

# SUSTech\_ACD API Document

## Json Object 内容表

需求汇总:

fund: 经费, 一笔经费有

一个唯一的经费号 fundID

经费名称 fundName

一个课题组 userID

经费总额度 totalQuota

已使用额度 usedQuota

经费内容摘要 abstract

经费备注 remark

example:

```
{
  "fund_id": 5433912,
  "fund_name": "国自然",
  "user_id": 012,
  "total_quota": 500,
  "used_quota": 10,
  "abstract": "this is a abstract",
  "remark": "no remark"
}
```

expense: 支出

唯一识别: 支出编号 expenseID

支出名称 expenseName

该支出对应的经费号 fundID

支出额度 amount

经办人 operator

支出类别一级 category1

支出类别二级 category2

内容摘要 abstract

备注 remark

申请状态 ApplicationState = {0未发送, 1正在申请中, 2被拒绝打回, 3申请成功}

example:

```
{
  "expenseID": 1827328,
  "expenseName": "办公用品采购",
  "fundID": 9374829,
  "amount": 50,
  "operator": "沈昀",
  "category1": "会议费",
  "category2": "学术会议费",
  "abstract": "购买会议必需品",
  "remark": "请尽快审批",
  "applicationState": 1
}
```

user: 课题组

唯一识别: 编号 userID

课题组名称: userName 如“唐老师”

密码: password

example:

```
{
  "userID": 014,
  "userName": "唐老师",
  "password": "12345678"
}
```

注意，一个支出只对应一笔经费，一笔经费可以由很多次支出完成。支出amount的总和不超过经费总额度totalQuota。

一笔经费属于一个课题组。

## 操作列表:

查询:

- 1、(课题组) 查询某课题组名下的全部经费的完成情况（支出情况），对应《财务系统.xlsx》多项经费使用一览表。
- 2、(管理员) 查询若干个fundID的使用情况和执行率（executionRate），对应《财务系统.xlsx》xx经费使用汇总表。
- 3、(课题组) 查询一笔支出（expenseID）的申请情况，包含该支出的具体内容，和该支出的审批情况。

审批：

（管理员）查询待审批的所有支出列表，包含该支出的具体情况，并可审批。

增加/删除：

- 1、（管理员）增加或删除一个经费。
- 2、（课题组）增加（或删除）一个支出。
- 3、（课题组）送审一个支出。

修改：

- 1、（课题组）修改一个支出，并重新送审。

## 后端API：

---

1. 获取课题组全部经费完成情况（支出情况）：

- 请求方式：GET
- 路径：/api/users/{userID}/funds
- 真实路径（捆死user）：/api/users/funds
- 参数：userID（课题组ID）
- 响应：返回一个json object `Items: fund Json Object, expenses: list of expense json object` )

该课题组下所有经费的完成情况和每一个经费对应的所有支出情况

/api/users/313/funds

example:

```
{
  "items": [
    {
      "expenses": [
        {expenseObject1},
        {expenseObject2},
        {expenseObject3}
      ],
      "fund": {fundObject}
    },
    {
      同上
    },
    {
      同上
    }
  ]
}
```

```
]
}
```

本方法还需要同步写一个返回excel的方法:

/api/users/{userID}/funds/excel

## 2. 查询若干个fundID的使用情况和执行率, 或是所有fundID的使用情况和执行率

- 请求方式: GET
- 路径: /api/funds/{ multiple fundIDs, 逗号分隔}
- 参数: fundIDs (多个fundID, 逗号分隔)
- 响应: 返回json object `Items: fund json Object`, 执行率前端自己算
- /api/funds/58293,38472,37492,34728
- 如果没有后面的多个ID则直接返回所有情况

```
{
  "items": [
    {fundObject1},
    {
      同上
    },
    {
      同上
    }
  ]
}
```

## 1. 查询一笔支出的申请情况:

- 请求方式: GET
- 路径: /api/expenses/{expenseID}
- 参数: expenseID (支出ID)
- 响应: 返回 `expense json Object`
- 真实内容:
- /api/expenses?expenseID=expenseID
- 否则返回当前用户下所有expenseID

## 1. 查询待审批的所有支出列表:

- 请求方式: GET
- 路径: /api/expenses/pending
- 响应: 返回待审批的支出列表及其详细情况
- 返回ApplicationState = 1的 `expense json Object`

## 2. 增加或删除一个经费：

- 增加：
  - 请求方式：POST
  - 路径：/api/funds/add
  - 参数：经费相关信息 (fundName、userID、totalQuota等)
  - 响应：返回成功状态
  - /api/funds/add?fundID=123123&fundName=国自然&user=01012&totalQuota=500
- 删除：
  - 请求方式：DELETE
  - 路径：/api/funds/delete/{fundID}
  - 参数：fundID (待删除的经费ID)
  - 响应：返回删除成功的消息或状态

## 3. 增加或删除一个支出：

- 增加：
  - 请求方式：POST
  - 路径：/api/expenses/add
  - 参数：支出相关信息 (expenseID、expenseName、fundID、amount等)
  - 响应：返回新增的支出信息
- 删除：
  - 请求方式：DELETE
  - 路径：/api/expenses/delete/{expenseID}
  - 参数：expenseID (待删除的支出ID)
  - 响应：返回删除成功的消息或状态

## 4. 送审一个支出：

- 请求方式：POST
- 路径：/api/expenses/submit/{expenseID}
- 参数：expenseID (待送审的支出ID)
- 响应：返回送审成功的消息或状态

## 5. 修改一个支出并重新送审：

- 请求方式：PUT
- 路径：/api/expenses/modify/{expenseID}
- 参数：expenseID (待修改的支出ID) , 支出相关信息 (expenseName、amount等)
- 响应：返回修改成功的支出信息

## 6. 用户操作

/api/users/add

/api/users/modify/{userID}

/api/users/delete/{userID}

/api/users/login?username={username}&passwd={passwd}

返回1,0: 1成功, 0失败

上面是假的。真的用这个: (绑死所有内容到后端)

GET /api/users/login?username={username}&passwd={passwd}

返回值: 1或者0。1成功, 0失败(不允许登录), 后端需要保存当前用户

GET /api/users/currentUser

返回值: 当前username (如果=admin则是admin, 否则是普通用户)

API路径中的占位符(如{userID}、{fundID}、{expenseID})应替换为实际的课题组ID、经费ID和支出ID。

增加API (待完成)

通过审批

拒绝审批

Excel