

API 接口文档

1. 一、通用规范

1.1 1.1 请求头

```
Content-Type: application/json
Authorization: Bearer <access_token> // 需要认证的接口
```

Fence 1

1.2 1.2 统一响应格式

成功响应:

```
{
  "code": 0,
  "message": "success",
  "data": {}
}
```

Fence 2

错误响应:

```
{
  "code": 10001,
  "message": "参数错误",
  "data": null
}
```

Fence 3

1.3 1.3 分页参数

参数	类型	必填	说明
page	int	否	页码, 默认 1
page_size	int	否	每页数量, 默认 10, 最大 100

分页响应格式:

```
{
  "code": 0,
  "message": "success",
  "data": {
    "list": [],
    "total": 100,
    "page": 1,
    "page_size": 10
  }
}
```

Fence 4

1.4 1.4 部门枚举值

枚举值	显示标签
backend	后端
frontend	前端
sre	SRE
product	产品
design	视觉设计
android	Android
ios	iOS

注意：API 响应中需同时返回 `department`（枚举值）和 `department_label`（中文标签）

2. 二、用户模块

2.1 2.1 用户注册

接口地址： `POST /user/register`

是否认证： 否

请求参数：

参数	类型	必填	说明
username	string	是	用户名
password	string	是	密码
nickname	string	是	昵称
department	string	是	部门

请求示例：

```
{
  "username": "xiaodeng001",
  "password": "123456",
  "nickname": "小登一号",
  "department": "backend"
}
```

Fence 5

响应示例:

```
{
  "code": 0,
  "message": "注册成功",
  "data": {
    "id": 1,
    "username": "xiaodeng001",
    "nickname": "小登一号",
    "role": "student",
    "department": "backend",
    "department_label": "后端"
  }
}
```

Fence 6

2.2 2.2 用户登录

接口地址: `POST /user/login`

是否认证: 否

请求参数:

参数	类型	必填	说明
username	string	是	用户名
password	string	是	密码

请求示例:

```
{
  "username": "xiaodeng001",
  "password": "123456"
}
```

Fence 7

响应示例:

```
{
  "code": 0,
  "message": "登录成功",
  "data": {
    "access_token": "eyJhbGciOiJIUzI1NiIs ... ",
    "refresh_token": "eyJhbGciOiJIUzI1NiIs ... ",
    "user": {
```

```
{
  "id": 1,
  "username": "xiaodeng001",
  "nickname": "小登一号",
  "role": "student",
  "department": "backend",
  "department_label": "后端"
}
```

Fence 8

2.3 2.3 刷新 Token

接口地址: `POST /user/refresh`

是否认证: 否

请求参数:

参数	类型	必填	说明
refresh_token	string	是	Refresh Token

响应示例:

```
{
  "code": 0,
  "message": "刷新成功",
  "data": {
    "access_token": "eyJhbGciOiJIUzI1NiIs ... ",
    "refresh_token": "eyJhbGciOiJIUzI1NiIs ... "
  }
}
```

Fence 9

2.4 2.4 获取当前用户信息

接口地址: `GET /user/profile`

是否认证: 是

响应示例:

```
{
  "code": 0,
  "message": "success",
  "data": {
    "id": 1,
    "username": "xiaodeng001",
    "nickname": "小登一号",
    "role": "student",
    "department": "backend",
    "department_label": "后端",
    "email": ""
  }
}
```

Fence 10

2.5 2.5 注销账号

接口地址: `DELETE /user/account`

是否认证: 是

请求参数:

参数	类型	必填	说明
password	string	是	当前密码 (二次确认)

响应示例:

```
{
  "code": 0,
  "message": "账号已注销",
  "data": null
}
```

Fence 11

3. 三、作业模块

3.1 3.1 发布作业

接口地址: `POST /homework`

是否认证: 是

权限要求: 老登 (admin)

请求参数:

参数	类型	必填	说明
title	string	是	作业标题
description	string	是	作业描述/要求
department	string	是	所属部门
deadline	string	是	截止时间
allow_late	bool	否	是否允许补交, 默认 false

请求示例:

```
{
  "title": "第一周作业: 实现简单的 HTTP 服务器",
  "description": "使用 Gin 框架实现一个简单的 RESTful API ... ",
  "department": "backend",
  "deadline": "2024-01-20 23:59:59",
  "allow_late": true
}
```

Fence 12

响应示例:

```
{
  "code": 0,
  "message": "发布成功",
  "data": {
    "id": 1,
    "title": "第一周作业: 实现简单的 HTTP 服务器",
    "department": "backend",
    "department_label": "后端",
    "deadline": "2024-01-20 23:59:59",
    "allow_late": true
  }
}
```

Fence 13

3.2 3.2 获取作业列表

接口地址: GET /homework

是否认证: 是

请求参数 (Query) :

参数	类型	必填	说明
department	string	否	部门筛选
page	int	否	页码
page_size	int	否	每页数量

响应示例:

```
{
  "code": 0,
  "message": "success",
  "data": {
    "list": [
      {
        "id": 1,
        "title": "第一周作业：实现简单的 HTTP 服务器",
        "department": "backend",
        "department_label": "后端",
        "creator": {
          "id": 10,
          "nickname": "后端讲师"
        },
        "deadline": "2024-01-20 23:59:59",
        "allow_late": true,
        "submission_count": 15
      }
    ],
    "total": 5,
    "page": 1,
    "page_size": 10
  }
}
```

Fence 14

3.3 3.3 获取作业详情

接口地址： GET /homework/:id

是否认证： 是

响应示例：

```
{
  "code": 0,
  "message": "success",
  "data": {
    "id": 1,
    "title": "第一周作业：实现简单的 HTTP 服务器",
    "description": "使用 Gin 框架实现 ...",
    "department": "backend",
    "department_label": "后端",
    "creator": {
      "id": 10,
      "nickname": "后端讲师"
    },
    "deadline": "2024-01-20 23:59:59",
    "allow_late": true,
    "submission_count": 15,
    "my_submission": {
      "id": 100,
      "score": 90,
      "is_excellent": false
    }
  }
}
```

```
}
```

Fence 15

说明: `my_submission` 字段仅对小登返回

3.4 3.4 修改作业

接口地址: `PUT /homework/:id`

是否认证: 是

权限要求: 老登 (admin) , 同部门均可修改

请求参数:

参数	类型	必填	说明
title	string	否	作业标题
description	string	否	作业描述
deadline	string	否	截止时间
allow_late	bool	否	是否允许补交

响应示例:

```
{
  "code": 0,
  "message": "修改成功",
  "data": {
    "id": 1,
    "title": "第一周作业: 实现简单的 HTTP 服务器",
    "deadline": "2024-01-22 23:59:59"
  }
}
```

Fence 16

3.5 3.5 删除作业

接口地址: `DELETE /homework/:id`

是否认证: 是

权限要求: 老登 (admin) , 同部门均可删除

响应示例:

```
{
  "code": 0,
  "message": "删除成功",
  "data": null
}
```

Fence 17

4. 四、作业提交模块

4.1 4.1 提交作业

接口地址: `POST /submission`

是否认证: 是

权限要求: 小登 (student)

请求参数:

参数	类型	必填	说明
homework_id	int	是	作业 ID
content	string	是	提交内容 (文本/链接)
file_url	string	否	附件地址

响应示例:

```
{
  "code": 0,
  "message": "提交成功",
  "data": {
    "id": 100,
    "homework_id": 1,
    "is_late": false,
    "submitted_at": "2024-01-15 18:30:00"
  }
}
```

Fence 18

业务逻辑:

- 1 检查作业是否存在
- 2 检查作业是否属于当前用户的部门
- 3 检查是否已过截止时间
- 4 如果已过截止时间, 检查是否允许补交
- 5 自动记录 `is_late` 状态

4.2 4.2 获取我的提交列表

接口地址: `GET /submission/my`

是否认证： 是

权限要求： 小登 (student)

响应示例：

```
{
  "code": 0,
  "message": "success",
  "data": {
    "list": [
      {
        "id": 100,
        "homework": {
          "id": 1,
          "title": "第一周作业",
          "department": "backend",
          "department_label": "后端"
        },
        "score": 90,
        "comment": "代码结构清晰",
        "is_excellent": false,
        "submitted_at": "2024-01-15 18:30:00"
      }
    ],
    "total": 3,
    "page": 1,
    "page_size": 10
  }
}
```

Fence 19

4.3 4.3 获取作业的所有提交

接口地址： GET /submission/homework/:homework_id

是否认证： 是

权限要求： 老登 (admin) , 同部门

响应示例：

```
{
  "code": 0,
  "message": "success",
  "data": {
    "list": [
      {
        "id": 100,
        "student": {
          "id": 1,
          "nickname": "小登一号",
          "department": "backend",
          "department_label": "后端"
        },
        "content": "GitHub 仓库地址: ...",

```

```
        "is_late": false,
        "score": null,
        "comment": null,
        "submitted_at": "2024-01-15 18:30:00"
    }
],
"total": 15,
"page": 1,
"page_size": 10
}
```

Fence 20

4.4 4.4 批改作业

接口地址: `PUT /submission/:id/review`

是否认证: 是

权限要求: 老登 (admin) , 同部门

请求参数:

参数	类型	必填	说明
score	int	否	分数, 0-100
comment	string	是	评语
is_excellent	bool	否	是否标记为优秀作业

请求示例:

```
{
  "score": 90,
  "comment": "代码结构清晰, 但缺少错误处理。",
  "is_excellent": false
}
```

Fence 21

响应示例:

```
{
  "code": 0,
  "message": "批改成功",
  "data": {
    "id": 100,
    "score": 90,
    "comment": "代码结构清晰, 但缺少错误处理。",
    "is_excellent": false,
    "reviewed_at": "2024-01-18 14:00:00"
  }
}
```

Fence 22

4.5 4.5 标记/取消优秀作业

接口地址: `PUT /submission/:id/excellent`

是否认证: 是

权限要求: 老登 (admin) , 同部门

请求参数:

参数	类型	必填	说明
is_excellent	bool	是	是否优秀作业

响应示例:

```
{
  "code": 0,
  "message": "标记成功",
  "data": {
    "id": 100,
    "is_excellent": true
  }
}
```

Fence 23

4.6 4.6 获取优秀作业列表

接口地址: `GET /submission/excellent`

是否认证: 是

请求参数 (Query) :

参数	类型	必填	说明
department	string	否	部门筛选
page	int	否	页码
page_size	int	否	每页数量

响应示例:

```
{
  "code": 0,
  "message": "success",
  "data": {
    "list": [
      {
        "id": 100,
        "homework": {
          "id": 1,
          "title": "第一周作业",

```

```
        "department": "backend",
        "department_label": "后端"
    },
    "student": {
        "id": 1,
        "nickname": "小登一号"
    },
    "score": 95,
    "comment": "非常优秀! "
}
],
"total": 10,
"page": 1,
"page_size": 10
}
}
```

Fence 24

5. 五、进阶接口（选做）

5.1 5.1 绑定邮箱

接口地址: `POST /user/bindEmail`

是否认证: 是

请求参数:

参数	类型	必填	说明
email	string	是	邮箱地址

5.2 5.2 AI 作业评价

接口地址: `POST /submission/:id/aiReview`

是否认证: 是

权限要求: 老登 (admin)

响应示例:

```
{
  "code": 0,
  "message": "success",
  "data": {
    "ai_comment": "代码分析结果: 项目结构清晰, 建议添加错误处理 ... ",
    "suggested_score": 85
  }
}
```

Fence 25

6. 六、接口汇总

6.1 用户模块

方法	路径	说明	认证	权限
POST	/user/register	用户注册	否	-
POST	/user/login	用户登录	否	-
POST	/user/refresh	刷新 Token	否	-
GET	/user/profile	获取用户信息	是	-
DELETE	/user/account	注销账号	是	-

6.2 作业模块

方法	路径	说明	认证	权限
POST	/homework	发布作业	是	老登
GET	/homework	获取作业列表	是	-
GET	/homework/:id	获取作业详情	是	-
PUT	/homework/:id	修改作业	是	老登+同部门
DELETE	/homework/:id	删除作业	是	老登+同部门

6.3 提交模块

方法	路径	说明	认证	权限
POST	/submission	提交作业	是	小登
GET	/submission/my	我的提交列表	是	小登
GET	/submission/homework/:id	作业的所有提交	是	老登+同部门
PUT	/submission/:id/review	批改作业	是	老登+同部门
PUT	/submission/:id/excellent	标记优秀	是	老登+同部门
GET	/submission/excellent	优秀作业列表	是	-