

字节后端训练营一班5组设计文档

小组成员：田震雄，周一魏，崔墨迪，胡耀天，吴欢栋

成员分工：

- 田震雄：项目代码的主要编写者，抢课模块的设计以及实现，参与了排课模块的算法设计以及实现，redis数据库的设计，实现了日志中间件，debug，项目后期对边界情况的分析、处理
- 周一魏：编写项目设计文档的登录模块、成员模块、排课模块，参与实现了部分登录模块、成员模块的curd以及排课求解器的算法
- 崔墨迪：参与实现了登录模块和成员、排课模块的curd，负责项目的测试，编写postman接口集合，编写抢课部分的设计文档
- 胡耀天：程序脚手架的编写，session功能封装，实现了部分查询功能
- 吴欢栋：数据库设计，实现登录注销接口和部分crud

1 登录模块

1.1 登录

接口说明

接口功能

- 用户登录
- 登录成功后会返回cookie，浏览器会自动保存，以后对于该项目的每次请求都会自动携带cookie

API

/auth/login

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注

username	用户名	是	string			
password	密码	是	string			

返回参数格式

JSON

返回参数说明

返回参数	说明	类型	备注
Code	状态码	int	0表示成功，5表示密码错误
Data	数据	json	

示例

登录成功

请求地址

/auth/login

请求参数

```
JSON

1 {
2     "username": "JudgeAdmin",
3     "password": "JudgePassword2022"
4 }
```

返回参数

```
JSON

1 {
2     "Code": 0,
3     "Data": {
4         "UserID": "1"
5     }
6 }
```

用户名或者密码错误

请求地址

/auth/login

请求参数

```
JSON

1  {
2      "username": "JudgeAdmin",
3      "password": "JudgePassword2021"
4  }
```

返回参数

```
JSON

1  {
2      "Code": 5,
3      "Data": {
4          "UserID": ""
5      }
6  }
```

1.2 登出

接口说明

接口功能

- 退出当前用户的登录状态
- 登出成功后cookie会被清空

API

/auth/logout

请求方法

POST

请求参数格式

JSON

返回参数格式

JSON

返回参数说明

- 若当前处于登录状态，成功返回0
- 若当前未处于登录状态并请求此接口，返回6

示例

登出成功

成功登出后浏览器cookie会被清空

JSON

```
1  {
2      "Code": 0
3  }
```

登出失败

用户未登录

JSON

```
1  {
2      "Code": 6
3  }
```

1.3 获取个人信息

接口说明

接口功能

登录后访问个人信息页可以查看自己的信息，包括用户ID、用户名称、用户昵称

API

/auth/whoami

请求方法

GET

请求参数格式

JSON

请求参数说明

请求不需要额外携带参数

- 若当前处于登录状态，返回用户ID、用户名称、用户昵称等信息
- 若当前未处于登录状态并请求此接口，会返回6

返回参数格式

JSON

返回参数说明

- 成功返回0
- 用户未登录返回6

示例

获取成功

JSON

```
1 {
2     "Code": 0,
3     "Data": {
4         "UserID": "1",
5         "Nickname": "",
6         "Username": "JudgeAdmin",
7         "UserType": 1
8     }
9 }
```

用户未登录

JSON

```
1 {
2     "Code": 6,
3     "Data": {
4         "UserID": "",
5         "Nickname": "",
6         "Username": "",
7         "UserType": 0
8     }
9 }
```

2 成员模块

2.1 创建成员

接口说明

接口功能

创建成员

- 参数不合法返回ParamInvalid状态码
- 只有管理员才有操作权限，无权限返回PermDenied状态码

API

/member/create

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
username	用户名	是	string		只支持大小写，长度不小于 8 位 不超过 20 位	
nickname	用户昵称	是	string		长度不小于 4 位 不超过 20 位	
password	密码	是	string		同时包括大小写、数字，长度不少于 8 位 不超过 20 位	
usertype	用户类型	是	int		枚举值 1、2、3	

返回参数格式

JSON

示例

创建成功

请求地址

/member/create

请求参数

JSON

```
1 {
2   "username": "NewUser1",
3   "nickname": "NewUser1",
4   "password": "abcAbc123",
5   "usertype": 1
6 }
```

返回参数

JSON

```
1 {
2   "Code": 0,
3   "Data": {
4     "UserID": "2"
5   }
6 }
```

参数不合法

请求地址

/member/create

请求参数

JSON

```
1 {
2   "username": "New",
3   "nickname": "NewUser2",
4   "password": "abcAbc123",
5   "usertype": 1
6 }
```

返回参数

JSON

```
1 {
2   "Code": 1
3 }
```

该 Username 已存在

请求地址

/member/create

请求参数

```
JSON

1  {
2      "username": "NewUser2",
3      "nickname": "NewUser2",
4      "password": "abcAbc123",
5      "usertype": 1
6  }
```

返回参数

```
JSON

1  {
2      "Code": 2
3  }
```

用户未登录

请求地址

/member/create

请求参数

```
JSON

1  {
2      "username": "NewUser1",
3      "nickname": "NewUser1",
4      "password": "abcAbc123",
5      "usertype": 1
6  }
```

返回参数

```
JSON

1  {
2      "Code": 6
3  }
```


没有操作权限

请求地址

/member/create

请求参数

JSON

```
1  {
2      "username": "NewUser1",
3      "nickname": "NewUser1",
4      "password": "abcAbc123",
5      "usertype": 1
6  }
```

返回参数

JSON

```
1  {
2      "Code": 10
3  }
```

2.2 更新成员

接口说明

接口功能

更新成员昵称

API

/member/update

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
------	----	----	----	-----	----	----

userid	用户ID	是	string			该项不可修改
nickname	用户昵称	是	string			

返回参数格式

JSON

示例

更新成功

请求地址

/member/update

请求参数

```
JSON

1  {
2      "userid": "2",
3      "nickname": "NewUser5"
4  }
```

返回参数

```
JSON

1  {
2      "Code": 0
3  }
```

昵称不合法

请求地址

/member/update

请求参数

```
JSON

1  {
2      "userid": "4",
3      "nickname": "aa"
4  }
```

返回参数

JSON	
1	{
2	"Code": 1
3	}

用户已删除

请求地址

/member/update

请求参数

JSON	
1	{
2	"userid": "2",
3	"nickname": "NewUser1"
4	}

返回参数

JSON	
1	{
2	"Code": 3
3	}

用户不存在

请求地址

/member/update

请求参数

JSON	
1	{
2	"userid": "999",
3	"nickname": "NewUser2"
4	}

返回参数

```
JSON

1  {
2      "Code": 4
3  }
```

2.3 删除成员

接口说明

接口功能

- 删除成员
- 成员删除后，该成员不能够被登录且不应该不可见

API

/member/delete

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
userid	用户ID	是	string			

返回参数格式

JSON

示例

删除成功

请求地址

/member/delete

请求参数

JSON

```
1 {
2   "userid": "2"
3 }
```

返回参数

JSON

```
1 {
2   "Code": 0
3 }
```

用户已删除

请求地址

/member/delete

请求参数

JSON

```
1 {
2   "userid": "2"
3 }
```

返回参数

JSON

```
1 {
2   "Code": 3
3 }
```

用户不存在

请求地址

/member/update

请求参数

JSON

```
1 {  
2   "userid": "999"  
3 }
```

返回参数

JSON

```
1 {  
2   "Code": 4  
3 }
```

2.4 获取单个成员

接口说明

接口功能

获取单个成员

- 如果用户已删除返回UserHasDeleted状态码
- 不存在返回UserNotExisted状态码

API

/member/?UserID=:id

请求方法

GET

请求参数格式

URL路径变量

请求参数说明

id为需要查询的用户

返回参数格式

JSON

示例

查询id为1的用户

请求地址

| /member?UserID=1

返回参数

JSON

```
1 {
2     "Code": 0,
3     "Data": {
4         "UserID": "1",
5         "Nickname": "",
6         "Username": "JudgeAdmin",
7         "UserType": 1
8     }
9 }
```

用户已删除

请求地址

| /member?UserID=2

返回参数

JSON

```
1 {
2     "Code": 3,
3     "Data": {
4         "UserID": "2",
5         "Nickname": "ADeletedUser",
6         "Username": "ADeletedUser",
7         "UserType": 1
8     }
9 }
```

用户不存在

请求地址

| /member?UserID=999

返回参数

JSON

```
1 {  
2     "Code": 4,  
3     "Data": {  
4         "UserID": "",  
5         "Nickname": "",  
6         "Username": "",  
7         "UserType": 0  
8     }  
9 }
```

2.5 批量获取成员

接口说明

接口功能

分页批量获取成员

API

/member/list?Offset=0&Limit=2

请求方法

GET

请求参数格式

URL路径变量

请求参数说明

id为需要查询的用户

返回参数格式

JSON

示例

获取所有用户

请求地址

/member/list

返回参数

JSON

```
1  {
2      "Code": 0,
3      "Data": {
4          "MemberList": [
5              {
6                  "UserID": "1",
7                  "Nickname": "",
8                  "Username": "JudgeAdmin",
9                  "UserType": 1
10             },
11             {
12                 "UserID": "4",
13                 "Nickname": "NewUser3",
14                 "Username": "NewUser3",
15                 "UserType": 1
16             }
17         ]
18     }
19 }
```

3 排课模块

3.1 创建课程

接口说明

接口功能

创建课程

API

/course/create

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
subject	课程名称	是	string			

capacity	课程容量	是	int			
----------	------	---	-----	--	--	--

返回参数格式

JSON

示例

创建成功

请求地址

/course/create

请求参数

JSON

```
1 {
2     "Name": "课程1",
3     "Cap": 150
4 }
```

返回参数

JSON

```
1 {
2     "Code": 0,
3     "Data": {
4         "CourseID": "1"
5     }
6 }
```

课程已存在

请求地址

/course/create

请求参数

JSON

```
1 {
2   "Name": "课程1",
3   "Cap": 200
4 }
```

返回参数

JSON

```
1 {
2   "Code": 255,
3   "Data": {
4     "CourseID": ""
5   }
6 }
```

3.2 获取单个课程

接口说明

接口功能

- 获取单个课程
- 不存在返回CourseNotExisted状态码

API

/course/get?CourseID=:id

请求方法

GET

请求参数格式

URL路径变量

请求参数说明

id为需要查询的课程

返回参数格式

JSON

示例

查询id为1的课程

请求地址

| /course/get?CourseID=1

返回参数

```
JSON

1  {
2      "Code": 0,
3      "Data": {
4          "CourseID": "1",
5          "Name": "课程1",
6          "TeacherID": "3"
7      }
8  }
```

课程不存在

请求地址

| /course/get?CourseID=999

返回参数

```
JSON

1  {
2      "Code": 12,
3      "Data": {
4          "CourseID": "",
5          "Name": "",
6          "TeacherID": ""
7      }
8  }
```

3.3 绑定课程

接口说明

接口功能

| 老师绑定课程

API

| /teacher/bind_course

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
courseid	课程ID	是	string			
teacherid	老师ID	是	string			

返回参数格式

JSON

示例

绑定成功

请求地址

/teacher/bind_course

请求参数

JSON

```
1 {
2     "courseid": "1",
3     "teacherid": "3"
4 }
```

返回参数

JSON

```
1 {
2     "Code": 0
3 }
```

被绑定的成员不是教师

请求地址

/teacher/bind_course

请求参数

JSON	
1	{
2	"courseid": "1",
3	"teacherid": "1"
4	}

返回参数

JSON	
1	{
2	"Code": 255
3	}

课程已绑定过

请求地址

| /teacher/bind_course

请求参数

JSON	
1	{
2	"courseid": "1",
3	"teacherid": "1"
4	}

返回参数

JSON	
1	{
2	"Code": 8
3	}

课程不存在

请求地址

| /teacher/bind_course

请求参数

JSON

```
1 {
2   "courseid": "999",
3   "teacherid": "1"
4 }
```

返回参数

JSON

```
1 {
2   "Code": 12
3 }
```

教师不存在

请求地址

/teacher/bind_course

请求参数

JSON

```
1 {
2   "courseid": "1",
3   "teacherid": "999"
4 }
```

返回参数

JSON

```
1 {
2   "Code": 4
3 }
```

3.4 解绑课程

接口说明

接口功能

老师解绑课程

API

/teacher/unbind_course

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
courseid	课程ID	是	string			
teacherid	老师ID	是	string			

返回参数格式

JSON

示例

解绑成功

请求地址

/teacher/unbind_course

请求参数

```
JSON

1 {
2     "courseid": "1",
3     "teacherid": "3"
4 }
```

返回参数

```
JSON

1 {
2     "Code": 0
3 }
```

课程未绑定过

请求地址

/teacher/unbind_course

请求参数

JSON	
1	{
2	"courseid": "1",
3	"teacherid": "3"
4	}

返回参数

JSON	
1	{
2	"Code": 9
3	}

课程不存在

请求地址

/teacher/unbind_course

请求参数

JSON	
1	{
2	"courseid": "999",
3	"teacherid": "1"
4	}

返回参数

JSON	
1	{
2	"Code": 9
3	}

教师不存在

请求地址

/teacher/unbind_course

请求参数

JSON	
1	{
2	"courseid": "1",
3	"teacherid": "999"
4	}

返回参数

JSON	
1	{
2	"Code": 4
3	}

3.5 获取教师课程

接口说明

接口功能

获取老师的所有课程

API

/teacher/get_course?TeacherID=:id

请求方法

GET

请求参数格式

URL路径变量

请求参数说明

id为需要查询的课程

返回参数格式

JSON

示例

查询id为3的老师的所有课程

请求地址

/teacher/get_course?TeacherID=2

返回参数

JSON	
1	{
2	"Code": 0,
3	"Data": {
4	"CourseList": [
5	{
6	"CourseID": "1",
7	"Name": "课程1",
8	"TeacherID": "2"
9	}
10]
11	}
12	}

教师不存在

请求地址

/teacher/get_course?TeacherID=999

返回参数

JSON	
1	{
2	"Code": 4
3	}

3.6 排课

接口说明

接口功能

返回教师与课程的二分图最大匹配

API

/course/schedule

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
TeacherCourseRelationShip	课程与教师的绑定关系关系	是	map			

返回参数格式

JSON

示例

教师1绑定课程1，3，4；教师2绑定课程2，3，4；教师3绑定课程1，2，3

请求地址

/course/schedule

请求参数

JSON

```
1  {
2      "TeacherCourseRelationShip":{
3          "1":["1","3","4"],
4          "2":["2","3","4"],
5          "3":["1","2","3"]
6      }
7  }
```

返回参数

```
JSON

1 {
2     "Code": 0,
3     "Data": {
4         "1": "3",
5         "2": "2",
6         "3": "1"
7     }
8 }
```

4 抢课模块

4.1 学生抢课

接口说明

接口功能

学生登录账号后抢课

API

/student/book_course

请求方法

POST

请求参数格式

JSON

请求参数说明

请求参数	说明	必选	类型	默认值	限制	备注
courseid	课程ID	是	string			
teacherid	老师ID	是	string			

返回参数格式

JSON

示例

请求地址

/teacher/unbind_course

抢课成功

请求参数

JSON	
1	{
2	"courseid": "1",
3	"teacherid": "3"
4	}

返回参数

JSON	
1	{
2	"Code": 0
3	}

课程已满

返回参数

JSON	
1	{
2	"Code": 7
3	}

学生不存在

返回参数

JSON	
1	{
2	"Code": 11
3	}

课程不存在

返回参数

JSON

```
1 {  
2     "Code": 12  
3 }
```

4.2 查看学生课表

接口说明

接口功能

查看学生课表

API

/student/course?StudentID=1

请求方法

GET

请求参数格式

URL路径变量

请求参数说明

StudentID为想要查询课表的学生id

返回参数格式

JSON

示例

查看课表成功

请求地址

/student/course?StudentID=14

返回参数

JSON

```
1  {
2      "Code": 0,
3      "Data": {
4          "CourseList": [
5              {
6                  "CourseID": "5",
7                  "Name": "课程9",
8                  "TeacherID": "15"
9              },
10             {
11                 "CourseID": "2",
12                 "Name": "课程1",
13                 "TeacherID": "12"
14             }
15         ]
16     }
17 }
```

学生不存在

返回参数

JSON

```
1  {
2      "Code": 11,
3      "Data": {
4          "CourseList": null
5      }
6  }
```

学生没有课程

返回参数

JSON

```
1  {
2      "Code": 13,
3      "Data": {
4          "CourseList": null
5      }
6  }
```