

# 用户管理系统接口需求文档

## 1. 项目概述

本项目为“用户管理系统”，提供一套 RESTful 风格的接口，用于实现用户的基本增删查改功能。接口返回数据统一使用 JSON 格式，遵循 OpenAPI 3.0 规范。

## 2. 功能模块概述

功能模块	描述
获取用户列表	查询所有用户数据
获取用户详情	根据 ID 查询单个用户
创建用户	添加新用户信息
更新用户	修改已有用户信息
删除用户	根据 ID 删除用户

## 3. 接口定义

### 3.1 获取所有用户

- 方法: GET
- 路径: /api/users
- 说明: 获取用户列表
- 响应:
  - 200 OK: 返回用户数组

#### 示例响应:

```
1  [
2    {
3      "id": 1,
4      "name": "张三",
5      "gender": "男",
6      "age": 30,
7      "email": "zhangsan@example.com",
8      "phone": "13800000000"
9    }
10 ]
11
```

## 3.2 创建用户

- **方法:** POST
- **路径:** /api/users
- **说明:** 提交用户信息进行创建
- **请求体 (必须) :**

字段	类型	是否必须	描述
name	string	是	用户名
gender	string	是	用户性别
age	integer	是	用户年龄
email	string	是	邮箱 (格式)
phone	string	是	电话

- **响应:**
  - 200 OK: 返回创建的用户对象
  - 400 Bad Request: 参数验证失败

## 3.3 获取用户详情

- **方法:** GET
- **路径:** /api/users/{id}
- **路径参数:**

参数名	类型	是否必须	描述
id	integer	是	用户 ID

- **响应:**

- 200 OK: 返回对应的用户信息
  - 404 Not Found: 用户不存在
- 

## 3.4 更新用户

- **方法:** PUT

- **路径:** /api/users/{id}

- **路径参数:**

参数名	类型	是否必须	描述
id	integer	是	用户 ID

- **请求体:** 同创建用户

- **响应:**

- 200 OK: 用户更新成功
  - 400 Bad Request: 参数验证失败
  - 404 Not Found: 用户不存在
- 

## 3.5 删除用户

- **方法:** DELETE

- **路径:** /api/users/{id}

- **路径参数:**

参数名	类型	是否必须	描述
id	integer	是	用户 ID

- **响应:**

- 200 OK: 用户删除成功
  - 404 Not Found: 用户不存在
- 

## 4. 数据结构定义

## User 对象结构

字段	类型	是否必须	描述	约束条件
id	integer	否	用户ID	系统生成
name	string	是	用户姓名	最长 50 字符
gender	string	是	性别	最长 10 字符
age	integer	是	年龄	$\geq 0$
email	string	是	邮箱	email 格式
phone	string	是	电话	最长 20 字符

## 5. 接口错误码定义 (建议扩展)

HTTP 状态码	含义	说明
200	OK	请求成功
400	Bad Request	请求参数格式有误
404	Not Found	请求的资源不存在
500	Internal Error	服务内部错误 (保留扩展)

## 6. Open API 源代码

```
1 openapi: 3.0.0
2 info:
3   title: 用户管理系统 API
4   version: 1.0.0
5   description: 用户管理系统的RESTful API接口文档
6
7 servers:
8   - url: http://localhost:8080
9     description: 本地开发服务器
10
11 paths:
12   /api/users:
13     get:
14       summary: 获取所有用户
15       description: 返回所有用户的列表
16       responses:
17         '200':
18           description: 成功获取用户列表
19           content:
20             application/json:
21               schema:
22                 type: array
23                 items:
24                   $ref: '#/components/schemas/User'
25     post:
26       summary: 创建用户
27       description: 创建一个新的用户
28       requestBody:
29         required: true
30         content:
31           application/json:
32             schema:
33               $ref: '#/components/schemas/User'
34       responses:
35         '200':
36           description: 用户创建成功
37           content:
38             application/json:
39             schema:
```

```
40             $ref: '#/components/schemas/User'
41
42     '400':
43         description: 请求参数验证失败
44
45 /api/users/{id}:
46
47     get:
48         summary: 获取用户
49         description: 根据ID获取用户信息
50         parameters:
51             - name: id
52                 in: path
53                 required: true
54                 schema:
55                     type: integer
56
57         responses:
58             '200':
59                 description: 成功获取用户信息
60                 content:
61                     application/json:
62                         schema:
63                             $ref: '#/components/schemas/User'
64
65             '404':
66                 description: 用户不存在
67
68     put:
69         summary: 更新用户
70         description: 更新指定ID的用户信息
71         parameters:
72             - name: id
73                 in: path
74                 required: true
75                 schema:
76                     type: integer
77
78         requestBody:
79             required: true
80             content:
81                 application/json:
82                     schema:
83                         $ref: '#/components/schemas/User'
84
85         responses:
86             '200':
```

```
80     description: 用户更新成功
81     '400':
82         description: 请求参数验证失败
83     '404':
84         description: 用户不存在
85     delete:
86         summary: 删除用户
87         description: 删除指定ID的用户
88         parameters:
89             - name: id
90                 in: path
91                 required: true
92         schema:
93             type: integer
94         responses:
95             '200':
96                 description: 用户删除成功
97             '404':
98                 description: 用户不存在
99
100    components:
101        schemas:
102            User:
103                type: object
104                required:
105                    - name
106                    - gender
107                    - age
108                    - email
109                    - phone
110            properties:
111                id:
112                    type: integer
113                    description: 用户ID
114                name:
115                    type: string
116                    description: 用户姓名
117                    maxLength: 50
118                gender:
119                    type: string
```

```
120     description: 用户性别
121         maxLength: 10
122     age:
123         type: integer
124         description: 用户年龄
125         minimum: 0
126     email:
127         type: string
128         description: 用户邮箱
129         format: email
130         maxLength: 100
131     phone:
132         type: string
133         description: 用户电话
134         maxLength: 20
```