

# Momento系统接口描述文档

## Momento系统接口描述文档

### 1.内部接口

#### 1.1 用户服务

- 接口一: register(RegisterUserReqDTO registerUserReqDTO)
- 接口二: updateUserInfo(UpdateUserInfoReqVO updateUserInfoReqVO)
- 接口三: updatePassword(UpdateUserPasswordReqDTO updateUserPasswordReqDTO)
- 接口四: findByEmail(FindUserByEmailReqDTO findUserByEmailReqDTO)
- 接口五: findByNickname(FindUserByNicknameReqDTO findUserByNicknameReqDTO)
- 接口六: findById(FindUserByIdReqDTO findUserByIdReqDTO)
- 接口七: findByIds(FindUsersByIdsReqDTO findUsersByIdsReqDTO)
- 接口八: getCurrentUserInfo()

#### 1.2 用户关系服务

- 接口一: follow(FollowUserReqVO followUserReqVO)
- 接口二: unfollow(UnfollowUserReqVO unfollowUserReqVO)
- 接口三: findFollowingList(FindFollowingListReqVO findFollowingListReqVO)
- 接口四: findFansList(FindFansListReqVO findFansListReqVO)

#### 1.3 笔记服务

- 接口一: publishNote(PublishNoteReqVO publishNoteReqVO)
- 接口二: findNoteDetail(FindNoteDetailReqVO findNoteDetailReqVO)
- 接口三: updateNote(UpdateNoteReqVO updateNoteReqVO)
- 接口四: deleteNote(DeleteNoteReqVO deleteNoteReqVO)
- 接口五: addComment(CommentReqVO commentReqVO)
- 接口六: deleteComment(Long commentId)
- 接口七: getCommentsByNotelId(Long notelId, int page, int size)
- 接口八: addNoteContent(AddNoteContentReqDTO addNoteContentReqDTO)
- 接口九: findNoteContent(FindNoteContentReqDTO findNoteContentReqDTO)

#### 1.4 计数服务

- 接口一: getNoteCountByNotelId(Long notelId)
- 接口二: getUserCountByUserId(Long userId)
- 接口三: void onMessage(String body)
- 接口四: void onMessage(List<String> bodys)
- 接口五: getNoteCount(@RequestBody Map<String, Long> request)
- 接口六: getUserCount(@RequestBody Map<String, Long> request)

#### 1.5 搜索服务

- 1. 搜索服务接口 (SearchService)
- 4. 搜索/推荐结果缓存接口 (SearchCacheService)

### 2.外部接口

#### 2.1 智能推荐服务接口 (RecommendService)

#### 2.2 AI 模型服务接口 (AIModelService)

Momento系统采用微服务架构，主要业务服务有：用户服务(momento-user)负责账号管理和用户信息维护；笔记服务(momento-note)处理内容发布、互动和评论功能；用户关系服务(momento-user-relation)管理用户间的社交关系；计数服务(momento-count)统一处理各类计数指标；AI搜索推荐服务(momento-ai-search)提供智能化的搜索和推荐功能。系统在技术实现上采用多级缓存提升性能，使用消息队列处理异步操作，通过令牌桶实现流量削峰，并运用事务管理确保数据一致性。接口设计遵循统一的响应格式，实现了DTO/VO分离，并提供规范的接口文档。特别是在AI搜索推荐服务中，引入大语言模型实现语义化搜索和个性化推荐，体现了系统的智能化特点。整体设计注重可扩展性和可维护性，通过服务解耦和合理的技术选型，构建了一个功能完整、性能优秀的社交平台后端系统。

# 1.内部接口

## 1.1 用户服务

### 接口一：register(RegisterUserReqDTO registerUserReqDTO)

项目	内容
方法签名	Response<Long> register(RegisterUserReqDTO registerUserReqDTO)
接口调用	userService.register()
实际调用	UserServiceImpl.register()
返回值	Response<Long>
描述	用户注册，返回用户ID

### 接口二：updateUserInfo(UpdateUserInfoReqVO updateUserInfoReqVO)

项目	内容
方法签名	Response<?> updateUserInfo(UpdateUserInfoReqVO updateUserInfoReqVO)
接口调用	userService.updateUserInfo()
实际调用	UserServiceImpl.updateUserInfo()
返回值	Response<?>
描述	更新用户基本信息，如昵称、头像等

### 接口三：updatePassword(UpdateUserPasswordReqDTO updateUserPasswordReqDTO)

项目	内容
方法签名	Response<?> updatePassword(UpdateUserPasswordReqDTO updateUserPasswordReqDTO)
接口调用	userService.updatePassword()
实际调用	UserServiceImpl.updatePassword()
返回值	Response<?>
描述	修改用户密码，需要验证原密码

接口四：findByEmail(FindUserByEmailReqDTO findUserByEmailReqDTO)

项目	内容
方法签名	<code>Response&lt;FindUserByEmailRspDTO&gt; findByEmail(FindUserByEmailReqDTO findUserByEmailReqDTO)</code>
接口调用	<code>userService.findByEmail()</code>
实际调用	<code>UserServiceImpl.findByEmail()</code>
返回值	<code>Response&lt;FindUserByEmailRspDTO&gt;</code>
描述	通过邮箱查询用户信息

接口五：findByNickname(FindUserByNicknameReqDTO findUserByNicknameReqDTO)

项目	内容
方法签名	<code>Response&lt;FindUserByNicknameRspDTO&gt; findByNickname(FindUserByNicknameReqDTO findUserByNicknameReqDTO)</code>
接口调用	<code>userService.findByNickname()</code>
实际调用	<code>UserServiceImpl.findByNickname()</code>
返回值	<code>Response&lt;FindUserByNicknameRspDTO&gt;</code>
描述	通过昵称查询用户信息

接口六：findById(FindUserByIdReqDTO findUserByIdReqDTO)

项目	内容
方法签名	<code>Response&lt;FindUserByIdRspDTO&gt; findById(FindUserByIdReqDTO findUserByIdReqDTO)</code>
接口调用	<code>userService.findById()</code>
实际调用	<code>UserServiceImpl.findById()</code>
返回值	<code>Response&lt;FindUserByIdRspDTO&gt;</code>
描述	通过用户ID查询用户信息

接口七：findByIds(FindUsersByIdsReqDTO findUsersByIdsReqDTO)

项目	内容
方法签名	Response<List<FindUserByIdRspDTO>> findByIds(FindUsersByIdsReqDTO findUsersByIdsReqDTO)
接口调用	userService.findByIds()
实际调用	UserServiceImpl.findByIds()
返回值	Response<List<FindUserByIdRspDTO>>
描述	批量查询用户信息，用于用户列表展示

接口八：getCurrentUserInfo()

项目	内容
方法签名	Response<UserDO> getCurrentUserInfo()
接口调用	userService.getCurrentUserInfo()
实际调用	UserServiceImpl.getCurrentUserInfo()
返回值	Response<UserDO>
描述	获取当前登录用户信息

1.2 用户关系服务

接口一：follow(FollowUserReqVO followUserReqVO)

项目	内容
方法签名	Response<?> follow(FollowUserReqVO followUserReqVO)
接口调用	relationService.follow()
实际调用	RelationServiceImpl.follow()
返回值	Response<?>
描述	关注用户，同时更新双方的关注/粉丝计数

接口二：unfollow(UnfollowUserReqVO unfollowUserReqVO)

项目	内容
方法签名	Response<?> unfollow(UnfollowUserReqVO unfollowUserReqVO)
接口调用	relationService.unfollow()

项目	内容
实际调用	<code>RelationServiceImpl.unfollow()</code>
返回值	<code>Response&lt;?&gt;</code>
描述	取消关注用户，同时更新双方的关注/粉丝计数

### 接口三：findFollowingList(FindFollowingListReqVO findFollowingListReqVO)

项目	内容
方法签名	<code>PageResponse&lt;FindFollowingUserRspVO&gt; findFollowingList(FindFollowingListReqVO findFollowingListReqVO)</code>
接口调用	<code>relationService.findFollowingList()</code>
实际调用	<code>RelationServiceImpl.findFollowingList()</code>
返回值	<code>PageResponse&lt;FindFollowingUserRspVO&gt;</code>
描述	分页查询用户的关注列表，包含关注用户的基本信息

### 接口四：findFansList(FindFansListReqVO findFansListReqVO)

项目	内容
方法签名	<code>PageResponse&lt;FindFansUserRspVO&gt; findFansList(FindFansListReqVO findFansListReqVO)</code>
接口调用	<code>relationService.findFansList()</code>
实际调用	<code>RelationServiceImpl.findFansList()</code>
返回值	<code>PageResponse&lt;FindFansUserRspVO&gt;</code>
描述	分页查询用户的粉丝列表，包含粉丝的基本信息

## 1.3 笔记服务

### 接口一：publishNote(PublishNoteReqVO publishNoteReqVO)

项目	内容
方法签名	<code>Response&lt;?&gt; publishNote(PublishNoteReqVO publishNoteReqVO)</code>
接口调用	<code>noteService.publishNote()</code>

项目	内容
实际调用	<code>NoteServiceImpl.publishNote()</code>
返回值	<code>Response&lt;?&gt;</code>
描述	发布笔记,支持图文和视频类型,同时会更新缓存和计数信息

## 接口二：findNoteDetail(FindNoteDetailReqVO findNoteDetailReqVO)

项目	内容
方法签名	<code>Response&lt;FindNoteDetailRspVO&gt; findNoteDetail(FindNoteDetailReqVO findNoteDetailReqVO)</code>
接口调用	<code>noteService.findNoteDetail()</code>
实际调用	<code>NoteServiceImpl.findNoteDetail()</code>
返回值	<code>Response&lt;FindNoteDetailRspVO&gt;</code>
描述	查询笔记详情,采用多级缓存(本地缓存->Redis->DB)

## 接口三：updateNote(UpdateNoteReqVO updateNoteReqVO)

项目	内容
方法签名	<code>Response&lt;?&gt; updateNote(UpdateNoteReqVO updateNoteReqVO)</code>
接口调用	<code>noteService.updateNote()</code>
实际调用	<code>NoteServiceImpl.updateNote()</code>
返回值	<code>Response&lt;?&gt;</code>
描述	更新笔记内容,同时更新缓存

## 接口四：deleteNote(DeleteNoteReqVO deleteNoteReqVO)

项目	内容
方法签名	<code>Response&lt;?&gt; deleteNote(DeleteNoteReqVO deleteNoteReqVO)</code>
接口调用	<code>noteService.deleteNote()</code>
实际调用	<code>NoteServiceImpl.deleteNote()</code>
返回值	<code>Response&lt;?&gt;</code>
描述	删除笔记,同时清理缓存和相关计数信息

接口五：addComment(CommentReqVO commentReqVO)

项目	内容
方法签名	Response<?> addComment(CommentReqVO commentReqVO)
接口调用	commentService.addComment()
实际调用	CommentServiceImpl.addComment()
返回值	Response<?>
描述	添加评论,支持一级评论和二级评论

接口六：deleteComment(Long commentId)

项目	内容
方法签名	Response<?> deleteComment(Long commentId)
接口调用	commentService.deleteComment()
实际调用	CommentServiceImpl.deleteComment()
返回值	Response<?>
描述	删除评论及其子评论(逻辑删除)

接口七：getCommentsByNoteId(Long noteId, int page, int size)

项目	内容
方法签名	Response<List<CommentWithUserVO>> getCommentsByNoteId(Long noteId, int page, int size)
接口调用	commentService.getCommentsByNoteId()
实际调用	CommentServiceImpl.getCommentsByNoteId()
返回值	Response<List<CommentWithUserVO>>
描述	分页查询笔记的一级评论,包含发表评论的用户信息

接口八：addNoteContent(AddNoteContentReqDTO addNoteContentReqDTO)

项目	内容
方法签名	Response<?> addNoteContent(AddNoteContentReqDTO addNoteContentReqDTO)

项目	内容
接口调用	<code>noteContentService.addNoteContent()</code>
实际调用	<code>NoteContentServiceImpl.addNoteContent()</code>
返回值	<code>Response&lt;?&gt;</code>
描述	存储笔记内容,使用 UUID 作为内容标识

### 接口九：findNoteContent(FindNoteContentReqDTO findNoteContentReqDTO)

项目	内容
方法签名	<code>Response&lt;FindNoteContentRspDTO&gt; findNoteContent(FindNoteContentReqDTO findNoteContentReqDTO)</code>
接口调用	<code>noteContentService.findNoteContent()</code>
实际调用	<code>NoteContentServiceImpl.findNoteContent()</code>
返回值	<code>Response&lt;FindNoteContentRspDTO&gt;</code>
描述	根据 UUID 查询笔记内容

## 1.4 计数服务

### 接口一：getNoteCountByNoteId(Long noteId)

项目	内容
方法签名	<code>NoteCountDO getNoteCountByNoteId(Long noteId)</code>
接口调用	<code>countService.getNoteCountByNoteId()</code>
实际调用	<code>CountServiceImpl.getNoteCountByNoteId()</code>
返回值	<code>NoteCountDO</code>
描述	查询笔记的计数信息（点赞数、收藏数、评论数等）

### 接口二：getUserCountByUserId(Long userId)

项目	内容
方法签名	<code>UserCountDO getUserCountByUserId(Long userId)</code>
接口调用	<code>countService.getUserCountByUserId()</code>



项目	内容
实际调用	<code>CountServiceImpl.getUserCountByUserId()</code>
返回值	<code>UserCountDO</code>
描述	查询用户的计数信息（关注数、粉丝数、获赞数等）

### 接口三：void onMessage(String body)

项目	内容
方法签名	<code>void onMessage(String body)</code>
接口调用	<code>CountNoteCollect2DBConsumer.onMessage()</code>
实际调用	<code>CountNoteCollect2DBConsumer.onMessage()</code>
返回值	<code>void</code>
描述	处理笔记收藏计数更新，使用令牌桶限流，批量聚合计数

### 接口四：void onMessage(List<String> bodys)

项目	内容
方法签名	<code>void onMessage(List&lt;String&gt; bodys)</code>
接口调用	<code>CountFansConsumer.onMessage()</code>
实际调用	<code>CountFansConsumer.onMessage()</code>
返回值	<code>void</code>
描述	处理用户粉丝计数更新，按目标用户分组聚合计数

### 接口五：getNoteCount(@RequestBody Map<String, Long> request)

项目	内容
方法签名	<code>Response&lt;NoteCountDO&gt; getNoteCount(@RequestBody Map&lt;String, Long&gt; request)</code>
接口调用	<code>POST /count/note</code>
实际调用	<code>CountController.getNoteCount()</code>
返回值	<code>Response&lt;NoteCountDO&gt;</code>
描述	HTTP接口：查询笔记计数信息

接口六：getUserCount(@RequestBody Map<String, Long> request)

项目	内容
方法签名	<code>Response&lt;UserCountDO&gt; getUserCount(@RequestBody Map&lt;String, Long&gt; request)</code>
接口调用	<code>POST /count/user</code>
实际调用	<code>CountController.getUserCount()</code>
返回值	<code>Response&lt;UserCountDO&gt;</code>
描述	HTTP接口：查询用户计数信息

1.5 搜索服务

1. 搜索服务接口 (SearchService)

项目	内容
方法签名	<code>Response&lt;List&lt;SearchNoteVO&gt;&gt; semanticSearchNotes(SemanticSearchReqVO searchReqVO)</code>
接口调用	<code>searchService.semanticSearchNotes()</code>
实际调用	<code>SearchServiceImpl.semanticSearchNotes()</code>
返回值	<code>Response&lt;List&lt;SearchNoteVO&gt;&gt;</code>
描述	基于语义的笔记搜索，支持自然语言查询，返回相关度排序的笔记列表

项目	内容
方法签名	<code>Response&lt;List&lt;SearchUserVO&gt;&gt; semanticSearchUsers(SemanticSearchReqVO searchReqVO)</code>
接口调用	<code>searchService.semanticSearchUsers()</code>
实际调用	<code>SearchServiceImpl.semanticSearchUsers()</code>
返回值	<code>Response&lt;List&lt;SearchUserVO&gt;&gt;</code>
描述	基于语义的用户搜索，支持通过兴趣、简介等信息搜索用户

4. 搜索/推荐结果缓存接口 (SearchCacheService)

项目	内容
方法签名	<code>void cacheSearchResults(String query, List&lt;SearchResultVO&gt; results)</code>
接口调用	<code>searchCacheService.cacheSearchResults()</code>
实际调用	<code>SearchCacheServiceImpl.cacheSearchResults()</code>
返回值	<code>void</code>
描述	缓存搜索结果，提高热门查询的响应速度

项目	内容
方法签名	<code>Response&lt;List&lt;SearchResultVO&gt;&gt; getCachedResults(String query)</code>
接口调用	<code>searchCacheService.getCachedResults()</code>
实际调用	<code>SearchCacheServiceImpl.getCachedResults()</code>
返回值	<code>Response&lt;List&lt;SearchResultVO&gt;&gt;</code>
描述	获取缓存的搜索结果

2.外部接口

2.1 智能推荐服务接口 (RecommendService)

项目	内容
方法签名	<code>Response&lt;List&lt;RecommendNoteVO&gt;&gt; getPersonalizedNoteRecommendations(Long userId, int size)</code>
接口调用	<code>recommendService.getPersonalizedNoteRecommendations()</code>
实际调用	<code>RecommendServiceImpl.getPersonalizedNoteRecommendations()</code>
返回值	<code>Response&lt;List&lt;RecommendNoteVO&gt;&gt;</code>
描述	基于用户兴趣和行为的个性化笔记推荐

项目	内容
方法签名	<code>Response&lt;List&lt;RecommendUserVO&gt;&gt; getPersonalizedUserRecommendations(Long userId, int size)</code>
接口调用	<code>recommendService.getPersonalizedUserRecommendations()</code>

项目	内容
实际调用	<code>RecommendServiceImpl.getPersonalizedUserRecommendations()</code>
返回值	<code>Response&lt;List&lt;RecommendUserVO&gt;&gt;</code>
描述	基于用户兴趣和社交网络的个性化用户推荐

## 2.2 AI 模型服务接口 (AIModelService)

项目	内容
方法签名	<code>Response&lt;String&gt; generateEmbedding(String content)</code>
接口调用	<code>aiModelService.generateEmbedding()</code>
实际调用	<code>AIModelServiceImpl.generateEmbedding()</code>
返回值	<code>Response&lt;String&gt;</code>
描述	生成文本的向量嵌入表示，用于语义搜索

项目	内容
方法签名	<code>Response&lt;List&lt;String&gt;&gt; generateTags(String content)</code>
接口调用	<code>aiModelService.generateTags()</code>
实际调用	<code>AIModelServiceImpl.generateTags()</code>
返回值	<code>Response&lt;List&lt;String&gt;&gt;</code>
描述	自动为内容生成标签，用于内容分类和推荐