

UserApi«interface»
+createUser(user:UserDto):void +updateUser(user:UserDto):void +deleteUser(userId:String):void +getUserById(userId:String):UserDto +addFollow(subscriberId:String, followedId:String):void +getFollow(userId:String, page:int, size:int):List<UserDto> +removeFollow(subscriberId:String, followedId:String):void +searchUser(userId:String, page:int, size:int):List<UserDto>

PublicRecipeApi«interface»
+addRecipe(recipe:PublicRecipeDto):void +updateRecipe(recipe:PublicRecipeDto):void +deleteRecipe(recipeId:int):void +getRecipe(recipeId:int):PublicRecipeDto +setRating(recipeId:int, userId:String, value:int):void +getRatingFromUser(userId:String, recipeId:int):int +getAverageRating(recipeId:int):double +search(title:String, tags:List<String>, ingredients:List<String>, creationDate: Date, alreadyLoad:int, readCount:int):List<PublicRecipe> +reportRecipe(recipeId:int):void

FriendRequestApi«interface»
+publishRequest(request: FriendRelationDto) +getOpenRequestsforMe(userId:String, page:int, count:int) : List<FriendRequest> +getFriends(userId:String, page:int, size:int) : List<String> +deleteFriend(enquirerId:String, requestedFriendId:String):void

GroupApi«interface»
+createGroup(group:GroupDto):void +updateGroup(group:GroupDto):void +deleteGroup(groupId:int):void +getGroupById(groupId:int):GroupDto +addUserToGroup(userId:String, groupId:int):void +getGroups(userId:String, page:int, size:int):List<GroupDto> +removeUserFromGroup(userId:String, groupId:int):void +getSharedRecipesFromGroup(groupId:int, page:int, size:int):List<SharedPrivateRecipe> +addSharedRecipeToGroup(groupId:int, sharedRecipe:SharedPrivateRecipeDto):void +deleteSharedRecipeFromGroup(groupId:int, sharedRecipeId:int):void +updateRecipe(groupId:int,recipe:SharedPrivateRecipeDto):void +reportGroup(groupId:int):void +reportSharedPrivateRecipe(recipeId:int):void

FavouritesApi«Interface»
+addFavRecipe(userId:String, recipeId:int):void +getFavRecipe(userId:String, page:int, size:int):List<PublicRecipe> +delFavRecipe(userId:String, recipeId):void

AdminApi«interface»
+getReportedPublicRecipe(size:int, page:int):List<Recipe> +getReportedSharedPrivateRecipe(size:int, page:int):List<SharedPrivateRecipe> +getReportedUsers(size:int, page:int):List<UserDto> +getReportedGroups(size:int, page:int):List<GroupDto> +getReportedComments(size:int, page:int):List<CommentDto> +deReportPublicRecipe(recipeId:int):void +deReportSharedPrivateRecipe(recipeId:int):void +deReportUser(userId:String):void +deReportGroups(groupId:int):void +deReportComment(commentId:int):void

CommentApi«interface»
+addComment(comment:CommentDto):void +updateComment(comment:CommentDto):void +deleteComment(commentId:int):void +getComments(recipeId:int, page:int, size:int):List<Comment> +reportComment(commitId:int):void