|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 路径 | 参数(eg) | 请求方法 | 响应 | 说明 |
| 用户登录 | /api/auth/login | {  "user\_name":"丁二",  "user\_pwd":"123456"  } | post | {  "code": 0,  "user\_auth": "试题录入员",  "user\_name": "丁二"  }  出错时  {  "code": -1  } | 返回json文件，并且用户名和用户权限加入到session  session['user\_name'] session['user\_auth'] |
| 用户登出 | /api/auth/logout |  |  | 1 | Clear session |
| 增加用户 | /api/auth/add\_user | {  "user\_name":"123",  "user\_pwd":"123456",  "user\_auth": "试题录入员"  }  user\_auth有三种，还有  “试题管理员”和”超级用户  ” | post | 正确返回  {  "code": 0  }  错误返回  {  "code": -1  } |  |
| 修改密码 | /api/auth/update\_pwd | {  "user\_name":"123",  "old\_pwd":"123456",  "new\_pwd": "654321",  "repeat\_pwd": "654321"  } | post | 正确返回  {  "code": 0  }  错误返回  {  "code": -1  }  其中两次密码不一致返回  {  "code": -2  } | 按理说 用户名是可以从session读取，不用传参…  这里测试没测…  检查两次密码一致不一致可以放在前端，去掉-2的响应 |
| 获取用户表所有记录 | /api/auth/getUsersMsg |  | get | [  {  "changeButton": "编辑",  "infoChangeSeen": false,  "lastUserName": "none",  "password": "654321",  "permission": "试题录入员",  "userName": "123"  },…  ] | username:用户名  password:密码  permission：权限，试题录入员，试题管理员，超级用户  注意，返回的是列表 |
| 更新用户表 | /api/auth/uploadUserMsg | {  "userName" : "xxx",  "password" : "xxx",  "permission" : "试题录入员",  "opt" : "add",  "lastUserName": "none"  } | post | 正确返回  {  "code": 0  }  错误返回  {  "code": -1  } | Opt del删除 update更新  add 添加  lastUserName在更新信息中表示更新前的用户名，方便数据库查找 |
| 试题库获取每页题目 | /api/get/Question | {  "questionOffset": 0,  "quetisonNum": 100,  "conditionForm": {  "keyword": "",  "paper\_grade": "12",  "paper\_subject": "4",  "knowledge\_point": ["32", "33"]  }  } | post | {  questions:[  {  question\_no: “0”,  question\_content: …,  paper\_name: ….,  figure\_url:…  },….  ]  }具体可见图  没有题目  {  "questions": []  } | 除了关键字外其余的都必须要填写 |
| 查看题目详情 | /api/get/QuestionDetail | {  "question\_no" = "2"  } | get | 获取成功  {  code: 0,  question\_diff:….,  question\_type: "0",  question\_content: ….,  question\_answer:…,  question\_analy:….,  figure\_url:.....,  paper\_name: ….,  knowledge\_point:[ "1", "3"],  "little\_question": []  }可见图  获取失败{  code: -1,  } | 返回题目详情  其中type参考相关字典  knowledge\_point 顺序为父、子、子的子..以及最终所属的编号。 |
| 获取所有知识点 | /api/get/KnowledgePoints |  | get | {  "know":[  {  "know\_name": "椭圆",  "know\_no": “13”,  "super\_name": "解析几何",  "super\_no": “12”  },  …  }  具体见图  一级学科“super\_no”: “0” |  |
| 上传试卷试题 | /uploadPaper | paperInfoForm: {  paper\_name: ….,  paper\_source:[0,1],  paper\_subject:….,  paper\_year: ….,  paper\_grade:….,  },  questionsForm: {  questions: {  question\_type: 0,  question\_point: [“0”,”1”],  question\_content: ….,  question\_answer: ….,  question\_analy: ....,  question\_diff:1,  figure\_url:[…],  }  } | post | {  code: 0  }  表示成功  或者  {  code: -1  }  表示失败 | 上传试卷及试题  其中paper\_source为城市-区-学校的下标，可能存在只有城市或城市和区的情况（即数组长度0<length<4）如[0,1]则代表上海市所属区列表中下标为1的区。（知识点跟他类似）  type参考相关字典  知识点取决于传给前端的顺序。（需讨论） |
| 收藏某题  （所有函数中优先级低） | /storeUp | ?quetsion\_no=1&user\_no=2 | get | {  code:0,  }  表示成功  {  code:-1,  }  表示失败 | 收藏试题的功能，可以放在最后实现 |
| 上传图片 | /uploadFig | {  figure: 文件  } | post | {  code:0,  figure\_url:…  },  表示成功  {  code:-1,  }  表示失败 | 上传图片，返回图片新的url |
| 上传md内，返回word文件 | /paperPreview/md2word | {  paperName:…,  questionContent:…,  } | post | 返回一个word文件 | 上传md文件内容，返回word文件  将questionContent存入一个md文件，再转换成word，其过程参考python\_script里的脚本 |

相关字典

subject:

0:不限

1:语文

2:数学

3.英语

4.物理

5.化学

6.生物

7.地理

8.历史

9.政治

type:

0:不限

1:填空题

2.单选题

3.多选题

4.应用题（简答题）

5.综合题

souce

0: 上海市 child{

0: 区按拼音排 child {

0. 某某学校(按拼音)

}

1: 区按拼音排

}

/api/upload/figure /api/upload/questions这样

不过那个就是 0初中：{

0：预备，1：初一。。。

}

1 高中 {

0 高一

}