

TeamNote接口详细设计

Service

（以下接口中的userId类型的参数具体是怎样还要看一下spring security再定——身份和权限验证）可能不需要动作发出者的Id作为参数

1.CreateNoteService:

- **createNotebook**: 前台发送创建者id和笔记本名称，在后台数据库中作出相关操作
- **uploadFileNote**: 前台发送笔记上传人id和、笔记所属笔记本id，笔记内容，上传时间。其中笔记内容的类型（File or Text）前台也要发送，在相应controller里面决定调用哪个function
- **uploadTextNote**: 与上面类似，不赘言

2.CooperateService:

- **inviteCooperator**: 前台发送邀请者id、邀请目标用户id、notebookid、邀请描述等信息，通过这些参数生成链接，通过Util中的NoticeUtil向指定用户发送通知，也就是对数据库作出相关修改
- **viewInvitation**: 被邀请用户点击收到通知中的URL，该函数返回URL中包含的邀请者，notebook、邀请描述等信息，用来呈现接受邀请页面
- **takeInvitation**: 受邀者在接受邀请页面，前台提供邀请相关信息和受邀者相关信息，以及是否接受邀请，后台对数据库进行相应修改
- **pushUpdate**: 笔记管理团队push更新，根据前台参数修改相关数据库
- **reset**: 版本回滚，根据参数对数据库作出相关修改
- **raiseAdvice**: 普通用户对笔记提出修改意见，根据参数生成链接，向笔记管理团队发送通知，修改数据库相关表

- **mergeAdvice**: 笔记管理团队采纳其他用户提出的意见，并将用户做出的更新merge到原来的版本上，根据链接中的参数（包括提出意见者，笔记管理者，笔记id和提出修改意见的时间，修改相关数据库
- **sendGroupChat**: 笔记管理团队成员之间的群聊，主要实现在前端，向后端发送参数主要是将群聊内容存到数据库中
- **getGroupChat**: 前端发来notebookid，从数据库里取出群聊内容
- **giveOwnership**: 前端发来新旧owner的id以及笔记本id，修改相关数据库，直接设置新owner不需要经过同意，（发送提醒）

3.RecommendService:

- **getRecommendNotebooks**: 前台提供用户id信息，该函数返回经过推荐算法得出的notebooks，以json array的形式返回给前台进行呈现
(request.setAttribute(xxx)的形式)

4.SearchService:

- **search**: 前台发送用户id，搜索关键词，该函数返回json array给controller用于设置model的attribute

5.EvaluateService:

- **upvote/downvote**: 对后台数据库作出相应修改即可
- **reportNote/reportComment**: 举报笔记or评论
- **reward**: 打赏，通过前端传来的打赏人Id和noteId，在数据库里寻找到笔记拥有者的支付宝二维码
- **comment**: 根据参数对数据库作出相应修改即可

6.LetterService:

- **sendLetter/readLetter**: 根据参数分别向数据库写相应数据or从数据库读出相应数据
- **getAllReceivedLetter/gitAllSentLetter**: 相当于是收件箱、发件箱列表

7.DownloadService:

- **downloadNote/downloadNotebook**: 分别返回文件型的笔记 / 笔记本内的所有笔记

8.NoteManageService:

- 主要包括对笔记和笔记本的增删查改等，这里不作更具体的说明

9.UserBasicService:

- 包括注册、登录登出（? ? ? ）、对个人信息的修改、管理自己感兴趣的标签等，主要也都是对数据库的直接操作，不作更具体说明

10.AdminService:

- 包括数据库里各种数据的管理（CRUD）、审核功能