

# ShareWallet Restful API Document

Version: v0.51  
UGeeker  
2014-1-1

- 所有token均应做为GET参数提交
- POST及PUT的数据数据使用JSON格式提交
- 文档中若有多个返回值时，则第一个返回为参数正确时的返回，其余为发生错误时的返回
- 关于返回结果的 HTTP 状态码
  - 如不声明，则返回默认为200
  - 若请求不符合标准则返回值为400
  - 若token错误或过期则返回值为401
  - 若权限错误则返回值为403
  - 若请求序号在数据库中不存在则返回值为404
- 所有时间戳均为ISO 8601格式
- Parameters 中[]为可选项

## Users

GET /users 获取全部用户列表

Parameters:

None

Returns:

```
[  
  {  
    "id": 1,  
    "name": "name" #user's name  
  },  
  ...  
]
```

---

POST /users 注册新用户

Parameters:

username  
password  
name

```
Returns:
{
  "id": id #id for new user
}, 201
{'status': 'username exist'}, 409
```

---

PUT /users 修改用户信息

```
Parameters:
  token
  [password] #新密码
  [name] #新名字
Returns:
  {'status': 'SUCCESS'}
```

---

GET /login 登陆并获得token

```
Parameters:
  username
  password
Returns:
{
  "status": "SUCCESS",
  "token": "", #your token
  "expire_time": "", #in iso format
  "user_id": "", #your user id
  "name": "", #your user's name
}
{'status': 'user not exist'}, 401
{'status': 'password not correct'}, 401
```

## Wallets

GET /wallets 返回用户参与的所有钱包列表

```
Parameters:
  token
Returns:
[
  {
    "id": 1,
    "name": "", #wallet name
    "description": "", # wallet description
    "timestamp": "", #create time of wallet, in iso
format
    "user_id": "", #creator of wallet
  },
  ...
]
```

---

POST /wallets 创建新钱包

```
Parameters:
  token
  name
  description
Returns:
{
  "id": id  #id for new wallet
}
```

---

GET /wallets/:wallet\_id 返回id为wallet\_id的钱包的信息

```
Parameters:
  token
Returns:
{
  "id": 1,
  "name": "", #wallet name
  "description": "", # wallet description
  "timestamp": "", #create time of wallet, in iso format
}
```

---

PUT /wallets/:wallet\_id 修改id为wallet\_id的钱包的信息

```
Parameters:
  token
  [name]
  [description]
Returns:
{'status': 'SUCCESS'}
```

## Wallet User

GET /wallets/:wallet\_id/users 得到钱包中所有参与者的列表

```
Parameters:
  token
Returns:
[
  {
    "id": 1,
    "name": "",
    "ballance": 3.23, #该用户的余额
    "role": [0/1], #0为普通, 1为管理员
  },
  ...
]
```

---

POST /wallets/:wallet\_id/users 向钱包中添加新用户

```
Parameters:
  token
```

```
    user_id
    role=[normal, admin] #表明身份
Returns:
    {'status': 'SUCCESS'}
    {'status': 'already exist'}, 409
```

---

PUT /wallets/:wallet\_id/users 修改钱包中已有用户的身份

```
Parameters:
    token
    user_id
    role=[normal, admin] #表明身份
Returns:
    {'status': 'SUCCESS'}
```

---

DELETE /wallets/:wallet\_id/users 从钱包中删除用户，删除时用户余额应为0

```
Parameters:
    token
Returns:
    {'status': 'SUCCESS'}
    {'status': 'still has money'}, 400 #若账户仍然有钱
```

## Event

GET /wallets/:wallet\_id/events 查看本钱包中全部交易

```
Parameters:
    token
Returns:
    [
        {
            "id": 1,
            "user_id": 1, #创建者id
            "name": "",
            "description": "",
            "timestamp": "" # 创建时间
        },
        ...
    ]
```

---

GET /wallets/:wallet\_id/events/:event\_id 查看钱包中某次交易的细节

```
Parameters:
    token
Returns:
    {
        "id": 1, #event_id
        "name": "",
        "description": "",
        "transaction":
```

```

    [
      {
        "user_id": 1, #交易用户id
        "name": "", #交易用户名字
        "amount": 2.34, #交易额, 用户收入为正
        "notes": "", #备注
      },
      ...
    ]
  }

```

---

POST /wallets/:wallet\_id/events 创建新的交易, 所有账目总和应该为0

#### Parameters:

```

token
name
description
transaction 为JSON对象, 包含全部交易信息
  [
    {
      "user_id": 1, #交易用户id
      "amount": 2.34, #交易额, 用户收入为正
      "notes": "", #备注
    },
    ...
  ]

```

#### Returns:

```

{"id": 1} #new event id
{"status": "user not exist"}, 400
{"status": "total money is not right"}, 400

```