

# StudyGo自习室预约系统后端接口文档

---

## 1. 用户管理接口 (UserController)

---

### 1.1 用户注册

- URL : /api/user/register
- 方法 : POST
- 请求体 : User对象
- 描述 : 用户注册
- 返回 : 注册结果

### 1.2 用户登录

- URL : /api/user/login
- 方法 : POST
- 请求体 : User对象 (包含username和password)
- 描述 : 用户登录
- 返回 : 登录结果和用户信息

### 1.3 获取用户信息

- URL : /api/user/info
- 方法 : GET
- 描述 : 获取当前登录用户的信息
- 返回 : 用户信息

### 1.4 更新用户信息

- URL : /api/user/update
- 方法 : PUT
- 请求体 : User对象
- 描述 : 更新当前登录用户的信息
- 返回 : 更新结果

### 1.5 修改密码

- URL : /api/user/password
- 方法 : PUT
- 请求体 : {"oldPassword": "旧密码", "newPassword": "新密码"}
- 描述 : 修改当前登录用户的密码
- 返回 : 修改结果

### 1.6 管理员获取用户列表

- URL : /api/admin/user/list

- 方法: GET
- 参数:
  - page: 页码 (默认1)
  - size: 每页数量 (默认10)
  - keyword: 关键词 (可选)
- 描述: 管理员获取用户列表
- 返回: 用户列表分页数据

## 1.7 管理员获取用户详情

- URL: /api/admin/user/{id}
- 方法: GET
- 参数: id: 用户ID
- 描述: 管理员获取指定用户详情
- 返回: 用户详情

## 1.8 管理员更新用户信息

- URL: /api/admin/user/{id}
- 方法: PUT
- 参数: id: 用户ID
- 请求体: User对象
- 描述: 管理员更新指定用户信息
- 返回: 更新结果

## 1.9 管理员删除用户

- URL: /api/admin/user/{id}
- 方法: DELETE
- 参数: id: 用户ID
- 描述: 管理员删除指定用户
- 返回: 删除结果

## 1.10 管理员添加用户

- URL: /api/admin/user/add
- 方法: POST
- 请求体: User对象
- 描述: 管理员添加用户
- 返回: 添加结果

# 2. 自习室管理接口 (StudyRoomController)

---

## 2.1 获取自习室列表

- URL: /api/studyroom/list
- 方法: GET
- 参数:

- page: 页码 (默认1)
  - size: 每页数量 (默认10)
  - status: 状态 (可选)
- 描述: 获取自习室列表
- 返回: 自习室列表分页数据

## 2.2 获取自习室详情

- URL : /api/studyroom/{id}
- 方法: GET
- 参数: id: 自习室ID
- 描述: 获取自习室详情
- 返回: 自习室详情

## 2.3 模糊查询自习室

- URL : /api/studyroom/search
- 方法: GET
- 参数:
  - keyword: 关键词
  - page: 页码 (默认1)
  - size: 每页数量 (默认10)
- 描述: 模糊查询自习室
- 返回: 自习室列表分页数据

## 2.4 获取自习室座位列表

- URL : /api/studyroom/{id}/seats
- 方法: GET
- 参数:
  - id: 自习室ID
  - date: 日期 (可选)
  - status: 状态 (可选)
- 描述: 获取自习室座位列表
- 返回: 座位列表

## 2.5 添加自习室

- URL : /api/admin/studyroom/add
- 方法: POST
- 请求体: StudyRoom对象
- 描述: 管理员添加自习室
- 返回: 添加结果

## 2.6 更新自习室信息

- URL : /api/admin/studyroom/{id}

- 方法 : PUT
- 参数 : id: 自习室ID
- 请求体 : StudyRoom对象
- 描述 : 管理员更新自习室信息
- 返回 : 更新结果

## 2.7 删除自习室

- URL : /api/admin/studyroom/{id}
- 方法 : DELETE
- 参数 : id: 自习室ID
- 描述 : 管理员删除自习室
- 返回 : 删除结果

## 2.8 更新自习室状态

- URL : /api/admin/studyroom/{id}/status
- 方法 : PUT
- 参数 :
  - id: 自习室ID
  - status: 状态
- 描述 : 管理员更新自习室状态
- 返回 : 更新结果

## 3. 座位管理接口 (SeatController)

---

### 3.1 管理员添加座位

- URL : /api/admin/seat/add
- 方法 : POST
- 请求体 : Seat对象
- 描述 : 管理员添加座位
- 返回 : 添加结果

### 3.2 管理员获取所有座位信息

- URL : /api/admin/seat/all
- 方法 : GET
- 描述 : 管理员获取所有座位信息
- 返回 : 所有座位列表

### 3.3 管理员更新座位信息

- URL : /api/admin/seat/{id}
- 方法 : PUT
- 参数 : id: 座位ID
- 请求体 : Seat对象
- 描述 : 管理员更新座位信息

- 返回：更新结果

### 3.4 管理员更新座位状态

- URL：/api/admin/seat/{id}/status
- 方法：PUT
- 参数：
  - id: 座位ID
  - status: 状态 0:可用 1:已预约 2:维护中
- 描述：管理员更新座位状态
- 返回：更新结果

### 3.5 管理员批量设置座位信息

- URL：/api/admin/seat/batch
- 方法：POST
- 参数：
  - roomId: 自习室ID
  - startNumber: 起始编号
  - endNumber: 结束编号
  - status: 状态 0:可用 1:已预约 2:维护中
- 描述：管理员批量设置座位信息
- 返回：设置结果

### 3.6 管理员删除座位

- URL：/api/admin/seat/delete/{id}
- 方法：POST
- 参数：id: 座位ID
- 描述：管理员删除座位
- 返回：删除结果

## 4. 预约管理接口 (ReservationController)

---

### 4.1 预约座位

- URL：/api/reservation/add
- 方法：POST
- 参数：
  - seatId: 座位ID
  - date: 预约开始日期
  - days: 预约天数
- 描述：用户预约座位
- 返回：预约ID

## 4.2 获取个人预约列表

- URL : /api/reservation/list
- 方法 : GET
- 参数 :
  - page: 页码 (默认1)
  - size: 每页数量 (默认10)
  - status: 状态 (可选)
- 描述 : 获取当前登录用户的预约列表
- 返回 : 预约列表分页数据

## 4.3 获取预约详情

- URL : /api/reservation/{id}
- 方法 : GET
- 参数 : id: 预约ID
- 描述 : 获取预约详情
- 返回 : 预约详情

## 4.4 取消预约

- URL : /api/reservation/{id}/cancel
- 方法 : PUT
- 参数 : id: 预约ID
- 描述 : 用户取消预约
- 返回 : 取消结果

## 4.5 管理员获取预约列表

- URL : /api/admin/reservation/list
- 方法 : GET
- 参数 :
  - page: 页码 (默认1)
  - size: 每页数量 (默认10)
  - userId: 用户ID (可选)
  - roomId: 自习室ID (可选)
  - status: 状态 (可选)
  - startDate: 开始日期 (可选)
  - endDate: 结束日期 (可选)
- 描述 : 管理员获取预约列表
- 返回 : 预约列表分页数据

## 4.6 管理员处理预约

- URL : /api/admin/reservation/{id}
- 方法 : PUT

- 参数：
  - id: 预约ID
  - status: 状态 0:待审核 1:已通过 2:已拒绝 3:已取消
- 描述：管理员处理预约
- 返回：处理结果

## 4.7 导出预约记录

- URL：/api/admin/reservation/export
- 方法：GET
- 参数：
  - startDate: 开始日期（可选）
  - endDate: 结束日期（可选）
  - roomId: 自习室ID（可选）
  - status: 状态（可选）
- 描述：管理员导出预约记录
- 返回：预约记录列表

## 5. 公告管理接口 (AnnouncementController)

---

### 5.1 获取公告列表

- URL：/api/announcement/list
- 方法：GET
- 参数：
  - page: 页码（默认1）
  - size: 每页数量（默认10）
- 描述：获取公告列表
- 返回：公告列表分页数据

### 5.2 获取公告详情

- URL：/api/announcement/{id}
- 方法：GET
- 参数：id: 公告ID
- 描述：获取公告详情
- 返回：公告详情

### 5.3 发布公告

- URL：/api/admin/announcement/add
- 方法：POST
- 请求体：Announcement对象
- 描述：管理员发布公告
- 返回：发布结果

## 5.4 更新公告

- URL : /api/admin/announcement/{id}
- 方法 : PUT
- 参数 : id: 公告ID
- 请求体 : Announcement对象
- 描述 : 管理员更新公告
- 返回 : 更新结果

## 5.5 删除公告

- URL : /api/admin/announcement/{id}
- 方法 : DELETE
- 参数 : id: 公告ID
- 描述 : 管理员删除公告
- 返回 : 删除结果

## 5.6 更新公告状态

- URL : /api/admin/announcement/{id}/status
- 方法 : PUT
- 参数 :
  - id: 公告ID
  - status: 状态 0:已下架 1:已发布
- 描述 : 管理员更新公告状态
- 返回 : 更新结果

# 6. 反馈管理接口 (FeedbackController)

---

由于FeedbackController的具体实现未在提供的代码中找到，但根据项目结构和命名规范，以下是可能的接口：

## 6.1 提交反馈

- URL : /api/feedback/add
- 方法 : POST
- 请求体 : Feedback对象
- 描述 : 用户提交反馈
- 返回 : 提交结果

## 6.2 获取个人反馈列表

- URL : /api/feedback/list
- 方法 : GET
- 参数 :
  - page: 页码 (默认1)
  - size: 每页数量 (默认10)
  - status: 状态 (可选)



- 描述：获取当前登录用户的反馈列表
- 返回：反馈列表分页数据

## 6.3 获取反馈详情

- URL：/api/feedback/{id}
- 方法：GET
- 参数：id: 反馈ID
- 描述：获取反馈详情
- 返回：反馈详情

## 6.4 管理员获取反馈列表

- URL：/api/admin/feedback/list
- 方法：GET
- 参数：
  - page: 页码（默认1）
  - size: 每页数量（默认10）
  - type: 反馈类型（可选）
  - status: 状态（可选）
- 描述：管理员获取反馈列表
- 返回：反馈列表分页数据

## 6.5 管理员处理反馈

- URL：/api/admin/feedback/{id}
- 方法：PUT
- 参数：id: 反馈ID
- 请求体：{"status": 状态, "reply": "回复内容"}
- 描述：管理员处理反馈
- 返回：处理结果