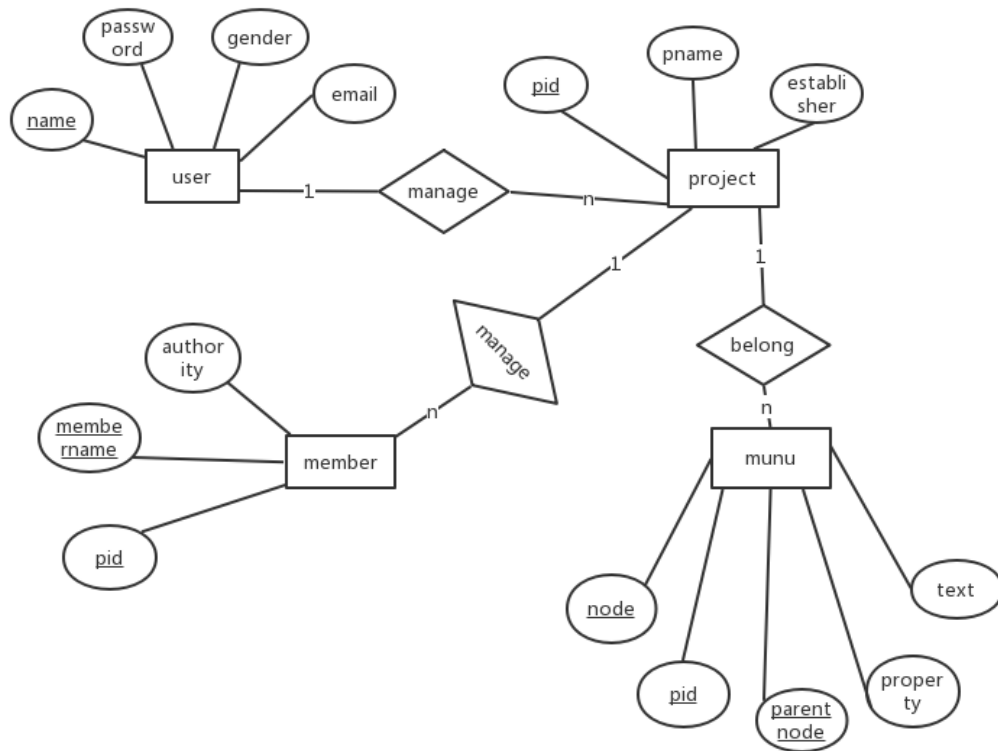


数据库设计:



用户表:

Name varchar[20]

Password varchar[20]

Gender integer //女 0; 男 1

Email varchar[30]

Name	password	gender	email
aa	*****	0	1234567788@163.com
bb	*****	1	1122334455@qq.com

项目表:

Pid integer

Pname varchar[20]

Establisher varchar[20]

Pid	pname	establisher
123	aa	ss

456	zz	xx
-----	----	----

项目成员及权限表：

Pid integer

Membername varchar[20]

Authority interger

pid	membername	authority
1122	asd	7367
3344	zxc	3452

项目目录表：

Pid integer

Node carchar[50]

Parentnode varchar[50]

Property interger//1 代表是文件夹 0 代表不是文件夹

Text mediumtext

pid	node	parentnode	property	text
156656	Fbgr4	Ver3	1	docx
556776	5tgef3	Sde23	1	txt

接口设计：

用户命令处理接口：

定义 status 类为返回类型，包含状态码，路由的 url 和异常说明

1. 登录：

1.1. 请求（GET / accounts / {name}）

Get 请求报文：

{"user_password":" "}

响应报文：

若成功，则返回：

{"code":"200"

"data":{"空

}"

若错误，则返回相应的错误码，

客户端错误状态码：

400：语法错误

401：当前请求需要验证

404：数据库没有该用户信息

406：用户请求格式不可得

服务端错误状态码（同适用于所有错误返回）：

500：服务器无法解析 ASP 代码

503：服务器不可用

data 里跟错误附加信息

错误报文以下接口均一样，以下不再赘述，只描述成功时 data 里具体的数据内容与特殊功能下的错误状态码

2. 注册：

2.1. 请求（POST / accounts）

Post 请求报文：

```
{"user_name":"","  
,"user_password":"","  
,"user_gender":" "  
,"user_email":" "}
```

响应报文：

正确 状态码 201data

```
{"user_name":"","  
,"user_password":"","  
,"user_gender":" "  
,"user_email":" "}
```

错误则返回相应的状态码及错误原因：

422：用户名已被占用,验证错误

400：语法错误

406：用户请求格式不可得

3. 个人信息管理

3.1. 请求（GET / datamanagers / {name}.

GET 请求报文：

无

响应报文：

正确 状态码 200data 为

```
    {"user_name":" "
    ,"user_password":" "
    ,"user_gender":" "
    ,"user_email"=" "}
```

失败则返回错误的状态码及错误原因

3.2. PATCH / datamanagers / {name}

请求报文：若不需要更改某一项更改为空即可

```
    {"user_name":" "
    ,"user_password":" "
    ,"user_gender":" "
    ,"user_email"=" "}
```

响应报文：

成功 状态码 201data 为

```
    {"user_name":" "
    ,"user_password":" "
    ,"user_gender":" "
    ,"user_email"=" "}
```

失败则返回错误的状态码及错误原因

404：请求修改的用户信息不存在

422：更改后用户名重复，

文档存储数据库访问接口：

4.工程管理：

4.1 新建工程：POST / projects

post 请求报文：

```
{
    "project_pname"=" "
    ,"project_establisher"=" "
    ,"project_describe"=" "
}
```

响应报文：

成功 状态码 201data 为

```
{
    "pid"=" "
    ,"project_pname"=" "
    ,"project_establisher"=" "
    ,"project_describe"=" "
```

}

失败则返回错误的状态码及错误原因

4.2 查看某一工程配置：GET/ projectsetting /{pid}

get 请求报文：无

响应报文

成功 状态码 200data 为

```
{"project_pname"=" "
,"project_establisher"=" "
,"project_describe"=" "
,"content":[{"project_member"=" "
,"project_authority"=" "
,"project_describe"=" "}]}
```

失败 则返回错误的状态码及错误原因

404：项目不存在

406：格式不正确

4.3 工程配置更改 PATCH / projectsettings / {pid}

patch 请求报文：若不需要更改某一项更改为空即可

```
{"project_pname"=" "
,"project_establisher"=" "
,"project_describe"=" "
,"content":[{"project_member"=" "
,"project_authority"=" "
,"project_describe"=" "}]}
```

响应报文

成功 状态码 201data 为

```
{"project_pname"=" "
,"project_establisher"=" "
,"project_describe"=" "
,"content":[{"project_member"=" "
,"project_authority"=" "
,"project_describe"=" "}]}
```

失败则返回错误的状态码及错误原因

404：项目不存在

406：格式不正确

4.4 查看用户参与的所有工程：GET / projects

get 请求报文：无

响应报文:

成功 状态码 200data 为

```
[{"project_member"=" "
,project_authority=" "
,"project_describe"=" "}….]
```

失败则返回错误的状态码及错误原因

404: 项目不存在

410: 项目已被永久删除

4.5 删除工程: DELETE / projects / {pid}

Delete 请求报文: 无

响应报文:

成功 状态码 204 data 为一个空项目

失败则返回错误的状态码及错误原因

403: 无权限

404: 数据库中无该项目

5 文档管理:

5.1 查看文档: GET / filemanagers / {fname} / file / {path}

get 请求报文: 无

GET 响应报文:

成功

```
{"file_text":" "}
```

错误的状态码及错误原因:

404: 数据库中无该项目

403: 无权限

5.2 修改文档: PUT / filemanagers / {fname} / file / {path}

PUT 请求报文:

```
{"file_text":" "}
```

响应报文:

成功 状态码 201data 为空

失败则返回错误的状态码及错误原因:

403: 无权限

404: 数据库中无该文档

5.3 新建文档: POST / filemanagers / {fname} / file / {path}

POST 请求报文:

```
{"file_node"=" " " "
```

```
, "file_fname" = " "
, "file_parentnode" = " "
, "file_property" = " "
, "file_text" = " " }
```

响应报文：

成功：状态码 201 data 为

```
{ "file_node" = " "
, "file_fname" = " "
, "file_parentnode" = " "
, "file_property" = " "
, "file_text" = " " }
```

失败则返回错误的状态码及错误原因：

403：无权限

404：数据库中无该项目

5.4 删除文档：DELETE / filemanagers / {fname} / file / {path}

DELETE 请求报文：空

响应报文：

成功状态码 204 及一个空文档

失败则返回错误的状态码及错误原因：

403：无权限

404：数据库中无该文档

6 目录管理：

1. 目录查看：GET / menus / {filenode} / filepaths / {path}

GET 请求报文：无

响应报文：

成功则返回状态码 200 data 为

```
[{"file_node": " ", "file_property": " "}, ...]
```

失败则返回错误的状态码及错误原因：

404：无该项目

2. 目录修改：PATCH / menus / {filenode} / filepaths / {path}

PATCH 请求报文：不需要修改则为空

```
{ "menu_text" = " " }
```

响应报文：

成功则返回状态码 201 data 为

```
{ "menu_text" = " " }
```

失败则返回错误的状态码及错误原因：

404: 无该项目

403: 无权限