网络交互接口设计

1. 系统管理接口

1.1 用户认证与权限管理接口 (`/api/auth`)

1.1.1 用户注册

- **URL**: `/api/auth/demo/register`
- **方法**: POST
- **请求体**:

```json
{
  "username": "string",
  "email": "string",
  "password": "string"
}
```

- **响应**:

```json
{
  "code": 200,
  "message": "注册成功",
  "data": "注册成功"
}
```

1.1.2 用户登录

- **URL**: `/api/auth/demo/login`
- **方法**: POST
- **请求体**:

```json
{
  "email": "string",
  "password": "string"
}
```

- **响应**:

```json
{
  "code": 200,
  "message": "success",
  "data": {
    "token": "string",
    "expiration": "long"
  }
}
```

1.1.3 获取路由信息

- **URL**: `/api/auth/demo/getRouters`

- **方法**: GET

- **响应**:

```json
{
  "code": 200,
  "message": "success",
  "data": [
    {
      "id": "long",
      "path": "string",
      "name": "string",
      "component": "string",
      "redirect": "string",
      "meta": {
        "title": "string",
        "icon": "string",
        "hidden": "boolean"
      },
      "children": []
    }
  ]
}
```

1.1.4 获取用户权限信息

- **URL**: `/api/auth/demo/getUserPrivateInformation`

- **方法**: GET

- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": ["string", "string"]
}
```

1.1.5 获取用户角色信息

- **URL**: `/api/auth/demo/getUserRole`
- **方法**: GET
- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": {
    "id": "long",
    "roleName": "string",
    "rolePermissions": ["string"]
  }
}
```

1.1.6 获取用户基本信息

- **URL**: `/api/auth/demo/getUserBaseInfo`
- **方法**: GET
- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": {
    "id": "long",
    "username": "string",
    "email": "string",
    "roleId": "long"
  }
}
```

1.1.7 编辑路由

- **URL**: `/api/auth/demo/editRouter`
- **方法**: POST
- **请求体**:

```json
{
  "id": "long",
  "path": "string",
  "name": "string",
  "component": "string",
  "redirect": "string",
  "meta": {
    "title": "string",
    "icon": "string",
    "hidden": "boolean"
  }
}
```

- **响应**:

```json
{
  "code": 200,
  "message": "编辑成功",
  "data": null
}
```

1.1.8 添加路由

- **URL**: `/api/auth/demo/addRouter`
- **方法**: POST
- **请求体**:

```
{
  "path": "string",
  "name": "string",
  "component": "string",
  "redirect": "string",
  "meta": {
    "title": "string",
    "icon": "string",
    "hidden": "boolean"
  }
}
```

- **响应**:

```
{
  "code": 200,
  "message": "添加成功",
  "data": null
}
```

1.1.9 删除路由

- **URL**: `/api/auth/demo/deleteRouter`
- **方法**: POST
- **请求体**:

```
{
  "id": "long"
}
```

- **响应**:

```
{
  "code": 200,
  "message": "删除成功",
  "data": null
}
```

1.1.10 获取角色列表

- **URL**: `/api/auth/demo/getRoleList`

- **方法**: GET
- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": [
    {
      "id": "long",
      "roleName": "string",
      "rolePermissions": ["string"]
    }
  ]
}
```

1.1.11 添加角色

- **URL**: `/api/auth/demo/addRole`
- **方法**: POST
- **请求体**:

```
{
  "roleName": "string",
  "rolePermissions": ["string"]
}
```

- **响应**:

```
{
  "code": 200,
  "message": "添加成功",
  "data": null
}
```

1.1.12 编辑角色

- **URL**: `/api/auth/demo/editRole`
- **方法**: POST
- **请求体**:

```json
{
  "id": "long",
  "roleName": "string",
  "rolePermissions": ["string"]
}
```

- **响应**:

```json
{
  "code": 200,
  "message": "编辑成功",
  "data": null
}
```

1.1.13 更新角色权限

- **URL**: `/api/auth/demo/updateRoleRouters`
- **方法**: POST
- **请求体**:

```json
{
  "roleId": "long",
  "permissions": ["string"]
}
```

- **响应**:

```json
{
  "code": 200,
  "message": "更新成功",
  "data": null
}
```

1.1.14 删除角色

- **URL**: `/api/auth/demo/deleteRole`
- **方法**: POST
- **请求体**:

```
{
  "id": "long"
}
```

- **响应**:

```
{
  "code": 200,
  "message": "删除成功",
  "data": null
}
```

1.2 角色管理接口 (`/api/role`)

1.2.1 获取角色列表

- **URL**: `/api/role/getRoleList`
- **方法**: GET
- **参数**:
  - `roleName` (可选): 角色名称
- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": [
    {
      "id": "long",
      "roleName": "string",
      "rolePermissions": ["string"]
    }
  ]
}
```

1.3 用户管理接口 (`/api/user`)

1.3.1 获取用户管理信息

- **URL**: `/api/user/getUserManageInfo`

- **方法**: POST
- **请求体**:

  ```
  {
    "pageNum": "int",
    "pageSize": "int",
    "query": {
      "username": "string",
      "email": "string"
    }
  }
  ```

- **响应**:

  ```
  {
    "code": 200,
    "message": "success",
    "data": {
      "total": "long",
      "list": [
        {
          "id": "long",
          "username": "string",
          "email": "string",
          "roleId": "long"
        }
      ],
      "pageNum": "int",
      "pageSize": "int"
    }
  }
  ```

1.3.2 删除用户

- **URL**: `/api/user/deleteUser`
- **方法**: POST
- **请求体**:

  ```
  {
    "id": "long",
    "password": "string"
  }
  ```

- **响应**:

  ```json
  {
    "code": 200,
    "message": "删除成功",
    "data": null
  }
  ```

1.3.3 更新用户信息

- **URL**: `/api/user/updateUser`
- **方法**: POST
- **请求体**:

  ```json
  {
    "id": "long",
    "username": "string",
    "email": "string",
    "roleId": "long"
  }
  ```

- **响应**:

  ```json
  {
    "code": 200,
    "message": "更新成功",
    "data": null
  }
  ```

2. Boss招聘数据接口

2.1 基础数据接口 (`/api/boss/basic`)

2.1.1 获取城市名称列表

- **URL**: `/api/boss/basic/city-name-list`
- **方法**: GET
- **响应**:

```
["北京", "上海", "广州", "深圳"]
```

2.1.2 获取职位名称列表

- **URL**: `/api/boss/basic/position-name-list`
- **方法**: GET
- **参数**:
  - `cityName` (可选, 默认值: "all"): 城市名称
- **响应**:

```
["Java开发", "前端开发", "算法工程师", "数据分析师"]
```

2.2 用户分析接口 (`/api/boss/user`)

2.2.1 获取薪资热门职位分析

- **URL**: `/api/boss/user/salary-hot-job`
- **方法**: GET
- **参数**:
  - `limit` (可选, 默认值: 20): 限制返回数量
- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": [
    {
      "jobName": "string",
      "avgSalary": "double",
      "jobCount": "int"
    }
  ]
}
```

2.2.2 获取二维分析图表数据

- **URL**: `/api/boss/user/two-dimensional-analysis-chart`
- **方法**: GET
- **参数**:
  - `cityName` (可选, 默认值: "all"): 城市名称
  - `positionName` (可选, 默认值: "all"): 职位名称
  - `xAxis` (可选, 默认值: "salary_value"): X轴数据
  - `yAxis` (可选, 默认值: "degree_value"): Y轴数据
- **响应**:

```
{
  "code": 200,
  "message": "success",
  "data": [
    {
      "xValue": "double",
      "yValue": "double",
      "count": "int"
    }
  ]
}
```

2.3 数据可视化接口 (`/api/boss`)

2.3.1 获取所有职位数据

- **URL**: `/api/boss/jobs`
- **方法**: GET
- **响应**:

```
[
  {
    "id": "string",
    "position": "string",
    "company": "string",
    "salary": "string",
    "city": "string",
    "experience": "string",
    "education": "string",
    "timeStamp": "string"
  }
]
```

2.3.2 按职位获取数据

- **URL**: `/api/boss/jobs/position/{position}`
- **方法**: GET
- **路径参数**:
  - `position`: 职位名称
- **响应**:

```
[
  {
    "id": "string",
    "position": "string",
    "company": "string",
    "salary": "string",
    "city": "string",
    "experience": "string",
    "education": "string",
    "timeStamp": "string"
  }
]
```

2.3.3 按职位获取数据(请求参数)

- **URL**: `/api/boss/jobs/position`
- **方法**: GET
- **参数**:

  - `position`: 职位名称

- **响应**:

```
[
  {
    "id": "string",
    "position": "string",
    "company": "string",
    "salary": "string",
    "city": "string",
    "experience": "string",
    "education": "string",
    "timeStamp": "string"
  }
]
```

2.3.4 按职位和时间段获取数据

- **URL**: `/api/boss/jobs/position/{position}/time/{timePeriod}`

- **方法**: GET

- **路径参数**:

  - `position`: 职位名称
  - `timePeriod`: 时间段

- **响应**:

```
[
  {
    "id": "string",
    "position": "string",
    "company": "string",
    "salary": "string",
    "city": "string",
    "experience": "string",
    "education": "string",
    "timeStamp": "string"
  }
]
```

2.3.5 按职位和时间段获取数据(请求参数)

- **URL**: `/api/boss/jobs/position/time`
- **方法**: GET
- **参数**:
  - `position`: 职位名称
  - `timePeriod`: 时间段
- **响应**:

```
[
  {
    "id": "string",
    "position": "string",
    "company": "string",
    "salary": "string",
    "city": "string",
    "experience": "string",
    "education": "string",
    "timeStamp": "string"
  }
]
```

2.3.6 获取所有职位列表

- **URL**: `/api/boss/positions`
- **方法**: GET
- **响应**:

```
["Java开发", "前端开发", "算法工程师", "数据分析师"]
```

2.3.7 获取所有时间段列表

- **URL**: `/api/boss/timePeriods`
- **方法**: GET
- **响应**:

```
["2023-01", "2023-02", "2023-03"]
```

2.3.8 获取所有城市列表

- **URL**: `/api/boss/cities`
- **方法**: GET
- **响应**:

  ```
  ["北京", "上海", "广州", "深圳"]
  ```