React综合项目接口文档

1. 项目业务

项目名称: 赚赚金融渠道管理系统

项目业务流程:

1. 用户先登陆( 填写账号,密码,验证码,发起登陆请求 ).
2. 用户可以分别查看所有贷款订单,转单订单,保险订单;
3. 用户可以设置个人头像;
4. 用户可以点击”退出”登陆;
5. 用户关闭页面;

开发工具: 任选( WebStorm,vs code,sublime )

框架&库: react , react-dom , react-router-dom , redux , react-redux , antd等

1. 项目接口

项目所有接口分为 **用户相关接口** , **订单管理相关接口** 两大部分.

所有接口:

code值为0 , 表示请求成功.

code值为1 , 表示请求失败.

**注意**: 所有api请求, 必需在请求发起前在请求头中添加**authorization**字段;

**authorization 字段的值为登陆成功后得到的sessionId值;**

/api/login 和 /api/checkCode 两个接口是在登陆前发起请求的,,请求头设置**authorization 为 “ ”**

**服务端程序需要设置允许跨域:**

app.js文件中,找到如下位置:

app.use((req,res,next)=>{  
 **if**(/^\/api/.test(req.path)){  
 **//在这里添加( 设置响应头 )** res.setHeader(**"Access-Control-Allow-Origin"**, **"\*"**)**;** next()**;** }  
 res.sendFile(***path***.join(\_\_dirname, **'public/index.html'**))**;**})**;**

### 用户相关

涉及接口有: 用户登陆 , 获取验证码;

#### 用户登陆

接口地址：http://localhost:3000/api/login

支持格式：JSON

请求方式：POST

请求示例：axios.post(‘http://localhost:3000/api/login’,{参数}).then().catch();

**请求参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| password | string | 登陆密码 |
| phone | string | 登陆账号 |
| checkcode | string | 验证码 |

**响应参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| code | number | 状态码 |
| message | string | 状态信息 |
| sessionId | string | 身份令牌( 密钥 ) |

例如: {"code":1,"message":"登录成功","sessionId":"eyJhbGciOiJIUzI1NiIsInR5cC”}

code值为0 , 表示登陆成功.

code值为1 , 表示登陆失败.

**注意**: sessionId是用户是否登陆的依据.. 在第一次登陆成功后,需要将token保存在cookie中.

#### 获取验证码

接口地址：http://localhost:3000/api/checkCode

支持格式：JSON

请求方式：GET

请求示例：axios.get(‘http://localhost:3000/api/checkCode’,{参数}).then().catch();

**请求参数:**

无

**响应参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| code | number | 状态码 |
| verification | string | 验证码字符串 |

例如: {"code":1,"verification":"666d"}

#### 判断用户是否登陆( 拿到用户信息 )

接口地址：http://localhost:3000/api/islogin

支持格式：JSON

请求方式：GET

请求示例：axios.get(‘http://localhost:3000/api/islogin’,{参数}).then().catch();

**请求参数:**

无需传递参数.

**响应参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| code | number | 状态码 |
| info | object | 用户信息 |

info参数是个对象 , 该对象包含如下属性:

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| facePhoto | string | 头像地址 |
| phone | string | 手机号 |
| password | string | 密码 |
| id | string | 用户编号 |
| exp | string | xxx |
| iat | string | 其他联系方式 |

#### 修改头像

接口地址：http://localhost:3000/api/facePhoto

支持格式：JSON

请求方式：POST

请求示例：axios.post(‘http://localhost:3000/api/facePhoto’,{参数}).then().catch();

**请求参数:**

无需传递参数.

**响应参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| file | 文件 | 文件数据 |

### 订单管理相关

涉及业务有: 查询贷款订单 , 查询转单订单, 查询保险订单 ;

#### 查询订单列表

接口地址：http://localhost:3000/api/list

支持格式：JSON

请求方式：GET

请求示例：axios.get(‘http://localhost:3000/api/list?order=1’).then().catch();

**请求参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| order | number | 订单类型 |

订单类型: 1贷款订单 , 2转单订单 , 3 保险订单

**响应参数:**

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| code | number | 状态码 |
| data | array | 订单列表 |

例如: { "code": 1 ,"data": [] }

data参数是个数组, 包含多个订单对象 , 每个订单对象属性如下:

|  |  |  |
| --- | --- | --- |
| 参数名称 | 类型 | 说明 |
| id | string | 订单编号 |
| customerName | string | 客户名称 |
| phone | string | 手机号 |
| money | number | 订单金额(万元) |
| date | string | 订单处理时间 |
| type | string | 产品类型 |
| interestRate | number | 贷款利率 |
| handleState | number | 订单处理状态(0,1,2,3,4) |
| serviceName | number | 客服 |
| order | string | xxx |

**订单状态**: 0新订单 , 1未审核 , 2已接单 , 3已完成 , 4暂无状态

**产品类型**: 信用贷 , 押房贷 , 房乐贷 , 车乐贷

**客服**: 李大维 , 李小冉 , 李莉 , 张玲 , 李家豪

1. 英文解释

login 登陆

password 密码

status 状态

type 类型

facePhoto 头像

nickname 昵称