

1. 登录与注册模块

用户注册

方法	POST
路径	/api/auth/register
说明	用户通过用户名、密码和邮箱进行注册
请求头	Content-Type: application/json
参数	username (string): 用户名 password (string): 密码 email (string): 邮箱
请求示例	json { "username": "JohnDoe", "password": "password123", "email": "johndoe@example.com" }
返回状态码	200: 注册成功 400: 参数错误
返回数据格式	json { "message": "注册成功", "userId": "12345" }
错误响应	json { "error": "邮箱已被注册" }
验证规则	密码至少包含 8 个字符，需含大小写字母和数字
安全性要求	密码需加密存储，使用 HTTPS
备注	无

用户登录

方法	POST
路径	/api/auth/login
说明	用户通过用户名和密码登录
请求头	Content-Type: application/json
参数	username (string): 用户名 password (string): 密码
请求示例	json { "username": "JohnDoe", "password": "password123" }
返回状态码	200: 登录成功 401: 登录失败

返回数据格式	json { "token": "abc.def.ghi", "userId": "12345" }
错误响应	json { "error": "用户名或密码错误" }
验证规则	用户名和密码不得为空
安全性要求	使用 HTTPS，令牌需设置过期时间
备注	无

获取用户信息

方法	GET
路径	/api/auth/userinfo
说明	获取当前登录用户信息
请求头	Authorization: Bearer {token}
参数	无
请求示例	无
返回状态码	200: 请求成功 401: 未登录
返回数据格式	json { "username": "JohnDoe", "email": "johndoe@example.com" }
错误响应	json { "error": "无效的令牌" }
验证规则	验证用户身份令牌
安全性要求	身份令牌需加密传输
备注	无

2. 骑行记录模块

上传骑行记录

方法	POST
路径	/api/riding/upload
说明	用户上传单次骑行记录
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	distance (float): 骑行距离 time (string): 骑行时长 date (string): 骑行日期

请求示例	json { "distance": 15.2, "time": "00:45:00", "date": "2024-11-27" }
返回状态码	200: 上传成功 400: 参数错误
返回数据格式	json { "message": "骑行记录已上传", "recordId": "67890" }
错误响应	json { "error": "参数缺失或无效" }
验证规则	骑行距离需为正数
安全性要求	数据传输需加密
备注	无

查询历史骑行记录

方法	GET
路径	/api/riding/history
说明	获取用户的所有骑行记录
请求头	Authorization: Bearer {token}
参数	startDate (string): 起始日期 endDate (string): 结束日期
请求示例	/api/riding/history?startDate=2024-11-01&endDate=2024-11-27
返回状态码	200: 请求成功 401: 未授权
返回数据格式	json [{ "distance": 10.5, "time": "00:30:00", "date": "2024-11-01" }]
错误响应	json { "error": "未提供身份令牌" }
验证规则	日期格式需为 YYYY-MM-DD
安全性要求	数据需按用户隔离
备注	无

3. 骑行社区模块

发布帖子

方法	POST
路径	/api/community/post
说明	用户发布社区帖子
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	title (string): 帖子标题 content (string): 帖子内容
请求示例	json { "title": "骑行技巧", "content": "保持骑行节奏, 注意呼吸调整" }
返回状态码	200: 发布成功 400: 参数错误
返回数据格式	json { "message": "帖子已发布", "postId": "12345" }
错误响应	json { "error": "帖子标题或内容不能为空" }
验证规则	标题长度需在 5-100 字符之间
安全性要求	用户需登录后才能发布
备注	无

获取帖子列表

方法	GET
路径	/api/community/posts
说明	获取所有社区帖子
请求头	无
参数	page (int): 页码 size (int): 每页条数
请求示例	/api/community/posts?page=1&size=10

返回状态码	200: 请求成功 400: 参数错误
返回数据格式	json [{ "postId": "12345", "title": "骑行技巧", "content": "保持节奏", "author": "JohnDoe" }]
错误响应	json { "error": "无效的页码或每页条数" }
验证规则	页码和每页条数需为正整数
安全性要求	无特殊要求
备注	支持分页查询

发布评论

方法	POST
路径	/api/community/comment
说明	用户对帖子发表评论
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	postId (string): 帖子 IDcontent (string): 评论内容
请求示例	json { "postId": "12345", "content": "谢谢分享!" }
返回状态码	200: 评论成功 400: 参数错误
返回数据格式	json { "message": "评论已发布", "commentId": "67890" }
错误响应	json { "error": "评论内容不能为空" }
验证规则	评论内容长度需在 1-300 字符之间
安全性要求	用户需登录后才能评论
备注	无

删除帖子

方法	DELETE
路径	/api/community/{id}
说明	删除指定社区帖子

方法	DELETE
请求头	Authorization: Bearer {token}
参数	路径参数: id (string): 帖子 ID
请求示例	/api/community/12345
返回状态码	200: 删除成功 403: 无权限删除
返回数据格式	json { "message": "帖子已删除" }
错误响应	json { "error": "未找到帖子或无权限操作" }
验证规则	用户只能删除自己发布的帖子
安全性要求	验证用户身份和权限
备注	无

4. 骑行二手交易模块

发布交易信息

方法	POST
路径	/api/trade/create
说明	用户发布二手骑行装备的交易信息
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	item (string): 商品名称 price (float): 商品价格 description (string): 商品描述
请求示例	json { "item": "山地车", "price": 1500.00, "description": "九成新, 附赠头盔" }
返回状态码	200: 发布成功 400: 参数错误
返回数据格式	json { "message": "交易信息已发布", "tradeId": "12345" }
错误响应	json { "error": "商品名称或价格无效" }
验证规则	商品价格需为正数, 描述长度不得超过 500 字符
安全性要求	用户需登录后才能发布
备注	无

获取交易列表

方法	GET
路径	/api/trade/list
说明	获取所有交易信息
请求头	无
参数	page (int): 页码 size (int): 每页条数
请求示例	/api/trade/list?page=1&size=10
返回状态码	200: 请求成功 400: 参数错误
返回数据格式	json [{ "tradeId": "12345", "item": "山地车", "price": 1500.00, "description": "九成新" }]
错误响应	json { "error": "无效的页码或每页条数" }
验证规则	页码和每页条数需为正整数
安全性要求	无特殊要求
备注	支持分页查询

查询单个交易信息

方法	GET
路径	/api/trade/{id}
说明	查询指定交易信息
请求头	无
参数	路径参数: id (string): 交易 ID
请求示例	/api/trade/12345
返回状态码	200: 请求成功 404: 未找到交易
返回数据格式	json { "tradeId": "12345", "item": "山地车", "price": 1500.00, "description": "九成新" }
错误响应	json { "error": "交易信息不存在" }
验证规则	路径参数需为有效的交易 ID

安全性要求	无特殊要求
备注	无

删除交易信息

方法	DELETE
路径	/api/trade/{id}
说明	删除指定交易信息
请求头	Authorization: Bearer {token}
参数	路径参数: id (string): 交易 ID
请求示例	/api/trade/12345
返回状态码	200: 删除成功 403: 无权限删除
返回数据格式	json { "message": "交易信息已删除" }
错误响应	json { "error": "未找到交易信息或无权限操作" }
验证规则	用户只能删除自己发布的交易信息
安全性要求	验证用户身份和权限
备注	无

5. 活动发布与管理模块

发布活动

方法	POST
路径	/api/activity/create
说明	用户创建骑行活动
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	title (string): 活动标题 description (string): 活动描述 date (string): 活动日期 location (string): 活动地点 maxParticipants (int): 最大参与人数
请求示例	json { "title": "秋季骑行活动", "description": "环湖骑行 50 公里", "date": "2024-12-01", "location": "XX 公园",

方法	POST
	"maxParticipants": 50 }
返回 状态 码	200: 活动发布成功 400: 参数错误
返回 数据 格式	json { "message": "活动发布成功", "activityId": "12345" }
错误 响应	json { "error": "活动标题或日期无效" }
验证 规则	活动日期需为未来日期，标题长度在 5-100 字符之间
安全 性要 求	用户需登录后才能发布
备注	无

获取活动列表

方法	GET
路径	/api/activity/list
说明	获取所有骑行活动信息
请求 头	无
参数	page (int): 页码 size (int): 每页条数
请求 示例	/api/activity/list?page=1&size=10
返回 状态 码	200: 请求成功 400: 参数错误
返回 数据 格式	json [{ "activityId": "12345", "title": "秋季骑行活动", "date": "2024-12-01", "location": "XX 公园", "participants": 25, "maxParticipants": 50 }]
错误	json { "error": "无效的页码或每页条数" }

响应	
验证规则	页码和每页条数需为正整数
安全性要求	无特殊要求
备注	支持分页查询

获取活动详情

方法	GET
路径	/api/activity/{id}
说明	查询指定活动的详细信息
请求头	无
参数	路径参数：id (string)：活动 ID
请求示例	/api/activity/12345
返回状态码	200：请求成功 404：活动不存在
返回数据格式	json { "activityId": "12345", "title": "秋季骑行活动", "description": "环湖骑行 50 公里", "date": "2024-12-01", "location": "XX 公园", "participants": 25, "maxParticipants": 50 }
错误响应	json { "error": "活动信息不存在" }
验证规则	活动 ID 需为有效格式
安全性要求	无特殊要求
备注	无

报名参加活动

方法	POST
路径	/api/activity/join
说明	用户报名参加活动
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	activityId (string): 活动 ID
请求示例	json { "activityId": "12345" }
返回状态码	200: 报名成功 400: 参数错误 403: 活动已满
返回数据格式	json { "message": "报名成功", "participants": 26 }
错误响应	json { "error": "活动已达到报名人数上限" }
验证规则	活动报名人数不得超过 maxParticipants
安全性要求	用户需登录后才能报名
备注	无

取消报名

方法	POST
路径	/api/activity/cancel
说明	用户取消已报名的活动
请求头	Authorization: Bearer {token}Content-Type: application/json
参数	activityId (string): 活动 ID
请求示例	json { "activityId": "12345" }
返回状态码	200: 取消成功 400: 参数错误
返回数据格式	json { "message": "取消报名成功", "participants": 24 }
错误响应	json { "error": "未找到报名记录" }
验证规则	仅允许报名用户取消
安全性要求	用户需验证身份
备注	无

删除活动

方法	DELETE
路径	/api/activity/{id}
说明	发布者删除指定活动
请求头	Authorization: Bearer {token}
参数	路径参数: id (string): 活动 ID
请求示例	/api/activity/12345
返回状态码	200: 删除成功 403: 无权限删除
返回数据格式	json { "message": "活动已删除" }
错误响应	json { "error": "未找到活动或无权限操作" }
验证规则	仅允许活动发布者删除
安全性要求	验证用户身份和权限
备注	无

接口模块总结

- 登录与注册模块：用户账户管理。
- 骑行记录模块：记录与查询骑行数据。
- 骑行社区模块：互动帖子与评论管理。
- 骑行二手交易模块：骑行装备交易信息管理。
- 活动发布与管理模块：骑行活动的组织与报名。