Web课设数据字典

# Ⅰ 数据库设计

## 1.1 users表

uid int //用户ID (自增长)

uname varchar(24) //用户名 (base64编码后存入)

passwd varchar(24) //用户密码 (base64编码后存入)

## 1.2 files表

fid int //文件id (自增长)

fname varchar(100) //文件名 (base64编码后存入)

type varchar(10) //文件类型

md5 varchar(32) //文件MD5

size int //文件大小

did int ->dir(did) //所属文件夹

## 1.3 dir表

did int //文件夹id (自增长)

uid int ->user(uid) //文件夹所属用户id

pnode int ->dir(did) //文件夹父节点

dname varchar(100) //文件夹名 (base64编码后存入)

# Ⅱ 接口设计

## 2.1 用户相关

### 2.1.1 用户名查重：GET

Api: /caniuse

Send:

?uname = xxxxxx //xxxxxx为用户名

Receive: (json)

{

“type” : “uname”,

“data” : {

“uname” : “xxxxxx”, //”xxxxxx”为用户名

“caniuse” : “true/false” //”true”为用户名可用，”false”为不可用

}

}

### 2.1.2 用户注册：POST

Api: /user

Send: (json)

{

“type” : “regist”,

“data” : {

“uname” : “xxxxxx”, //用户名

“passwd” : “xxxxxx” //密码

}

}

Receive: (json)

{

“regist” : “true/false”, //”true”为注册成功，”false”为注册失败

“data” : {

“uname” : “xxxxxx”, //”xxxxxx”为用户名

“code” : xxxxxx //错误代码

}

}

错误代码:

2120 注册成功

2121 用户名已被使用

### 2.1.3 用户登录：POST

Api: /user

Send: (json)

{

“type” : “login”,

“data” : {

“uname” : “xxxxxx”, //用户名

“passwd” : “xxxxxx” //密码

}

}

Receive: (json)

{

“login” : “true/false”, //”true”为登录成功，”false”为登录失败

“data” : {

“uname” : “xxxxxx”, //用户名

“code” : xxxxxx //错误代码

}

}

错误代码:

2130 成功

2131 用户名或密码错误

2132 用户不存在

### 2.1.4 用户登出：GET

Api: /user

Send: (json)

{

“type” : “logout”

}

Receive: (json)

{}

## 2.2 文件相关

### 2.2.1 文件上传：POST

Api: /upload

Send:

?did=xxxxxx

md5=xxxxxx

FromData

Receive:

{

“upload” : “true/false”,

data : {

“code” : xxxxxx

}

}

### 2.2.2 文件下载：GET

Api: /download

Send:

?fid=xxxxxx

Receive:

文件

### 2.2.3 文件list：GET

Api: /getlist

Send:

did = xxxxxx //文件夹id，为-1则获取根目录下的文件

Receive:

{

“getlist” : “true/false”, //”true”为获取成功，”false”为失败

“data” : {

“code” : xxxxxx, //错误代码

“dirlist” : [

{

“did” : xxxxxx, //文件夹id

“dname” : “xxxxxx” //文件夹名

},

…

],

“fileslist” : [

{

“fid” : xxxxxx, //文件id

“fname” : “xxxxxx”, //文件名

“type” : “xxxxxx” //文件类型

},

…

]

}

}

### 2.2.4 新建文件夹：GET

Api : /newdir

Send :

pnode = xxxxxx //父节点did，为-1则在根节点下创建

dname = xxxxxx //文件夹名

Receive:

{

“newdir” : “true/false”, //创建成功/失败

“data” : {

“code” : xxxxxx, //错误代码

}

}

### 2.2.5 删除：GET

Api :　/del

Send :

type = dir/file //删除类型 dir/file

id = xxxxxx //删除id

Receive:

{

“del” : “true/false”, //删除成功/失败

“data” : {

“code” : xxxxxx //错误代码

}

}