user/loginWithCode"两个接口也不需要登录码。

注意，微信小程序api四月末有登陆相关的改版，要求开发者不必要不要获取用户的基本信息。因此user表只记录了授权基本信息的用户。在visitor发起提问时，我们会照常用code记录其openId——但是可能在 user表找不到他。

前端：

用了colorUI，要继续用的话自行搜搜了解，弃用的话不用管现在的导入啥的。

因为管理员和用户都用这个小程序，因此tarBar没用微信小程序原生的，而是用组件的方式动态显示。

项目运行

首先确保数据库端口号、数据库名称、数据库登陆的账号密码都已配置正确!

在idea中打开改项目，首次打开maven要下载相关包，要耐心等待；有报错自行百度就好。

然后在启动项目前，在dos里执行redis-server启动redis。

点击后端的启动项目。

确保小程序的ajax.js后端地址正确，就可以运行了。

祝一切顺利！